The Multivariate price formation process and cross-impact

Mehdi Tomas

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2022IPPAX021

Thèse de doctorat

The multivariate price formation process and cross-impact

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'Ecole Polytechnique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 30 Mars 2022, par

MEHDI TOMAS

Composition du Jury :

Damien Challet Professeur, MICS, CentraleSupélec	Président
Jean-François Muzy Directeur de Recherche, CNRS & Université de Corse	Rapporteur
Alexander Schied Professor, Department of Statistics and Actuarial Science, Waterloo University	Rapporteur (absent)
Aurélien Alfonsi Professeur, CERMICS, École Nationale des Ponts et Chaussées	Examinateur
Caroline Hillairet Professeur, ENSAE-CREST, ENSAE Paris	Examinatrice
Sophie Laruelle Professeur associé, LAMA, Université Paris-Est Créteil	Examinatrice
Fabrizio Lillo Professor, Dipartimento di Matematica, Università di Bologna	Examinateur
Mathieu Rosenbaum PRCE, CMAP, École Polytechnique	Directeur de thèse
Michael Benzaquen Professeur, CNRS & LadHyX, École Polytechnique	Co-directeur de thèse
Jean-Philippe Bouchaud Directeur de recherche, Capital Fund Management & Académie des Sciences	Invité
Blanka Horvath Professur, Finanzmathematik, Technische Universität München	Invitée
Iacopo Mastromatteo Chercheur, Capital Fund Management	Invité

Remerciements

Je souhaite d'abord exprimer ma plus grande gratitude envers Michael Benzaquen et Mathieu Rosenbaum pour avoir accepté de diriger ma thèse et envers Iacopo Mastromatteo pour son encadrement. Votre confiance, conseils et bienveillance durant ces trois années les ont rendu enrichissantes et passionnantes. Je remercie tout particulièrement Jean-Philippe Bouchaud pour avoir accepté de co-financer cette thèse dans le cadre de la chaire Éconophysique et Systèmes Complexes. Sans ce soutien, cette thèse n'aurait pas pu voir le jour.

Je remercie grandement Jean-François Muzy et Alexander Schied d'avoir accepté d'être rapporteurs pour cette thèse. Je remercie également Aurélien Alfonsi, Damien Challet, Sophie Laruelle, Caroline Hillairet et Fabrizio Lillo pour avoir accepté de faire partie du jury. Les résultats de leur recherche ayant profondément nourri ce manuscrit, je suis honoré de l'intérêt porté pour mes travaux.

Cette thèse a bénéficié de plusieurs collaborations et je remercie mes co-auteurs: Blanka Horvath, Aitor Muguruza, Christian Bayer et Benjamin Stemper. Je remercie chaleureusement Blanka Horvath et Aitor Muguruza pour m'avoir encouragé à poursuivre une thèse et pour leur recommandations qui m'ont amené à rencontrer Mathieu et Michael.

Merci à Eduardo Abi-Jaber, Jean-Philippe Bouchaud, Zoltán Eisler et Bence Tóth pour les discussions scientifiques qui m'ont permis de gagner en compréhension à la fois en finance et en mathématiques.

Je remercie le LadHyX et le CMAP pour leur accueil durant cette thèse. Un grand merci à l'équipe des doctorants en mathématiques financières avec qui j'ai eu le plaisir de partager ces années: Aditi, Bastien, Heythem, Paul, Pamela, Marcos et Othmane. De même, un grand merci à Antoine, Armine, Christian, Cécilia, Frederic, Federico, José, Jérôme, Johannes, Karl, Michele, Pierre Mergny et Pierre Lecointre, Pierre-Philippe, Rudy, Riccardo, Samy, Théo et Valerio. Je remercie en particulier Antoine, José, Michele et Frédéric pour nos discussions enrichissantes sur les problèmes scientifiques rencontrées durant cette thèse.

Je remercie la société Capital Fund Management de m'avoir fourni un support de travail essentiel a ma thèse.

Je tiens à remercier Nasséra Naar et Alexandra Noiret du secrétariat du CMAP ainsi que Delphine L'Huillier du secrétariat de la chaire pour leur disponibilité, aide et patience durant ces trois ans.

Enfin, je remercie mes parents, ma famille et mes amis pour leur soutien.

Résumé

Cette thèse comprend six parties. La première lie les flux d'ordres anonymes et les variations de prix à l'aide de modèles d'impact croisé statiques et linéaires. Nous énumérons les propriétés souhaitables de ces modèles, caractérisons ceux qui les satisfont et les testons sur différents marchés. La deuxième partie étend cette approche aux produits dérivés afin d'obtenir une méthode d'estimation pour l'impact croisé que nous appliquons aux options SP500 et aux contrats à terme VIX. Dans la troisième partie, nous généralisons les modèles précédents pour prendre en compte l'influence des ordres passés sur les prix. La quatrième partie utilise des données de méta-ordres sur les actions et les contrats à terme pour proposer une formule d'impact croisé qui généralise la loi de la racine carrée de l'impact. Dans la cinquième partie, nous présentons un modèle tick-par-tick pour la dynamique des prix multivariés en utilisant les processus de Hawkes. La dernière partie résout le problème de la calibration des modèles de volatilité en utilisant des réseaux de neurones.

La première partie étudie les modèles statiques linéaires pour l'impact croisé. Ces modèles dépendent des covariances des transactions et des prix. Nous introduisons des propriétés pour qu'un modèle se comporte correctement dans différentes conditions de marché. Nous montrons qu'il existe un seul modèle qui satisfait toutes ces propriétés. Nous appliquons différents modèles sur des actions et des contrats à terme. Le modèle précédent est l'un des deux modèles robustes sur les marchés étudiés. Il s'agit donc d'un bon candidat pour une vision unifiée du processus de formation des prix.

La deuxième partie généralise l'approche précédente aux produits dérivés. Nous dérivons une méthode d'estimation pour l'impact croisé à partir de covariances de faible dimension. Sur des données de produits dérivés sur le SP500 et des contrats à terme VIX, le modèle explique en partie les fluctuations du sous-jacent et de la surface de volatilité implicite.

Dans la deuxième partie, nous étudions des modèles linéaires à noyaux pour l'impact croisé. Nous examinons deux classes de modèles : ceux qui anticipent le flux d'ordres pour fixer des prix martingale et ceux qui empêchent l'arbitrage statistique. Nous montrons qu'il existe au plus un noyau appartenant aux deux classes mais qu'il n'empêche pas nécessairement l'arbitrage. Pour résoudre ce problème, nous introduisons un second noyau qui empêche l'arbitrage statistique et qui est le plus proche possible à donner des prix martingale. Enfin, nous calibrons ces noyaux sur des données de contrats à terme.

La troisième partie mesure l'impact croisé avec deux bases de données d'ordres envoyés par des gestionnaires d'actifs sur des actions et des contrats à terme. Nous proposons une formule pour l'impact croisé qui généralise la loi de la racine carrée et donne des prédictions plus précises sur nos données.

Dans la quatrième partie, nous modélisons le processus de prix tick-par-tick avec des processus

de Hawkes. Pour capturer l'endogénéité des marchés financiers, nous étudions la limite où la norme L^1 du rayon spectral du noyau de Hawkes devient égale à un. Certains modèles de volatilité rugueuse multivariée émergent alors comme la limite macroscopique de la dynamique microscopique des prix. Le processus de volatilité de ces modèles est une combinaison de facteurs de variance entraînés par un mouvement brownien fractionnaire d'indice de Hurst commun.

Enfin, la dernière partie de cette thèse examine la calibration des modèles de volatilité à l'aide de réseaux de neurones. Nous approchons la fonction donnant les prix des contrats à partir des paramètres du modèle en utilisant des réseaux de neurones. Cette approximation est ensuite utilisée pour obtenir les paramètres du modèle à partir de prix de marché des contrats. Nous mettons en évidence l'applicabilité de la méthode en utilisant des données de marché synthétiques et réelles.

Abstract

This thesis comprises six parts. The first relates anonymous order flow and price changes using static, linear cross-impact models. We list desirable properties of such models, characterise those which satisfy them and test their predictions on different markets. The second part extends this approach to derivatives to obtain a tractable estimation method for cross-impact which is applied to SP500 options and VIX futures. In the third part, we generalise the previous setup to derive and estimate cross-impact models which account for the influence of past trades on current prices. The fourth part uses meta-order databases on stocks and futures to propose a formula for cross-impact which generalises the square-root law of market impact. In the fifth part, we propose a tick-by-tick model for price dynamics using Hawkes processes. We investigate scaling limits of prices in the high endogeneity regime to derive multivariate macroscopic price dynamics of rough Heston type. Finally, the last part solves the calibration problem of volatility models using neural networks.

In the first part, we study linear cross-impact models which relate asset prices to anonymous order flow. These models are functions of the covariances of these variables. We introduce properties models should satisfy to behave well across market conditions and show that there exists a unique model which satisfies all such properties. We apply models on stocks and futures and find that the latter model is one of two robust across markets. Thus, it is a good candidate model for a unifying view of the price formation process on stocks and futures.

The second part leverages the candidate model identified in the first part to extend the previous setup to derivatives. We derive an estimation method for the large cross-impact matrix which depends on low-dimensional covariances. Using SP500 options and VIX futures data, we show cross-impact captures salient features of the price formation process on derivatives.

The second part examines cross-impact kernels, which account for the lasting influence of past trades on current prices. We focus on two kernel classes: kernels that anticipate future order flow to set martingale prices and those that prevent statistical arbitrage. We show that there is at most one kernel belonging to both classes. This kernel sets martingale prices but may not prevent arbitrage. To fix this, we introduce a methodology to obtain a second kernel which prevents statistical arbitrage and is the closest to setting martingale prices. Finally, we derive a calibration methodology for both kernels and apply it to futures data.

The third part measures cross-impact from using two databases of proprietary orders sent by asset managers on U.S stocks and futures. These databases allow us to study the cross-impact of individual investor orders. We propose a formula for cross-impact which generalises the square-root law to account for price and order correlations. On both stocks and futures, we find that this generalisation gives more precise predictions than the square-root law.

In the fourth part, we model the tick-by-tick price process using Hawkes processes. To capture the high endogeneity of financial markets, we investigate the limit where the L^1 norm of the spectral radius of the Hawkes kernel goes to one. We show that some multivariate rough volatility models emerge as the macroscopic limit of the microscopic price dynamics. In these models, volatility is a combination of underlying variance factors, each driven by a fractional Brownian motion of common Hurst index.

Finally, the last part examines the calibration of volatility models by using neural networks. We first approximate the map from model parameters to contract prices using neural networks. This approximation can then be used to recover model parameters given market prices of contracts. We highlight the applicability of the method using synthetic and real market data.

List of papers being part of this thesis

- M. Tomas, I. Mastromatteo, M. Benzaquen, *How to build a cross-impact model from first principles: theoretical requirements and empirical results*, to appear in Quantitative Finance, 2022.
- M. Tomas, I. Mastromatteo, M. Benzaquen, *Cross impact in derivative markets*, submitted, 2021.
- M. Rosenbaum, M. Tomas, *A characterisation of cross-impact kernels*, submitted, 2021.
- M. Tomas, I. Mastromatteo, M. Benzaquen, *Cross-impact on meta-orders*, working paper, 2021.
- M. Rosenbaum, M. Tomas, *From microscopic price dynamics to multidimensional rough volatility models*, Advances in Applied Probability, 53(2), 425-462, 2021.
- B. Horvath, A. Muguruza, M. Tomas, *Deep learning volatility: a deep neural network perspective on pricing and calibration in (rough) volatility models*, Quantitative Finance, 21(1), 11-27, 2021.
- C. Bayer, B. Horvath, A. Muguruza, B. Stemper, M. Tomas, *On deep calibration of (rough) stochastic volatility models*, submitted, 2020.

Table of contents

Introduction

The general objective of this thesis is to understand the multivariate price formation process: how and why prices of financial assets move, and what makes them move together. Our goal is to build models which are theoretically plausible, can be estimated on empirical data and produce predictions in line with observations. These models can be used in applications where execution costs play an important role. For example, regulators may leverage estimates of the liquidation costs of large portfolios to assess systemic risk. Investors may use these models to optimize their trading strategies and reduce trading costs. To build these models, a large part of the thesis focuses on the link between orders and prices. We begin by presenting and motivating the different questions on which we want to shed some light in this thesis.

Motivations

A key stylized fact of price formation is that on average, trades on an asset move its price. In fact, a large fraction of the fluctuations of an asset's price can be explained solely from trades on that asset. Thus, trades form a critical part of the price formation process. But models relating trades on an asset to its price most often ignore the influence of other assets. To understand how and why asset prices move together, we need to consider *cross-impact*: how trades on one asset push the price of another asset. Anonymous order flow data, both plentiful and public, is an ideal candidate to begin studying this question. However, even in a static setting where we ignore the lasting influence of past trades, there are many possible models for cross-impact. We might thus ask ourselves the following question:

Question 1. *How can we model static cross-impact from anonymous order flow?*

Among the possible models for cross-impact, our goal is to find a unifying model, applicable across assets. This includes derivatives, which poses challenges for two main reasons. First, in an efficient market, the prices of derivatives should be locked by no-arbitrage. This should constrain cross-impact. Second, for a given underlying, there are thousands of individually illiquid derivatives. Therefore, to estimate cross-impact, we need to aggregate the liquidity of many instruments. This leads us to ask:

Question 2. *How can we estimate cross-impact from anonymous order flow on derivatives?*

A unifying static cross-impact framework is a first step towards understanding the multivariate price formation process across asset classes. However, we also need to understand the dynamics

of the price formation process. In particular, the average price change induced by a trade on the traded asset is maximum right after the trade and decays over time. We expect the same effect to be true of trades on one asset on the price of another. This raises the issue of accounting for this property:

Question 3. *How can we model dynamic cross-impact from anonymous order flow?*

Questions 1 and 3 analyse cross-impact from anonymous order flow. This aggregates the actions of all market participants to explain price moves. To understand how individual agents shape prices by trading, we need to analyze labeled order flow, where we know which market participants sent orders. This would give us a microscopic perspective on the price formation process so that we are interested in this effect:

Question 4. *How can we measure cross-impact from labeled order flow?*

All previous questions explain the price formation process through anonymous or labeled order flow. We may ask which price dynamics we can derive solely from properties of asset prices, beginning at the high-frequency scale, where prices evolve on a discrete grid, and examining the resulting dynamics at the macroscopic scale, where asset prices appear continuous. This leads to the following question:

Question 5. *What macroscopic multivariate price dynamics emerge from microscopic stylized facts?*

A key ingredient in the macroscopic price dynamics of the previous question, and in most models of asset prices, is the volatility process. While increasingly accurate models for volatility have emerged, these models have proved hard to adopt because of the long computation time required to obtain market prices of derivatives. In turn, this makes calibrating the parameters of these models from market data prohibitive. To make these volatility models applicable, we should answer the following question:

Question 6. *How can we quickly calibrate computationally-intensive volatility models?*

Outline

Each question presented above corresponds to a part of the thesis.

Chapter I answers Question 1 by viewing cross-impact models as functions of covariances which are measured on empirical data. This allows us to abstract the particular numerical calibration of a model on a dataset and look for robust models across asset classes. Chapter I introduces desirable properties cross-impact models should satisfy to behave well across markets. We show that there exists a unique cross-impact model which satisfies all such properties. Then, we stress test a variety of cross-impact models on stocks and futures and find that this model is one of two robust across markets.

Using the candidate model identified in Chapter I, Chapter II models cross-impact on derivatives to answer Question 2. In this setting, we find a dimensional reduction recipe to aggregate

non-stationary liquidity of thousands of options into few, stationary liquidity factors. We apply this framework on SP500 options and VIX futures data and find that the cross-impact model explains fluctuations of the underlying and the implied volatility surface. Therefore, the candidate model of Chapter I provides a unifying view of the price formation process on stocks, futures and derivatives.

Chapter III treats Question 3 by setting up a toy market model which reproduces the long-range auto-correlation property of order flow. Within this market, we focus on two distinct classes of propagator models for cross-impact: those that anticipate future order flow to set martingale prices (martingale-admissible kernels) and those that prevent statistical arbitrage. Each class embodies a different notion of price efficiency. We find that there is at most one cross-impact kernel belonging to both classes, which sits as an ideal candidate for a cross-impact model. However, depending on the auto-correlation structure of trades, there may be no model belonging to both classes. Therefore, we introduce a methodology to obtain two plausible cross-impact kernels. The first is martingale-admissible and satisfies necessary conditions for no-arbitrage. The second prevents statistical arbitrage and is the closest kernel (in a certain norm) to kernels leading to martingale prices. We find that the boundary values of both of these kernels can be interpreted as the candidate cross-impact model of Chapters I and II with different input parameters. Finally, we derive a calibration methodology for both kernels and apply it to empirical data.

Chapter IV tackles Question 4 by using two databases of proprietary orders sent by asset managers on U.S. stocks and futures. These databases allow us to study the cross-impact of individual investor orders. We introduce the concept of cross-impact diagonalisation to tackle the curse of dimensionality. Using that principle, we propose a generalisation of the square-root law which accounts for price and order correlations. On both stocks and futures, this generalisation gives more accurate predictions than the square-root law.

Chapter V examines Question 5 by modelling tick-by-tick price dynamics using multivariate Hawkes processes. To capture the high endogeneity of financial markets, we investigate the limit where the L^1 norm of the spectral radius of the Hawkes kernel goes to one. We show that some specific multivariate rough volatility models emerge as the macroscopic limit of the microscopic price dynamics. In these models, volatility is a combination of underlying variance factors, each of which is driven by a fractional Brownian motion of common Hurst index. Furthermore, we are able to conciliate clusters of asset jumps on short time scales with rough volatility and asset return correlations structures on larger time scales.

Finally, Chapters VIA and VIB treat Question 6 by replacing computationally expensive numerical methods to estimate option prices from parameters with neural networks. Then, in the calibration step, this fast approximation is used to search the space of parameters to fit market data. Chapter VIA introduces this approach for different volatility models and shows that it achieves similar levels of precision to standard computational methods while providing a significant speedup of the computation time. We illustrate it by calibrating parameters of a

computationally intensive model on historical data. Chapter VIB examines variations of the setup, including leveraging priors on the distribution of model parameters in the calibration step.

Let us now rapidly review the main results of the thesis.

1 Chapter I: Static cross-impact models

Chapter I studies linear static cross-impact models. These models classically assume that the net anonymous order flow pushes prices linearly, so that all information concerning cross-impact can be encoded in a single matrix, called the *cross-impact matrix*. In practice, most simple cross-impact matrices can be expressed as a function of statistics of order flows and prices. We dub these functions *cross-impact models* and introduce axioms: desirable properties of cross-impact models. We show that only one cross-impact model satisfies all axioms and give its form. Finally, we stress-test cross-impact models on various asset classes to identify which models best explain price changes from anonymous order flow.

1.1 Cross-impact models as functions

To relate anonymous order flow to market prices, we assume that we observe anonymous trades and prices on a universe of d assets. Both are binned on a regular time interval of length Δt. We denote by q_t the net market order flow traded in the time window $[t, t+\Delta t]$. On a given trading day, our goal is to relate the time series of prices $\{p_0, p_{\Delta t}, \cdots, p_T\}$ with the time series of order flows $\{q_0, q_{\Delta t}, \cdots, q_T\}$. For a given asset, it is classically admitted that price changes and net order flow are linearly related [PB18], although the linear relationship breaks down for large order flow values. Inspired by this, we also assume this is true across assets.

Assumption 1. *Price changes $\Delta p_t := p_{t+\Delta t} - p_t$ and order flow imbalances q_t are linearly related, i.e.*

$$\Delta p_t = \Lambda(\Sigma_t, \Omega_t, R_t) q_t + \eta_t, \tag{1}$$

where $\Sigma_t = \mathrm{Cov}(\Delta p_t)$, $\Omega_t := \mathrm{Cov}(q_t)$, $R_t = \mathbb{E}[\Delta p_t q_t^\top]$, Λ is a matrix-valued function of these second-order statistics called a cross impact model and $\eta_t = (\eta_t^1, \ldots, \eta_t^d)$ is a vector of zero-mean random variables representing exogenous noise.

In Equation (1), price changes and order flow imbalances are known and we have to choose the cross-impact model Λ. Assumption 1 states that cross-impact models only depend on parameters of the true data generating process, which is not assumed to be stationary. When price changes, order flows and exogenous noise are Gaussian random variables, then it only depends on $(\Sigma_t, \Omega_t, R_t)$, which motivates the simplification of Assumption 1.

The previous assumption discards the role of past order flow imbalances to focus on the influence of same-time order flows. Thus, these models are dubbed *static* models. Past order flow imbalances do play a role [BMEB17, SL19, WNG17], but we focus on equal-time

relationships between prices and volumes with the time resolution of $\Delta t = 1$ minute. In this regime, the effect of order flow imbalances of previous time windows is typically much smaller than that of the order flow imbalance of the current time window. We will study in detail the influence of past order flows in Chapter III.

A cross-impact model is a function of the three key second-order statistics which describe our system. Thus, we are not only interested in a particular model designed specifically for a given market (i.e., for a given set of second-order statistics). Rather, this view allows us to examine how a model behaves in a variety of market conditions.

1.2 Axioms: the desirable properties of cross-impact models

Given the many possible choices for cross-impact models, we are concerned with identifying those which are theoretically more plausible. To do so, we introduce axioms which reflect desirable properties of cross-impact models. Each axiom can be expressed mathematically as a constraint on the cross-impact model. For brevity, we present a selection of different axioms, the full list being detailed in Chapter I.

1.2.1 Symmetry axioms

We begin with symmetry axioms which constrain the behaviour of cross-impact models under natural transformations of second-order statistics. One example is that the cross-impact model should adapt to the re-ordering of the assets, which we call *permutation invariance*. We can translate this mathematically: a cross impact model Λ is permutation-invariant if, for any permutation matrix P and $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$,

$$\Lambda(P\Sigma P^\top, P\Omega P^\top, PRP^\top) = P\Lambda(\Sigma, \Omega, R)P^\top. \tag{2}$$

Other axioms can be translated mathematically, but we omit these details in this introduction for conciseness.

While the previous axiom is fairly simple, other symmetry axioms may be more intricate. For example, stock splits can double the number of outstanding shares and halve their values. The split modifies the second-order statistics of the system. However, barring microstructural effects, cross-impact should be consistent between before and after the stock split. This axiom is dubbed *split invariance*.

Overall, symmetry axioms are simple (although important) properties we should ask of cross-impact models.

1.2.2 Fragmentation axioms

Cross-impact models which respect fragmentation invariance are adapted to scenarios of extreme price correlations. For example, consider a stock traded on multiple markets (say, Apple traded on the Nasdaq and on the Bats venues). In our framework, this corresponds to

two different instruments. However, we expect that, abstracting microstructural effects, the prices on both venues is the same. Thus, buying a volume $q = q_{\text{Nasdaq}} + q_{\text{Bats}}$ of Apple stock should yield the same cost no matter how one fragments the q_{Nasdaq} units bought on Nasdaq and the q_{Bats} units bought on Bats. For this reason, this class of axiom is dubbed *fragmentation invariance*.

Fragmentation invariance axioms are critical to apply cross-impact models in strong correlation regimes.

1.2.3 Stability axioms

Stability axioms control cross impact models when instruments have different levels of liquidity. Intuitively, it should not be possible to manipulate the price of a liquid product by trading an illiquid product (and vice-versa). Stability axioms prevent this.

Stability axioms are particularly important in markets where some assets are much more liquid than others.

1.2.4 No-arbitrage axioms

No-arbitrage axioms prevent the presence of statistical arbitrages: trading strategies which generate profits on average, as defined in [Gat10]. We distinguish between two types of statistical arbitrages. Static statistical arbitrages generate positive average profits on a single time period and dynamic statistical arbitrages, which generate positive average profits by trading on multiple time periods.

No-arbitrage axioms are particularly useful for practical applications such as portfolio construction or optimal execution since they exclude trading strategies which manipulate prices to make a profit.

Again, recall that each axiom can be expressed mathematically as a constraint on the cross-impact model, in the spirit of Equation (2).

1.2.5 Characterisation of cross-impact models

Finally, Chapter I shows that the set of cross-impact models which satisfies all axioms reduces to a singleton, which is the topic of the next result.

Result 1. *There is a unique cross-impact model which satisfies all axioms, the* `kyle` *model, defined for any* $(\Sigma, \Omega, R) \in (\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{kyle}}(\Sigma, \Omega, R) := (\mathcal{L}^\top)^{-1} \sqrt{\mathcal{L}^\top \Sigma \mathcal{L}}\, \mathcal{L}^{-1}, \tag{3}$$

where $\mathcal{L}$ *is a matrix such that* $\mathcal{L}\mathcal{L}^\top = \Omega$ *and* $\sqrt{M}$ *is the symmetric square root of the matrix* M*.*

The model from Result 1 is dubbed `kyle` because it was first derived from a multivariate generalisation of the Kyle insider trading framework [Kyl85, CK94, GdMMBB20]. Interestingly, the `kyle` model only depends on covariances of prices and order flows. Furthermore, a key matrix in the `kyle` model is $\sqrt{\mathscr{L}^\top \Sigma \mathscr{L}}$. This matrix can be interpreted as the risk-weighted liquidity which mixes information about volatility and liquidity to estimate cross-impact. We will see that the `kyle` model appears in an even more general setting, as in Chapter III. Result 1 provides some theoretical explanation for its seemingly fundamental role.

Beyond the special role of the `kyle` model among all cross-impact models, we are interested in stress-testing models on empirical data to assess their robustness.

1.3 Fit on empirical data

Chapter I analyses a variety of cross-impact models, but we focus our results on four key models, which correspond to different choices of the function Λ:

1. The `r-direct` cross-impact model which ignores cross-sectional effects and serves as a benchmark.

2. The `r-el` model, suggested in [BMEB17], which pools liquidity of different assets using the structure of the price change correlation matrix.

3. The `kyle` model first introduced in [GdMMBB20, CK94], which is the only model that satisfies all axioms.

4. The `ml` model which maximises empirical fit.

Given a cross-impact model Λ, a time series of net traded order flows $\{q_t\}_{t=1}^T$ and a time series of price changes $\{\Delta p_t\}_{t=1}^T$ along with the second-order statistics Σ_t, Ω_t, R_t we build the residuals:

$$\varepsilon_t := \Delta p_t - \Lambda(\Sigma_t, \Omega_t, R_t) q_t \,.$$

To compare impact models, we use three different indicators of performance which emphasize different aspects of prediction errors. For a given matrix M, the M-weighted generalized $\mathscr{R}^2$ is defined as

$$\mathscr{R}^2(M) := 1 - \frac{\sum_{1 \le t \le T} \varepsilon_t^\top M \varepsilon_t}{\sum_{1 \le t \le T} \Delta p_t^\top M \Delta p_t} \,.$$

To highlight different sources of error, we consider the following choices of M:

1. To measure the average errors relative to the typical deviation of each asset, we choose $M = I_\sigma := \mathrm{diag}(\sigma^{-2})$. This score gives a proxy for the fraction of volatility explained by the cross-impact model for each asset.

2. To measure the model's accuracy to explain global price changes, we choose $M = J_\sigma := (\Sigma_{ii}^{-1/2} \Sigma_{jj}^{-1/2})_{1 \le i,j \le m}$. This score highlights how well the cross-impact model explains the price changes of a risk-weighted index made up of all assets.

Model	Crude Futures			Bond and index Futures			Stocks		
	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$
`r-direct`	0.27 ± 0.01	0.40 ± 0.01	$-\infty$	0.23 ± 0.01	0.27 ± 0.02	-1.71 ± 0.05	0.274 ± 0.004	0.479 ± 0.010	0.180 ± 0.003
`r-el`	0.37 ± 0.01	0.46 ± 0.01	0.31 ± 0.02	0.27 ± 0.01	0.17 ± 0.02	0.13 ± 0.03	0.249 ± 0.003	0.644 ± 0.005	0.124 ± 0.004
`kyle`	0.35 ± 0.01	0.46 ± 0.01	0.29 ± 0.02	0.38 ± 0.01	0.29 ± 0.02	0.11 ± 0.03	0.336 ± 0.003	0.808 ± 0.004	0.211 ± 0.003
`ml`	0.37 ± 0.01	0.45 ± 0.01	0.31 ± 0.02	0.40 ± 0.01	0.30 ± 0.02	0.20 ± 0.03	0.358 ± 0.003	0.803 ± 0.004	0.208 ± 0.003

Table .1: **Goodness of fit of cross impact models on different datasets**.
Goodness of fit was measured using two years of data sampled at a time interval of one minute. Out-of-sample goodness of fit was obtained by applying the calibrated models on never seen before data. We reported as ∞ the scores of models which are numerically infinite, but due to clipping appear finite.

3. To measure how well the model explains price changes of portfolios of varying volatility, we choose $M = \Sigma^{-1}$. This score emphasizes the errors of cross-impact models in directions of small volatility and downplays errors in directions of large volatility.[1]

Table .1 summarises the scores for the different models on three different datasets. To account for varying daily volatility and liquidity, the second-order statistics Σ, Ω, R are reestimated for each trading day. Cross-impact models on a given trading day are computed using the daily estimators for Σ, Ω, R. Table .1 highlights the following result.

Result 2. *The `kyle` and `ml` models are robust across futures and stocks and outperform models which ignore cross-sectional effects.*

This empirical analysis, combined with the result that the `kyle` model is the only model which satisfies all axioms, suggests that it is an ideal candidate to understand static cross-impact from anonymous order flow across asset classes.

2 Chapter II: Static cross-impact on derivatives

To further assess if the `kyle` model is robust across asset classes, Chapter II models cross-impact on derivatives with the `kyle` model, in a slightly generalised setup of Chapter I. We show that, within this framework, the cross-impact matrix can be estimated using a low-dimensional formula which circumvents the estimation of very large covariances. We apply our methodology on SP500 options and VIX futures and find that it explains subtle fluctuations of the implied volatility surface. This reinforces the finding of Chapter I that the `kyle` cross-impact model is an ideal candidate for a unifying view of cross-impact across asset classes.

We begin with the setup for our results.

[1] Note that this measure strongly penalizes models violating fragmentation invariance: errors along modes of zero risk should *a-priori* be enhanced by an infinite amount.

2.1 Setup

We consider a universe comprising two classes of financial instruments, that we refer to as *factors* and *derivatives*. *Factors* are modeled by N stochastic processes and we denote their prices by $p_t = (p_t^1, \dots, p_t^N)$. Factors correspond to all stochastic factors on which derivative prices depend, such as the spot, the forward variance curve, the yield curve, etc. *Derivatives* are a set of M instruments, whose prices $P_t = (P_t^1, \dots, P_t^M)$ at maturity depend on the values of the factors.

As in Chapter I, we assume that impact is linear in the total signed flows of market orders and we denote by $q_t = (q_t^1, \dots, q_t^N)$ the stochastic process corresponding to the net traded order flows on factors and by $Q_t = (Q_t^1, \dots, Q_t^M)$ the stochastic process corresponding to the net traded order flows on derivatives. As we are interested in a simple, infinitesimal framework for the influence of trades on prices, we assume that order flow dynamics on each asset are continuous stochastic processes driven by Brownian motions, described in the next assumption.

Assumption 2. *The order flow follows the following stochastic dynamics*

$$\begin{aligned} dq_t &= \nu_q(p_t, t)dt + \mathscr{L}_{qq}(p_t, t)dZ_t^q + \mathscr{L}_{Qq}(p_t, t)dZ_t^Q \\ dQ_t &= \nu_Q(p_t, t)dt + \mathscr{L}_{Qq}(p_t, t)dZ_t^q + \mathscr{L}_{QQ}(p_t, t)dZ_t^Q, \end{aligned}$$

where Z^q, Z^Q are uncorrelated standard Brownian motions, $\nu_q \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ and $\nu_Q \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ encode order flow drift, $\mathscr{L}_{qq} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_N(\mathbb{R})$, $\mathscr{L}_{QQ} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_M(\mathbb{R})$, $\mathscr{L}_{Qq} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{M,N}(\mathbb{R})$, $\mathscr{L}_{qQ} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{N,M}(\mathbb{R})$ encode co-trading among assets.

Assumption 2 allows for co-trading of factor and derivatives, which is a typical feature expected in derivative markets. However, the continuous modeling framework for order flows is somewhat unrealistic. At the high-frequency scale, trades arrive punctually and trade arrivals are clustered so that a more realistic modeling is driven by point processes which respect this property. Such modeling requires lengthy mathematical treatment and is addressed in Chapter III.

We now move to the dynamics of factor prices in our setup.

Assumption 3. *We assume that factor prices are given by*

$$dp_t = \mu_p(p_t, t)dt + \mathscr{G}_p(p_t, t)dW_t + \Lambda_{pq}(p_t, t)dq_t + \Lambda_{pQ}(p_t, t)dQ_t \tag{4}$$

where W is a standard N-dimensional Brownian motion, $\mu_p \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ is the factor drift, $\mathscr{G}_p \colon \mathbb{R}^N \times \mathbb{R} \to \mathrm{GL}_N(\mathbb{R})$ is the factor diffusion matrix, and $\Lambda_{pq} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_N(\mathbb{R})$, $\Lambda_{pQ} \colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{N,M}(\mathbb{R})$ capture cross-impact.

The factor dynamics of Assumption 3 are quite general. They encompass traditional stochastic volatility models and allow for cross-impact between factors and between factors and derivatives.

We assume that factors are chosen so that derivatives are priced in a Markovian manner with respect to the factors, which leads us to the next assumption.

Assumption 4. *There exists a function $F\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$, twice differentiable with respect to the first argument and differentiable with respect to the second argument, such that $P_t = F(p_t, t)$.*

Note that we could relax Assumption 4 to include the influence of order flows without much modification of our results. Applying Ito's formula to Equation (4) and using Assumptions 2 to 4, we obtain the following corollary.

Corollary 1. *The derivative dynamics are given by*

$$dP_t = \mu_P(p_t, t)dt + \mathcal{G}_P(p_t, t)dW_t + \Lambda_{Pq}(p_t, t)dq_t + \Lambda_{PQ}(p_t, t)dQ_t\,, \tag{5}$$

where $\mu_P\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ is the derivative drift, $\mathcal{G}_P\colon \mathbb{R}^N \times \mathbb{R} \to \mathcal{M}_{M,N}(\mathbb{R})$ is the derivative diffusion matrix, $\Lambda_{Pq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathcal{M}_{M,N}(\mathbb{R})$ and $\Lambda_{PQ}\colon \mathbb{R}^N \times \mathbb{R} \to \mathcal{M}_M(\mathbb{R})$ encode cross-impact. In particular, we have the constraints $\Lambda_{Pq} = \Xi\Lambda_{pq}$, $\Lambda_{PQ} = \Xi\Lambda_{pQ}$, $\mathcal{G}_P = \Xi\mathcal{G}_p$, where $\Xi := (\frac{\partial p^i}{\partial p^j})_{i,j}$ is the M by N sensitivity matrix.

Corollary 1 does not make explicit the dependence of the derivative drift on other variables as it will not play an important role.

Cross-impact matrices can be compactly rearranged into a single matrix, Λ, which we refer to as the cross-impact matrix since it describes the cross-impact of the complete system

$$\Lambda(p_t, t) := \begin{pmatrix} \Lambda_{pq} & \Lambda_{pQ} \\ \Lambda_{Pq} & \Lambda_{PQ} \end{pmatrix}(p_t, t)\,. \tag{6}$$

The cross-impact matrix may depend, along with covariances, on prices of the factors and may evolve over time, which generalises the framework of Chapter I.

The static cross-impact model we choose involves two parameters, the price covariance matrix and the order flow covariance matrix. The factor-factor price covariance matrix $\Sigma_{pp}\colon \mathbb{R}^N \times \mathbb{R} \to \mathcal{S}_N^+(\mathbb{R})$ and factor-factor order flow covariance matrix $\Omega_{qq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathcal{S}_N^+(\mathbb{R})$ are defined as

$$\Sigma_{pp}(p_t, t)\mathrm{d}t := \mathrm{d}\langle p, p\rangle_t\,, \qquad \Omega_{qq}(p_t, t)\mathrm{d}t := \mathrm{d}\langle q, q\rangle_t$$

where $\langle X, Y\rangle$ is the quadratic covariation of the processes X, Y. Using similar notations for other covariance matrices (such as $\Omega_{qQ}(p_t, t)\mathrm{d}t := \mathrm{d}\langle q, Q\rangle_t$), introduced below, the covariance structure of prices and order flows for the whole system can be arranged compactly as

$$\Sigma(p_t, t) = \begin{pmatrix} \Sigma_{pp} & \Sigma_{pP} \\ \Sigma_{pP}^\top & \Sigma_{PP} \end{pmatrix}(p_t, t) \qquad \Omega(p_t, t) = \begin{pmatrix} \Omega_{qq} & \Omega_{qQ} \\ \Omega_{qQ}^\top & \Omega_{QQ} \end{pmatrix}(p_t, t)\,.$$

Motivated by the results of Chapter I, we choose to use the `kyle` model to capture cross-impact.

Assumption 5. *The cross-impact matrix Λ is of the form*

$$\Lambda := \sqrt{Y}(\mathscr{L}^{-1})^\top \sqrt{\mathscr{L}^\top \Sigma \mathscr{L}}\, \mathscr{L}^{-1},$$

where $\mathscr{L}$ is a matrix such that $\mathscr{L}\mathscr{L}^\top = \Omega$, Y is a constant such that $0 < Y < 1$, and we have omitted the dependence on (p_t, t) for compactness.

The choice of the `kyle` model has important consequences, which we detail in the next section.

2.2 Properties of the chosen cross-impact model

A main result implied by our choice of cross-impact model is a derivation of Λ which is tractable for applications. We first show that the influence of order flows can be absorbed in Brownian motions.

Result 3. *Using the notation of Assumptions 3 and 5 and Corollary 1, we have*

$$dp_t = \tilde{\mu}_p(p_t, t)dt + \frac{1}{\sqrt{1-Y}}\mathscr{G}_p(p_t, t)dB_t \tag{7}$$

$$dP_t = \tilde{\mu}_P(p_t, t)dt + \frac{1}{\sqrt{1-Y}}\mathscr{G}_P(p_t, t)dB_t, \tag{8}$$

where $\tilde{\mu}_p \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$, $\tilde{\mu}_P \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ and B is a standard N dimensional Brownian motion.

Result 3 underscores a convenient property of our framework: the sensitivities of derivative prices with respect to factors are independent of the order flow dynamics. In particular, the Greeks can be computed using traditional derivative pricing methods since derivative prices satisfy the classic stochastic differential system of Result 3. The next result expresses the large $N + M \times N + M$ cross-impact matrix Λ as a function of the much smaller $N \times N$ cross-impact matrix Λ_{pq} and the derivative sensitivities Ξ.

Result 4. *We have*

$$\Lambda(p_t, t) = \begin{pmatrix} \Lambda_{pq} & \Lambda_{pq}\Xi^\top \\ \Xi\Lambda_{pq} & \Xi\Lambda_{pq}\Xi^\top \end{pmatrix}(p_t, t), \tag{9}$$

where we recall that $\Xi := (\frac{\partial P^i}{\partial p^j})_{i,j}$ is the M by N sensitivity matrix.

Note that Result 4 shows that the chosen cross-impact matrix satisfies the constraints of Corollary 1. Thanks to Result 3, the sensitivity matrix can be computed as the usual Greeks of our derivative pricing model. Thus, given a formula for Λ_{pq}, the cross-impact matrix Λ can be computed. The next result shows that the cross-impact matrix Λ_{pq} can be expressed as a function of the factor price covariance matrix Σ_{pp} and a modified covariance matrix of order flows.

Result 5. *We have*

$$\Lambda_{pq} = \sqrt{Y}(\mathscr{L}_\Xi^{-1})^\top \sqrt{\mathscr{L}_\Xi^\top \Sigma_{pp} \mathscr{L}_\Xi}\, \mathscr{L}_\Xi^{-1}, \tag{10}$$

where we have omitted the dependence on (p_t, t), $\Omega_\Xi := \Omega_{qq} + \Xi^\top \Omega_{QQ}\Xi + \Xi^\top \Omega_{Qq} + \Omega_{qQ}\Xi$, and $\mathscr{L}_\Xi$ is a matrix such that $\mathscr{L}_\Xi \mathscr{L}_\Xi^\top = \Omega_\Xi$.

Combined, Results 4 and 5 give a formula for the cross-impact matrix Λ as a function of the measurable quantities Σ_{pp} and Ω_{Ξ}. Furthermore, by Result 3, Ξ can be computed using usual derivative pricing methods. Overall, we have thus derived a scheme for estimating the cross-impact matrix Λ on derivatives.

A relevant insight of Equation (10) for applications is that even if factors are not traded, as long as derivatives with sensitivities to these factors are traded, i.e. $\Xi^\top \Omega_{QQ} \Xi$ is positive definite, then the inverses appearing in Equation (10) are well-defined. This is not obvious from the form of the Kyle cross-impact matrix in Assumption 5. This property is important for applications where most factors correspond to non-tradeable instruments, such as volatility factors.

2.3 Application to SP500 options and VIX futures

We illustrate our approach on a set of instruments which includes includes the front-month maturity of SP500 futures, the two front-month VIX futures and a large set of SP500 options. We bin price and order flow data on a 5 minute timescale using the same approach as in Chapter I. We begin by describing the modeling choices for the factors of this system.

2.3.1 A simple factor model for the implied volatility surface

To analyse this set of instruments, we take as factors the price of the SP500 future closest to expiry, which we refer as the spot factor, and a set of Q volatility factors $\varsigma_t = (\varsigma_t^1, \dots, \varsigma_t^Q)$. We assume that the implied volatility surface is completely described by these factors, i.e. that the implied volatility $\hat{\sigma}_t^i$ of Derivative i is such that

$$\hat{\sigma}_t^i = F^i(\varsigma_t)\,,$$

where $F^i : \mathbb{R}^Q \to \mathbb{R}$ is some function of these factors. We consider a linear approximation of the implied volatility surface with volatility factors, so that $F^i(\varsigma) = \sum_{q=1}^Q \beta^{iq} \varsigma^q$, where (β^{iq}) are parameters to be calibrated. We choose $Q = 3$ and fit these parameters using a principal component analysis of the implied volatility surface, as in [CFD02]. More details are given in Chapter II. The resulting factors are described in Figure .1.

The first volatility factor is a classic implied volatility *level* factor and we make the rough approximation that the price changes of VIX futures solely depend on this factor. The second volatility factor corresponds to the skew of the implied volatility surface, referred to as the *skew* factor hereafter. The third volatility factor explains the term structure of the implied volatility, hence the name the *term* factor in the following.

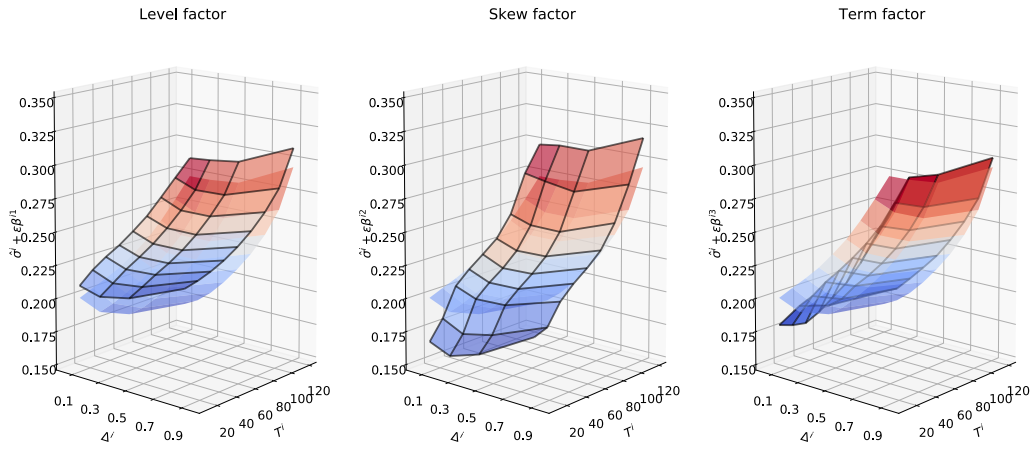


Figure .1: **Effect of the different volatility factors on the implied volatility surface.** Starting from a historical implied volatility surface ($\hat{\sigma}^i$), we show the modified implied volatility surface ($\hat{\sigma}^i + \varepsilon\beta^{iq}$) after adding a contribution of size ε from the volatility factor q. The original (non-modified) implied volatility surface is shown in light opacity for reference.

2.3.2 Quality of fit

To assess quality of fit, we use the same methodology as in Chapter I. In this case, we focus on the quality of the predictions of each model on the spot and volatility factors. We highlight three different types of models, whose expressions are given in Chapter II but omitted here for conciseness:

1. The `black-scholes` model which accounts for impact only in the spot direction. It serves as a benchmark model if we believe option prices have the same, constant, implied volatility.

2. The `direct-2d` model uses order flow on the spot and level directions to compute impact, but ignores cross-sectional effects. The `direct-4d` model does the same with all four directions.

3. Finally, the `kyle-2d` model only accounts for spot and level modes while the `kyle-4d` model accounts for all directions.

Model	Scores			
	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{spot}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{level}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{skew}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{term}})$
`black-scholes`	0.18 ± 0.01	-0.00 ± 0.02	-0.00 ± 0.01	-0.00 ± 0.02
`direct-2d`	0.18 ± 0.01	-0.03 ± 0.02	-0.01 ± 0.01	0.00 ± 0.02
`direct-4d`	0.18 ± 0.01	-0.03 ± 0.02	-0.14 ± 0.02	-0.26 ± 0.02
`kyle-2d`	0.20 ± 0.01	0.12 ± 0.01	-0.01 ± 0.01	0.01 ± 0.02
`kyle-4d`	0.20 ± 0.01	0.14 ± 0.01	-0.12 ± 0.02	0.04 ± 0.01

Table .2: **Scores of different cross-impact models**.
All scores were computed in-sample using the same data used for the calibration of the cross-impact models.

Table .2 suggests that, as expected from the liquidity structure, most price changes on the spot are explained by trades made on the spot or the delta-weighted contribution of trades made on options. Remarkably, only models which account for cross-impact are able to explain a large fraction of the level direction. This is the topic of the next result.

Result 6. *Fluctuations of the level factor are mostly explained through trades on the spot or delta-weighted contributions of trades on options.*

Furthermore, we observe a slight score improvement for models which account for trades in the level direction. This suggests that there is a small explanatory power in trades in the level direction.

Result 7. *A small fraction of fluctuations of the spot can be explained through trades on VIX futures or vega-weighted contributions of trades on options.*

Finally, scores improve when we consider a more detailed description of the dynamics of the implied volatility surface. This suggests that the `kyle` cross-impact model becomes more accurate as our description of option dynamics become more precise, and there may be room for improvement beyond the simple volatility factor modelling that we have chosen.

Result 8. *The `kyle` model becomes more accurate when the description of the implied volatility surface becomes more precise.*

Overall, our approach reinforces the result of Chapter I that the `kyle` model is an ideal candidate to understand static cross-impact from anonymous order flow across asset classes, including on complex assets such as derivatives.

3 Chapter III: Dynamic cross-impact models from anonymous order flow

Chapter III examines dynamic models for cross-impact from anonymous order flow. These models capture the lasting influence of past trades on current prices. We setup a toy market model and look for dynamic cross-impact models, dubbed *cross-impact kernels*, which yield

realistic price dynamics. We derive two such cross-impact kernels and calibrate them on empirical data. Interestingly, the boundary values of these cross-impact kernels are related to the `kyle` model of Chapters I and II.

3.1 Setup

We consider a market made of d different assets, of prices $p := (p^1, \cdots, p^d)$, and we denote the cumulative anonymous order flow at the ask (resp. bid) by q^a (resp. q^b) and the net traded volume by $q := q^a - q^b$.

Although ignored for simplicity in Chapters I and II, anonymous order flow is highly persistent [BBDG18]. To account for this, we classically model order flow dynamics with a multivariate Hawkes process [Haw71a]. More precisely we assume that the number of orders at the ask N^a and the bid N^b is a counting process $N = (N^a, N^b)$ of intensity λ which satisfies

$$\lambda_t = \mu + \int_0^t \Phi(t-s)\mathrm{d}N_s\,,$$

where μ is a vector of positive entries and Φ a positive matrix-valued function with integrable entries such that the spectral radius of $\|\Phi\|_1$ is below unity. Hawkes processes are classical models in finance to capture self-excitation and cross-excitation across time and instruments [BMM15] and allow for rich order flow dynamics. To simplify results, we assume that each order on Asset i is of constant size v_i so that $q_i^a = v_i N^a, q_i^b = v_i N^b$.

Our goal is to generalise the static cross-impact models of Chapters I and II to account for the influence of past order flows. To do so, *kernel* or *propagator* models are popular in the single-asset case [BBDG18] and have recently been examined in the multivariate case [AKS16, BMEB17, SL19]. Thus, we focus on *cross-impact kernels*, as described in the next assumption.

Assumption 6 (Price dynamics). *There exists some function $K\colon t \in \mathbb{R}_+ \mapsto K(t) \in \mathcal{M}_d(\mathbb{R})$, called cross-impact kernel, such that the price process p satisfies, for all $t \in \mathbb{R}_+$*

$$p_t = p_0 + \int_0^t K(t-s)\,dq_s\,, \tag{11}$$

and $K(t) \underset{t\to\infty}{\to} \Lambda$, where Λ is an invertible $d \times d$ matrix called the permanent cross-impact matrix.

The matrix Λ is called the permanent cross-impact matrix since Λ_{ij} quantifies how much the price of Asset i is moved by the net order flow on Asset j after a long period.

Our goal is to find suitable choices of cross-impact kernels. To do so, we focus on those which yield realistic price dynamics. We examine two notions of efficiency: martingale prices and absence of statistical arbitrage, that is of round-trip trading strategies which generate a profit on average.

We begin by introducing the class of cross-impact kernels which gives martingale prices. Since it also includes trivial examples such as $K = 0$, requiring martingale prices only does not necessarily lead to relevant kernels. To focus on pertinent kernels, we introduce *martingale-admissible* kernels, which anticipate the impact contribution of the order flow and lead to martingale prices. Such kernels generate non-trivial price dynamics since they incorporate the impact contribution of trades in prices. In the univariate case $d = 1$, Theorem 2.1 of [Jai15] shows that when prices are martingales and trades impact prices, we have

$$p_t - p_0 = \kappa \lim_{s\to\infty} \mathbb{E}[q_s^a - q_s^b \mid \mathscr{F}_t]\,,$$

where $\kappa > 0$ is the permanent market impact contribution. This motivates the following definition for *martingale-admissible* kernels.

Definition 1 (Martingale-admissible kernels). *A cross-impact kernel K is said to be martingale-admissible if*

$$p_t - p_0 = \int_0^t K(t-s)\,dq_s = \Lambda \lim_{s\to\infty} \mathbb{E}[q_s \mid \mathscr{F}_t]\,. \tag{12}$$

Martingale-admissible kernels set prices according to the linear permanent market impact induced by the predictable component of all future trades. We show in Chapter III that martingale-admissible kernels indeed lead to martingale prices. Such kernels prevent agents who successfully forecast order flow to trade profitably. However, they do not forbid statistical arbitrages entirely. We call *no-statistical-arbitrage-admissible* (or *nsa-admissible* for short) cross-impact kernels that prevent statistical arbitrage.

Definition 2 (No-statistical-arbitrage-admissible kernels). *A cross-impact kernel K is said to be no-statistical-arbitrage-admissible, or nsa-admissible for short, if there are no possible statistical arbitrages, i.e. no round-trip trading strategies with average negative cost.*

We now describe the main results of Chapter III in the next section.

3.2 Characterisation of martingale-admissible and nsa-admissible kernels

We show in Chapter III that martingale-admissible kernels must satisfy constraining conditions but that their boundary values are not fixed. On the other hand, nsa-admissible kernels must satisfy a soft constraint but their boundary values are fixed. In particular, the values at zero of nsa-admissible kernels can be expressed compactly, which is the topic of the next result.

Result 9. *For any nsa-admissible kernel K, we have*

$$K(0) = \frac{1}{\sqrt{2}} (\mathscr{L}_0^{-1})^\top \sqrt{\mathscr{L}_0^\top \Sigma \mathscr{L}_0}\, \mathscr{L}_0^{-1}\,, \tag{13}$$

where

1. *the matrix $\mathscr{L}_0$ is any matrix such that $\mathscr{L}_0 \mathscr{L}_0^\top = \operatorname{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$,*

2. *the matrix* $\Sigma := \lim_{t\to\infty} \mathbb{E}[d\langle P, P\rangle_t]$ *is loosely speaking the stationary instantaneous covariance matrix of returns. The existence of this limit is shown in Chapter III.*

We further show that the value at infinity of cross-impact kernels which are both martingale-admissible and nsa-admissible is constrained and has an explicit expression given in the next result.

Result 10. *For any nsa-admissible, martingale-admissible kernel K, we have*

$$\lim_{t\to\infty} K(t) = \Lambda = \frac{1}{\sqrt{2}} (\mathscr{L}_\infty^{-1})^\top \sqrt{\mathscr{L}_\infty^\top \Sigma \mathscr{L}_\infty} \mathscr{L}_\infty^{-1}, \tag{14}$$

where the matrix $\mathscr{L}_\infty$ is any matrix such that $\mathscr{L}_\infty \mathscr{L}_\infty^\top = \int_0^\infty \Omega(ds)$, where Ω is the reduced covariance measure of the stationary version of the point process q (see Chapter III for more details). We can interpret $\int_0^\infty \Omega(ds)$ as the stationary total autocovariance matrix of order flows.

The matrices $K(0)$ and $\lim_{t\to\infty} K(t)$ have an interpretation in terms of the `kyle` model discussed in Chapters I and II. The matrix $K(0)$ corresponds to the `kyle` model where the price-covariance matrix is Σ and the order-flow covariance matrix is $\mathrm{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$, which is actually the reduced covariance measure evaluated at the singleton zero: $\Omega(\{0\}) = \mathrm{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$. On the other hand, $\lim_{t\to\infty} K(t)$ corresponds to the `kyle` model where the price-covariance matrix is Σ and the order-flow covariance matrix is $\int_0^\infty \Omega(\mathrm{d}s)$. Thus, both $K(0)$ and $\lim_{t\to\infty} K(t)$ can be interpreted as `kyle` models with the same price-covariance matrix but with differing order-flow covariance matrices. We can interpret the order-flow covariance matrix associated to $K(0)$ as the immediate available liquidity and the order-flow covariance matrix associated to $\lim_{t\to\infty} K(t)$ as the total market liquidity, combining immediate available liquidity with liquidity due to the order-flow correlation structure.

Results 9 and 10 show that any martingale and nsa-admissible kernel has fixed boundary values. The next result shows that there is at most one martingale and nsa-admissible kernel.

Result 11. *There exists a unique cross-impact kernel K that is martingale-admissible and which satisfies the necessary conditions for arbitrage-admissibility outlined in Equations* (13) *and* (14)*. Its expression is given in Chapter III. Therefore, there exists at most one martingale and nsa-admissible kernel.*

Given Result 11, there exists only one martingale-admissible kernel which satisfies the boundary conditions of nsa-admissible kernels and may additionally be nsa-admissible. This kernel is of particular interest, and we dub it the K^1 kernel.

As we have no guarantee that K^1 is nsa-admissible, this poses issues in certain applications. For example, in portfolio optimization, a trading cost model which allows for arbitrages induces spurious round-trip strategies, as shown in [AKS16]. Thus, we introduce a regularisation method to find the closest nsa-admissible kernel to K^1, which we write K^2. The

regularisation method is motivated and explained in Chapter III.

We now turn to applications of our framework on market data.

3.3 Estimation of cross-impact kernels

A key result of Chapter III is a numerical method for calibration of K^1 and K^2. We leave the details in Chapter III and show the calibration results on two maturities of SP500 futures in Figure .2. The K^1 kernel shows a power-law decay consistent with previous studies [SL19, BMEB17]. Slightly more surprising however is the shape of the nsa-admissible kernel K^2. We believe this is a numerical effect due to the regularisation procedure to compute K^2, which involves Fourier transforms.

Despite the different shapes of the two kernels, their predictions on empirical data are markedly similar, as shown in Figure .3.

4 Chapter IV: Measuring cross-impact from labeled order flow

Chapters I to III focused on calibrating cross-impact models from anonymous order flow. This part examines how indidivual agents push prices when trading, by measuring cross-impact from labeled order flow. To do so, we use two databases of proprietary orders from asset managers. We propose measures of cross-impact in the bases of cross-impact models studied in Chapter I. These yield generalisations of the square-root law of market impact which fit data more precisely by accounting for price and order flow correlations.

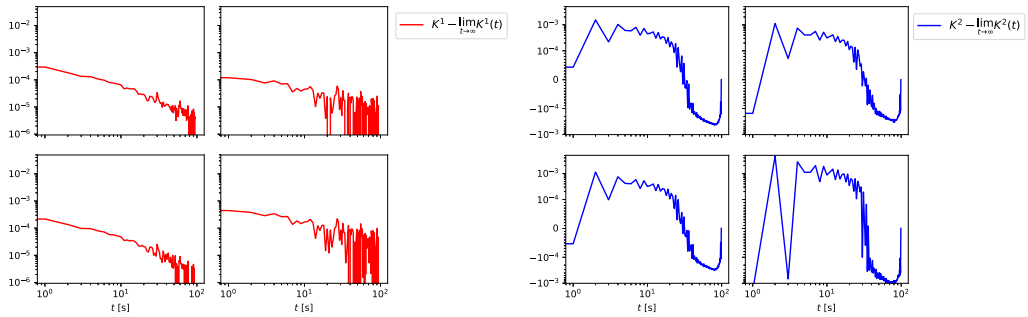


Figure .2: **Values of the K^1 and K^2 kernels.**
We report the values of the transient part of the martingale-admissible $K^1 - \lim_{t\to\infty} K^1(t) = K^1 - \Lambda$ (left, red) and of the transient part of the nsa-admissible kernel $K^2 - \lim_{t\to\infty} K^2(t) = K^2 - \Lambda$ (right, blue). Each subplot shows the matrix elements of the kernels. For instance, the top left plot of the left inset shows $K^1_{11} - \Lambda_{11}$ and the top right shows $K^1_{12} - \Lambda_{12}$. The permanent cross-impact matrix Λ has been removed to highlight the power-law decay of the cross-impact kernels toward their limit.

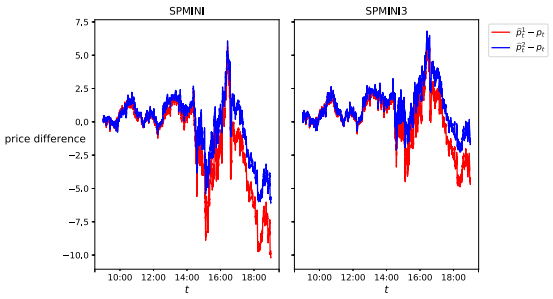


Figure .3: **Example of predicted prices from K^1 and K^2.**
Difference between realized price and the predicted impact-induced price from Equation (2) with the martingale-admissible cross-impact kernel K^1, given by $\widehat{p}^1_t = p_0 + \sum_{s<t} K^1(t-s)q_s$, (red) and with the nsa-admissible cross-impact kernel K^2, given by $\widehat{p}^2_t = p_0 + \sum_{s<t} K^2(t-s)q_s$, (blue), where (q_t) are measured signed volumes of market orders. Price difference is reported in dollars per lot.

4.1 Dealing with the curse of dimensionality

We begin with the definition of cross-impact from labeled order flow.

Definition 3. *An asset manager decides to buy or sell a portfolio $Q = (Q_1, \cdots, Q_d)$ of d different assets, each made up of Q_i dollar-risks of Asset i where* dollars of risk := number of shares × volatility × daily opening price. *He, or his broker, splits his meta-order in smaller orders and executes them on the market. The cross-impact of a meta-order of the portfolio Q is*

$$\mathscr{C}\mathscr{I}(Q) := \mathbb{E}[s \mid Q], \tag{15}$$

where $s = (s_1, \cdots, s_d)$ are the normalized daily returns of the assets, $s_i := \frac{S^i_c - S^i_o}{\sigma_i S^i_o}$ being the normalized daily asset return of Asset i, S^i_c is the price of Asset i at the end of the trading day, S_o is the price of Asset i at the beginning of the trading day, $\sigma_i := (S^i_{\text{high}} - S^i_{\text{low}})/S^i_o$ is a proxy for the daily return volatility of Asset i.

Directly measuring cross-impact as defined in Equation (15) is arduous since we need to measure the average of a random variable conditioned with d random variables, where d is large. To tackle this problem, the next section introduces the concept of diagonalisation of cross-impact.

4.2 Diagonalisation of cross-impact and proposed cross-impact law

Before introducing the concept of diagonalisation of cross-impact, we note that we can apply a change of basis to simplify the problem. Let $\Pi := (\pi_1, \cdots, \pi_d)$ be an orthonormal basis of $\mathbb{R}^d$. Then, since Π is a basis of $\mathbb{R}^d$, there is a one to one correspondence between the distribution of the family of N random variables $\mathbb{E}[s_i \mid Q_1, \cdots, Q_d]$ and the distribution of the family of N random variables

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}],$$

where $s_{\pi_i} := \pi_i^\top s$ is the portfolio return and $Q_{\pi_i} := \pi_i^\top Q$ is the traded volume of dollar-risk of the portfolio. If, for this portfolio, we had

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}] = \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}],$$

then the price impact on portfolio π_i would only depend on what is traded on portfolio π_i. This leads to the following definition.

Definition 4 (Diagonal basis for cross-impact). *An orthonormal basis $\Pi = (\pi_1, \cdots, \pi_d)$ of $\mathbb{R}^d$ is said to diagonalise cross-impact if, for all $1 \leq i \leq d$, we have*

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}] = \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}], \tag{16}$$

where $s_{\pi_i} := \pi_i^\top s$ is the daily normalized return of the portfolio and $Q_{\pi_i} := \pi_i^\top Q$ is the traded volume of dollar-risk of the portfolio.

Naturally, the challenge is then to find a basis such that Equation (16) holds. Inspired by the previous definition, for a given orthonormal basis Π, we associate the following prediction for cross-impact:

$$\widehat{\mathscr{C}\mathscr{I}}_\Pi(Q) := \sum_{k=1}^{d} \mathscr{I}_{\pi_k}(Q_{\pi_k})\pi_k, \tag{17}$$

where $\mathscr{I}_{\pi_k}(Q_{\pi_k}) := \mathbb{E}[s_{\pi_k} \mid Q_{\pi_k}]$ are functions to be estimated. This prediction is motivated by the fact that when a basis respects Equation (16), cross-impact can be reduced to Equation (17). We have restricted our study to orthonormal bases: the portfolios of these bases have unit risk and no overlap, *i.e.* $\pi_i^\top \pi_j = 0$ for two different portfolios π_i and π_j. Their components can be interpreted as positions in each asset expressed in units of risk.

Depending on the choice of basis Π, Equation (17) may yield sensible predictions or not. The next sections examine relevant choices of bases.

4.3 Bases of interest

4.3.1 Canonical basis

The first basis examined is the canonical basis which ignores cross-sectional effects and we thus dub the `direct` basis:

$$\Pi_{\texttt{direct}} := (e_1, \cdots, e_d),$$

where the e_i are the canonical vectors of $\mathbb{R}^d$. This basis uses the predictions for market impact on each asset to predict cross-impact. The functions $\mathscr{I}_{e_i}$ correspond to the usual market impact and are well described by the square-root law [BBDG18, ATHL05, Tor97, ZTFL15, BBLB18, BBLB19, BR13, BILL15].

4.3.2 Random basis

The second basis of interest is the `random` basis defined as

$$\Pi_{\texttt{random}} := (o_1, \cdots, o_d)\,,$$

where the basis $(o_1, \cdots, o_d)$ has been randomly sampled in the orthogonal group. This basis serves as a zero-intelligence basis which accounts for cross-sectional effects. Contrary to the `direct` basis where it is known that the square-root law applies, we have no prior knowledge of the functions $\mathscr{I}_{o_i}$, so that we will need to estimate it using meta-order data.

4.3.3 Eigenliquidity model basis

The third selected basis accounts for return correlations. It is taken using the eigenvectors of the normalized return correlation matrix $\varrho := \mathrm{Corr}(s)$:

$$\Pi_{\texttt{el}} := \mathrm{eigenvectors}(\varrho)\,.$$

This basis is dubbed eigenliquidity model (`el` for short) basis since it is inspired by the eigenliquidity cross-impact model [TMB20, BMEB17, MTB14] which has the same eigenvectors as ϱ. Given the goodness-of-fit of the linear eigenliquidity model on anonymous data, the cross-impact predictions associated to this basis may fit empirical data well. As in the `random` basis, we will have to estimate empirically the functions $\mathscr{I}_\pi$ where π is in the basis $\Pi_{\texttt{el}}$.

4.3.4 Kyle model basis

The final basis chosen uses both return and order flow structure, encoded in the normalized return correlation matrix ϱ and the co-trading effects encoded in the traded risk covariance matrix $\Omega := \mathrm{Cov}(q)$, where $q := (q_1, \cdots, q_d)$ is the daily signed traded risk of each asset. The `kyle` basis is defined as

$$\Pi_{\texttt{kyle}} := \mathrm{eigenvectors}((\mathscr{L}^{-1})^\top \sqrt{\mathscr{L}^\top \varrho \mathscr{L}}\, \mathscr{L}^{-1})\,,$$

where $\mathscr{L}$ is such that $\mathscr{L}\mathscr{L}^\top = \Omega$. We refer to this basis as the `kyle` basis since comes from the multivariate Kyle cross-impact model studied in Chapters I and II. Once again, the functions $\mathscr{I}_\pi$ where π is in the basis $\Pi_{\texttt{kyle}}$ have to be estimated.

4.4 Meta-order datasets

This study uses two databases of proprietary meta-orders to analyse cross-impact. We present the characteristics of each dataset below.

4.4.1 ANcerno dataset

The first database is made available by ANcerno, formerly Abel Noser Corporation, which is a leading consulting firm that works with institutional investors to monitor their equity trading

costs. This database contains orders executed by different institutional investors in the U.S. equity market. Our sample of this database contains stocks and orders from 2000 to 2014.

In the following we will identify a meta-order as a series of consecutive orders performed by a single investor, through a single broker within a single day, on a given stock and with a fixed direction (buy/sell). To avoid an elaborate analysis of when precisely each meta-order starts and ends, how they overlap and which reference price to take, we consider impact at the daily scale. Doing so, we aggregate the meta-orders of different asset managers on a given stock. For a given trading day and asset, the signed daily risk is the sum of the signed risk of all meta-orders on that day. We assume that the daily meta-order starts when the market opens and finishes when the market closes.

The final dataset contains 2708 stocks, distributed among different sectors. With our daily aggregation methodology, we are left with around 2.8 million meta-orders. To keep the number of distinct stocks tractable, we select 1000 random stocks among our dataset. We are left with 1.6 million meta-orders.

4.4.2 CFM Futures dataset

The second dataset employed contains data on the proprietary meta-orders executed by the asset manager Capital Fund Management (CFM) on the futures market. After cleaning, the dataset covers around 250 different futures, with approximately 200,000 meta-orders from 2012 to 2019.

Contrary to the ANcerno dataset, meta-orders of this dataset are only those executed by CFM. Thus, a meta-order is identified with a future code, the signed number of exchanged dollars of risk (counted positively for buy orders and negatively for sell orders), executed during a physical time interval $[t_s, t_e]$ with t_s the start time and t_e the end time of the execution. To apply a consistent methodology with the ANcerno dataset, we aggregate orders at the daily scale.

4.5 Diagonalisation in bases of linear cross-impact models

This section examines how close different bases are to satisfying the diagonalisation condition of Equation (16). To do so, for a given orthonormal basis Π, we introduce the residual differences

$$Z_{ij}^{\Pi} := \left(\mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}, Q_{\pi_j}] - \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}]\right)^2, \tag{18}$$

which measure the error of the approximation. A global metric for our diagonalisation approximation error in the basis Π is then

$$Z^{\Pi} = \frac{1}{d} \sum_{\substack{1 \le i,j \le d \\ i \neq j}} Z_{ij}^{\Pi}, \tag{19}$$

which measures the average error made with the approximation. Table .3 reports the separation metric for the chosen bases and underscores the following result.

Result 12. *The bases* `el` *and* `kyle` *bases are much closer to safisfying the diagonalisation condition than the* `direct` *and* `random` *bases.*

This result is interesting because the `random` basis is a benchmark which should be close to satisfying the diagonalisation condition since both sides of Equation (18) should be close to zero.

Dataset	Basis			
	`direct`	`random`	`el`	`kyle`
ANcerno stocks	11.0 ± 0.9	0.61 ± 0.01	0.50 ± 0.05	0.54 ± 0.05
CFM futures	2.02 ± 0.01	0.977 ± 0.003	0.55 ± 0.02	0.68 ± 0.02

Table .3: **Separation metric Z^{Π} in the bases of different impact models**.
The table reports the separation metric Z^{Π} on both datasets. For a given pair of portfolios π_i, π_j from a given basis, Z^{Π}_{ij} is computed by estimating both sides of Equation (18) using empirical averages. Then, due to the large number of portfolio pairs to examine to compute Equation (19), Z^{Π} is estimated by averaging Z^{Π}_{ij} on a smaller sample of portfolio pairs and the error bars correspond to the 95% confidence interval for our subsampled estimate of Z^{Π}.

4.6 Estimation of the functions $\mathscr{I}_\pi$

The previous section showed that the `random`, `el` and `kyle` bases were much closer to satisfying the diagonalisation condition than the `direct` basis. For a given basis $\Pi = (\pi_1, \cdots, \pi_d)$, the associated predicted cross-impact of Equation (17) requires estimating the functions $\mathscr{I}_{\pi_k}(Q_{\pi_k}) := \mathbb{E}[s_{\pi_k} \mid Q_{\pi_k}]$. These functions are already known for the `direct` basis, since they correspond to market impact and are described by the square-root law. However, they need to be estimated for the `random`, `el` and `kyle` bases. Figure .4 reports the estimation of the functions for different bases and shows the following result.

Result 13. *For each portfolio π in the* `random`*,* `el` *and* `kyle` *bases, the function $\mathscr{I}_\pi$ is well-described by a square-root law like fit:*

$$\mathbb{E}[s_\pi \mid Q_\pi] \approx Y \sigma_\pi \left(\frac{Q_\pi}{V_\pi} \right)^{\bullet \delta}, \tag{20}$$

where $\delta \approx 0.4 - 0.6$, Y is a dimensionless prefactor (where $Y \approx 0.3$ on stocks and $Y \approx 0.5$ on futures), $V_\pi^2 := \pi^\top \Omega \pi$ is a proxy for the liquidity in the direction of the portfolio, $\sigma_\pi^2 := \pi^\top \varrho \pi$ is a proxy for the volatility of the portfolio.

A key difference with the square-root law is the scale of the dimensionless prefactor Y. Although it is independent of the portfolio, it changes depending on the dataset and, in particular, the number of assets d.

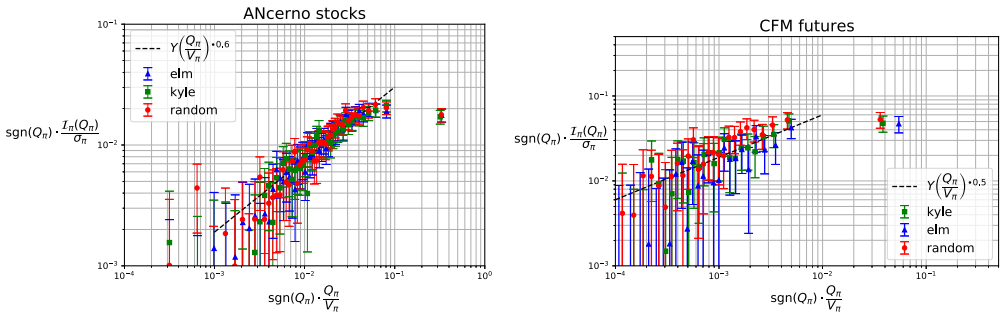


Figure .4: **Market impact in different bases on both datasets.**
Left: ANcerno stocks dataset. **Right**: CFM futures dataset. Each point for the different bases `el` (blue triangles), `kyle` (green squares) and `random` (red circles) is computed by averaging normalized returns on binned traded risk. Error bars shown are computed using 95% confidence intervals. The ANcerno dataset has significantly more data, which allows us to obtain more precision. On both datasets, for participation ratios $\eta_\pi := \frac{Q_\pi}{V_\pi}$ in the range $10^{-3} \lesssim \eta_\pi \lesssim 10^{-1}$, the square-root law fit is a good approximation.

4.7 Goodness-of-fit of the proposed cross-impact law

Using the results of the previous section we can plug the functional form of Equation (20) in the prediction of cross-impact of Equation (17) to associate with a given basis Π among the `direct`, `random`, `el` and `kyle` bases the following prediction for cross-impact:

$$\widehat{\mathscr{CI}}_\Pi(Q) \approx Y \sum_{k=1}^{d} \pi_k \sigma_{\pi_k} \left(\frac{Q_{\pi_k}}{V_{\pi_k}}\right)^{\bullet\delta} = Y \sum_{k=1}^{d} \pi_k \sqrt{\pi_k^\top \varrho \pi_k} \left(\frac{\pi_k^\top Q}{\sqrt{\pi_k^\top \Omega \pi_k}}\right)^{\bullet\delta}. \tag{21}$$

Equation (21) accounts for all portfolios, many of which may have small volatility and provide negligible contributions. To assess the quality of fit of Equation (21) as a function of the number of portfolios considered, we consider the truncated predictions $\widehat{\mathscr{CI}}_{\Pi,n}(Q)$ defined as

$$\widehat{\mathscr{CI}}_{\Pi,n}(Q) := Y \sum_{k=1}^{n} \pi_k \sigma_{\pi_k} \left(\frac{Q_{\pi_k}}{V_{\pi_k}}\right)^{\bullet\delta} = Y \sum_{k=1}^{n} \pi_k \sqrt{\pi_k^\top \varrho \pi_k} \left(\frac{\pi_k^\top Q}{\sqrt{\pi_k^\top \Omega \pi_k}}\right)^{\bullet\delta}, \tag{22}$$

where the portfolios π_k are sorted in decreasing order of volatility $\pi_k^\top \varrho \pi_k$. To evaluate the goodness-of-fit of the predictions obtained with Equation (22), we compute the r^2 between realized returns and the predictions given by $\widehat{\mathscr{CI}}_{\Pi,n}(Q)$ as

$$r^2 = 1 - \frac{1}{d} \sum_{i=1}^{d} \frac{\mathrm{Var}(s_i - e_i^\top \widehat{\mathscr{CI}}_{\Pi,n}(Q))}{\mathrm{Var}(s_i)},$$

where $\mathrm{Var}(X)$ is the variance of the random variable X. To favour the `direct` basis, the i-th variance in the sum is estimated on empirical data only using days where a meta-order

on Asset i was executed. Figure IV.3 reports the estimated r^2 as a function of the fraction of explained variance $\frac{\sum_{i=1}^{n} \pi_i^\top \varrho \pi_i}{\sum_{i=1}^{N} \pi_i^\top \varrho \pi_i}$ associated to the chosen basis. In particular, it shows the following result.

Result 14. *The proposed cross-impact law associated to the* `el` *and* `kyle` *bases has better empirical fit on both datasets than the single-asset square-root law.*

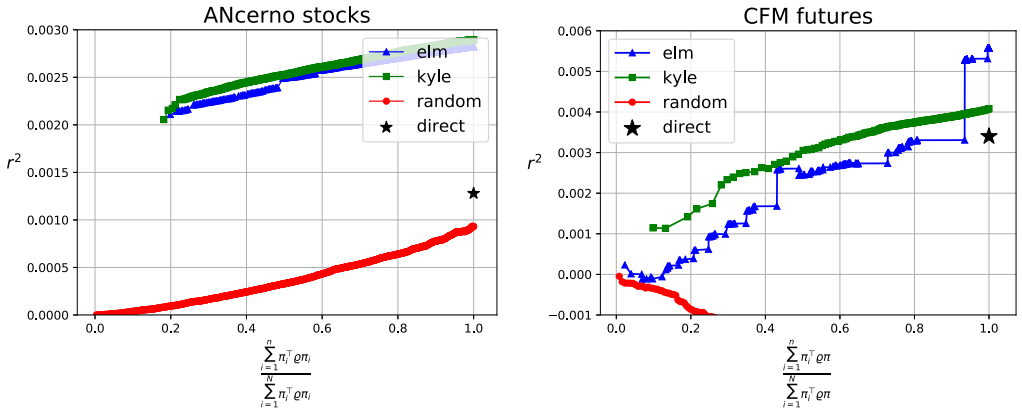


Figure .5: **Fit of different impact models on both datasets.**
Left: ANcerno stocks dataset. **Right**: CFM futures dataset. The r^2-squared measure of fit is computed for the `el` (blue triangles), `kyle` (green squares), `random` (red circles) and `direct` (black star) bases as a function of the cumulative explained variance. As expected, the random model shows poor performance on both datasets. Strikingly, both cross-impact bases outperform the `direct` basis. The difference is more pronounced on stocks than on futures. The quality of fit improves for both models as the explained variance grows.

Our analysis on labeled meta-orders proposed a cross-impact law which generalises the square-root law while accounting for return and order flow correlations. These results confirm the importance of the linear cross-impact models which have been used throughout Chapters I to III.

5 Chapter V: Microfounding multivariate price dynamics without order flows

Chapters I to IV examined the links between trades and prices to understand the price formation process. Chapter V focuses on finding micro-founded multivariate price dynamics. We introduce a tick-by-tick model for prices at the microscopic level and study macroscopic limits of this price process.

5.1 A microscopic model for prices

As in previous parts, we model the prices of a universe of d assets. At the high-frequency scale, the price of each asset is a piecewise constant process with upward and downward jumps denoted by the counting process N^{i+}, N^{i-}. The global counting process is a $2d$-dimensional process $N = (N^{1+}, N^{1-}, \dots, N^{d+}, N^{d-})$.

To capture self and mutually exciting features of the trade process, we model the counting process as a Hawkes process so that its intensity λ satisfies

$$\lambda_t = \mu_t + \int_0^t \Phi(t-s)\mathrm{d}N_s,$$

where μ is a vector-valued function of positive entries and Φ a positive matrix-valued function with integrable entries such that the spectral radius of $\|\Phi\|_1$ is below unity. However, calibration results suggest, that for models of the above form to fit data, the spectral radius of the L^1 norm of the kernel must be close to one [LC14, FS12, HBB13]. This motivated a ***nearly-unstable*** framework where it is close to one [JR16, JR15, EEFR18]. In this framework, the dynamics of the market up to time T are modeled with a Hawkes process N^T of baseline μ^T and kernel Φ^T, so that its intensity λ^T satisfies

$$\lambda_t^T = \mu_t^T + \int_0^t \Phi^T(t-s)\mathrm{d}N_s^T.$$

In order to obtain macroscopic dynamics, as time horizon T goes to infinity, the spectral radius of $\|\Phi^T\|_1$ goes to one. In a simplified framework examined in [EEFR18], a single eigenvalue almost saturates the stability condition, which leads to a single volatility factor at the limit for all assets. To obtain more general dynamics, we assume the Hawkes kernel is of the form, using block matrix notation

$$\Phi^T(t) = O\begin{pmatrix} A^T(t) & 0 \\ B^T(t) & C^T(t) \end{pmatrix} O^{-1},$$

where O is an invertible $2d \times 2d$ matrix, $A^T : \mathbb{R}_+ \to \mathscr{M}_{d_c}(\mathbb{R})$, $B^T : \mathbb{R}_+ \to \mathscr{M}_{2d-d_c,d_c}(\mathbb{R})$ and $C^T : \mathbb{R}_+ \to \mathscr{M}_{2d-d_c}(\mathbb{R})$. This block structure allows for richer limiting dynamics. We assume that the stability condition is saturated at the speed $T^{-\alpha}$ so that the saturation condition translates to

$$T^\alpha\big(I - \int_0^\infty A^T(s)\mathrm{d}s\big) \underset{T\to\infty}{\to} K,$$

where K is an invertible matrix. We encode the long memory property of the order flow by imposing a heavy-tail condition for $A := \lim_{T\to\infty} A^T$ with the same exponent α:

$$\alpha x^\alpha \int_x^\infty A^T(s)\mathrm{d}s \underset{x\to\infty}{\to} M,$$

where M is an invertible matrix.

5.2 Multivariate rough Heston

In the framework described above, we show that the macroscopic limit of prices is a multivariate version of the rough Heston model introduced in [EEGRR20, EEGR19], where the volatility process is a solution of a multivariate rough stochastic Volterra equation.

We introduce the rescaled microscopic price process P^T defined for $t \in [0,1]$ as

$$P_t^T = \frac{1}{T^{2\alpha}}(N_{tT}^{T,1+} - N_{tT}^{T,1-}, \cdots, N_{tT}^{T,d+} - N_{tT}^{T,d}). \tag{23}$$

The rescaled microscopic price process corresponds to the tick-by-tick price of each asset, properly scaled so that it admits a non-trivial limit as T goes to infinity. This limit is dubbed the macroscopic price process. We show in Chapter V the following result.

Result 15. *Under some technical and no statistical arbitrage assumptions, there exists a d_c dimensional process $\tilde{V}$, matrices $\Theta^1 \in \mathcal{M}_{d_c}(\mathbb{R}), \Theta^2 \in \mathcal{M}_{n-d_c}(\mathbb{R}), \Gamma_0 \in \mathcal{M}_{d_c}(\mathbb{R}), \Gamma_1 \in \mathcal{M}_{d_c}(\mathbb{R}), \Gamma_2 \in \mathcal{M}_{d_c,n-d_c}(\mathbb{R}), \theta_0 \in \mathbb{R}^{d_c}$, with explicit expressions given in Chapter V, and two independent Brownian motions W, Z such that*

- *Any macroscopic limit point P of the sequence P^T satisfies*

$$P_t = \Sigma \int_0^t \mathrm{diag}\sqrt{\Theta^1 \tilde{V}_s} dW_s + \Sigma \int_0^t \mathrm{diag}\sqrt{\Theta^2 \tilde{V}_s} dZ_s,$$

  where $\tilde{V}$ is defined below.

- *$\tilde{V}$ has Hölder regularity $\alpha - 1/2 - \epsilon$ for any $\epsilon > 0$.*

- *For any t in $[0,1]$, $\tilde{V}$ satisfies*

$$\tilde{V}_t = \int_0^t (t-s)^{\alpha-1}(\theta_0 - \Gamma_0 \tilde{V}_s) ds + \int_0^t (t-s)^{\alpha-1} \Gamma_1 \mathrm{diag}\sqrt{\Theta^1 \tilde{V}_s} dW_s + \int_0^t (t-s)^{\alpha-1} \Gamma_2 \mathrm{diag}\sqrt{\Theta^2 \tilde{V}_s} dZ_s.$$

Thus the volatility process is driven by $\tilde{V}$, which represents volatility factors, of which there are as many as there are critical directions. The matrices $\theta_0, \Theta^1, \Theta^2, \Gamma_0, \Gamma_1, \Gamma_2, \Sigma$ are given explicitly in Chapter V.

We can use this result to provide microscopic foundations for some empirical properties of correlation matrices. Informally, the parameters appearing in the previous result are connected to the limiting matrices K and M. We show in Chapter V that coupling price jumps of multiple assets at the high-frequency scale, roughly mimicking the co-jumps observed in [BCT+15, CBT+18], leads to Σ which has one very large eigenvalue followed by smaller eigenvalues that we can interpret as due to the presence of sectors and a bulk of eigenvalues much smaller than the others, which is typical of stock correlation matrices [LCBP99].

6 Chapters VIA and VIB: Deep learning volatility

Multivariate rough volatility models emerged in Chapter V from tick-by-tick models for prices. Such models are computationally intensive and thus hard to calibrate on market data. Chapters VIA and VIB addresses this issue. We begin by introducing the setup of the calibration problem.

6.1 The calibration problem

In plain words, for a given set of parameters of a volatility model, one can compute a set of contract prices. The calibration procedure of volatility models is meant to fix parameters such that the model yields prices as close as possible to market prices.

We formalise this by setting the notation $\mathscr{M} := \mathscr{M}(\theta)_{\theta\in\Theta}$ which represents an abstract model with parameters θ in the set $\Theta \subset \mathbb{R}^n$, for some $n \in \mathbb{N}$. Thus the model $\mathscr{M}(\theta)$ and the corresponding prices of financial contracts are fully specified by the choice of the parameter combination $\theta \in \Theta$. Furthermore, we introduce a pricing map $P : \mathscr{M}(\theta,\zeta) \to R^m$, where $\zeta : (C(\mathbb{R}) \to \mathbb{R}^m)$, $m \in \mathbb{N}$ denote the financial products we aim to price, such as vanilla options for (a set of) given maturities and strikes. We denote the observed market data corresponding to the contracts by $\mathscr{P}^{MKT}(\zeta) \in \mathbb{R}^m$, $m \in \mathbb{N}$. The generic setup of the calibration problem is introduced in the next definition.

Definition 5. *The parameter configuration $\hat{\theta}$ solves an (ideal) δ-calibration problem for a model $\mathscr{M}(\Theta)$ for the conditions $\mathscr{P}^{MKT}(\zeta)$ if*

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(P(\mathscr{M}(\theta),\zeta), \mathscr{P}^{MKT}(\zeta)) \tag{24}$$

where $\delta(\cdot,\cdot)$ is a suitable choice of metric for the financial contract ζ at hand.

Equation (24) represents an idealised form of the calibration problem as in practice there rarely exists an analytical formula for the option price $P(\mathscr{M}(\theta),\zeta)$ and for the vast majority of financial models it needs to be computed by some numerical approximation scheme, which motivates the following definition.

Definition 6. *We say that the parameter configuration $\hat{\theta} \in \Theta$ solves an approximate δ-calibration problem for the model $\mathscr{M}(\Theta)$ for the conditions $\mathscr{P}^{MKT}(\zeta)$ if*

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(\widetilde{P}(\mathscr{M}(\theta),\zeta), \mathscr{P}^{MKT}(\zeta)) \tag{25}$$

where $\delta(\cdot,\cdot)$ is a suitably chosen metric and $\widetilde{P}$ is a numerical approximation of the pricing map P.

The next section introduces the two-step approach to solve the approximate calibration problem.

6.2 The two step approach

Whenever for a stochastic volatility model the computation time of option prices is slow, it causes a bottleneck in calibration time. This is the case in particular for the family of rough volatility models (see [AGLM18, BFG16, EER19, GJR18], among others). To address this issue, we propose to split the calibration procedure into two parts:

- We first approximate the pricing map by a neural network that maps parameters of a stochastic model to option prices and store this map during an off-line training procedure.
- We calibrate the volatility model using the deterministic approximation of the price map, which speeds up the on-line calibration by orders of magnitude.

To formalise the two step approach, we write for a payoff ζ and a model $\mathscr{M}$ with parameters $\theta \in \Theta$ the two steps

$$\textbf{(i) Learn: } \widetilde{F}(\Theta,\zeta) = \widetilde{P}(\mathscr{M}(\Theta,\zeta)) \qquad \textbf{(ii) Calibrate: } \hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(\widetilde{F}(\theta,\zeta), \mathscr{P}^{MKT}(\zeta)). \tag{26}$$

Part **(i)** in (26) denotes an approximation of the pricing map through a neural network, which is calibrated in a supervised training procedure using the original numerical pricing maps for training. We give details for the training procedure in Chapters VIA and VIB.

Part **(ii)** of (26) we essentially replaced $\widetilde{P}(\mathscr{M}(\Theta,\zeta))$ in Equation (25) by a neural-network $\widetilde{F}(\Theta,\zeta)$ from **(i)**. Therefore, this second calibration is considerably faster than calibration of all those traditional stochastic models which involve numerical simulation of the expected payoff $P(\mathscr{M}(\theta,\zeta)) = \mathbb{E}[\zeta(X(\theta))]$ for some underlying stochastic process $X(\theta)$.

In the two step approach, we distinguish between two approximation methods which yields distinct results: the pointwise and grid-based approximations. The pointwise approach learns the map from model parameters and contract characteristics (such as strike and time to maturity) to the contract's price. The grid-based approach fixes a priori a set of options of varying characteristics (strikes and maturities) and learns the map of model parameters to these option prices. The details of each method are presented in Chapters VIA and VIB.

The next section tests the accuracy of both stages of the two step approach.

6.3 Accuracy of the method

The methodology can be applied to any volatility model but to illustrate the approach we focus on one particular model, the rough Bergomi model [BFG16], which we introduce below.

Definition 7. *In the abstract model framework, the rough Bergomi model is represented by* $\mathscr{M}^{rBergomi}(\Theta^{rBergomi})$, *with parameters* $\theta = (\xi_0, \nu, \rho, H) \in \Theta^{rBergomi}$. *On a given filtered proba-*

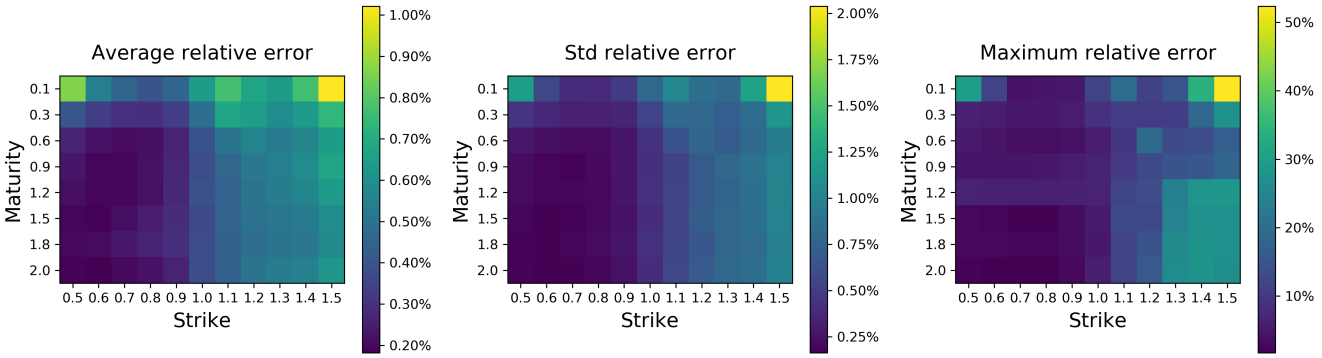


Figure .6: **Relative errors of the neural network approximation**.
We compare surface relative errors of the neural network approximator against the Monte Carlo benchmark across all training data (68,000 random parameter combinations) in the rough Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right).

bility space $(\Omega, \mathscr{F}, (\mathscr{F}_t)_{t\geq 0}, \mathbb{P})$ *the model corresponds to the following system*

$$\begin{aligned} dX_t &= -\frac{1}{2}V_t\,dt + \sqrt{V_t}\,dW_t, \quad \text{for } t>0, \quad X_0 = 0, \\ V_t &= \xi_0(t)\mathscr{E}\left(\sqrt{2H}\nu\int_0^t (t-s)^{H-1/2}\,dZ_s\right), \quad \text{for } t>0, \quad V_0 = v_0 > 0 \end{aligned} \tag{27}$$

where $H \in (0,1)$ *denotes the Hurst parameter,* $\nu > 0$, $\mathscr{E}(\cdot)$ *the stochastic exponential [DD70], and* $\xi_0(\cdot) > 0$ *denotes the initial forward variance curve (see [Ber15, Section 6]), and* W *and* Z *are correlated standard Brownian motions with correlation parameter* $\rho \in [-1,1]$. *To fit the model parameters into our abstract model framework* $\Theta^{rBergomi} \subset \mathbb{R}^n$ *for some* $n \in \mathbb{N}$, *the initial forward variance curve* $\xi_0(\cdot) > 0$ *is approximated by a piecewise constant function.*

The numerical computation of option prices from the rough Bergomi model is computationally expensive but feasible through Monte-Carlo methods [HJM17].

6.3.1 Approximation step

To assess the quality of the neural network approximation, we compare predicted option prices of the neural network with prices obtained via Monte Carlo methods. The experiment details are given in Chapters VIA and VIB. Figure .6 shows the accuracy of the neural network used to approximate the pricing function in the grid-based approach. In particular, it highlights the following result.

Result 16. *The average error of the neural network approximation for the grid-based approach is within the bid-ask spread of option prices.*

Given Result 16, our fast approximation of option prices may be accurate enough for calibration, which we examine in the next step.

6.3.2 Calibration step

To assess the precision of the calibration results, we generate an implied volatility surface from a given set of parameters, obtain a set of parameters using the two-step approach and finally compare the real parameters and the estimated parameters. The precision of the calibration of the pointwise method in Figure .7 and the precision of the grid-based method is given in Figure .8. The numerical experiments suggest the following result.

Result 17. *The calibration method for the pointwise and grid-based methods recover parameters close to the original set of model parameters.*

6.4 Historical calibration

Finally, to illustrate the suitability of our approach for practical settings, we show numerical calibration results on real option prices. We showcase the pointwise approach with a liquidity-weighted Bayesian regression against SPX implied volatilities from 19th May 2017 in Figure .9 and the grid-based approach on a historical implied volatilities of the SPX in Figure .10. The calibration results highlight the following result.

Result 18. *The calibration results on historical data yield model parameters in line with typical parameter values and evolving regularly in time.*

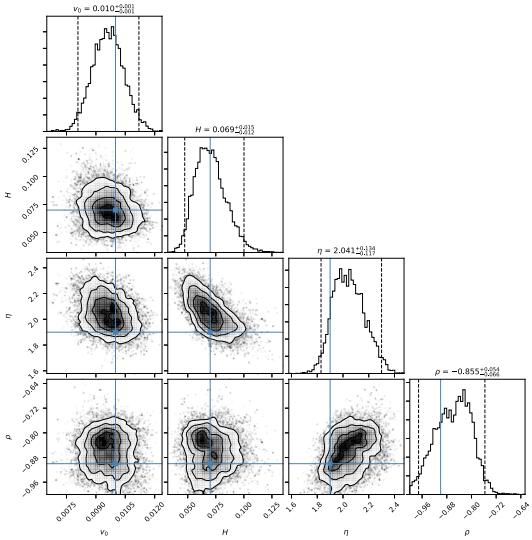


Figure .7: **Synthetic parameter calibration in the pointwise approach.**
Bayes calibration in the pointwise approach against synthetic implied volatility surface. The posterior distribution of estimated parameters is shown in black, while solid vertical blue lines indicate true parameter values. Details for the construction of the posterior distributions of the parameters given the data are given in Chapter VIB. The histograms show that the posterior distribution calibration is centered around true parameter values, thus being close to the exact parameters.

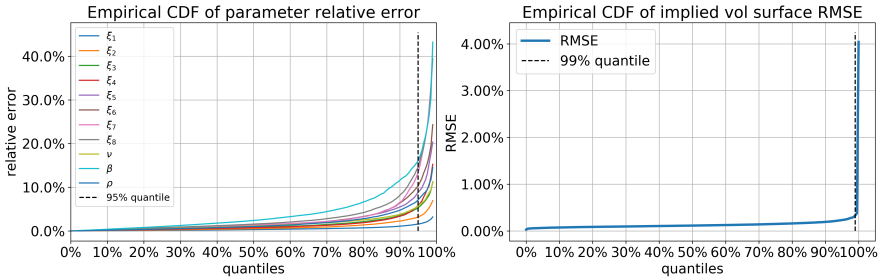


Figure .8: **Synthetic parameter and implied volatility surface error in the grid-based approach.**
Cumulative Distribution Function (CDF) of 1 Factor Bergomi parameter relative errors (left) and RMSE (right) after Levengerg-Marquardt calibration across test set random parameter combinations.

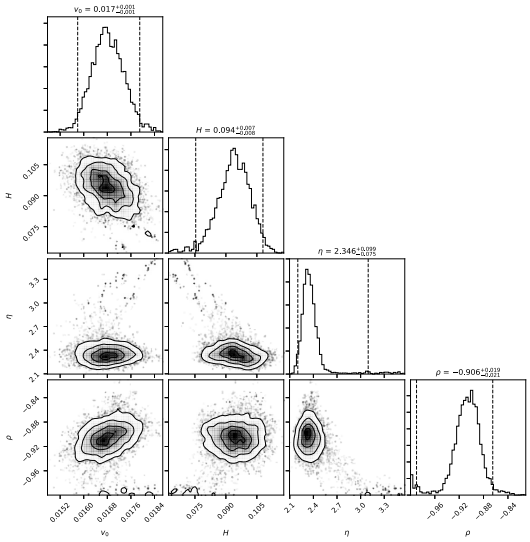


Figure .9: **Historical parameter calibration in the pointwise approach.**
Liquidity-weighted Bayes calibration against SPX market implied volatility surface from 19th May 2017. Liquidity proxies are given by inverse bid-ask-spreads of market prices. The posterior distribution of estimated parameters is shown in black, while solid vertical blue lines indicate true parameter values. Details for the construction of the posterior distributions of the parameters given the data are given in Chapter VIB.

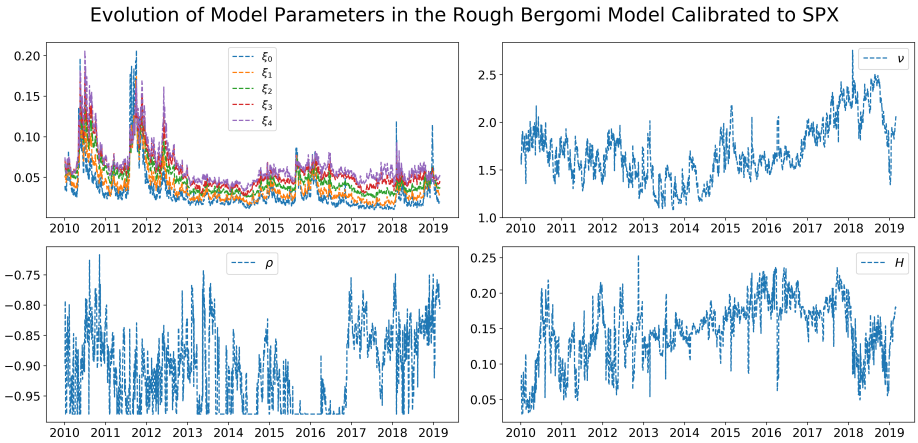


Figure .10: **Historical parameter calibration in the grid-based approach.**
Historical Evolution of parameters of the rough Bergomi model with a piecewise constant forward variance term structure calibrated on SPX.

CHAPTER I

Static cross-impact models

Abstract

Trading a financial instrument pushes its price and those of other assets, a phenomenon known as cross-impact. To be of use, cross-impact models must fit data and be well-behaved so they can be applied in applications such as optimal trading. To address these issues, we introduce a set of desirable properties which constrain cross-impact models. We classify cross-impact models according to which properties they satisfy and stress them on three different asset classes to evaluate goodness-of-fit. We find that two models are robust across markets, but only one satisfies all desirable properties and is appropriate for applications.

From:
How to build a cross-impact model: Theoretical requirements and empirical results
M. Tomas, I. Mastromatteo, M. Benzaquen

Introduction

Trading pressure moves prices, a now well-established phenomenon known as *market impact* [BBDG18, ATHL05, Tor97]. In fact, market impact has been measured in many independent studies and it is robust across assets, time periods and markets. A more subtle effect is that trading pressure from one asset can move the price of another. This effect has been dubbed *cross-impact*. cross-impact transmits information across assets and amplifies market shocks. Many papers incorporate cross-impact in applications but assume that the parameters of the model are known [ELMK18, EMK19, GP16, TWG19]. To be applied in practice, they require a calibration technique to estimate cross-impact from data.

The importance of cross-impact has sparked recent interest in devising calibration methodologies from empirical data [HS01, PV15, BMEB17, WSG16, SL19, TMB21, RT21a]. However, empirical studies focus on goodness-of-fit to calibrate cross-impact models or make simplifying assumptions which may work well on certain asset classes but break on others. Therefore, from the literature, we cannot determine whether there exists a universally robust and statistically accurate cross impact model, nor more specifically one which is suitable in other applications,

such as optimal trading.

This chapter seeks to bridge this gap by introducing desirable properties of cross-impact models, classifying models according to which properties they satisfy and stress-testing these models on different markets to assess which provide good empirical fit. In particular, we find that there is a single cross-impact model robust across asset classes and well-suited for applications.

We briefly comment on the links between our approach and the literature.

The paper most related to our work is [RT21a]. There, the authors characterise suitable cross-impact models within a market where trades are modeled with Hawkes processes. The resulting cross-impact models are well-behaved and can be calibrated, which the authors illustrate on E-Mini S&P 500 futures.

This chapter stress-tests many different cross-impact models and thus contributes to the literature on the calibration and goodness-of-fit testing of cross-impact models [BMEB17, SL19, RT21a]. However, papers on the literature focus on one (or few) models at a time on a particular asset class. Thus, this chapter shines light on which models perform best on a variety of markets.

A set of theoretical studies have attempted to characterise suitable cross-impact models from certain properties. Unfortunately, a pure no-arbitrage framework as in [AKS16, SL19] is not sufficiently restrictive to prescribe a calibration methodology. We use some of their results to classify static cross-impact models which allow for arbitrage.

Other studies obtain cross-impact models via interactions of different agents. While the mean-field framework for optimal execution of [LM19] provides one explanation of the many possible phenomena underlying cross-impact, it does not provide a recipe one may use on empirical data. In the optimal market making literature, [BEGV18] finds that the liquidation costs of a market maker faces when he holds an inventory q is of the form $-q^\top \Lambda q$, where Λ can be estimated in practice. In fact, the same Λ can be derived from the multivariate Kyle framework [GdMMBB20, CK94]. We show that this Λ plays a special role, as it is the only cross-impact model which satisfies all desirable properties. This partially explains why this cross-impact model appears in a variety of seemingly unrelated settings.

The chapter is organized as follows. Section 1 introduces the setup of the chapter. Section 2 lists axioms and Section 3 cross-impact models of interest, along with the axioms they satisfy. Section 4 stress-tests cross-impact models on a variety of markets. We conclude by stressing the main contributions of the chapter and discussing open questions and directions for future work.

Notation

The set of n by n real-valued square matrices is denoted by $\mathcal{M}_n(\mathbb{R})$, the set of orthogonal matrices by $\mathcal{O}_n$, the set of real symmetric positive semi-definite matrices by $\mathcal{S}_n^+(\mathbb{R})$ and the set of real symmetric positive definite matrices by $\mathcal{S}_n^{++}(\mathbb{R})$. Given a matrix A in $\mathcal{M}_n(\mathbb{R})$, $A^\top$ denotes its transpose. Given A in $\mathcal{S}_n^+(\mathbb{R})$, we write $A^{1/2}$ for a matrix such that $A^{1/2}(A^{1/2})^\top = A$ and $\sqrt{A}$ for the matrix square root, the unique positive semi-definite symmetric matrix such that $(\sqrt{A})^2 = A$. We write $\ker(M)$ for the null space of a matrix $M \in \mathcal{M}_n(\mathbb{R})$, Π_V for the projector on a linear subspace of $V \in \mathbb{R}^n$ and $\bar{\Pi}_V = I - \Pi_V$ for the orthogonal projector. Finally, given a vector $v \in \mathbb{R}^n$, we write $v = (v_1, \ldots, v_n)$ and $\mathrm{diag}(v)$ for the diagonal matrix with diagonal components the components of v.

1 Cross-impact models as functions of market data

To relate trades to prices, we observe the mid-prices and trades of d different assets, both being binned on a regular time interval of length Δt. We denote by p_t^i the opening price of Asset i in the time window $[t, t+\Delta t]$ and by $p_t = (p_t^1, \cdots, p_t^d)$ the vector of asset prices. We write q_t^i the net market order flow traded in the same period, i.e. the signed sum of the volumes of all trades in that window, counting buy trades volume positively and sell trades volume negatively. Similarly, $q_t = (q_t^1, \cdots, q_t^d)$ is the vector of net traded order flow.

On a given trading day, our goal is to relate the time series of net order flows $\{q_0, q_{\Delta t}, \cdots, q_T\}$ with the time series of prices $\{p_0, p_{\Delta t}, \cdots, p_T\}$. For a given asset, it is classically admitted that price changes and net order flow are linearly related, although the linear relationship breaks down for large order flow values [PB18]. Inspired by this, we also assume this is true across assets. Furthermore, to emphasize the cross-sectional features of the problem, we discard the influence of past order flows. This leads to the following assumption.

Assumption 1. *Price changes $\Delta p_t := p_{t+\Delta t} - p_t$ and order flow imbalances q_t are linearly related, i.e.*

$$\Delta p_t = \Lambda_t q_t + \eta_t\,, \tag{1}$$

where the $d \times d$ matrix Λ_t is called the cross-impact matrix and $\eta_t = (\eta_t^1, \ldots, \eta_t^d)$ is a vector of zero-mean random variables representing exogenous noise.

In Equation (1), price changes and order flow imbalances are known and we have to choose the cross-impact matrix Λ_t. Our choice of cross-impact matrix influences the quality of fit of the model via the size of the residuals of Equation (1). The focus of this chapter is to find the right choice for the cross-impact matrix Λ_t.

While many choices for the cross-impact matrix are possible, we focus on those which depend on the parameters of the true data generating process. In the particular context where all random variables are Gaussian, then second-order statistics are sufficient statistics. This motivates the next assumption.

Assumption 2. *In the rest of the chapter, we assume that price dynamics are given by*

$$\Delta p_t = \Lambda(\Sigma_t, \Omega_t, R_t) q_t + \eta_t, \tag{2}$$

where $\Lambda\colon \mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}) \to \mathscr{M}_n(\mathbb{R})$ *is called a* cross-impact model, $\Sigma_t = \mathrm{Cov}(\Delta p_t)$ *is the price change covariance matrix,* $\Omega_t := \mathrm{Cov}(q_t)$ *is the order flow covariance matrix,* $R_t = \mathbb{E}\left[(\Delta p_t - \mathbb{E}[\Delta p_t])(q_t - \mathbb{E}[q_t])^\top\right]$ *is the response matrix and* $\eta_t = (\eta_t^1, \dots, \eta_t^d)$ *is a vector of zero-mean random variables representing exogenous noise.*

The cross-impact model Λ of Equation (2) is a function of the three key second-order statistics which describe our market. Indeed, Σ_t quantifies the co-variation of returns, Ω_t captures co-trading of different assets, and R_t reflects the average change of asset prices with traded order flow. By examining cross-impact models, we abstract the particular value of a cross-impact model for a given set of second-order statistics. Instead, this allows us to examine the cross-impact model across markets.

The main purpose of this chapter is to find a suitable cross-impact impact model Λ in Equation (2) given a set of observations of market data and corresponding statistics Σ_t, Ω_t, R_t. The next section discusses how to choose a proper cross-impact model Λ. First, we comment on the validity of our assumptions.

Remark 1 (Validity of the static approximation). *Overall, Assumption 1 is relevant to describe price impact shortly induced after trading for small portfolios. To assess its validity, we compare our setup to models which account for past order flow imbalances [SL19, AKS16, BMEB17] where price dynamics are*

$$p_t = \sum_{s \le t} G(t-s) q_s + \xi_t,$$

where $G\colon t \mapsto G(t) \in \mathscr{M}_d(\mathbb{R})$ *captures the dependence on past order flow and* ξ_t *is a vector of zero-mean random variables. Then*

$$\Delta p_t = G(0) q_t + \eta_t + \sum_{s<t} \left(G(t+\Delta t - s) - G(t-s)\right) q_s,$$

where $\eta_t := \xi_{t+\Delta t} - \xi_t$ *is a vector of zero-mean random variables. Assumption 1 ignores the last term of the right hand side in the above equation. Therefore, we can measure the validity of our approximation by comparing* $\mathscr{G}_{ij} := \sum_{s<t} G_{ij}(t+\Delta t - s) - G_{ij}(t-s)$ *and* $G_{ij}(0)$*, say with* $\kappa_{ij} := \frac{\mathscr{G}_{ij}}{G_{ij}(0)}$.

For $\Delta t = 5$ *minutes and on stocks, Figure 3 of [BMEB17] shows* $\kappa_{ij} \approx 20\%$. *Figure 5 of [SL19] works in transaction time on bonds but a rough estimate for* $\Delta t = 1$ *minute yields* $\kappa_{ij} \approx 30\%$. *This indicates our setup is relevant to capture the salient features of cross-impact.*

Before proceeding, we comment on the structure of the trading costs in our setup.

Remark 2 (Trading costs). *Equation* (2) *gives a prediction of portfolio trading costs. In particular, if one assumes that the difference between the arrival price and the execution price is given by* Δp_t*, the cost incurred after the execution of the portfolio* ξ *is*

$$\mathscr{C}(\xi) = \xi^\top \Delta p_t = \xi^\top \Lambda_t \xi + \xi^\top \Lambda_t \bar{q}_t + \xi^\top \eta_t, \tag{3}$$

where $\bar{q}_t$ is the order flow imbalance due to trades of other market participants. Thus

$$\mathbb{E}[\mathscr{C}(\xi)] = \xi^\top \Lambda_t \xi + \xi^\top \Lambda_t \mathbb{E}[\bar{q}_t \mid \xi] + \xi^\top \eta_t \,,$$

where $\mathbb{E}[\bar{q}_t \mid \xi]$ represents the alignment of the market trades' conditioned to the traded portfolio. This may be non-zero because of herding, where our trades cause other investors to trade. The treatment of this term depends on the trading strategy and is outside of the scope of this chapter. Thus, we assume that $\mathbb{E}[\bar{q}_t \mid \xi] = \mathbb{E}[\bar{q}_t] = 0$ so that the average impact costs of trading the portfolio ξ in our setting is $\mathbb{E}[\mathscr{C}(\xi)] = \xi^\top \Lambda_t \xi$.

The next section examines desirable properties we want cross-impact models to satisfy.

2 Axioms: the desirable properties of cross-impact models

To find a proper cross-impact model for Equation (2), we have two potentially conflicting objectives. The first is goodness of fit, or how well our model captures the influence of order flow to explain price changes. Cross-impact models are often selected on this basis alone [BMEB17, SL19, WSG16]. This may yield good empirical fit but these models neglect another important aspect: the theoretical implications of our market model of Equation (2). For example, does our choice of cross-impact model imply that agents can abuse cross-impact to make a profit on average? These properties are critical if we want to use cross-impact models in applications, such as [ELMK18, EMK19, GP16, TWG19].

To select a model on the basis of its implications, we need to establish which properties we would like a cross-impact model to satisfy. To address this issue, this section proposes a list of desirable properties of cross-impact models, which we dub *axioms*. Each axiom translates a desired behaviour of cross-impact model, grounded in the implications of the cross-impact model for the evolution of prices.

2.1 Symmetry

The first type of axioms we introduce are *symmetry* axioms. They ensure that the cross-impact is well-behaved under financially-grounded transformations of its variables. First, the cross-impact model should adapt to a re-ordering of the assets. This yields the following axiom.

Axiom 1 (Permutational invariance). *A cross-impact model Λ is* permutation-invariant *if, for any permutation matrix P and $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$,*

$$\Lambda(P\Sigma P^\top, P\Omega P^\top, PRP^\top) = P\Lambda(\Sigma, \Omega, R)P^\top.$$

In the absence of any return or order flow correlation among assets, we expect price changes to be independent. The cross-impact model should then respect this property. This motivates the following axiom.

Axiom 2 (Direct invariance). *A cross-impact model* Λ *is* direct-invariant *if, for any* $\sigma, \omega \in \mathbb{R}^n_+$ $r \in \mathbb{R}^n$,

$$\Lambda(\mathrm{diag}(\sigma)^2, \mathrm{diag}(\omega)^2, \mathrm{diag}(r)) = \sum_{i=1}^{n} \Lambda(\sigma_i^2 e_i e_i^\top, \omega_i^2 e_i e_i^\top, r_i e_i e_i^\top),$$

where e_i *is the* i*-th element of the canonical basis.*

Impact is expressed in a choice of currency units. However, the chosen currency should not matter and cross-impact models should adapt accordingly. The next axiom translates this property.

Axiom 3 (Cash invariance). *A cross-impact model* Λ *is* cash-invariant *if, for any* $\alpha > 0$*, and* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$,

$$\Lambda(\alpha^2 \Sigma, \Omega, \alpha R) = \alpha \Lambda(\Sigma, \Omega, R).$$

Similarly, cross-impact models should account for changes in volume units. For example, stock splits can double the number of outstanding shares and halve their values (if we ignore microstructural effects such as tick size and lot rounding). This leads to the following axiom.

Axiom 4 (Split invariance). *A cross-impact model* Λ *is* split-invariant *if, for any diagonal matrix of positive elements* $D \in \mathscr{M}_n(\mathbb{R})$ *and* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$,

$$\Lambda(D^{-1}\Sigma D^{-1}, D\Omega D, D^{-1} R D) = D^{-1} \Lambda(\Sigma, \Omega, R) D^{-1}.$$

The profit and loss of traders is invariant under orthogonal transformations (see Equation (3)). It is natural to look for cross-impact models that share this property. As before, this ignores microstructural effects such as exchange trading fees, bid-ask spreads, etc. The following axiom introduces this property.

Axiom 5 (Rotational invariance). *A cross-impact model* Λ *is* rotation-invariant *if, for any real orthogonal matrix* $O \in \mathscr{O}_n$ *and* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$,

$$\Lambda(O\Sigma O^\top, O\Omega O^\top, ORO^\top) = O\Lambda(\Sigma, \Omega, R) O^\top.$$

We say of a model which does not satisfy Axiom 5 that it has a privileged basis. Note that any cross-impact model which satisfies Axioms 4 and 5 is invariant under the action of any non-singular matrix M.

Among symmetry axioms, we expect permutational, direct and cash invariance (Axioms 1 and 3) to be of critical importance as models which do not respect them would behave oddly. Split invariance (Axiom 4) is realistic but it may break on small timescales due to microstructural effects. On the other hand, rotational invariance (Axiom 5) is less plausible because markets have transaction costs, leverage constraints and other effects which break this symmetry.

2.2 Arbitrage

This family of axioms clarifies what properties a cross-impact model should satisfy to excludes any statistical arbitrage in the sense of [Gat10], i.e. round-trip trading strategies with positive average profit. The first axiom involves *static* arbitrages: single period trading strategies with average negative costs.

Axiom 6 (Positive semi-definiteness). *The cross-impact model Λ takes values in the space of positive semi-definite matrices.*

The next axiom involves *dynamic* arbitrages, i.e. multi-period trading strategies in the spirit of [AKS16, Gat10, SL19]. Even though these arbitrages cannot be exploited in our single-period setup, they would emerge by generalizing our setup to the multi-period setting as shown in [SL19]. This is why we choose to also consider this class of arbitrages.

Axiom 7 (Symmetry). *The cross-impact model Λ takes values in the space of symmetric matrices.*

Axioms 6 and 7 together are sufficient to guarantee absence of statistical arbitrages. Arbitrage-related axioms are of great important in applications where the presence of arbitrages leads to odd behaviour. For example, [AKS16] highlights how arbitrageable cross-impact models lead to ill-behaved optimal trading strategies. Although outside the scope of this chapter, it is interesting to assess if real markets admit some kind of statistical arbitrage, and whether these hold when factoring other transaction costs (see [SL19]).

2.3 Fragmentation

While the previous axioms ruled out statistical arbitrage, another related issue is what happens when trading assets (or combination of assets) which have constant prices. For example, consider a stock traded on multiple markets (say, Apple traded on the Nasdaq and on the Bats venues). For a reasonably large interval of time Δt (and abstracting microstructural effects), we expect $p_{\text{Nasdaq}} - p_{\text{Bats}} = 0$. Thus, buying a volume $q = q_{\text{Nasdaq}} + q_{\text{Bats}}$ of Apple stock should yield the same cost no matter how one fragments the q_{Nasdaq} units bought on Nasdaq and the q_{Bats} units bought on Bats. For this reason, this axiom is dubbed *fragmentation invariance*.

We distinguish between three different forms of fragmentation invariance. The first, *weak fragmentation invariance*, concerns the price changes given by a cross-impact model and is detailed in the next Axiom.

Axiom 8 (Weak fragmentation invariance). *A cross-impact model Λ is* weakly fragmentation invariant *if, for any $(\Sigma, \Omega, R) \in (\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$ and $\emptyset \subset V \subseteq \ker \Sigma$,*

$$\Pi_V \Lambda(\Sigma, \Omega, R) = 0\,,$$

where we recall that Π_V denotes the projector on the linear subspace V.

In practice, if the price of a linear combination of assets is constant, weak fragmentation invariance guarantees that its price cannot be moved through trading.

Remark 3. *From now on, we will implicitly assume that* $\ker(\Sigma) \subseteq \ker(R^\top)$*, which is consistent with the interpretation of* Σ *and* R *as covariations of prices and order flows. This implies that from the point of view of the fragmentation-related axioms, any condition involving the the kernel of* Σ *will be naturally related to the kernel of* $R^\top$ *as well.*

We obtain a stronger condition if we require volume traded in zero-volatility directions to induce no price impact. This leads to the following Axiom.

Axiom 9 (Semi-Strong fragmentation invariance). *A cross-impact model satisfies* semi-strong fragmentation invariance *if, besides satisfying the weak fragmentation invariance Axiom 8, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *and* $\emptyset \subset V \subseteq \ker \Sigma$,

$$\Lambda(\Sigma, \Omega, R)\Pi_V = 0\,.$$

We can go one step further by ensuring that the cross-impact model itself should also not depend on how zero-volatility directions are traded by *other* market members. This is *strong fragmentation invariance*, the subject of the next Axiom.

Axiom 10 (Strong fragmentation invariance). *A cross-impact model* Λ *is* strongly fragmentation invariant *if, besides satisfying semi-strong fragmentation invariance (Axiom 9), for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *and* $\emptyset \subset V \subseteq \ker(\Sigma)$,

$$\Lambda(\Sigma, \Omega, R) = \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V)\,.$$

Weak fragmentation invariance (Axiom 8) is critical since it prevents models from predicting price changes for zero-volatility instruments. Furthermore, it properly aggregates liquidity of an asset traded on multiple venues. For the same reasons, semi-strong and strong fragmentation invariance (Axioms 9 and 10) should be of crucial importance.

2.4 Stability

Fragmentation invariance axioms constrain cross-impact models in extreme regimes of price correlations. Similarly, stability axioms control behaviour in extremes of liquidity. Intuitively, price manipulation of liquid products using illiquid instruments should be excluded.

We model this by defining a set V of illiquid instruments. We consider the matrix $\bar{\Pi}_V + \varepsilon \Pi_V$ that multiplies by $\varepsilon \ll 1$ the liquidity of all instruments belonging to V. After multiplying the traded order flow by this matrix, the observables become

$$\begin{aligned}
\Sigma_\varepsilon^q &:= \Sigma \\
\Omega_\varepsilon^q &:= (\bar{\Pi}_V + \varepsilon \Pi_V)\Omega(\bar{\Pi}_V + \varepsilon \Pi_V) \\
R_\varepsilon^q &:= R(\bar{\Pi}_V + \varepsilon \Pi_V)\,.
\end{aligned}$$

We are now ready to formulate liquidity axioms. First, trading illiquid instruments should not lead to large impact on liquid instruments. We would otherwise be able to manipulate the prices of liquid instruments. The converse should be true: we should not be able to manipulate prices of illiquid instruments by trading liquid instruments. This motivates the next axiom.

Axiom 11 (Weak Cross-Stability). *A cross-impact model* Λ *is* weakly cross-stable *if, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *and linear subspace* V *and using the above notations,*

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V \underset{\varepsilon\to 0}{=} O(1) \tag{4}$$

$$\Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{=} O(1)\,. \tag{5}$$

We can formulate a stronger cross-stability property. The next axiom formalizes the intuition that impact among liquid assets should be independent of the behavior of illiquid assets.

Axiom 12 (Strong Cross-Stability). *A cross-impact model* Λ *is* strongly cross-stable *if, in addition to satisfying weak-cross stability (Axiom 11), for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *and linear subspace* V *and using the above notations,*

$$\bar{\Pi}_V \Lambda\left(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q\right)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} \bar{\Pi}_V \Lambda\left(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V\right)\bar{\Pi}_V$$

An unresolved question is the effect of trading illiquid instruments on illiquid products. The following axiom deals with this issue.

Axiom 13 (Self-Stability). *A cross-impact model is* self-stable *if, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$*, subspace* V *and using the above notations,*

$$\Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V \underset{\varepsilon\to 0}{=} O(1). \tag{6}$$

Intuitively we want to avoid this property since it indicates that, even though a product is illiquid ($q \propto \varepsilon$, so that one would expect a diverging impact) the predicted cost of trading such product can be finite.

We believe weak cross-stability (Axiom 11) is fundamental. Indeed, it should be impossible to manipulate prices from liquid assets by trading illiquid assets and vice-versa. We would also like the stronger version of this axiom (Axiom 12) to hold: liquid instruments should be insensitive to trading on illiquid ones. On the other hand, self-stability (Axiom 13) does not penalize trading illiquid instruments. Thus, it is undesirable in applications.

2.5 Predicted covariance

Finally, it can be interesting to consider whether a cross-impact model predicts a contribution to the return covariance that is proportional to Σ or not.

Axiom 14 (Return covariance consistency). *A cross-impact model* Λ *is* return covariance consistent *if, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$*, it satisfies (up to a multiplicative constant):*

$$\Sigma = \Lambda(\Sigma, \Omega, R)\Omega\Lambda(\Sigma, \Omega, R)^\top.$$

This axiom is motivated by the fact that under the model in Equation (2), we expect return covariances to be given by

$$\Sigma = \mathrm{Cov}(\Delta p) = \Lambda \Omega \Lambda^\top + \mathrm{Cov}(\eta)\,,$$

so that if one assumes that the fundamental return covariance is proportional to the predicted one, i.e. $\mathbb{E}[\eta\eta^\top] \propto \Sigma$, one would recover return covariance consistency. This property controls the predicted price changes of the model, but we have no strong reason to believe cross-impact models should satisfy it.

2.6 Link between axioms

Fragmentation and cross-stability are related for split and rotation-invariant cross-impact models. The next proposition shows that fragmentation invariance implies cross-stability properties for continuous cross-impact models.

Proposition 1. *Let Λ be a jointly continuous cross-impact model which satisfies split and rotational invariance (Axioms 4 and 5). Then*

1. *If Λ satisfies semi-strong fragmentation invariance (Axiom 9), then it is weakly cross-stable (Axiom 11).*
2. *If Λ is strongly fragmentation invariant (Axiom 10), then it is strongly cross-stable (Axiom 12).*

We prove Proposition 1 in Section I.A.1. While the converse is not true, the next proposition shows that, given an additional regularity condition, cross-stability implies fragmentation invariance.

Proposition 2. *Let Λ be a jointly continuous cross-impact model which satisfies split and rotational invariance (Axioms 4 and 5). We further assume that, for any linear subspace V and using the notations of the previous section, $\varepsilon^2 \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) \underset{\varepsilon \to 0}{\to} 0$. Then*

1. *If Λ is weakly cross-stable (Axiom 11), then it satisfies semi-strong fragmentation invariance (Axiom 9).*
2. *If Λ is strongly cross-stable (Axiom 12), then it is strongly fragmentation invariant (Axiom 10).*

We prove Proposition 2 in Section I.A.1. A particularly interesting result of Propositions 1 and 2 is that for continuous cross-impact models which satisfy the regularity property of Proposition 2, fragmentation invariance and cross-stability are equivalent.

3 Candidate cross-impact models

Now that we have characterized the desirable properties of cross-impact models, we provide a set of cross-impact models and detail which axioms they satisfy. Their empirical performance will be assessed in Section 4. We divide these models in two classes; those that are based on

the return covariance Σ and those based on the response R.

Before presenting the different cross-impact models, we introduce some notation. For convenience, we will note the price volatility $\sigma := (\sqrt{\Sigma_{11}}, \cdots, \sqrt{\Sigma_{dd}})$, the signed order flow volatility $\omega := (\sqrt{\Omega_{11}}, \cdots, \sqrt{\Omega_{dd}})$, and the price and flow correlations $\rho := \mathrm{diag}(\sigma)^{-1}\Sigma\mathrm{diag}(\sigma)^{-1}$, $\rho_\Omega := \mathrm{diag}(\omega)^{-1}\Omega\mathrm{diag}(\omega)^{-1}$.

3.1 Return covariance based models

Let us start with the simplest possible linear impact model: one without cross-impact.

Definition 1 (`direct` model). *The* `direct` *model is defined for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *as*

$$\Lambda_{\texttt{direct}}(\Sigma, \Omega, R) := \mathrm{diag}(\sigma)^{1/2}\mathrm{diag}(\omega)^{-1/2}. \tag{7}$$

To generalize this model to the multivariate setting while respecting cash invariance, weak fragmentation invariance and consistency with correlations, a first idea is to use the matrices $\Sigma^{1/2}$ and $\Omega^{-1/2}$. Since $\Omega^{-1/2}q$ is a whitening transformation, this model is referred to as the `whitening` model.

Definition 2 (`whitening` model). *Recall that given* $M \in \mathscr{S}_n^+(\mathbb{R})$, $M^{1/2}$ *indicates a symmetric matrix factorization (i.e.,* $M^{1/2}(M^{1/2})^\top = I$*). The* `whitening` *model*[1] *is defined, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$, *as*

$$\Lambda_{\texttt{whitening}}(\Sigma, \Omega, R) := \Sigma^{1/2}\Omega^{-1/2}. \tag{8}$$

Unfortunately, this model does not respect symmetry, positive-definiteness, strong fragmentation invariance or weak cross-stability (Axioms 6, 7, 10 and 11). To impose symmetry and strong fragmentation invariance, the `el` model[2] proposed in [MBEB17] is directly expressed in the basis of the return covariance matrix.

Definition 3 (`el` model). *The eigenliquidity (*`el`*) model is defined, for any* $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$, *as*

$$\Lambda_{\texttt{el}}(\Sigma, \Omega, R) := \sum_{a=1}^{n} s_a \frac{\sqrt{\lambda_a}}{(s_a^\top \Omega s_a)^{1/2}} s_a^\top, \tag{9}$$

where we have introduced the eigenvalue decomposition of $\Sigma = \sum_{a=1}^n s_a \lambda_a s_a^\top$.

The `el` model is cross-stable, self-stable (Axioms 11 to 13) and is return covariance inconsistent (Axiom 14). As mentioned above, there is in fact only one model which satisfies all the axioms that we have provided: the so-called multivariate `kyle` model, see [GdMMBB20, CK94].

[1] The `whitening` model is not independent of the symmetric factorization chosen for Σ and Ω. As convention, we will take the square root obtained by an orthogonal decomposition of each matrix and the square root of their eigenvalues.

[2] The model proposed in [MBEB17] is actually the response-based one, referred later as `r-el`$\star$ model.

Definition 4 (`kyle` model). *The* `kyle` *model is defined, for any* $(\Sigma, \Omega, R) \in (\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{kyle}}(\Sigma, \Omega, R) := (\Omega^{-1/2})^\top \sqrt{(\Omega^{1/2})^\top \Sigma \Omega^{1/2}} \Omega^{-1/2}. \tag{10}$$

The `kyle` model plays a fundamental role as it is the only model which satisfies all axioms. This may explain why it appears in many different settings, seemingly unrelated to the Kyle insider trading setup [Gué17, EV18, RT21a]. The next proposition shows it is the only model which satisfies arbitrage axioms and return covariance consistency.

Proposition 3. *Let* Λ *be a symmetric, positive-semidefinite and return covariance consistent cross-impact model (Axioms 6, 7 and 14). Then* $\Lambda = \Lambda_{\texttt{kyle}}$ *up to a multiplicative constant.*

The proof of Proposition 3 is given in Section I.A.2. The next proposition further shows that the `kyle` model is also the only return covariance based model which satisfies all symmetry axioms.

Proposition 4. *A return covariance based cross-impact model* Λ *that is both split-invariant and rotation-invariant (Axioms 4 and 5) can always be written in the form*

$$\Lambda(\Sigma, \Omega) = \mathscr{L}^{-\top} U F(\mu) U^\top \mathscr{L}^{-1},$$

where

$$\Omega = \mathscr{L}\mathscr{L}^\top \quad ; \quad \hat{\Sigma} := \mathscr{L}^\top \Sigma \mathscr{L} \quad ; \quad U^\top \hat{\Sigma} U := \operatorname{diag}(\mu) \quad ; \quad F(\mu) := \Lambda(\operatorname{diag}(\mu), I).$$

Furthermore, if Λ *is cash-invariant and direct-invariant (Axioms 2 and 3), then* $F(\mu) = \operatorname{diag}(\mu)^{1/2}$ *up to a multiplicative constant and* $\Lambda = \Lambda_{\texttt{kyle}}$ *up to a multiplicative constant.*

The proof of Proposition 4 is given in Section I.A.2.

3.2 Response based models

All the models presented above assume that it is possible to relate the effect of the order flow imbalance solely with the return and order flow covariances. This section examines models which also use the response matrix R. First, we can define a response-based direct impact model similar to Equation (7).

Definition 5 (`r-direct` model). *The response direct (*`r-direct`*) model is defined, for any* $(\Sigma, \Omega, R) \in (\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{r-direct}}(\Sigma, \Omega, R) := \operatorname{diag}(R_{11}, \cdots, R_{dd}) \operatorname{diag}(\omega)^{-1}.$$

This model corresponds to the maximum likelihood estimator of the cross-impact matrix Λ under the constraint $\Lambda_{ij} = 0$ for $i \neq j$. Removing this constraint, one obtains the multiavariate maximum likelihood estimator defined below.

Definition 6 (`ml` model). *The maximum likelihood (*`ml`*) model is defined, for any* (Σ, Ω, R) *in* $(\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{ml}}(\Sigma, \Omega, R) := R\Omega^{-1}.$$

The `ml` does not satisfy desirable arbitrage or liquidity axioms. Thus, for similar reasons the `el` was introduced, we introduce a `r-el` model, so to have a response-based model satisfying more axioms while coinciding with the `ml` model when R and Ω commute.

Definition 7 (`r-el` model). *The response-based eigenliquidity (`r-el`) model is defined, for any* (Σ, Ω, R) *in* $(\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{r-el}}(\Sigma, \Omega, R) := \sum_a s_a \frac{s_a^\top R s_a}{s_a^\top \Omega s_a} s_a^\top, \tag{11}$$

where s_a *are the eigenvectors of* Σ*.*

Finally, we can replicate the construction of the `kyle` estimator in a response-based context to obtain the following model.

Definition 8 (`r-kyle` model). *The response-based Kyle (`r-kyle`) model is defined, for any* (Σ, Ω, R) *in* $(\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$*, as*

$$\Lambda_{\texttt{r-kyle}}(\Sigma, \Omega, R) := (\Omega^{-1/2})^\top \sqrt{(\Omega^{1/2})^\top R \Omega^{-1} R^\top \Omega^{1/2}} \Omega^{-1/2}. \tag{12}$$

3.3 The $\star$ transformation

Some of the models defined in the previous section (`whitening`, `el`, `r-el`) violate split invariance even though they are well-behaved under rotation. We can trade one for the other through the following transformation.

Definition 9 (The $\star$ transformation). *Given a cross-impact model* Λ*, the starred version of* Λ*, written* $\Lambda^\star$*, is a cross-impact model defined for any* (Σ, Ω, R) *in* $(\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$ *as*

$$\Lambda^\star(\Sigma, \Omega, R) := \mathrm{diag}(\sigma) \Lambda(\rho, \Omega^\star, R^\star) \mathrm{diag}(\sigma),$$

where we have defined $\Omega^\star = \mathrm{diag}(\sigma)\Omega\mathrm{diag}(\sigma)$ *and* $R^\star = \mathrm{diag}(\sigma)^{-1} R \mathrm{diag}(\sigma)$*.*

In practice, the starred version of a cross-impact model applies the original cross-impact model after rescaling all the observables in units of risk via a multiplication by the volatility σ. Naturally, this transformation has no effect on models that satisfy split invariance.

3.4 Axioms satisfied by each model

Table I.1 summarises the axioms satisfied by each model. Most results are straightforward and omitted for conciseness. We include some of the slightly less trivial proofs of the axioms satisfied by the `kyle` model in Section I.A.3. We summarise some of the connections between Axioms in Table I.2.

Model	Symmetries					Arbitrage		Fragmentation			Liquidity			Covariances
	PI	DI	CI	SI	RI	SA	DA	WFI	SSFI	SFI	WCS	SCS	SS	PCC
`direct`	✓	✓	✓	✓	✗	✓	✓	✗	✗	✗	✓	✓	✗	✗
`whitening`	✓	✓	✓	✗	✓	✗	✗	✓	✗	✗	✗	✗	✗	✓
`whitening⋆`	✓	✓	✓	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗	✓
`el`	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
`el⋆`	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✗
`kyle`	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
`r-direct`	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗	✓	✓	✗	✗
`ml`	✓	✓	✓	✓	✓	✗	✗	✓	✗	✗	✗	✗	✗	✗
`r-el`	✓	✓	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓	✓	✗
`r-el⋆`	✓	✓	✓	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓	✗
`r-kyle`	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗

Table I.1: **Summary of axioms satisfied by different cross-impact model.**
We use the symbol ✓ for axioms that are satisfied and ✗ for axioms that are violated. We use the color green in order to label a desirable property of the model, red for an undesirable property of the model. Yellow is used for properties/models whose violation might not be particularly relevant in order to explain empirical data, although they are interesting to consider. Axioms are grouped by category and the order in which they were presented in the text.

Result	Symmetries					Arbitrage		Fragmentation			Liquidity			Covariances
	PI	DI	CI	SI	RI	SA	DA	WFI	SSFI	SFI	WCS	SCS	SS	PCC
Proposition 1 (1)				H	H				H		✓			
Proposition 1 (2)				H	H					H		✓		
Proposition 2 (1)				H	H				✓		H			
Proposition 2 (2)				H	H					✓		H		
Proposition 3	✓	✓	✓	✓	✓	H	H	✓	✓	✓	✓	✓	✗	H
Proposition 4	✓	H	H	H	H	✓	✓	✓	✓	✓	✓	✓	✗	✓

Table I.2: **Salient relations among the axioms introduced in the chapter.**
The table summarises the results of different propositions relating axioms together. For a given result, we use the symbol H to denote a condition that holds by hypothesis. On the same row, we mark satisfied axioms using the notation of Table I.1.

4 Goodness-of-fit of cross-impact models

Sections 2 and 3 listed desirable properties of cross-impact models and examined which were satisfied by a variety of candidate models. This enabled us to understand the theoretical implications of a given cross-impact model. However, well-behaved models which poorly explain data are of little use. The goal of this section is to assess the goodness-of-fit of the

cross-impact models listed in Section 3 to understand which models satisfy desirable properties and fit data well.

4.1 Methodology

To assess goodness-of-fit, we select the timescale Δt to be one minute in order to avoid microstructural effects while being small. For a given cross-impact model Λ, the predicted price change for the time window $[t, t+\Delta t]$ due to the measured order flow imbalance q_t on that time window is

$$\widehat{\Delta p}_t := \Lambda(\Sigma_t, \Omega_t, R_t) q_t \,,$$

where Σ_t, Ω_t, R_t are the covariances defined in Assumption 2, which we will estimate using empirical data.

To evaluate quality of fit of the cross-impact model Λ, we compare the predicted price changes $\widehat{\Delta p}_t$ to the realised price changes Δp_t, using three different indicators of performance which emphasize different aspects of prediction errors. All three indicators are parametrized by a symmetric, positive definite matrix $M \in \mathcal{S}_n^+(\mathbb{R})$, $M \neq 0$. Given a realization of the price process $\{\Delta p_t\}_{t=1}^T$ of length T and a corresponding series of predictions $\{\widehat{\Delta p}_t\}_{t=1}^T$, the M-weighted generalized $\mathscr{R}^2$ is defined as

$$\mathscr{R}^2(M) := 1 - \frac{\sum_{1\leq t\leq T} (\Delta p_t - \widehat{\Delta p}_t)^\top M (\Delta p_t - \widehat{\Delta p}_t)}{\sum_{1\leq t\leq T} \Delta p_t^\top M \Delta p_t} \,.$$

The close the score is to one, the better the fit to empirical data. To highlight different sources of error, we consider the following choices of M:

1. $M = I_\sigma := \mathrm{diag}(\sigma)^{-1}$, to account for errors relative to the typical deviation of the asset considered. This type of error is relevant for strategies predicting idiosyncratic moves of the constituents of the basket, rather than strategies betting on correlated market moves.
2. $M = J_\sigma := (\Sigma_{ii}^{-1/2} \Sigma_{jj}^{-1/2})_{1\leq i,j\leq m}$, to check if the model successfully forecasts the overall direction of all assets. This is relevant for strategies predicting global moves of the constituents of the basket.
3. $M = \Sigma^{-1}$, to consider how well the model predicts the individual modes of the return covariance matrix. This would be the relevant error measure for strategies that place a constant amount of risk on the modes of the correlation matrix, leveraging up combinations of products with low volatility and scaling down market direction that exhibit large fluctuations. [3]

Given $M \in \mathcal{S}_n^+(\mathbb{R})$, $M \neq 0$, we compute scores on empirical data in the following manner. First, we divide data into two subsets of roughly equal length: data from 2016 on the one

[3] Note that this measure strongly penalizes models violating fragmentation invariance: errors along modes of zero risk should *a-priori* be enhanced by an infinite amount. In this study we have decided to clip the eigenvalues of Σ to a small, non-zero amount equal to 10^{-15}.

hand and in 2017 on the other hand. Given data from year X and year Y, we calibrate estimators and cross-impact models on year X and use models to predict price changes in year Y, writing $\mathscr{R}^2_{X\to Y}(M)$ for the average score. In-sample scores are defined as $\mathscr{R}^2_{\text{in}}(M) := \frac{1}{2}(\mathscr{R}^2_{2016\to 2016}(M) + \mathscr{R}^2_{2017\to 2017}(M))$ while out-of-sample scores are defined as $\mathscr{R}^2_{\text{out}}(M) := \frac{1}{2}(\mathscr{R}^2_{2016\to 2017}(M) + \mathscr{R}^2_{2017\to 2016}(M))$.

4.2 Data used

To assess goodness-of-fit in a variety of different conditions, we stress-test models on three different markets with different key characteristics. We detail each dataset here and the reason we chose them. Detailed descriptions of each dataset, the estimation of covariance matrices and of the cross-impact models is given in Section II.B.1.

The first dataset comprises two NYMEX Crude Oil future contracts and the corresponding Calendar Spread contract. The first two contracts (respectively, CRUDE0 and CRUDE1) entail an agreement to buy or sell 1000 barrels of oil either at the next month or at the subsequent month. The Calendar Spread CRUDE1_0 swaps the front month future with the contract settling on the following month. Because of the strong correlations among the two futures, the price of the calendar spread has very small fluctuations. This dataset allows us to test the importance of fragmentation axioms. Further details about this data are given in Section I.B.1.

While relevant to illustrate the importance of fragmentation invariance, the previous dataset corresponds to a pathological case where Σ has only one large non-zero eigenvalue, so that cross-impact models give similar results. To circumvent this issue, we look at 10-year US Treasury note futures and E-Mini S&P500 futures. We collect data from the Chicago Mercantile Exchange and use the first two upcoming maturities of both contracts (respectively called SPMINI and SPMINI3 for E-Mini S&P500 futures and 10USNOTE and 10USNOTE3 for 10-year US treasury notes). Further details about this data are given in Section I.B.2.

The previous datasets give us no clear conclusion on the role of stability axioms. In both examples illiquid assets were highly correlated to other liquid assets. This extreme regime of correlations makes it harder to analyse the role of liquidity. To circumvent this issue, we study the behavior of cross-impact models in the low-correlation, many assets regime, using stocks data. Further details about this data are given in Section I.B.3.

4.3 Goodness-of-fit

The goodness-of-fit results for each model, dataset and score are presented in Table I.4 in Section I.C. Overall, on all datasets, the cross-impact models with the best goodness-of-fit are the `r-el`, `kyle` and `ml` models. They significantly outperform models which do not account for cross-sectional effects, such as the `r-direct` model. In high-correlation regimes, such as on the Crude and Bonds and Indices datasets, this gap is more pronounced. Among these models, it is remarkable that the `kyle` model satisfies all axioms and achieves comparable

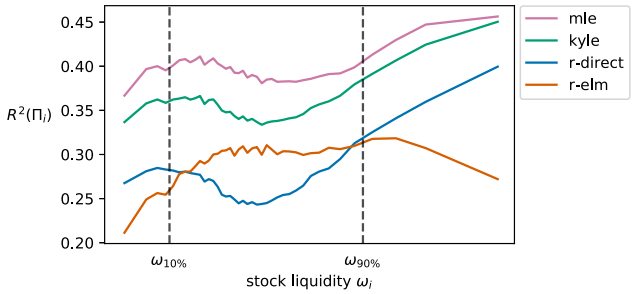


Figure I.1: **Idiosyncratic scores as a function of liquidity.**
For each stock in our dataset, we compute the the in-sample stock-specific scores $\mathscr{R}^2(\Pi_i)$ scores on 2016 data. We then represent the average in-sample stock-specific score as a function of the liquidity ω_i, binning data by ω_i to smooth out noise. Results for the `ml` (in pink), `kyle` (in green), `r-direct` (in blue) and `r-el` (in orange) models are shown. We have further indicated the 10% and 90% quantiles of liquidity $\omega_{10\%}$ and $\omega_{90\%}$.

performance to the `ml` model, which maximises empirical fit but has issues related to arbitrage.

Given these results, we focus on analysing the influence of certain market parameters on the goodness-of-fit of cross-impact models. The next section examines the role of the liquidity.

4.4 Goodness-of-fit relative to liquidity

An interesting feature of our stocks dataset is the heterogeneous liquidity among assets. This allows us to explore the influence of the liquidity of a given stock on the performance of different models. Figure I.1 shows the results of this analysis. Consistent with Table I.4, we find that overall, in score terms, `ml>kyle>r-direct>r-el`. The `r-direct` model fares better for liquid stocks, where a larger fraction of variance can be explained by same-stock trades. Surprisingly, the same holds for `ml` and `kyle` models. The `r-el` model stands as an exception. It better explains price moves for stocks which are within the band of typical liquidity, between $\omega_{10\%}$ and $\omega_{90\%}$. This makes sense since the `r-el` model is self-stable as it aggregates liquidity of all stocks. Thus, though this assumption is justified for stocks of liquidity close to the average, it is violated outside of this zone. The `ml` and `kyle` models are not self-stable and better deal with very liquid or illiquid stocks. To further reinforce this point, for stocks of liquidity close to the average in our pool of stocks, the difference scores of the `el` and `kyle` models reach a minimum. This is consistent with the fact that in the approximation $\Omega \approx \omega_{50\%}^2 I$, the two models coincide. Thus, violating self-stability (Axiom 13) is key to explain price changes for all ranges of liquidity within a basket of instruments.

4.5 Robustness of goodness-of-fit

The previous section compared the descriptive power of different cross-impact models. However, robustness of the different cross-impact models is also of interest. In Figure I.2, we show

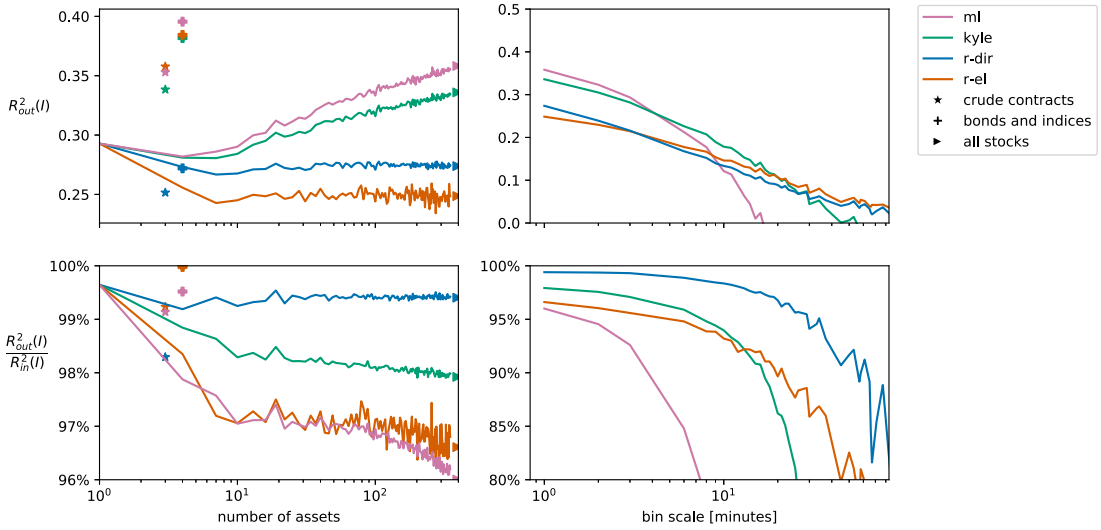


Figure I.2: **Idiosyncratic score and overfitting as a function of the number of assets and bin timescale.**
Left column: average out-of-sample idiosyncratic score $\mathscr{R}^2_{\text{out}}(I_\sigma)$ (top left) and overfitting coefficient $\frac{\mathscr{R}^2_{\text{out}}(I_\sigma)}{\mathscr{R}^2_{\text{in}}(I_\sigma)}$ (bottom left) computed using stocks data. Out-of-sample and in-sample scores were computed by randomly selecting a subset of stocks and computing scores on the given subset, repeating the procedure more when there are fewer stocks are selected than when a large proportion of stocks from our sample is considered. The average score for each models across all samples is then shown. Scores are shown for the `ml` (in pink), the `kyle` (in green), `r-direct` (in blue) and `r-el` (in orange). Stars show results for crude contracts, crosses for bonds and indices and triangles for all 393 stocks of our sample. **Right column**: idiosyncratic scores (top right) and overfitting coefficient (bottom right) as a function of the bin timescale. Scores were computed using the same procedure described in Section I.C, varying the bin parameter from 10 seconds up to around an hour.

the out-of-sample score and overfitting coefficient for idiosyncratic price changes for our set of 393 stocks, as a function of the bin timescale and number of instruments.

As expected, the number of degrees of freedom controls the overfitting of different models. This explains why, in terms of overfitting with respect to the number of instruments at the minute timescale, `r-direct` < `kyle` < `ml` ≈ `r-el`. In contrast, models overfit less on futures, which suggests that overfitting decreases as the pairwise correlation between instruments increases. Furthermore, out-of-sample idiosyncratic scores for the `ml` and `kyle` model increase with the number of assets. A somewhat surprising result, despite the small pairwise correlation of instruments in our stock dataset and the large number of stocks considered in this study, is that idiosyncratic scores appear to keep increasing for more than 400 assets. This suggests that there is still latent explanatory power in the dataset but only two models manage to extract it.

Focusing on the influence of the bin timescale, there is little overfitting at the minute timescale but it increases with the bin timescale. In particular, the good fit of the `ml` at small timescales quickly breaks down for larger timescales. On the other hand, both the `r-el` and `kyle` models are quite robust up until the 10 minute timescale. At this timescale, we expect our static approximation described in Section 1 to break down.

Conclusion

Let us summarize what we have achieved. Our main objective was to find suitable static cross-impact models given a set of empirical observations encoded in the sufficient statistics (Σ, Ω, R) which fit data well and led to well-behaved market dynamics. To do so, we introduced axioms, desirable properties of cross-impact models. We classified existing cross-impact models and characterised those which satisfy certain subset of axioms.

In all markets studied, our analysis confirms that cross-impact models are well suited to explain price changes, showing significant improvement compared to impact models in which cross-sectional effects are disregarded (see Table I.4). However, only the `kyle` and `ml` models perform well on all markets studied, whereas only the `kyle` model prevents arbitrage and is well-behaved when trading both liquid and illiquid instruments. This makes it an ideal model for other applications, such as optimal execution.

Independently of our specific model implementations, we also established a few characterisation results of cross-impact models from axioms. In particular, we showed that symmetry axioms (Axioms 1 to 5) alone completely characterise return-based cross-impact models and that the `kyle` model is the only model to satisfy all cross-impact axioms.

Even though we have considered a linear, single-period scenario, the ideas introduced in this chapter could be generalised. For instance, the framework can be adapted to deal with derivatives [TMB21]. Another topic is the generalisation of this framework to account for the auto-correlation of the order flow. This question is examined in [RT21b, RT21a] but an axiom-first approach is still lacking.

Finally, our results can be used to choose adequate cross-impact models for applications discussed in the literature. For optimal trading applications [ELMK18, EMK19, GP16, LM19], it is natural to favor models which satisfy arbitrage axioms to prevent ill-behaved trading strategies. When modeling cross-impact at the microstructural level [TWG19] then arbitrage, fragmentation and liquidity axioms are all important to rule out price manipulation. For each domain, we highlighted which cross-impact models would be good candidates.

I.A Proofs

This section contains proofs of some results stated in the main text.

I.A.1 Proof of Propositions 1 and 2

In this section, we establish some links between fragmentation and liquidity axioms. To do so, in the rest of this section, we will make use of two kinds of regularised covariance and response matrices. Given a linear subspace V, we first introduce the order flow regularised estimators:

$$\begin{aligned}\Sigma_\varepsilon^q &:= \Sigma\\ \Omega_\varepsilon^q &:= (\bar{\Pi}_V + \varepsilon\Pi_V)\Omega(\bar{\Pi}_V + \varepsilon\Pi_V)\\ R_\varepsilon^q &:= R(\bar{\Pi}_V + \varepsilon\Pi_V)\,.\end{aligned}$$

These correspond to the multiplication of liquidity of instruments in V by ε. Similarly, we introduce the price regularised estimators:

$$\begin{aligned}\Sigma_\varepsilon^p &:= (\bar{\Pi}_V + \varepsilon\Pi_V)\Sigma(\bar{\Pi}_V + \varepsilon\Pi_V)\\ \Omega_\varepsilon^p &:= \Omega\\ R_\varepsilon^p &:= (\bar{\Pi}_V + \varepsilon\Pi_V)R\,.\end{aligned}$$

These correspond to the multiplication of price fluctuations of instruments in V by ε. We begin with a convenient lemma that relates liquidity properties to fragmentation properties.

Lemma 1. *Let Λ be a split-invariant and rotation-invariant (Axioms 4 and 5) cross-impact model and a linear subspace V such that $\emptyset \subset V \subseteq \mathbb{R}^n$. Then, for all $\varepsilon > 0$, we have*

$$\begin{aligned}\Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) =& \bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V\\ &+ \varepsilon^{-1}\left[\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\Pi_V + \Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V\right]\\ &+ \varepsilon^{-2}\Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\Pi_V\,.\end{aligned}$$

Proof. Let $(u_1, \cdots, u_k)$ an orthonormal basis for the linear subspace V, where $k = \dim(V)$ and $(u_1, \cdots, u_d)$ a completed orthonormal basis on $\mathbb{R}^d$. We write $A := (u_1 \mid u_2 \mid \cdots \mid u_k) \in \mathscr{M}_{d,k}(\mathbb{R})$ and $U := (u_1 \mid u_2 \mid \cdots \mid u_d) \in \mathscr{M}_{d,d}(\mathbb{R})$. Then $\Pi_V = AA^\top$ and $\bar{\Pi}_V = I - AA^\top$. By rotation invariance, we have

$$U\Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)U^\top = \Lambda(U\Sigma_\varepsilon^q U^\top, U\Omega_\varepsilon^q U^\top, UR_\varepsilon^q U^\top)\,.$$

Since $(u_1, \cdots, u_d)$ is an orthonormal basis, UA only has non-zero entries along the diagonal. Writing $\widehat{A} := UA$ we can apply split invariance with $D = (I - \widehat{A}\widehat{A}^\top + \varepsilon\widehat{A}\widehat{A}^\top)$ to obtain

$$U\Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)U^\top = D\Lambda(D^{-1}U\Sigma_\varepsilon^q U^\top D^{-1}, DU\Omega_\varepsilon^q U^\top D, D^{-1}UR_\varepsilon^q U^\top D)D\,.$$

Straightforward computations show that

$$D^{-1}U\Sigma_\varepsilon^q U^\top D^{-1} = U\Sigma_\varepsilon^p U^\top \quad DU\Omega_\varepsilon^q U^\top D = U\Omega_\varepsilon^p U^\top \quad D^{-1}UR_\varepsilon^q U^\top D = UR_\varepsilon^p U^\top .$$

Therefore

$$U\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)U^\top = (I-\widehat{A}\widehat{A}^\top+\varepsilon^{-1}\widehat{A}\widehat{A}^\top)\Lambda\left(U\Sigma_\varepsilon^p U^\top, U\Omega_\varepsilon^p U^\top, UR_\varepsilon^p U^\top\right)(I-\widehat{A}\widehat{A}^\top+\varepsilon^{-1}\widehat{A}\widehat{A}^\top).$$

Applying rotational invariance once again we get

$$\begin{aligned}U\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)U^\top =& U(I-AA^\top)\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)(I-AA^\top)U^\top\\ &+\varepsilon^{-1}U\left[(I-AA^\top)\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)AA^\top + AA^\top\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)(I-AA^\top)\right]U^\top\\ &+\varepsilon^{-2}UAA^\top\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)AA^\top U^\top .\end{aligned}$$

This finally yields

$$\begin{aligned}\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q) =& \bar{\Pi}_V\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)\bar{\Pi}_V\\ &+\varepsilon^{-1}\left[\bar{\Pi}_V\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)\Pi_V+\Pi_V\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)\bar{\Pi}_V\right]\\ &+\varepsilon^{-2}\Pi_V\Lambda\left(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p\right)\Pi_V,\end{aligned}$$

which concludes the proof. □

In a similar fashion as Lemma 1, one can prove the following Lemma.

Lemma 2. *Let Λ be a split-invariant and rotation-invariant (Axioms 4 and 5) cross-impact model and a subspace V such that $\emptyset \subset V \subseteq \mathbb{R}^n$. Then we have*

$$\begin{aligned}\Lambda(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p) =& \bar{\Pi}_V\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)\bar{\Pi}_V\\ &+\varepsilon^{-1}\left[\bar{\Pi}_V\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)\Pi_V+\Pi_V\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)\bar{\Pi}_V\right]\\ &+\varepsilon^{-2}\Pi_V\Lambda(\Sigma_\varepsilon^q,\Omega_\varepsilon^q,R_\varepsilon^q)\Pi_V.\end{aligned}$$

Lemmas 1 and 2 enable us to relate cross-stability to fragmentation invariance. This is the topic of the next proposition.

Proposition 1. *Let Λ be a jointly continuous cross-impact model which satisfies split and rotational invariance (Axioms 4 and 5). Then*

1. *If Λ satisfies semi-strong fragmentation invariance (Axiom 9), then it is weakly cross-stable (Axiom 11).*
2. *If Λ is strongly fragmentation invariant (Axiom 10), then it is strongly cross-stable (Axiom 12).*

Proof. We first prove (1). Since the cross-impact model Λ is continuous and satisfies semi-strong fragmentation invariance we have

$$\begin{aligned}\Lambda(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p)\Pi_V &\underset{\varepsilon\to 0}{\to} 0\\ \Pi_V\Lambda(\Sigma_\varepsilon^p,\Omega_\varepsilon^p,R_\varepsilon^p) &\underset{\varepsilon\to 0}{\to} 0.\end{aligned}$$

Plugging the above in the results of Lemma 2 yields

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\Pi_V = \varepsilon^{-1}\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V \underset{\varepsilon\to 0}{\to} 0$$
$$\Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V = \varepsilon^{-1}\Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} 0\,.$$

Thus Λ is weakly cross-stable. We now prove (2). Continuity at $\varepsilon = 0$ and strong fragmentation invariance yield

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} \bar{\Pi}_V \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V)\bar{\Pi}_V\,.$$

Plugging the above into the results of Lemma 2 gives

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V = \bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} \bar{\Pi}_V \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V)\bar{\Pi}_V\,.$$

This implies that Λ is strongly cross-stable. □

Interestingly, the converse of Proposition 1 does not hold, thus indicating that the fragmentation invariance properties play a more fundamental role than liquidity related axioms. The next proposition shows the converse, provided some additional regularity of the cross-impact model.

Proposition 2. *Let Λ be a jointly continuous cross-impact model which satisfies split and rotational invariance (Axioms 4 and 5). We further assume that, for every linear subspace V and using the previous notations, $\varepsilon^2 \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) \underset{\varepsilon\to 0}{\to} 0$. Then*

1. *If Λ is weakly cross-stable (Axiom 11), then it satisfies semi-strong fragmentation invariance (Axiom 9).*
2. *If Λ is strongly cross-stable (Axiom 12), then it is strongly fragmentation invariant (Axiom 10).*

Proof. We first prove (1). Since the cross-impact model Λ is weakly cross-stable we have

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V \underset{\varepsilon\to 0}{\to} 0$$
$$\Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} 0\,.$$

Furthermore, by assumption we have

$$\varepsilon^2 \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) \underset{\varepsilon\to 0}{\to} 0.$$

Plugging the above in the results of Lemma 1 yields

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V = \varepsilon^{-1}\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\Pi_V \underset{\varepsilon\to 0}{\to} 0$$
$$\Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V = \varepsilon^{-1}\Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} 0$$
$$\varepsilon^2 \Pi_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V = \Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} 0\,.$$

Combining the above and using continuity, we obtain

$$\Pi_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p) \underset{\varepsilon\to 0}{\to} 0 = \Pi_V \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \Omega, \bar{\Pi}_V R)$$
$$\Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\Pi_V \underset{\varepsilon\to 0}{\to} 0 = \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \Omega, \bar{\Pi}_V R)\Pi_V\,.$$

Thus this proves that Λ is semi-strongly fragmentation invariant. We now prove (2). Continuity at $\varepsilon = 0$ and strong cross-stability yield

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} \bar{\Pi}_V \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V)\bar{\Pi}_V\,.$$

Plugging the above into the results of Lemma 1 gives

$$\bar{\Pi}_V \Lambda(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V = \bar{\Pi}_V \Lambda(\Sigma_\varepsilon^p, \Omega_\varepsilon^p, R_\varepsilon^p)\bar{\Pi}_V \underset{\varepsilon\to 0}{\to} \bar{\Pi}_V \Lambda(\bar{\Pi}_V \Sigma \bar{\Pi}_V, \bar{\Pi}_V \Omega \bar{\Pi}_V, \bar{\Pi}_V R \bar{\Pi}_V)\bar{\Pi}_V\,.$$

This implies that Λ is strongly fragmentation invariant. □

Propositions 1 and 2 show that fragmentation and cross-stability axioms are related. Furthermore, for cross-impact models which satisfy the regularity property of Proposition 2, the two sets of axioms are equivalent.

I.A.2 Proof of Propositions 3 and 4

In this section, we characterise the models which satisfy the axioms introduced in Section 2. We begin with the following proposition, the proof of which is heavily inspired by [CK94, GdMMBB20].

Proposition 3. *Let Λ be a symmetric, positive-semidefinite and return covariance consistent cross-impact model (Axioms 6, 7 and 14). Then $\Lambda = \Lambda_{\texttt{kyle}}$ up to a multiplicative constant.*

Proof. Let Λ be a cross-impact model which satisfies Axioms 6 and 14 and $(\Sigma, \Omega, R) \in (\mathcal{S}_n^+(\mathbb{R}) \times \mathcal{S}_n^{++}(\mathbb{R}) \times \mathcal{M}_n(\mathbb{R}))$. We assume for convenience that the multiplicative constant in Axiom 14 is one. Writing Λ for $\Lambda(\Sigma, \Omega, R)$, and $\mathscr{L}$ for a matrix such that $\Omega = \mathscr{L}\mathscr{L}^\top$, we have

$$\Sigma = \Lambda\Omega\Lambda^\top = \Lambda\mathscr{L}\mathscr{L}^\top\Lambda^\top = (\Lambda\mathscr{L})(\Lambda\mathscr{L})^\top.$$

Thus, by unicity up to a rotation of the square root decomposition, writing $\mathscr{G}$ for a matrix such that $\Sigma = \mathscr{G}\mathscr{G}^\top$, there exists a rotation O such that $\Lambda = \mathscr{G}O\mathscr{L}^{-1}$. Furthermore, since Λ is symmetric,

$$\mathscr{G}O\mathscr{L}^{-1} = (\mathscr{G}O\mathscr{L}^{-1})^\top.$$

Rewriting, we find

$$\mathscr{L}^\top\mathscr{G}O = O^\top\mathscr{G}^\top\mathscr{L},$$

so that the matrix $\mathscr{L}^\top\mathscr{G}O$ is symmetric and satisfies

$$(\mathscr{L}^\top\mathscr{G}O)(\mathscr{L}^\top\mathscr{G}O)^\top = (\mathscr{L}\mathscr{G})(\mathscr{L}^\top\mathscr{G})^\top\,.$$

Since $(\mathscr{L}^\top \mathscr{G})(\mathscr{L}^\top \mathscr{G})^\top$ is symmetric positive semi-definite, the symmetric square root is unique and

$$\mathscr{L}^\top \mathscr{G} O = \sqrt{(\mathscr{L}^\top \mathscr{G})(\mathscr{L}^\top \mathscr{G})^\top}.$$

Plugging this back into the expression of the cross-impact matrix yields the result:

$$\Lambda = \mathscr{G} O \mathscr{L}^{-1} = \mathscr{L}^{-\top} \sqrt{(\mathscr{L}^\top \mathscr{G})(\mathscr{L}^\top \mathscr{G})^\top}\, \mathscr{L}^{-1} = \mathscr{L}^{-\top} \sqrt{\mathscr{L}^\top \Sigma \mathscr{L}}\, \mathscr{L}^{-1}.$$

□

Hence, there is a single symmetric, positive-semidefinite, covariance-consistent, cross-impact model. Given that the fragmentation-related axioms seem so fundamental, one might wonder how many models one can build that satisfy that family of properties. Surprisingly, we find that the class of models enjoying both split invariance and rotational invariance is quite small, as shown in the next lemma.

Lemma 3. *Let Λ be a cross-impact model which satisfies Axioms 4 and 5. Then, for all $(\Sigma, \Omega, R) \in (\mathscr{S}_n^+(\mathbb{R}) \times \mathscr{S}_n^{++}(\mathbb{R}) \times \mathscr{M}_n(\mathbb{R}))$, it can be written as*

$$\Lambda(\Sigma, \Omega, R) = \mathscr{L}^{-\top} U \Lambda(U^\top \hat{\Sigma} U, I, U^\top \hat{R} U) U^\top \mathscr{L}^{-1},$$

where

$$\begin{aligned} \Omega &= \mathscr{L}\mathscr{L}^\top \\ \hat{\Sigma} &= \mathscr{L}^\top \Sigma \mathscr{L} \\ \hat{R} &= \mathscr{L}^\top R \mathscr{L}^{-\top} \end{aligned}$$

and U is an orthogonal matrix (i.e., $UU^\top = I$).

Proof. The lemma is obtained by applying sequentially rotational invariance, split invariance and again rotational invariance. The first two transformations can be used in order to remove the dependency in Ω as the second argument of the $\Lambda(\Sigma, \Omega, R)$ function. □

When one discards the influence of the response matrix, the model can further be characterised as shown by the next proposition.

Proposition 4. *A return covariance based cross-impact model Λ that is both split-invariant and rotation-invariant (Axioms 4 and 5) can always be written in the form*

$$\Lambda(\Sigma, \Omega) = \mathscr{L}^{-\top} U F(\mu) U^\top \mathscr{L}^{-1},$$

where

$$\Omega = \mathscr{L}\mathscr{L}^\top \quad ; \quad \hat{\Sigma} := \mathscr{L}^\top \Sigma \mathscr{L} \quad ; \quad U^\top \hat{\Sigma} U := \operatorname{diag}(\mu) \quad ; \quad F(\mu) := \Lambda(\operatorname{diag}(\mu), I).$$

Furthermore, if Λ is cash-invariant and direct-invariant Axioms 2 and 3, then $F(\mu) \propto \operatorname{diag}(\mu)^{1/2}$ and $\Lambda = \Lambda_{\texttt{kyle}}$ up to a multiplicative constant.

Proof. For a return covariance based model, we can simply choose from Lemma 3 to fix U as the rotation that diagonalizes the symmetric matrix $\hat{\Sigma}$, obtaining:

$$U^\top \hat{\Sigma} U = \mathrm{diag}(\mu)\,.$$

This choice implies

$$\Lambda(\Sigma, \Omega) = \mathscr{L}^{-\top} U \Lambda(\mathrm{diag}(\mu), I) U^\top \mathscr{L}^{-1},$$

which yields the result of the first part of the proposition. Furthermore, if we assume Λ is cash-invariant and direct-invariant (Axioms 2 and 3), we have

$$\Lambda(\mathrm{diag}(\mu), I) = \sum_{i=1}^{d} \sqrt{\mu_i}\, \Lambda(e_i e_i^\top, e_i\,, e_i^\top)$$

which yields the `kyle` model up to a constant. □

The above shows that the only return-based cross-impact model which satisfies all symmetry axioms Axioms 1 to 5 is the `kyle` model.

I.A.3 Proof of important properties of the `kyle` model

This section is dedicated to showing that the `kyle` model satisfies all the axioms outlined in section Section 2. As the fragmentation and invariance axioms were discussed in the previous section, the next lemma shows that the `kyle` model is also cross-stable.

Lemma 4. *The `kyle` model is strongly cross-stable in the sense of Axioms 12 and 13 and is not self-stable in the sense of Axiom 13.*

Proof. Let V be a linear subspace of $\mathbb{R}^n$) and $\varepsilon > 0$. Note that, writing $\mathscr{G}$ for a matrix such that $\mathscr{G}\mathscr{G}^\top = \Sigma$, for any matrix $\mathscr{L}_\varepsilon$ such that $\mathscr{L}_\varepsilon \mathscr{L}_\varepsilon^\top = \Omega_\varepsilon^q$, we previously showed that there exists a rotation matrix $O_\varepsilon = (\mathscr{L}_\varepsilon^\top \mathscr{G})^{-1} \sqrt{(\mathscr{L}_\varepsilon^\top \mathscr{G})(\mathscr{L}_\varepsilon^\top \mathscr{G})^\top}$ such that we have

$$\Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) = \mathscr{G} O_\varepsilon \mathscr{L}_\varepsilon^{-1}.$$

However, $\Omega_\varepsilon^q = (\bar{\Pi}_V + \varepsilon \Pi_V)\Omega(\bar{\Pi}_V + \varepsilon\Pi_V) = (\bar{\Pi}_V + \varepsilon\Pi_V)\mathscr{L}\mathscr{L}^\top(\bar{\Pi}_V + \varepsilon\Pi_V) = [(\bar{\Pi}_V + \varepsilon\Pi_V)\mathscr{L}][(\bar{\Pi}_V + \varepsilon\Pi_V)\mathscr{L}]^\top$. Thus,

$$\begin{aligned}
\Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) &= \mathscr{G} O_\varepsilon [(\bar{\Pi}_V + \varepsilon \Pi_V)\mathscr{L}]^{-1} \\
&= \mathscr{G} O_\varepsilon \mathscr{L}^{-1} (\bar{\Pi}_V + \frac{1}{\varepsilon}\Pi_V) \\
&= \mathscr{G} O_\varepsilon \mathscr{L}^{-1} \bar{\Pi}_V + \frac{1}{\varepsilon} \mathscr{G} O_\varepsilon \mathscr{L}^{-1} \Pi_V.
\end{aligned}$$

Using the symmetry of the `kyle` model, the above yields:

$$\Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q) = \bar{\Pi}_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top + \frac{1}{\varepsilon} \Pi_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top.$$

Thus, we have

$$\bar{\Pi}_V \Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V = \bar{\Pi}_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top \Pi_V$$
$$\Pi_V \Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V = \Pi_V \mathscr{G} O_\varepsilon \mathscr{L}^{-1} \bar{\Pi}_V .$$

Since $O_\varepsilon^\top$ is an orthogonal matrix, we have

$$\bar{\Pi}_V \Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\Pi_V \underset{\varepsilon\to 0}{=} O(1)$$
$$\Pi_V \Lambda_{\texttt{kyle}}(\Sigma_\varepsilon^q, \Omega_\varepsilon^q, R_\varepsilon^q)\bar{\Pi}_V \underset{\varepsilon\to 0}{=} O(1),$$

which proves that `kyle` satisfies Axiom 11. Furthermore,

$$\Pi_V \Lambda_{\texttt{kyle}} \Pi_V = \frac{1}{\varepsilon} \Pi_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top \Pi_V,$$

so that unless $\Pi_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top \Pi_V = 0$, we have:

$$||\Pi_V \Lambda_{\texttt{kyle}} \Pi_V|| = \varepsilon^{-1} ||\Pi_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top \Pi_V|| \underset{\varepsilon\to 0}{\to} \infty.$$

Choosing diagonal Σ and Ω such that $\Pi_V \mathscr{L} \neq 0$ and $\mathscr{G}\Pi_V \neq 0$, we see that $\Pi_V \mathscr{L}^{-\top} O_\varepsilon^\top \mathscr{G}^\top \Pi_V = 0$ cannot hold for all Σ, Ω. This shows that `kyle` does not satisfy Axiom 13. Finally, notice that by using Lemma 3 one can make Ω appear only in the combination $\mathscr{L}^\top \Sigma \mathscr{L}$, which is insensitive to the components of Ω belonging to the kernel of Σ, which proves strong cross-stability (Axiom 12). □

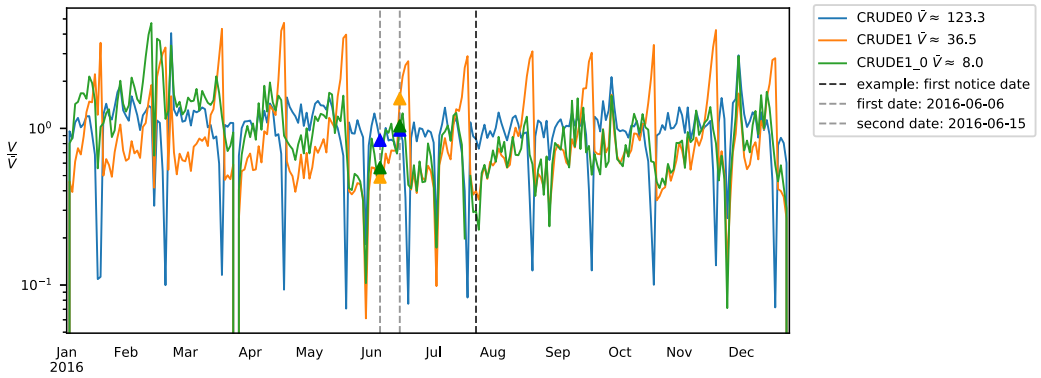


Figure I.3: **Number of traded NYMEX Crude oil futures and Calendar Spread contracts (in thousands) relative to daily number of traded contracts.**
The number of contracts sold relative to the daily average is shown for the front month contract (in blue), the subsequent month (in orange) and the Calendar Spread (in green). The average number of traded NYMEX Crude oil futures and Calendar Spread contracts $\bar{V}$ over 2016 is shown in the upper right corner. Vertical dashed lines show specific dates. An example of first notice date for the front month contract is shown in bold black. After the first notice date, holders of the future contract may ask for physical delivery of the underlying. We also show two dates away from a first notice date: the 6th and 15th of June 2016. Colored triangles show the relative number of contracts exchanged on these dates. Note that the number of contracts is represented in thousands and was not adjusted by the basis point, so that the underlying of each contract is 1000 barrels of oil.

I.B Data

This appendix contains details on the datasets and processing used to apply the different models.

I.B.1 Crude contracts

Description of the dataset We collected trades and quotes data from January 2016 to December 2017, between 9:30AM to 7:30PM UTC, where most of the trading takes place in our dataset, removing 30 minutes around the opening of trading hours to mitigate intraday seasonality. After filtering and processing, we have a total of 430 days in our sample (237 in 2016 and 193 in 2017). We highlight below two important features of our pre-processing for the estimation of Σ, Ω and R.

Pre-processing: accounting for non-stationarity Overall, the front month contract CRUDE0 is by far the most liquid, followed by the subsequent month contract CRUDE1 and the calendar spread CRUDE1_0. However, there are strong seasonal dependencies which are shown in Figure I.3. For example, the subsequent month contract becomes more liquid as

one approaches the maturity of the front month contract. Global estimators of Σ, Ω and R would thus be biased by this varying liquidity ω (σ also appears to follow a non-stationary pattern, but is not shown here). Thus, we used local (daily) estimators of price volatility σ_t and liquidity ω_t, and built local covariance estimators Σ_t and Ω_t by assuming stationarity of the correlations $\varrho = \mathrm{diag}(\sigma_t)^{-1}\Sigma_t\mathrm{diag}(\sigma_t)^{-1}$ and $\varrho_\Omega = \mathrm{diag}(\omega_t)^{-1}\Omega_t\mathrm{diag}(\omega_t)^{-1}$. We estimate volatility and liquidity with a simple standard deviation: $\sigma^2_{i,t} = \langle \Delta p^2_{i,t} \rangle$ and $\omega^2_{i,t} = \langle q^2_{i,t} \rangle$, where the average $\langle \cdot \rangle$ is computed using data on day t.

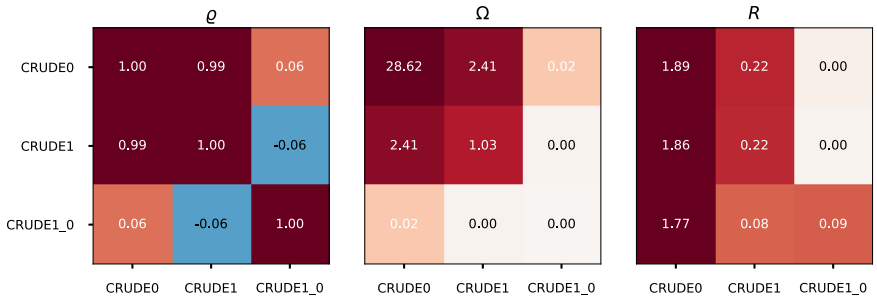


Figure I.4: **Estimates of ϱ, Ω and R for Crude contracts (in MUSD).**
The return correlation matrix ρ (left), order flow covariance matrix Ω (center) and response matrix R (right) were estimated using 2016 data and computed on the 6th of June 2016. This date represents the typical behavior of these contracts far away from the first notice date, before rolling effects become relevant. To highlight the amount of notional traded, order flow is reported in millions of exchanged dollars according to the average value of each contract on the 6th of June 2016. Though non-nill, order flow covariance of Calendar Spread thus appears small because traded notional is much smaller than on each leg of the futures contract.

Structure of ρ, Ω and R Figure I.4 reports the estimators of ϱ, Ω and R for the 6th of June 2016. The figure shows ϱ has one zero-volatility direction and one direction of very small fluctuations. Thus models which satisfy fragmentation invariance (Axioms 8 to 10) should give better predictions. On the other hand, Ω highlights the difference in liquidity of our assets. Thus, we should be cautious of models which do not satisfy stability axioms. Indeed, these will not penalize trading directions of small liquidity.

Pre-processing: cleaning estimators As illustrated in Figure I.4, where the structure of Σ, Ω and R are shown for a typical day, one can appreciate that the correlation between the two future contracts CRUDE0 and CRUDE1 is close to one, whereas the correlation with the Calendar Spread contract is very small, due to the small volatility of the fluctuations along the relative mode. Because of these effects, the sign of the Calendar Spread correlations with CRUDE0 and CRUDE1 is non-trivial to estimate: due to microstructural effects, the measured correlation is dominated by tick-size related effects [4]. In fact, empirical price changes of the

[4]To test this hypothesis, we estimated the empirical smallest eigenvalue of the covariance matrix for multiple futures contract as a function of relative tick size (not shown). If price changes of the Calendar Spread were

Calendar Spread are not given by the difference of price changes of the legs. To solve this issue, we impose the price changes of the Calendar Spread according to the price changes of the futures contracts.

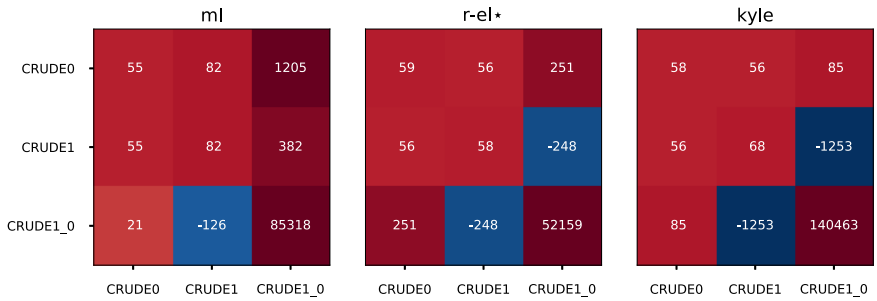


Figure I.5: **Values of different cross-impact models for Crude contracts.**
We report the values of the `ml` (left), `r-el⋆` (center) and `kyle` (right) cross-impact models for the covariances of the 6th of June 2016. Units are chosen to represent the relative price change in basis points (10^{-4} of the asset price) by hundred million USD worth of contract traded.

Cross-impact models for Crude oil contracts Figure I.5 shows the calibrated `ml`, `r-el⋆` and `kyle` models. Each satisfies weak fragmentation invariance (Axiom 8). Therefore, they prevent arbitrage which would trade the physical Calendar Spread contract against the synthetic Calendar Spread (made up of CRUDE0 and CRUDE1). However, `ml` and `kyle` are not self-stable (Axiom 13) while the `r-el⋆` model is. This explains why impact from trading the illiquid Calendar Spread is much larger in the `ml` and `kyle` models than in the `r-el⋆` model.

I.B.2 Bonds and indices

Description of the dataset We look at 10-year US Treasury note futures and the E-MINI futures. We collect data from the Chicago Mercantile Exchange and use the first two upcoming maturities of both contracts (respectively called SPMINI and SPMINI3 for E-MINI contracts and 10USNOTE and 10USNOTE3 for 10-year US treasury notes). E-Mini futures are quarterly, financially settled contracts with maturities in March, June, September and December. At expiry, the final settlement price of E-MINI futures is a proxy for the S&P500 index using the opening prices of the underlying stocks belonging to the index. Similarly, the 10-year treasury note futures are quaterly, financially settled contracts with maturities in March, June, September and December. At expiry, the final settlement price is volume weighted average

given by the legs of the contract, this eigenvalue should be equal to zero. However, we found that as the tick size increases, so does the smallest eigenvalue away from zero. This thus validates our hypothesis and justifies the need for additional processing of futures data.

price of past trades on the underlying treasury note.[5] We collected trades and quotes data from January 2016 to December 2017, between 9AM to 7PM UTC, where most of the trading takes place in our dataset. After filtering days for which data for one product was missing, we keep a total of 160 days (75 in 2016 and 85 in 2017). We highlight below one important pre-processing step for the estimation of Σ, Ω and R.

Pre-processing: accounting for non-stationarity The same non-stationary behavior observed for Crude Oil futures contract is observed here. Thus we adopt the same estimation procedure for the local covariance estimators Σ_t and Ω_t by assuming stationarity of the correlations $\varrho = \mathrm{diag}(\sigma_t)^{-1}\Sigma_t\mathrm{diag}(\sigma_t)^{-1}$ and $\varrho_\Omega = \mathrm{diag}(\omega_t)^{-1}\Omega_t\mathrm{diag}(\omega_t)^{-1}$.

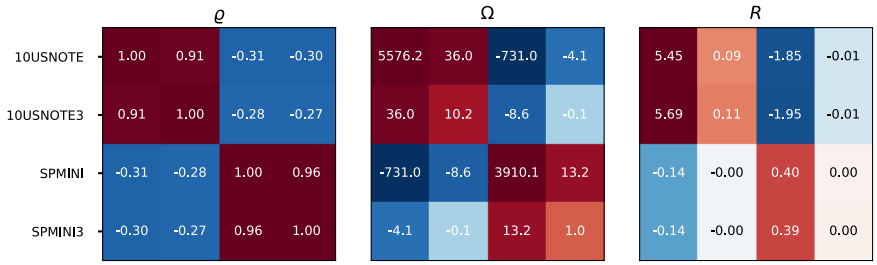


Figure I.6: **Estimates of ϱ, Ω and R for bonds and indices (in MUSD).**
The return correlation matrix ρ (left), order flow covariance matrix Ω (center) and response matrix R (right) were estimated using 2016 data and computed on the 17th of August 2016. To highlight the amount of notional traded, order flow is reported in millions of exchanged dollars according to the average value of each contract on the 17th of August 2016. Basis points were accounted for, so that one traded unit of the futures contracts entitles the owner to one unit of the underlying.

Structure of ρ, Ω and R Figure I.6 shows the estimators of ϱ, Ω and R for the 17th of August 2016. Contracts with the same underlying are strongly correlated. Thus, ρ shows 2 by 2 blocks of strongly correlated contracts and an anti-correlation between bonds and futures. Liquidity is heterogeneous as front month contracts are more actively traded. In this configuration, the discriminating factor between models should be stability axioms rather than fragmentation axioms.

[5] This is a simplification of the settlement rules to emphasize the expected value of the final settlement price. Further details about the final settlement price of E-MINI futures and 10-year US Treasury Note futures can be found in the CME Rulebook.

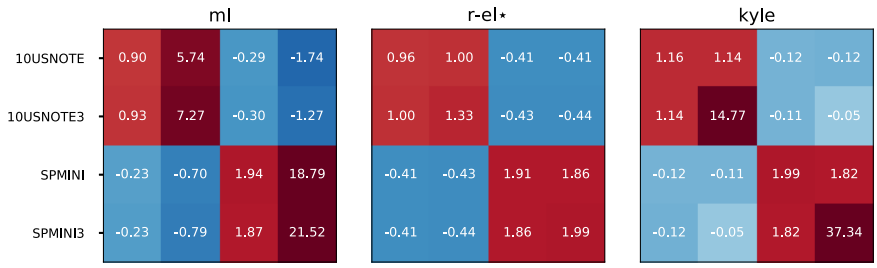


Figure I.7: **Values of different cross-impact models for bonds and indices.**
We report the values of the `ml` (left), `r-el⋆` (center) and `kyle` (right) cross-impact models for the covariances of the 17th of August 2016. Units are chosen to represent the relative price change in basis points (10^{-4} of the asset price) by hundred million USD worth of contract traded.

Cross-impact models for bonds and indices Figure I.7 shows the `ml`, `r-el⋆` and `kyle` models calibrated on bonds and indices. The `r-el` and `kyle` models are weakly cross-stable while the `ml` model is not. Thus `ml` assigns large impact to the less liquid contracts 10USNOTE3 and SPMINI3. Similarly, the self-stability of `r-el` explains the small impact predicted if one trades illiquid contracts. Reassuringly, all models correctly capture the negative index-bonds correlation.

I.B.3 Stocks

Description of the dataset We chose stocks which were in the S&P500 index between January 2016 and December 2017. The resulting universe is made up of with 393 high market cap and liquid stocks. We chose such stocks to build a similar asset universe as in previous studies [WSG15, WSG16, WNG17, BMEB17, PV15]. We collect trades and quotes data between 2PM and 9:30PM UTC, removing the beginning and end of the trading period to focus on the intraday behavior of liquidity and volatility and circumvent intraday non-stationary issues. We collected trades and quotes data from January 2016 to December 2017, between 2PM and 9:30PM UTC, to focus on the intraday behavior of liquidity and volatility and circumvent intraday non-stationary issues. After filtering days for which data for one product was missing, we keep a total of 302 days (154 in 2016 and 148 in 2017). Some summary characteristics of our sample are presented in Table I.3. The distribution of stocks in each sector is given in Figure I.8.

	Quantile		
	10%	50%	90%
Relative tick size (in %)	1.6	2.5	4.6
Number of trades per day (in thousands)	5.9	12.6	29.4
Daily turnover (in MUSD)	28.5	56.1	116.2

Table I.3: **Summary statistics for our sample of stocks.**

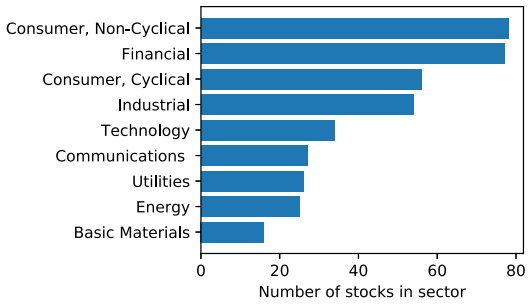


Figure I.8: **Sector breakdown for the 393 of stocks used in the stocks dataset.**

Pre-processing To a lesser degree than on the previous datasets, the stock dataset shows non-stationarity in both volatility and liquidity. Thus we adopt the same estimation procedure for the local covariance estimators Σ_t and Ω_t by assuming stationarity of the correlations $\varrho = \mathrm{diag}(\sigma_t)^{-1}\Sigma_t\mathrm{diag}(\sigma_t)^{-1}$ and $\varrho_\Omega = \mathrm{diag}(\omega_t)^{-1}\Omega_t\mathrm{diag}(\omega_t)^{-1}$.

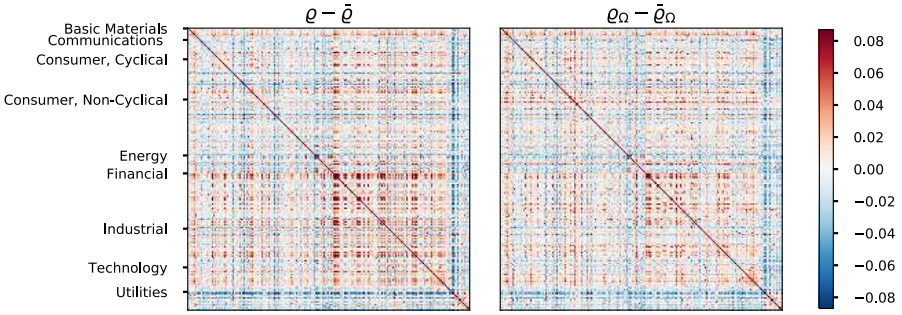


Figure I.9: **Estimated price and order flow correlation matrices ϱ, ϱ_Ω for stocks.**
We represent the return correlation matrix ρ (left), order flow correlation matrix ϱ_Ω (right) estimated on 2016. To highlight the amount of notional traded, order flow is reported in millions of exchanged dollars according to the average value of each contract on the 17th of August 2016. Correlation matrices were represented instead of covariance matrices due to the large volume heterogeneities between stocks. Stocks were grouped by sectors to highlight the blockwise structure of these matrices.

Structure of ϱ, ϱ_Ω and R Figure I.9 shows estimators of ϱ, ϱ_Ω. We report correlations instead of covariances to highlight the blockwise structure of these matrices. For the same reasons, R is not shown but presents a bandwise structure one expects from heterogeneity in liquidity. Pairwise price and order flow correlations between assets are small. Thus, improvement of cross-impact models over direct models should be lower than in previous applications. For more details about the structure of the price and volume covariance matrices, see [BMEB17].

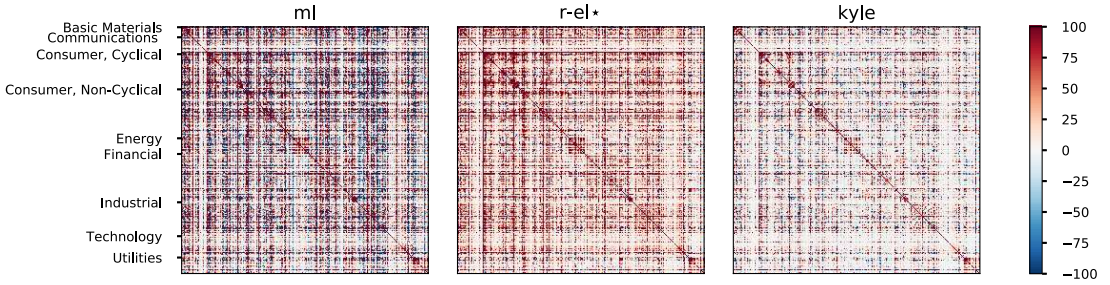


Figure I.10: **Values of different cross-impact models for stocks.**
We report the values of the `ml` (left), `r-el⋆` (center) and `kyle` (right) cross-impact models. Units are chosen to represent the relative price change in basis points (10^{-4} of the asset price) by hundred million USD worth of instruments traded.

Cross-impact models for stocks Figure I.10 shows the `ml`, `r-el⋆` and `kyle` models calibrated on the stocks dataset. At first glance, each model appears to present a blockwise structure similar to that of ρ, ρ_Ω. However, the `ml` model does not satisfy weak cross-stability and thus predicts large impact on liquid stocks if one trades illiquid stocks. By construction the `r-el⋆` model weighs most impact on the market mode. Finally, the `kyle` model looks like a symmetrized version of the `r-el⋆` model.

I.C Goodness-of-fit

We report in Table I.4 the numerical results of the goodness-of-fit tests run on each model and dataset.

Dataset	Score	direct	whitening	whitening⋆	el	el⋆	kyle	r-direct	ml	r-el	r-el⋆	r-kyle
Crude Futures												
	$\mathscr{R}^2_{\text{in}}(I_\sigma)$	0.01 ± 0.01	0.03 ± 0.01	0.06 ± 0.01	0.18 ± 0.01	0.18 ± 0.01	0.35 ± 0.01	0.27 ± 0.01	0.37 ± 0.01	0.37 ± 0.01	0.37 ± 0.01	0.22 ± 0.01
	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	0.01 ± 0.01	0.04 ± 0.01	0.06 ± 0.01	0.18 ± 0.01	0.18 ± 0.01	0.35 ± 0.01	0.27 ± 0.01	0.37 ± 0.01	0.37 ± 0.01	0.37 ± 0.01	0.22 ± 0.01
	$\mathscr{R}^2_{\text{in}}(J_\sigma)$	0.33 ± 0.01	0.32 ± 0.01	0.22 ± 0.01	0.27 ± 0.01	0.27 ± 0.01	0.46 ± 0.01	0.40 ± 0.01	0.45 ± 0.01	0.46 ± 0.01	0.46 ± 0.01	0.31 ± 0.01
	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	0.33 ± 0.01	0.32 ± 0.01	0.22 ± 0.01	0.27 ± 0.01	0.27 ± 0.01	0.46 ± 0.01	0.40 ± 0.01	0.45 ± 0.01	0.46 ± 0.01	0.46 ± 0.01	0.31 ± 0.01
	$\mathscr{R}^2_{\text{in}}(\Sigma^{-1})$	$-\infty$	-0.05 ± 0.02	-0.01 ± 0.02	0.07 ± 0.02	0.07 ± 0.02	0.29 ± 0.02	$-\infty$	0.32 ± 0.02	0.31 ± 0.02	0.31 ± 0.02	0.16 ± 0.02
	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$	$-\infty$	-0.05 ± 0.02	-0.01 ± 0.02	0.07 ± 0.02	0.07 ± 0.02	0.29 ± 0.02	$-\infty$	0.31 ± 0.02	0.31 ± 0.02	0.31 ± 0.02	0.16 ± 0.02
Bonds and indices												
	$\mathscr{R}^2_{\text{in}}(I_\sigma)$	-0.11 ± 0.02	0.03 ± 0.02	0.05 ± 0.02	0.19 ± 0.01	0.02 ± 0.02	0.38 ± 0.01	0.23 ± 0.01	0.40 ± 0.01	0.38 ± 0.01	0.27 ± 0.01	0.25 ± 0.01
	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	-0.11 ± 0.02	0.03 ± 0.02	0.04 ± 0.02	0.19 ± 0.01	0.02 ± 0.02	0.38 ± 0.01	0.23 ± 0.01	0.40 ± 0.01	0.38 ± 0.01	0.27 ± 0.01	0.24 ± 0.01
	$\mathscr{R}^2_{\text{in}}(J_\sigma)$	0.09 ± 0.02	-0.09 ± 0.03	-0.05 ± 0.03	0.09 ± 0.02	-0.21 ± 0.03	0.29 ± 0.02	0.27 ± 0.02	0.30 ± 0.02	0.29 ± 0.02	0.17 ± 0.02	0.14 ± 0.02
	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	0.09 ± 0.02	-0.10 ± 0.03	-0.05 ± 0.03	0.09 ± 0.02	-0.21 ± 0.03	0.29 ± 0.02	0.27 ± 0.02	0.30 ± 0.02	0.29 ± 0.02	0.17 ± 0.02	0.14 ± 0.02
	$\mathscr{R}^2_{\text{in}}(\Sigma^{-1})$	-7.24 ± 0.21	-0.37 ± 0.04	-0.36 ± 0.04	-0.26 ± 0.03	-0.37 ± 0.03	0.11 ± 0.03	-1.69 ± 0.05	0.20 ± 0.03	0.19 ± 0.03	0.13 ± 0.03	0.07 ± 0.03
	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$	-7.23 ± 0.21	-0.37 ± 0.04	-0.36 ± 0.04	-0.26 ± 0.03	-0.37 ± 0.03	0.11 ± 0.03	-1.71 ± 0.05	0.20 ± 0.03	0.19 ± 0.03	0.13 ± 0.03	0.07 ± 0.03
Stocks												
	$\mathscr{R}^2_{\text{in}}(I_\sigma)$	0.038 ± 0.004	-0.025 ± 0.004	0.059 ± 0.004	-0.631 ± 0.010	-0.128 ± 0.008	0.343 ± 0.003	0.276 ± 0.004	0.373 ± 0.003	0.257 ± 0.003	0.236 ± 0.004	0.239 ± 0.004
	$\mathscr{R}^2_{\text{out}}(I_\sigma)$	0.038 ± 0.004	-0.031 ± 0.004	0.047 ± 0.004	-0.642 ± 0.010	-0.133 ± 0.008	0.336 ± 0.003	0.274 ± 0.004	0.358 ± 0.003	0.249 ± 0.003	0.227 ± 0.004	0.232 ± 0.004
	$\mathscr{R}^2_{\text{in}}(J_\sigma)$	0.732 ± 0.006	-0.047 ± 0.012	0.277 ± 0.010	-1.770 ± 0.038	0.727 ± 0.005	0.822 ± 0.003	0.480 ± 0.010	0.829 ± 0.003	0.661 ± 0.005	0.753 ± 0.004	0.788 ± 0.004
	$\mathscr{R}^2_{\text{out}}(J_\sigma)$	0.732 ± 0.006	-0.192 ± 0.013	0.152 ± 0.012	-1.785 ± 0.038	0.701 ± 0.005	0.808 ± 0.004	0.479 ± 0.010	0.803 ± 0.004	0.644 ± 0.005	0.733 ± 0.005	0.776 ± 0.004
	$\mathscr{R}^2_{\text{in}}(\Sigma^{-1})$	-0.311 ± 0.004	-0.061 ± 0.003	-0.056 ± 0.003	-0.262 ± 0.005	-0.369 ± 0.008	0.214 ± 0.003	0.180 ± 0.003	0.215 ± 0.003	0.126 ± 0.004	0.090 ± 0.004	0.082 ± 0.004
	$\mathscr{R}^2_{\text{out}}(\Sigma^{-1})$	-0.293 ± 0.004	-0.061 ± 0.003	-0.056 ± 0.004	-0.260 ± 0.005	-0.360 ± 0.008	0.211 ± 0.003	0.180 ± 0.003	0.208 ± 0.003	0.124 ± 0.004	0.089 ± 0.004	0.081 ± 0.004

Table I.4: **Goodness-of-fit scores for each model and dataset.**
Goodness of fit was measured using two years of data sampled at a time interval of one minute. In-sample data was used to calibrate each cross impact model. Out-of-sample goodness of fit was obtained by applying the calibrated models on never seen before data. We reported as ∞ the scores of models which are numerically infinite, but due to clipping appear finite.

CHAPTER II

Static cross-impact on derivatives

Abstract

Trading a financial asset pushes its price as well as the prices of other assets, a phenomenon known as cross-impact. The empirical estimation of this effect on complex financial instruments, such as derivatives, is an open problem. To address this, we consider a setting in which the prices of *derivatives* is a deterministic function of stochastic *factors* where trades on both factors and derivatives induce price impact. We show that a specific cross-impact model satisfies key properties which make its estimation tractable in applications. Using E-Mini futures, European call and put options and VIX futures, we estimate cross-impact and show our simple framework successfully captures some of the empirical phenomenology. Our framework for estimating cross-impact on derivatives may be used in practice for estimating hedging costs or building liquidity metrics on derivative markets.

From:
Cross-impact in derivative markets
M. Tomas, I. Mastromatteo, M. Benzaquen

Introduction

Market impact describes how trades on one asset translate into its price. The many studies on the topic have deepened our understanding of how markets digest trades into prices [BBDG18, ATHL05, Tor97]. However, market impact ignores the effects of order flows of other assets on an asset's price. This phenomena, dubbed *cross-impact*, has received growing attention in recent years [HS01, PV15, SL19, WSG15, TMB20, RT21a]. Estimating models which account for cross-impact poses new challenges and led the authors to investigate in [TMB20] which cross-impact models satisfy desirable properties while fitting empirical data well.

This chapter contributes to the literature on estimating cross-impact by focusing on a particular asset class: derivatives. In this asset class, cross-sectional impact effects are expected to play a major role. Indeed, in an efficient market, the prices of derivatives should be locked by

no-arbitrage. Therefore, the impact of trades on different derivatives should be related. Furthermore, since there are thousands of strongly correlated and individually illiquid derivatives, we need to aggregate the liquidity of instruments to properly estimate impact.

We set up a continuous-time, state-dependent static cross-impact framework which generalises the setup of [TMB20]. In this framework, derivative prices are functions of stochastic factors and trades impact prices of all instruments. The main result of this chapter is a calibration formula for cross-impact on derivatives, which we find provides better goodness-of-fit than simple impact models.

We now comment on the links between our approach and the literature.

The methodology of this chapter is most similar to a previous paper on cross-impact by the authors [TMB20]. This chapter introduces an infinitesimal extension of the framework of the previous paper in which cross-impact also depends on state variables (such as factor prices). These are slight modifications to handle the specific challenges posed by derivatives.

The cross-impact model used in this chapter is the Kyle cross-impact model, first derived using a multivariate version of the Kyle insider trading framework [CK94, Kyl85, GdMMBB20]. The properties of this model were examined in [TMB20] and this chapter leverages some results from [TMB20].

This chapter focuses on linearly relating anonymous trade data to option prices. This linear approximation is roughly justified for anonymous order flow but not for a particular investor's trades. The papers [TEB16, Sai19] study this question by using databases of proprietary investor orders to estimate the price impact of trades on the implied volatility surface. In particular, [TEB16] shows that option trades have a vega-weighted impact on the level of the implied volatility surface. The work [Sai19] finds that trades also impact the skew or term structure of the implied volatility surface. When applying our framework to empirical data, we recover a linear version of these two effects, along with an estimation of the importance of each effect's contributions.

Finally, this chapter focuses on how market prices of options are affected by trades and ignores the intricacies of the replication problem. Thus, it is separate from works which model how price impact modifies option prices and the associated hedging strategies under different replication constraints [Loe18, BLZ17]. However, the current framework may be useful to give empirical estimates to the impact coefficients appearing in these papers.

The chapter is organized as follows. Section 1 presents our modeling framework. Section 2 derives a tractable calibration formula for cross-impact. Section 3 provides illustrative examples of the framework. Section 4 applies the setup to estimate cross-impact on options. Finally, we conclude on the contributions of the chapter, open questions, and directions for future work.

Notation

The set of n by n real-valued square matrices is denoted by $\mathcal{M}_n(\mathbb{R})$, the set of orthogonal matrices by $\mathcal{O}_n$, the set of real non-singular n by n matrices by $\mathrm{GL}_n(\mathbb{R})$, the set of real n by n symmetric positive semi-definite matrices by $\mathcal{S}_n^+(\mathbb{R})$, and the set of real n by n symmetric positive definite matrices by $\mathcal{S}_n^{++}(\mathbb{R})$. Further, given a matrix A in $\mathcal{M}_n(\mathbb{R})$, $A^\top$ denotes its transpose. Given A in $\mathcal{S}_n^+(\mathbb{R})$, we write $A^{1/2}$ for a matrix such that $A^{1/2}(A^{1/2})^\top = A$ and $\sqrt{A}$ for the matrix square root, the unique positive semi-definite symmetric matrix such that $(\sqrt{A})^2 = A$.

All stochastic processes in the text are defined on a probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in\mathbb{R}}, \mathbb{P})$ and adapted to the filtration $(\mathcal{F}_t)_{t\in\mathbb{R}}$ unless stated otherwise. Standard Brownian motions are defined with respect to the probability measure $\mathbb{P}$. All stochastic differential equations introduced will be assumed to have a unique strong solution and correspondingly the functions appearing in these equations will be assumed to be sufficiently regular for this to be true. We denote by $\mathbb{E}$ the expectation with respect to the probability measure $\mathbb{P}$. We denote by $\langle X, Y\rangle$ the quadratic covariation of the continuous stochastic processes X and Y.

1 Setup

We consider a universe comprising two classes of financial instruments, that we will refer to as *factors* and *derivatives*.

Factors represent a set of N stochastic processes from which one can derive derivative prices. These may include the factor of derivative contracts, such as the spot, as well as stochastic or local volatility, forward variance, yield curves, etc. The prices of factors at time t is denoted by $p_t = (p_t^1, \ldots, p_t^N)$. Derivatives are a set of M contracts, whose prices at maturity depend on the values of the factors. We write $P_t = (P_t^1, \ldots, P_t^M)$ for the prices of these instruments.

Factors and derivatives are traded continuously and we denote by $q_t = (q_t^1, \ldots, q_t^N)$ the stochastic process corresponding to the market net traded order flows on factors and by $Q_t = (Q_t^1, \ldots, Q_t^M)$ the stochastic process corresponding to the market net traded order flows on derivatives. These are aggregate order flow, corresponding to the sum of all trades sent by market participants.

As we are interested in a simple, infinitesimal framework for the influence of trades on prices, we assume that order flow dynamics on each asset are continuous stochastic processes driven by Brownian motions.

Assumption 1 (Order flow dynamics). *The order flow follows the following stochastic dynamics*

$$dq_t = \nu_q(p_t, t)dt + \mathcal{L}_{qq}(p_t, t)dZ_t^q + \mathcal{L}_{Qq}(p_t, t)dZ_t^Q \tag{1}$$

$$dQ_t = \nu_Q(p_t, t)dt + \mathcal{L}_{Qq}(p_t, t)dZ_t^q + \mathcal{L}_{QQ}(p_t, t)dZ_t^Q, \tag{2}$$

where Z^q, Z^Q are uncorrelated standard Brownian motions, $\nu_q\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ and $\nu_Q\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ encode order flow drift, $\mathscr{L}_{qq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_N(\mathbb{R})$, $\mathscr{L}_{QQ}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_M(\mathbb{R})$, $\mathscr{L}_{Qq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{M,N}(\mathbb{R})$, $\mathscr{L}_{qQ}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{N,M}(\mathbb{R})$ encode co-trading among assets.

Assumption 1 allows for correlations of order flows between assets, so that our model accounts for co-trading of factor and derivatives, which is a typical feature expected in derivative markets. However, the continuous modeling framework for order flows Assumption 1 is somewhat unrealistic. Indeed, at the high-frequency scale, trades arrive punctually and trade arrival are auto-correlated so that a more realistic modeling is driven by point processes which respect this property. Such modeling requires lengthy mathematical treatment and is outside the scope of this chapter, so that we choose a simple continuous order flow model. We refer the reader to [RT21a] for a model which accounts for these effects.

We now move to the dynamics of factor prices in our setup.

Assumption 2 (Factor dynamics). *We assume that factor prices are given by*

$$dp_t = \mu_p(p_t, t)dt + \mathscr{G}_p(p_t, t)dW_t + \Lambda_{pq}(p_t, t)dq_t + \Lambda_{pQ}(p_t, t)dQ_t \tag{3}$$

where W is a standard N-dimensional Brownian motion, $\mu_p\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ are the factor drift, $\mathscr{G}_p\colon \mathbb{R}^N \times \mathbb{R} \to \mathrm{GL}_N(\mathbb{R})$ are the factor and derivative diffusion matrices, and $\Lambda_{pq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_N(\mathbb{R})$, $\Lambda_{pQ}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{N,M}(\mathbb{R})$ capture cross-impact.

The factor dynamics of Assumption 2 are quite general. They encompass local and stochastic volatility models and allow for cross-impact between factors and between factors and derivatives. We assume that factors are chosen so that derivatives are priced in a Markovian manner with respect to the factors, which leads us to the next assumption.

Assumption 3 (Derivative prices). *There exists a function $F\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$, twice differentiable with respect to the first argument and differentiable with respect to the second argument, such that $P_t = F(p_t, t)$.*

Applying Ito's formula to Equation (3) and using Assumptions 1 and 3, we obtain the following corollary.

Corollary 1. *The derivative dynamics are given by*

$$dP_t = \mu_P(p_t, t)dt + \mathscr{G}_P(p_t, t)dW_t + \Lambda_{Pq}(p_t, t)dq_t + \Lambda_{PQ}(p_t, t)dQ_t\,, \tag{4}$$

where $\mu_P\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ is the derivative drift, $\mathscr{G}_P\colon \mathbb{R}^N \times \mathbb{R} \to \mathrm{GL}_M(\mathbb{R})$ is the derivative diffusion matrix, $\Lambda_{Pq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{M,N}(\mathbb{R})$ and $\Lambda_{PQ}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{M}_{N,M}(\mathbb{R})$ encode cross-impact. In particular, we have the constraints $\Lambda_{Pq} = \Xi\Lambda_{pq}$, $\Lambda_{PQ} = \Xi\Lambda_{pQ}$, $\mathscr{G}_P = \Xi\mathscr{G}_p$, where $\Xi := (\frac{\partial P^i}{\partial p^j})_{i,j}$ is the M by N sensitivity matrix.

Corollary 1 does not make explicit the dependence of the derivative drift on other variables as it will not play an important role.

For convenience cross-impact matrices appearing in Equations (3) and (4) can be compactly rearranged into a single matrix, Λ, which we refer to as the cross-impact matrix since it describes the cross-impact of the complete system

$$\Lambda(p_t, t) := \begin{pmatrix} \Lambda_{pq} & \Lambda_{pQ} \\ \Lambda_{Pq} & \Lambda_{PQ} \end{pmatrix}(p_t, t)\,. \tag{5}$$

The impact model we propose involves two parameters, the return covariance matrix and the order flow covariance matrix.

The factor-factor return covariance matrix $\Sigma_{pp}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{S}_N^+(\mathbb{R})$ is defined as

$$\Sigma_{pp}(p_t, t)\mathrm{d}t := \mathrm{d}\langle p, p\rangle_t\,, \tag{6}$$

and we similarly denote $\Sigma_{pP}, \Sigma_{Pp} = \Sigma_{pP}^\top, \Sigma_{PP}$ for the factor-derivative, derivative-factor and derivative-derivative return covariance matrices. Naturally, since derivative prices are deterministic function of factors, these matrices are all related to Σ_{pp}.

We denote by $\Omega_{qq}\colon \mathbb{R}^N \times \mathbb{R} \to \mathscr{S}_N^+(\mathbb{R})$ the factor-factor order flow covariance matrix

$$\Omega_{qq}(p_t, t)\mathrm{d}t := \mathrm{d}\langle q, q\rangle_t\,, \tag{7}$$

and we denote Ω_{qQ}, $\Omega_{Qq} = \Omega_{qQ}^\top$ and Ω_{QQ} for the factor-derivative, derivative-factor and derivative-derivative order flow covariances. Contrary to return covariance matrices, there are no constraints betweeen these order flow covariance matrices and Ω_{qq}. The covariance structure of returns and flows for the whole system can be arranged compactly as

$$\Sigma(p_t, t) = \begin{pmatrix} \Sigma_{pp} & \Sigma_{pP} \\ \Sigma_{pP}^\top & \Sigma_{PP} \end{pmatrix}(p_t, t) \qquad \Omega(p_t, t) = \begin{pmatrix} \Omega_{qq} & \Omega_{qQ} \\ \Omega_{qQ}^\top & \Omega_{QQ} \end{pmatrix}(p_t, t)\,.$$

The cross-impact matrix that we propose to use, dubbed the *Kyle cross-impact model*, was first derived in [CK94] and then analyzed in [GdMMBB20, TMB20]. The next assumption makes explicit the formula of the Kyle cross-impact matrix within our framework.

Assumption 4 (Cross-impact matrix). *The cross-impact matrix Λ is of the form*

$$\Lambda := \sqrt{Y}(\mathscr{L}^{-1})^\top \sqrt{\mathscr{L}^\top \Sigma \mathscr{L}}\,\mathscr{L}^{-1}\,, \tag{8}$$

where $\mathscr{L}$ is a matrix such that $\mathscr{L}\mathscr{L}^\top = \Omega$, Y is a constant such that $0 < Y < 1$, and we have omitted the dependence on (p_t, t) for compactness.

The choice of the Kyle cross-impact matrix is motivated by the goodness-of-fit observed in previous work [TMB20] and some key properties it satisfies, which will be outlined in the next section. Note that the cross-impact matrix of Assumption 4 must, at the very least, be compatible with the constraints on the cross-impact matrix outlined in Corollary 1. We will show that this is true in the next section.

Our framework being set up, we move to the derivation of key properties of our cross-impact framework in the next section. These properties motivate the choice of the Kyle cross-impact model as the only well-behaved model for cross-impact on derivatives.

2 A practical formula for the cross-impact matrix

This section shows how one can derive the cross-impact matrix Λ in a tractable form for applications.

The next proposition shows that the impact contributions to factor and derivative prices can be absorbed inside Brownian motions.

Proposition 1. *Using the notation of Assumptions 2 and 4 and Corollary 1, we have*

$$dp_t = \tilde{\mu}_p(p_t, t)dt + \frac{1}{\sqrt{1-Y}}\mathcal{G}_p(p_t, t)dB_t \tag{9}$$

$$dP_t = \tilde{\mu}_P(p_t, t)dt + \frac{1}{\sqrt{1-Y}}\mathcal{G}_P(p_t, t)dB_t\,, \tag{10}$$

where $\tilde{\mu}_p \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$, $\tilde{\mu}_P \colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^M$ *and* B *is a standard* N *dimensional Brownian motion.*

The proof of Proposition 1 is given in Section II.A. Proposition 1 underscores a convenient property of our framework: the sensitivities of derivative prices with respect to factors are independent of the order flow dynamics. In particular, the Greeks can be computed using traditional methods since derivative prices satisfy the classic stochastic differential system of Proposition 1.

The next proposition shows that the large cross-impact matrix Λ can be expressed solely as a function of the derivative sensitivities and the compact cross-impact matrix Λ_{pq}.

Proposition 2. *We have*

$$\Lambda(p_t, t) = \begin{pmatrix} \Lambda_{pq} & \Lambda_{pq}\Xi^\top \\ \Xi\Lambda_{pq} & \Xi\Lambda_{pq}\Xi^\top \end{pmatrix}(p_t, t)\,, \tag{11}$$

where we recall that $\Xi := (\frac{\partial P^i}{\partial p^j})_{i,j}$ *is the* M *by* N *sensitivity matrix.*

The proof of Proposition 2 is given in Section II.A. Note that Proposition 2 shows that the chosen cross-impact model satisfies the constraints of Corollary 1. Furthermore, it expresses the large $N+M$ by $N+M$ cross-impact matrix Λ as a function of the much smaller N by N cross-impact matrix Λ_{pq} and the derivative sensitivities Ξ. Thanks to Proposition 1, the latter can be computed as the usual Greeks of our derivative pricing model. Thus, given a formula for Λ_{pq}, we could express Λ. The next proposition proves that the cross-impact matrix Λ_{pq} can be expressed as a function of the factor return covariance matrix Σ_{pp} and a modified covariance matrix of order flows.

Proposition 3. *We have*

$$\Lambda_{pq} = \sqrt{Y}(\mathscr{L}_\Xi^{-1})^\top \sqrt{\mathscr{L}_\Xi^\top \Sigma_{pp} \mathscr{L}_\Xi}\, \mathscr{L}_\Xi^{-1}\,, \tag{12}$$

where we have omitted the dependence on (p_t, t), $\Omega_\Xi := \Omega_{qq} + \Xi^\top \Omega_{QQ}\Xi + \Xi^\top \Omega_{Qq} + \Omega_{qQ}\Xi$, *and* $\mathscr{L}_\Xi$ *is a matrix such that* $\mathscr{L}_\Xi \mathscr{L}_\Xi^\top = \Omega_\Xi$.

The proof of Proposition 3 is given in Section II.A. Combined, Propositions 2 and 3 give a formula for the cross-impact matrix Λ as a function of the measurable quantities Σ_{pp} and Ω_{Ξ}. Furthermore, by Proposition 1, Ξ can be computed using usual derivative pricing methods. Overall, we have thus derived a scheme for estimating a cross-impact matrix on derivatives.

A relevant insight of Equation (12) for applications is that even if factors are not traded, as long as derivatives with sensitivities to these factors are traded, i.e. $\Xi^\top \Omega_{QQ} \Xi$ is positive definite, then the inverses appearing in Equation (12) are well-defined. This is not obvious from the form of the Kyle cross-impact matrix in Assumption 4. This property is important for applications where most factors correspond to non-tradeable instruments, such as volatility factors.

3 Examples

To illustrate the flexibility of our setup and the usefulness in practice of the properties of the Kyle cross-impact model, we discuss examples in increasing complexity below.

3.1 Futures

For our first example, we consider a universe of $N = M = 1$ instruments, consisting in a spot with price p_t and a futures contract delivering one unit of the spot and expiring at a later time T. By assuming a constant, continuously compounded, deterministic interest rate r the derivative price is given by

$$P(p_t, t) = e^{r(T-t)} p_t .$$

Therefore, in this case $\Xi(p_t, t) = \partial_p P(p_t, t) = e^{r(T-t)}$ and Equation (11) yields

$$\Lambda(p_t, t) = \sqrt{Y} p_t \frac{\sigma(p_t, t)}{\omega(p_t, t)} \begin{pmatrix} 1 & e^{r(T-t)} \\ e^{r(T-t)} & e^{2r(T-t)} \end{pmatrix},$$

where $\sigma^2(p_t, t)\mathrm{d}t := \mathrm{d}\langle p, p \rangle_t$ and $\omega^2(p_t, t) := (1, e^{r(T-t)})^\top \Omega(p_t, t)(1, e^{r(T-t)})$. The meaning of this formula is simple: in a constant interest-rate model, the market liquidity should mix liquidity traded the spot and on the future, after properly adjusting for the interest rate.

3.2 Black-Scholes model

We now consider a system with a single factor ($N = 1$) and M derivatives. The factor is the spot with price p_t and the derivatives are a set of European call or put options with different strikes and maturities. We assume that the spot price p_t follows log-normal dynamics with cross-impact, given by

$$\mathrm{d}p_t = \mu p_t \mathrm{d}t + \sqrt{1 - Y} \sigma p_t \mathrm{d}W_t + \Lambda_{pq} \mathrm{d}q_t + \Lambda_{pQ} \mathrm{d}Q_t ,$$

where W_t is a one-dimensional standard Brownian motion, μ is the drift and σ is the implied volatility. The parameter σ is the implied volatility since Proposition 1 implies that there exists

some drift $\tilde{\mu}$ such that

$$\mathrm{d}p_t = \tilde{\mu} p_t \mathrm{d}t + \sigma p_t \mathrm{d}B_t\,,$$

where B is a one-dimensional standard Brownian motion. Then, with the usual notation for the Black-Scholes Δ, we have

$$\Xi^i(p_t,t) = \partial_p P^i(p_t,t) = \Delta^i(p_t,t)\,,$$

and, writing $\Delta := (\partial_p P^1(p_t,t),\cdots,\partial_p P^M(p_t,t))$, Equation (11) yields

$$\Lambda(p_t,t) = \sqrt{Y} p_t \frac{\sigma}{\omega(p_t,t)} \begin{pmatrix} 1 & \Delta^\top \\ \Delta & \Delta\Delta^\top \end{pmatrix}(p_t,t)\,,$$

where

$$\omega^2(p_t,t) = (1,\Delta(p_t,t))^\top \Omega(p_t,t)(1,\Delta(p_t,t))\,.$$

Thus, as in the previous example, there is a single liquidity pool, with volumes traded on options adjusted for the options' Δ. Volume traded on deep in-the-money options ($\Delta^i \approx 1$) contribute to the overall liquidity pool as if it was the spot itself that was traded, whereas deeply out-of-the-money options ($\Delta^i \approx 0$) give negligible contributions.

3.3 Volatility factors

3.3.1 General setup

We build on our previous example and consider a spot and a strip of M European call and put options, with different implied volatilities for each option. The spot price in this example is denoted by s_t, while option prices are given by $P^i_t(s_t,\hat{\sigma}^i_t)$ where $\hat{\sigma}^i_t$ is the implied volatility of this option.

In order to reduce the dimensionality of the M implied volatilities $\hat{\sigma}^i_t$, we assume that they are completely described by a set of volatility factors $\varsigma_t = (\varsigma^1_t,\ldots,\varsigma^Q_t)$ such that the implied volatility of each option is given by

$$\hat{\sigma}^i_t = F^i(\varsigma_t)\,, \tag{13}$$

where $F^i\colon \mathbb{R}^Q \to \mathbb{R}$ is some function of these factors. With some abuse of notation, we write

$$P^i(s_t,\hat{F}^i(\varsigma_t),t) = P^i(s_t,\varsigma_t,t)\,,$$

and we will employ a similar notation for other functions of the implied volatility $\hat{\sigma}^i_t$. Our instruments thus comprise $N = 1+Q$ factors, of which only one is tradeable (the spot), and where the other Q factors correspond to non-tradeable volatility factors. The sensitivities of the system in this case correspond to

$$\Xi^{i1}_t(s_t,\varsigma_t,t) = \Delta^i_t(s_t,\varsigma_t,t) := \frac{\partial P^i(s_t,\varsigma_t,t)}{\partial s}$$

$$\Xi^{i(q+1)}_t(s_t,\hat{\sigma}^i_t,t) = \frac{\partial P^i(s_t,\hat{\sigma}^i_t,t)}{\partial \hat{\sigma}^i} \frac{\partial F^i(\varsigma_t)}{\partial \varsigma^q} := \mathcal{V}^i_t(s_t,\varsigma_t,t)\beta^{iq}(s_t,\varsigma_t,t) =: \Upsilon^{iq}\,,$$

where $q = 1, \dots, Q$ and where, as it is customary in the literature on option pricing, we have introduced the *vega*

$$\mathcal{V}_t^i(s_t, \hat{\sigma}_t^i, t) = \frac{\partial P_t^i(s_t, \hat{\sigma}_t^i, t)}{\partial \hat{\sigma}^i},$$

and the sensitivities of the volatility surface to ς_t

$$\beta^{iq}(s_t, \varsigma_t, t) = \frac{\partial F^i(\varsigma_t)}{\partial \varsigma^q}.$$

This simple setup allows us to capture some of the salient implied volatility surface dynamics and we will make use of it in the next section.

3.3.2 Single factor model

This section examines the particular case when the volatility surface depends on a single volatility factor, i.e. $Q = 1$. The following lemma shows that given some additional assumptions, we can derive an explicit formula for Λ_{pq}.

Lemma 1. *We denote by* $\Delta_c := (1, 0, \frac{\partial P^1}{\partial s}, \cdots, \frac{\partial P^M}{\partial s})$, $\mathcal{V}_c := (0, 1, \frac{\partial P^1}{\partial \varsigma}, \cdots, \frac{\partial P^M}{\partial \varsigma})$ *the vectors of sensitivities. Then, if* $\Delta_c^\top \Omega \mathcal{V}_c = 0$, *we have*

$$\Lambda_{pq}(s_t, \varsigma_t, t) = \frac{\sqrt{Y}}{\sqrt{\sigma^2 \omega_\Delta^2 + \xi^2 \omega_\mathcal{V}^2 + 2\sigma\xi\rho\omega_\Delta\omega_\mathcal{V}}} \begin{pmatrix} \sigma^2 + \frac{\omega_\mathcal{V}}{\omega_\Delta}\sigma\xi\sqrt{1-\rho^2} & \sigma\xi\rho \\ \sigma\xi\rho & \xi^2 + \frac{\omega_\Delta}{\omega_\mathcal{V}}\sigma\xi\sqrt{1-\rho^2} \end{pmatrix} (s_t, \varsigma_t, t), \tag{14}$$

where $\omega_\Delta^2 := \Delta_c^\top \Omega \Delta_c$ *is the delta-aggregated liquidity,* $\omega_\mathcal{V}^2 := \mathcal{V}_c^\top \Omega \mathcal{V}_c$ *is the vega-aggregated liquidity,* $\sigma^2(s_t, \varsigma_t, t)dt := d\langle s, s\rangle_t$ *is the spot volatility,* $\xi^2(s_t, \varsigma_t, t)dt := d\langle \varsigma, \varsigma\rangle_t$ *is the volatility of volatility and* $\rho(s_t, \varsigma_t, t)dt := \dfrac{d\langle \varsigma, s\rangle_t}{\xi(s_t, \varsigma_t, t)\sigma(s_t, \varsigma_t, t)}$ *is the spot-vol correlation.*

The proof of Lemma 1 is given in Section II.A. An interesting result of Equation (14) is that a delta-hedged trade induces impact on the spot because of the negative spot-vol correlation. We will make use of this single factor model in the next section.

4 Empirical Results

We now illustrate our setup with an empirical analysis of cross-impact on derivatives markets which makes use of the results derived in Section 2. Section 4.1 describes the universe of instruments and the chosen derivative modeling. Section 4.2 shows the empirical observables Σ_{pp} and Ω_Ξ used in Section 4.3 to compute the resulting cross-impact matrix Λ_{pp}. Finally, Section 4.4 stress-tests the fit of cross-impact models and Section 4.5 examines non-parametric evidence of cross-impact.

4.1 Setup

The universe of instruments is made up of (i) the front-month E-mini future, (ii) the two front-month VIX futures, (iii) a set of $M-2$ call and put options on the E-mini. We thus have M derivatives.

We bin returns and order flows on a time window Δt of five minutes. We write δp_t for the factor price change between time t and time $t+\Delta t$, δq_t for the signed order flow traded on factors within that time window and δQ_t for the signed order flow traded on derivatives. Prices and order flows for these instruments are taken from trades and quotes data and more detail on the dataset is provided in Section II.B.1.

We consider a linear approximation of the implied volatility surface with volatility factors, so that using the notations of Section 3.3, we have $F^i(\varsigma) = \sum_{q=1}^{Q} \beta^{iq}\varsigma^q$ where $i = 1,\dots,M$. To fit surfaces, we choose $Q = 3$ and perform a principal component analysis of implied volatility surface returns (see, for example, [CFD02]). These factors are given Figure II.1. The first factor is a classic implied volatility level factor and we make the rough approximation that VIX futures are solely explained by such *level* factor. The second factor corresponds to the skew of the implied volatility surface, referred to as the *skew* factor hereafter. The third factor explains the term structure of the implied volatility, hence the name *term* factor in the following.

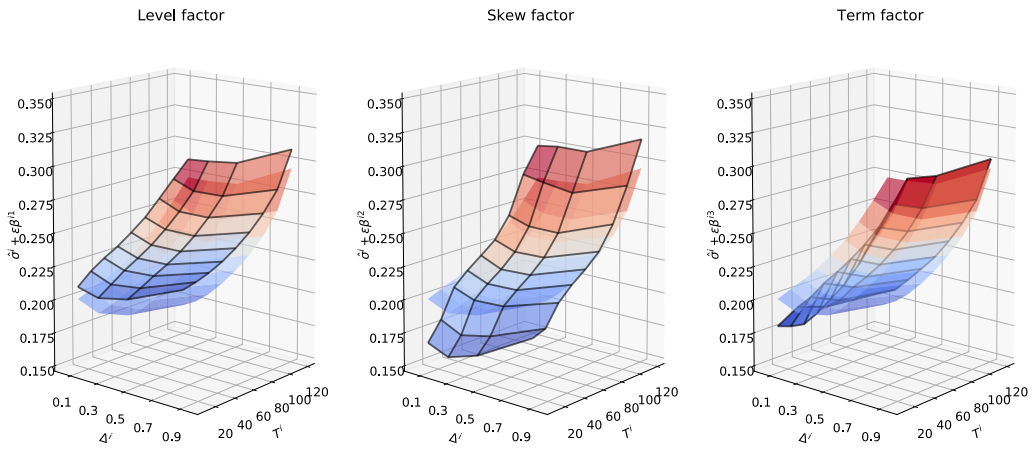


Figure II.1: **Effect of the different volatility factors on the implied volatility surface.** Starting from a historical implied volatility $\hat{\sigma}^i$, we show the modified implied volatility surface after adding a small contribution from the factor q: $\hat{\sigma}^i + \epsilon\beta^{iq}$. The original (non-modified) implied volatility surface is shown in light opacity for reference.

4.2 Estimated covariances

Proposition 3 shows that we need to estimate Σ_{pp} and Ω_Ξ to compute Λ_{pq}. We detail the estimation procedure in Section II.B.2. Figure II.2 shows the estimated factor return correlation matrix $\varrho_{pp} := \text{diag}\sigma^{-1}\Sigma_{pp}\text{diag}\sigma^{-1}$ and the risk order flow covariance matrix $\Omega_\Xi^{\text{risk}} := \text{diag}\sigma\Omega_\Xi\text{diag}\sigma$ where $\sigma = ((\Sigma_{pp}^{11})^{1/2}, \cdots, (\Sigma_{pp}^{NN})^{1/2})$ is the factor volatility.

The factor return correlation matrix correlation matrix shows strong negative correlation between the spot and level mode. This is a well-known stylised fact, sometimes referred to as the "leverage effect". This will play an important role in the form of the cross-impact model, as highlighted in Equation (18). Unsurprisingly, the correlation between spot and level order flow is much smaller, although still noticeable (around -0.15%).

The traded risk (volatility times liquidity) is concentrated on the spot and level directions. This justifies approximating cross-impact on options using solely the spot and the level factor, which we delve in more detail in Section 4.3. The traded risk in the skew direction is much smaller than all other directions and is thus expected to play a lesser role.

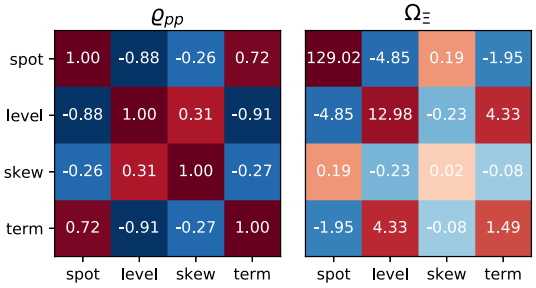


Figure II.2: **Empirical estimates of the factor return correlation matrix ϱ and aggregate order flow covariance Ω_Ξ.**
The factor return correlation matrix ϱ (left) and the aggregate order flow covariance matrix Ω (right) estimates on our dataset. The order flow is reported in thousands of dollars of risk.

4.3 Cross-impact models

We can now use the empirical estimates of Σ_{pp} and Ω_Ξ from the previous section to compute the cross-impact matrix Λ. For comparison purposes, we also introduce other choices of cross-impact matrices. All cross-impact matrices involve a prefactor Y which is calibrated to maximise goodness-of-fit. The first cross-impact matrix used for comparison is the Black-Scholes cross-impact model introduced in Section 3.2 which has a single factor: the spot. It is defined as

$$\Lambda_{\text{bs}}(p_t, t) := \sqrt{Y}\frac{\sigma}{\sqrt{\Delta_{\text{c}}^\top \Omega \Delta_{\text{c}}}}\Delta_{\text{c}}\Delta_{\text{c}}^\top, \tag{15}$$

where $\Delta_c := (1,0,0,0,0,0,\Delta^3(p_t,t),\cdots,\Delta^M(p_t,t))$ is the delta vector, which places one on the spot, zero on the three volatility factors, zero on the two VIX futures, and the usual Black-Scholes delta on put and call options. The Black-Scholes model coincides with the Kyle cross-impact model if all the liquidity is concentrated on the spot. In particular, this model is unable to account for changes in the volatility factors. We thus introduce the two-dimensional direct model $\Lambda_{\text{direct-2d}}$ which accounts for the spot and implied volatility factor but ignores cross-sectional effects, defined as

$$\Lambda_{\text{direct-2d}}(p_t,t) := \sqrt{Y}\frac{\sigma}{\sqrt{\Delta_c^\top \Omega \Delta_c}}\Delta_c\Delta_c^\top + \sqrt{Y}\frac{\xi}{\sqrt{\mathcal{V}_c^\top \Omega \mathcal{V}_c}}\mathcal{V}_c\mathcal{V}_c^\top, \tag{16}$$

where $\mathcal{V}_c := (0,1,0,0,1,1,\mathcal{V}^3(p_t,t),\cdots,\mathcal{V}^M(p_t,t)$ is the vega vector, which places zero on the spot, one on the level factor, zero on the other two volatility factors, and the usual Black-Scholes vega on put and call options. To account for all factor without correcting for cross-sectional effects, we introduce the four-dimensional direct model

$$\Lambda_{\text{direct-4d}}(p_t,t) := \sqrt{Y}\frac{\sigma}{\sqrt{\Delta_c^\top \Omega \Delta_c}}\Delta_c\Delta_c^\top + \sqrt{Y}\frac{\xi}{\sqrt{\mathcal{V}_c^\top \Omega \mathcal{V}_c}}\mathcal{V}_c\mathcal{V}_c^\top + \sqrt{Y}\sum_{i=3}^{Q+1}\sqrt{\frac{\Sigma_{pp}^{ii}}{\Omega_\Xi^{ii}}}\Xi^{\cdot i}(p_t,t)(\Xi^{\cdot i}(p_t,t))^\top. \tag{17}$$

Direct models ignore the off-diagonal structure of Σ_{pp} and Ω_Ξ. In particular they do not account for the leverage effect, which is an essential characteristic of the factor return covariance matrix Σ_{pp}. To fix this, we introduce the two-dimensional Kyle cross-impact model Λ_{2d} which captures the two dominating factor of the system: the spot and level factor. Since Figure II.2 shows that the delta and vega order flow correlation is small (around -0.15%) and $\xi\omega_{\mathcal{V}} \ll \sigma\omega_\Delta$, we can use Equation (14) to obtain the approximation

$$\Lambda_{2d}(p_t,\varsigma_t,t) \approx \sqrt{Y}\frac{\sigma}{\sqrt{\Delta_c^\top \Omega \Delta_c}}\Delta_c\Delta_c^\top + \sqrt{Y}\frac{\xi\sqrt{1-\rho^2}}{\sqrt{\mathcal{V}_c^\top \Omega \mathcal{V}_c}}\mathcal{V}_c\mathcal{V}_c^\top + \sqrt{Y}\frac{\xi\rho}{\sqrt{\Delta_c^\top \Omega \Delta_c}}(\mathcal{V}_c\Delta_c^\top + \Delta_c\mathcal{V}_c^\top). \tag{18}$$

The two-dimensional Kyle cross-impact model predicts that when trading options, one pushes the price in the amount of notional $\mathcal{V}$ traded divided by the typical $\mathcal{V}$ liquidity, which is compatible with findings from the meta-order study [TEB16].

The full, four-dimensional Kyle cross-impact model Λ_{4d} (with calibrated prefactor $Y = 0.5$) is shown in Figure II.3. Compared to the two-dimensional Kyle cross-impact model, it decouples the contribution of options on the level mode depending on the direction. This increases the explanatory power of the model, as will be clear in the next section.

4.4 Explanatory power of cross-impact models

For practical applications, a good cross-impact model should explain realized price changes from order flows. Thus to compare the models previously introduced, we now examine their explanatory power on empirical data. Given a realization of the factor price process

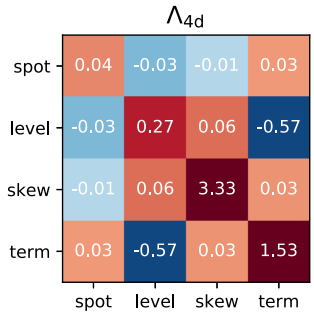


Figure II.3: **Four dimensional Kyle cross-impact model on options.**
We report the four dimensional Kyle model estimated using empirical estimates of the covariances of Figure II.2. The cross-impact matrix are reported in units of risk and in basis points so that Λ_{ij} encodes by how many basis points of volatility Asset i is pushed by trading one dollar of risk on Asset j.

$(\delta p_t)_{1\leq t\leq T}$ of length T, a corresponding series of predictions $(\widehat{\delta p}_t)_{1\leq t\leq T}$ and a symmetric positive semi-definite matrix M, we introduce the generalized $\mathscr{R}^2_{\text{in}}(M)$ error as

$$\mathscr{R}^2_{\text{in}}(M) := 1 - \frac{\sum_{1\leq t\leq T}(\delta p_t - \widehat{\delta p}_t)^\top M(\delta p_t - \widehat{\delta p}_t)}{\sum_{1\leq t\leq T}\delta p_t^\top M \delta p_t}.$$

The matrix M is used to examine a model's predictive power for different portfolios. As the factor of our system are natural directions to consider, we report $\mathscr{R}^2_{\text{in}}(M)$ in Table II.1 for $e_1e_1^\top =: \Pi_{\text{spot}}$, $e_2e_2^\top =: \Pi_{\text{level}}$, $e_3e_3^\top =: \Pi_{\text{skew}}$ and $e_4e_4^\top =: \Pi_{\text{term}}$.

Model	Scores			
	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{spot}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{level}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{skew}})$	$\mathscr{R}^2_{\text{in}}(\Pi_{\text{term}})$
Λ_{bs}	0.18 ± 0.01	-0.00 ± 0.02	-0.00 ± 0.01	-0.00 ± 0.02
$\Lambda_{\text{direct-2d}}$	0.18 ± 0.01	-0.03 ± 0.02	-0.01 ± 0.01	0.00 ± 0.02
$\Lambda_{\text{direct-4d}}$	0.18 ± 0.01	-0.03 ± 0.02	-0.14 ± 0.02	-0.26 ± 0.02
Λ_{2d}	0.20 ± 0.01	0.12 ± 0.01	-0.01 ± 0.01	0.01 ± 0.02
Λ_{4d}	0.20 ± 0.01	0.14 ± 0.01	-0.12 ± 0.02	0.04 ± 0.01

Table II.1: **Scores of different cross-impact models.**
All scores were computed in-sample using the same data used for the calibration of the cross-impact models.

The goodness-of-fit score in the spot direction $\mathscr{R}^2_{\text{in}}(\Pi_{\text{spot}})$ is similar for all models, with cross-impact models being slightly better. Furthermore, there is no difference between Λ_{2d} and Λ_{4d}. This is consistent with the liquidity reported in Figure II.6. Indeed, most of the liquidity is placed on the spot and the order flow traded on other factors is small in comparison. Models which only take into account the spot thus capture most of the order flow explanatory power.

There is also a small advantage in using order flow on the level mode since Λ_{2d} and Λ_{4d} score better, but using term and skew order flow provides no improvement.

While using solely spot liquidity to explain spot returns is a good approximation, $\mathscr{R}^2_{\text{in}}(\Pi_{\text{level}})$ shows the same is not true for the level factor. Indeed, only models with cross-impact between spot and level factors properly explain changes in the level factor. This is natural as most of the traded order flow is on the spot but there is a high negative correlation between spot and level factor (see Figure II.6). Unfortunately, all models fail to explain skew returns. We suspect this comes from the low signal to noise ratio and low liquidity (in risk terms) of the skew factor (see Figure II.6).

On all metrics, Λ_{4d} performs at least as well as Λ_{2d}, which shows that the model is able to combine additional factors without suffering from noise. The additional factor also help weigh trades appropriately on the implied volatility surface, which improves the $\mathscr{R}^2_{\text{in}}(\Pi_{\text{level}})$ score.

Finally, we report the expected realized return conditional on the prediction of Λ_{4d} in Figure II.4. This shows that, skew aside, Λ_{4d} provides a good fit for the realized returns of the different factor as $\mathbb{E}[\delta p_t \mid \delta \hat{p}_t] \approx \delta p_t$.

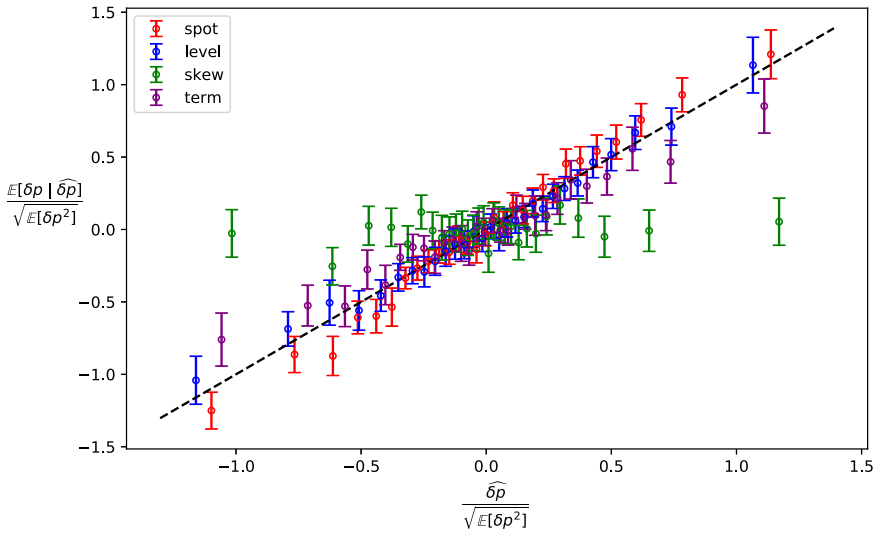


Figure II.4: **Predictions of the four-dimensional Kyle model on the main directions of the system.**
We report the expected price change conditional on the predicted price change of the four-dimensional Kyle model for the four main directions of the system: in red for the spot, blue for the level, green for the skew and purple for the term structure. Predicted price changes and conditional averages are both normalized by the standard deviation of price changes along the given direction.

4.5 Non-parametric evidence of cross-impact on options

Section 4.4 showed that only cross-impact models are able to explain returns for the level and term factor. Aside from this explanatory power, this section tests their ability to explain other features of our data. To do so, we introduce the cross aggregate impact metric. The cross aggregate impact induced from the portfolio $u \in \mathbb{R}^{N+M}$ on the return of the portfolio $v \in \mathbb{R}^{N+M}$ is

$$\mathrm{Agg}_{u,v}(x) := \mathbb{E}[v^\top(\delta p_t, \delta P_t) \mid u^\top(\delta q_t, \delta Q_t) = x] .$$

If returns are given by a linear cross-impact model Ψ and if we further assume $(\delta q_t, \delta Q_t)$ is a zero-mean Gaussian, then

$$\mathrm{Agg}_{u,v}(x) = \mathbb{E}[v^\top \Psi(\delta q_t, \delta Q_t) \mid u^\top(\delta q_t, \delta Q_t)] := \mathrm{Agg}^{\Psi}_{u,v}(x) = a_\Psi x ,$$

where the slope a_Ψ depends on the cross-impact model Ψ and on the order flow covariance. Even in the absence of cross-impact, the presence of order flow correlations between two portfolios u and v may lead to a non-zero cross aggregate impact. Thus, to test whether there is cross-impact, we compare the empirically measured $\mathrm{Agg}_{u,v}$ to the prediction $\mathrm{Agg}^{\Psi}_{u,v}$ for different cross-impact models Ψ. We differentiate models between those which have no off-diagonal contributions ($\Lambda_{\text{bs}}, \Lambda_{\text{direct-4d}}, \Lambda_{\text{direct-2d}}$) and thus ignore cross-impact and those that take it into account ($\Lambda_{\text{2d}}, \Lambda_{\text{4d}}$).

We report $\mathrm{Agg}_{u,v}$ in Figure II.4 for different portfolios u, v described in Table II.2. Diagonal plots show aggregate direct impact. As expected, buying the E-Mini increases, on average, the price of the E-Mini as shown by the $u, v =$ spot plot (first row, first column). We see from the $u, v =$ level plot (second row, second column) that buying options and VIX futures increases, on average, the implied volatility. Furthermore, buying options and VIX futures decreases, on average, the E-Mini price as shown by the $u =$ level, $v =$ spot plot (first row, second column). This same plot, among others of Figure II.4, shows that direct models provide a poor fit for cross aggregate impact. This suggests that the cross aggregate impact can only be explained by using a cross-impact model with off-diagonal elements, such as Λ_{4d}. Further, the fit is noticeably better for Λ_{4d} than Λ_{2d} which highlights the importance of taking into account the skew and term factors.

Name	Components			
	spot	VIX_0	VIX_1	options
spot	(1,	0,	0,	$0, \cdots, 0)$
level	(0,	β^{11},	β^{21},	$\beta^{31}, \cdots, \beta^{M1})$
skew	(0,	β^{12},	β^{22},	$\beta^{32}, \cdots, \beta^{M2})$
term	(0,	β^{13},	β^{23},	$\beta^{33}, \cdots, \beta^{M3})$

Table II.2: **Description of different directions used in this section.**

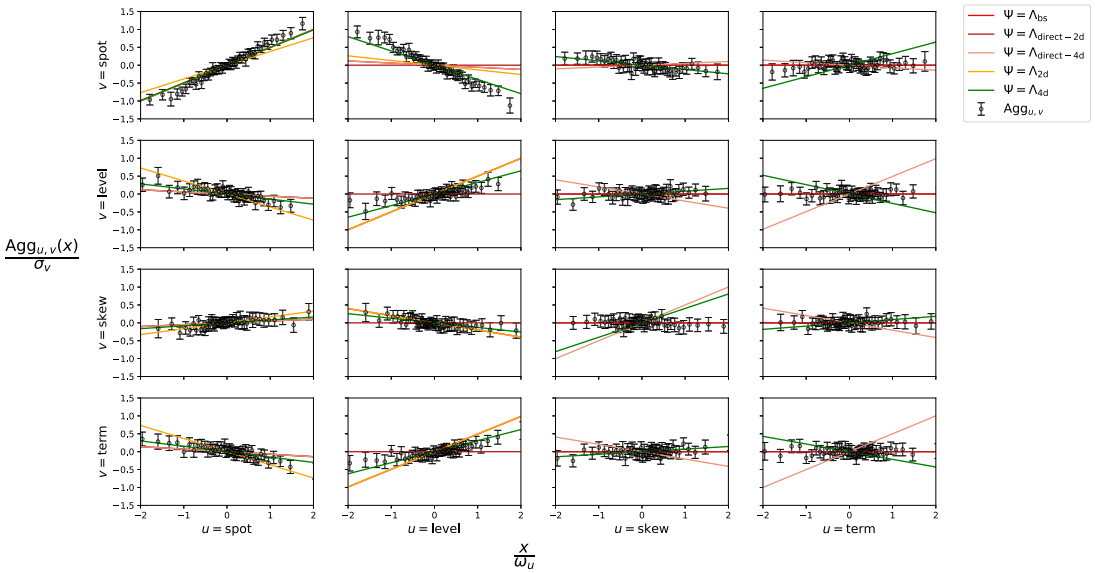


Figure II.5: **Normalized cross aggregate impact curves.**
We report the cross aggregate cross impact curves for the spot, level, skew and term structure directions. Aggregate traded volumes are normalized by the typical deviations $\omega_u^2 := \mathbb{E}_t[(u^\top(\delta q_t, \delta Q_t))^2]$ and portfolio returns by the typical deviations $\sigma_v^2 := \mathbb{E}_t[(v^\top(\delta p_t, \delta P_t))^2]$. Estimated cross aggregate impact $\mathrm{Agg}_{u,v}$ is reported along with predicted cross aggregate impact $\mathrm{Agg}_{u,v}^{\Psi}$ for different choices of linear cross-impact models Ψ.

Conclusion

The goal of this chapter was to derive an estimation methodology for cross-impact on derivatives. To do so, we introduced a market where derivatives and factors, variables which determine the prices of derivatives are co-traded and trades on one instrument induce impact on all instruments. We derived an explicit formula for cross-impact which is consistent with Ito's lemma and can be leveraged in practice. We applied the framework to E-Mini and VIX futures along with call and put options on the E-Mini.

One key result of this chapter is an estimation formula for linear cross-impact which is adaptable to the derivative modeling framework and is compatible with popular stochastic volatility models. This formula fits in conveniently with existing option pricing frameworks and could be readily adapted in applications, provided one has access to the net traded order flow on derivatives.

Finally, our framework gives a recipe for aggregating liquidity of derivative markets which accounts for the joint dynamics of order flows and is tractable in practice. This is a topic of interest for regulators and practitioners alike, as option liquidity is very fragmented.

II.A Proofs

This section contains proofs of the results of Sections 2 and 3. We begin with the proof of Proposition 1.

Proposition 1. *Using the notation of Assumptions 2 and 4 and Corollary 1, we have*

$$dp_t = \tilde{\mu}_p(p_t,t)dt + \frac{1}{\sqrt{1-Y}}\mathcal{G}_p(p_t,t)dB_t$$
$$dP_t = \tilde{\mu}_P(p_t,t)dt + \frac{1}{\sqrt{1-Y}}\mathcal{G}_P(p_t,t)dB_t\,,$$

where $\tilde{\mu}_p\colon \mathbb{R}^N\times\mathbb{R}\to\mathbb{R}^N$, $\tilde{\mu}_P\colon \mathbb{R}^N\times\mathbb{R}\to\mathbb{R}^M$ *and* B *is a standard* N *dimensional Brownian motion.*

Proof. We have, from Equations (3) and (4)

$$\mathrm{d}p_t = \mu_p(p_t,t)\mathrm{d}t + \mathcal{G}_p(p_t,t)\mathrm{d}W_t + \Lambda_{pq}(p_t,t)\mathrm{d}q_t + \Lambda_{pQ}(p_t,t)\mathrm{d}Q_t$$
$$\mathrm{d}P_t = \mu_P(p_t,t)\mathrm{d}t + \mathcal{G}_P(p_t,t)\mathrm{d}W_t + \Lambda_{Pq}(p_t,t)\mathrm{d}q_t + \Lambda_{PQ}(p_t,t)\mathrm{d}Q_t\,.$$

Using the order flow dynamics given in Assumption 1, we have

$$\begin{aligned}\mathrm{d}p_t &= \big(\mu_p(p_t,t) + \Lambda_{pq}(p_t,t)\nu_q(p_t,t) + \Lambda_{pQ}(p_t,t)\nu_Q(p_t,t)\big)\,\mathrm{d}t\\ &\quad + \mathcal{G}_p(p_t,t)\mathrm{d}W_t + \Lambda_{pq}(p_t,t)\mathcal{L}_{qq}(p_t,t)\mathrm{d}Z^q_t + \Lambda_{pQ}(p_t,t)\mathcal{L}_{qQ}(p_t,t)\mathrm{d}Z^Q_t\\ \mathrm{d}P_t &= \big(\mu_P(p_t,t) + \Lambda_{Pq}(p_t,t)\nu_q(p_t,t) + \Lambda_{PQ}(p_t,t)\nu_Q(p_t,t)\big)\,\mathrm{d}t\\ &\quad + \mathcal{G}_P(p_t,t)\mathrm{d}W_t + \Lambda_{Pq}(p_t,t)\mathcal{L}_{Qq}(p_t,t)\mathrm{d}Z^q_t + \Lambda_{PQ}(p_t,t)\mathcal{L}_{QQ}(p_t,t)\mathrm{d}Z^Q_t\,.\end{aligned}$$

Thus, defining $\tilde{\mu}_p := \mu_p + \Lambda_{pq}\nu_q + \Lambda_{pQ}\nu_Q$ and $\tilde{\mu}_P := \mu_P + \Lambda_{Pq}\nu_q + \Lambda_{PQ}\nu_Q$, the above reduces to

$$\begin{aligned}\mathrm{d}p_t &= \tilde{\mu}_p(p_t,t)\mathrm{d}t + \mathcal{G}_p(p_t,t)\mathrm{d}W_t + \Lambda_{pq}(p_t,t)\mathcal{L}_{qq}(p_t,t)\mathrm{d}Z^q_t + \Lambda_{pQ}(p_t,t)\mathcal{L}_{qQ}(p_t,t)\mathrm{d}Z^Q_t\\ \mathrm{d}P_t &= \tilde{\mu}_P(p_t,t) + \mathcal{G}_P(p_t,t)\mathrm{d}W_t + \Lambda_{Pq}(p_t,t)\mathcal{L}_{Qq}(p_t,t)\mathrm{d}Z^q_t + \Lambda_{PQ}(p_t,t)\mathcal{L}_{QQ}(p_t,t)\mathrm{d}Z^Q_t\,.\end{aligned}$$

The Kyle model is covariance-consistent (see [TMB20] for a proof): we have $\Lambda(p_t,t)\Omega(p_t,t)\Lambda^\top(p_t,t) = Y\Sigma(p_t,t)$. By construction

$$\langle(p,P)\rangle = \begin{pmatrix}\mathcal{G}_p\mathcal{G}_p^\top + & \mathcal{G}_p\mathcal{G}_P^\top\\ \mathcal{G}_P\mathcal{G}_p^\top & \mathcal{G}_P\mathcal{G}_P^\top\end{pmatrix} + \Lambda\Omega\Lambda^\top = \Sigma$$

So that we have

$$\begin{pmatrix}\mathcal{G}_p\mathcal{G}_p^\top + & \mathcal{G}_p\mathcal{G}_P^\top\\ \mathcal{G}_P\mathcal{G}_p^\top & \mathcal{G}_P\mathcal{G}_P^\top\end{pmatrix} = (1-Y)\Sigma$$

Therefore, there exists a standard Brownian motion B such that

$$\mathrm{d}p_t = \tilde{\mu}_p(p_t,t)\mathrm{d}t + \frac{1}{\sqrt{1-Y}}\mathcal{G}_p(p_t,t)\mathrm{d}B_t$$
$$\mathrm{d}P_t = \tilde{\mu}_P(p_t,t)\mathrm{d}t + \frac{1}{\sqrt{1-Y}}\mathcal{G}_P(p_t,t)\mathrm{d}B_t\,.$$

□

We now prove Proposition 2.

Proposition 2. *We have*

$$\Lambda(p_t,t)=\begin{pmatrix}\Lambda_{pq} & \Lambda_{pq}\Xi^\top\\ \Xi\Lambda_{pq} & \Xi\Lambda_{pq}\Xi^\top\end{pmatrix}(p_t,t)\,,$$

where we recall that $\Xi:=(\frac{\partial P^i}{\partial p^j})_{i,j}$ *is the* M *by* N *sensitivity matrix.*

Proof. First, note that by Assumption 2 combined with Proposition 1, we have that $P_t=F(p_t,t)$ and

$$\begin{aligned}\mathrm{d}p_t&=\tilde{\mu}_p(p_t,t)\mathrm{d}t+\frac{1}{\sqrt{1-Y}}\mathscr{G}_p(p_t,t)\mathrm{d}B_t\\ \mathrm{d}P_t&=\tilde{\mu}_P(p_t,t)\mathrm{d}t+\frac{1}{\sqrt{1-Y}}\mathscr{G}_P(p_t,t)\mathrm{d}B_t\,.\end{aligned}$$

Therefore, from using Ito's lemma, we have that $\mathscr{G}_P=\Xi\mathscr{G}_p$, where $\Xi_{ij}:=\dfrac{\partial P^i}{\partial p^j}$. Therefore, as shown in the proof of Proposition 1, we have

$$\begin{pmatrix}\mathscr{G}_p\mathscr{G}_p^\top+ & \mathscr{G}_p\mathscr{G}_P^\top\\ \mathscr{G}_P\mathscr{G}_p^\top & \mathscr{G}_P\mathscr{G}_P^\top\end{pmatrix}=\begin{pmatrix}\mathscr{G}_p\mathscr{G}_p^\top & \mathscr{G}_p\mathscr{G}_p^\top\Xi^\top\\ \Xi\mathscr{G}_p\mathscr{G}_p^\top & \Xi\mathscr{G}_p\mathscr{G}_p\Xi^\top\end{pmatrix}=(1-Y)\Sigma$$

Thus, the factor covariance matrix has the form

$$\Sigma=\begin{pmatrix}\Sigma_{pp} & \Sigma_{pp}\Xi^\top\\ \Xi\Sigma_{pp} & \Xi\Sigma_{pp}\Xi^\top\end{pmatrix}.$$

The Kyle cross-impact model is fragmentation invariant (see [TMB20] for a proof): if x is in $\ker(\Sigma)$ then x is in $\ker(\Lambda)$. The above implies that, for every $u\in\mathbb{R}^M$, the vector $(\Xi^\top u,u)$ is in $\ker(\Sigma)$, so that, by fragmentation invariance, $(\Xi^\top u,u)$ is in $\ker(\Lambda)$. Therefore, this implies that $\Lambda_{pQ}=\Lambda_{pq}\Xi^\top$.

Since Λ is symmetric, we also have that Λ_{pq} is symmetric and that $\Lambda_{Pq}=\Lambda_{pQ}^\top=\Xi\Lambda_{pq}$. Another application of fragmentation invariance implies that $\Lambda_{PQ}=\Lambda_{Pq}\Xi^\top=\Xi\Lambda_{pq}\Xi^\top$. Summarising, we have

$$\Lambda=\begin{pmatrix}\Lambda_{pq} & \Lambda_{pq}\Xi^\top\\ \Xi\Lambda_{pq} & \Xi\Lambda_{pq}\Xi^\top\end{pmatrix}$$

□

Finally, we prove the last proposition, Proposition 3.

Proposition 3. *We have*

$$\Lambda_{pq}=\sqrt{Y}(\mathscr{L}_\Xi^{-1})^\top\sqrt{\mathscr{L}_\Xi^\top\Sigma_{pp}\mathscr{L}_\Xi}\,\mathscr{L}_\Xi^{-1}\,,$$

where we have omitted the dependence on (p_t,t), $\Omega_\Xi:=\Omega_{qq}+\Xi^\top\Omega_{QQ}\Xi+\Xi^\top\Omega_{Qq}+\Omega_{qQ}\Xi$, *and* $\mathscr{L}_\Xi$ *is a matrix such that* $\mathscr{L}_\Xi\mathscr{L}_\Xi^\top=\Omega_\Xi$.

Proof. From Proposition 2, we have that

$$\Lambda = \begin{pmatrix} \Lambda_{pq} & \Lambda_{pq}\Xi^\top \\ \Xi\Lambda_{pq} & \Xi\Lambda_{pq}\Xi^\top \end{pmatrix}$$

The Kyle cross-impact model is covariance-consistent (see [TMB20] for a proof), i.e. we have

$$\begin{pmatrix} \Sigma_{pp} & \Sigma_{pP} \\ \Sigma_{Pp} & \Sigma_{PP} \end{pmatrix} = Y\Lambda\Omega\Lambda^\top.$$

Therefore, we obtain from the above that

$$\Sigma_{pp} = Y\Lambda_{pq}\Omega_\Xi\Lambda_{pq}^\top,$$

where $\Omega_\Xi := \Omega_{qq} + \Xi^\top\Omega_{QQ}\Xi + \Xi^\top\Omega_{Qq} + \Omega_{qQ}\Xi$. In particular, note that Ω_Ξ is a symmetric positive definite matrix. Since Λ is a symmetric positive definite matrix (see [TMB20] for a proof), so is Λ_{pq}. From [TMB20], we know that Λ_{pq} is the unique symmetric positive definite solution of the system $\Sigma_{pp} = \Lambda_{pq}\Omega_\Xi\Lambda_{pq}^\top$ and that its form is given by

$$\Lambda_{pq} = \sqrt{Y}(\Omega_\Xi^{-1/2})^\top\sqrt{(\Omega_\Xi^{1/2})^\top\Sigma_{pp}\Omega_\Xi^{1/2}}\Omega_\Xi^{-1/2}.$$

□

Lemma 1. *We denote by* $\Delta_c := (1, 0, \frac{\partial P^1}{\partial s}, \cdots, \frac{\partial P^M}{\partial s})$, $\mathcal{V}_c := (0, 1, \frac{\partial P^1}{\partial \varsigma}, \cdots, \frac{\partial P^M}{\partial \varsigma})$ *the vectors of sensitivities. Then, if* $\Delta_c^\top\Omega\mathcal{V}_c = 0$, *we have*

$$\Lambda_{pq}(s_t, \varsigma_t, t) = \frac{\sqrt{Y}}{\sqrt{\sigma^2\omega_\Delta^2 + \xi^2\omega_\mathcal{V}^2 + 2\sigma\xi\rho\omega_\Delta\omega_\mathcal{V}}}\begin{pmatrix} \sigma^2 + \frac{\omega_\mathcal{V}}{\omega_\Delta}\sigma\xi\sqrt{1-\rho^2} & \sigma\xi\rho \\ \sigma\xi\rho & \xi^2 + \frac{\omega_\Delta}{\omega_\mathcal{V}}\sigma\xi\sqrt{1-\rho^2} \end{pmatrix}(s_t, \varsigma_t, t),$$

where $\omega_\Delta^2 := \Delta_c^\top\Omega\Delta_c$ *is the delta-aggregated liquidity,* $\omega_\mathcal{V}^2 := \mathcal{V}_c^\top\Omega\mathcal{V}_c$ *is the vega-aggregated liquidity,* $\sigma^2(s_t, \varsigma_t, t)dt := d\langle s, s\rangle_t$ *is the spot volatility,* $\xi^2(s_t, \varsigma_t, t)dt := d\langle\varsigma, \varsigma\rangle_t$ *is the volatility of volatility and* $\rho(s_t, \varsigma_t, t)dt := \dfrac{d\langle\varsigma, s\rangle_t}{\xi(s_t, \varsigma_t, t)\sigma(s_t, \varsigma_t, t)}$ *is the spot-vol correlation.*

Proof. Using the results of Proposition 3, the cross-impact matrix Λ_{pq} is of the form

$$\Lambda_{pq}(s_t, \varsigma_t, t)\sqrt{Y}(\mathscr{L}_\Xi^{-1})^\top\sqrt{\mathscr{L}_\Xi^\top\Sigma_{pp}\mathscr{L}_\Xi}\mathscr{L}_\Xi^{-1}(s_t, \varsigma_t, t),$$

where $\Omega_\Xi := \Omega_{qq} + \Xi^\top\Omega_{QQ}\Xi + \Xi^\top\Omega_{Qq} + \Omega_{qQ}\Xi$, and $\mathscr{L}_\Xi$ is a matrix such that $\mathscr{L}_\Xi\mathscr{L}_\Xi^\top = \Omega_\Xi$. Using the notations of the lemma, we have

$$\Sigma_{pp}(s_t, \varsigma_t, t) = \begin{pmatrix} \sigma^2 & \sigma\xi\rho \\ \sigma\xi\rho & \xi^2 \end{pmatrix}(s_t, \varsigma_t, t) \qquad \Omega_\Xi(s_t, \varsigma_t, t) = \begin{pmatrix} \omega_\Delta^2 & 0 \\ 0 & \omega_\mathcal{V}^2 \end{pmatrix}(s_t, \varsigma_t, t).$$

Plugging the above into the formula for Λ_{pq} and applying some straightforward linear algebra gives the result. □

II.B Empirical details

II.B.1 Data

This section gives motivation about our choice of instruments, details on the data and methodology which were omitted in the main text for conciseness.

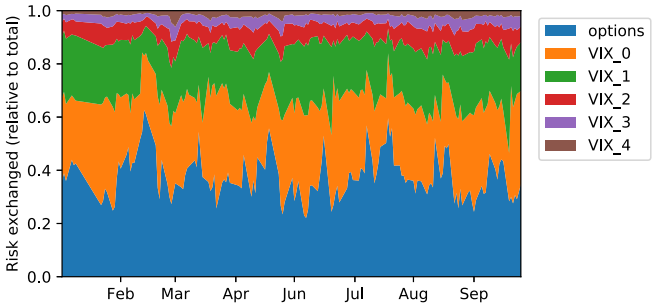


Figure II.6: **Distribution of liquidity among VIX futures and options.**

Choice of instruments To stress-test our approach, we sought an actively traded derivative market with many derivatives. Thus we considered E-mini vanilla options and their factor (both quoted on the CME), the front-month futures contract. However, a large fraction of the traded risk in derivatives comes from VIX futures (quoted on the CBOE) as shown in Figure II.6. The VIX index is computed using options with maturities between 23 and 37 days and is meant to track the level of the implied volatility for options expiring in one month. Thus, because of their liquidity and close relationship with the implied volatility of options, order flow traded on VIX futures play an important role and should not be ignored.

Filtering instruments Given the very large number of options quoted on the market, we kept options within a given range of strikes and maturities to limit the size of the data set.

Resolution and time frame Our dataset contains the trades and quotes of all previously selected products, at the five minute time scale, from January 2019 to September 2019. This time frame was chosen because of the large level of noise on derivatives' prices and the size of the data set which encumbered analysis. In a given five minute bin, signed trades were aggregated on their volumes, so that we have the opening and closing prices of instruments along with the aggregated signed traded order flow. We considered hours where both options and their factor are liquid, further removing 30 minutes around opening and closing for stationarity purposes. Doing so, data ranges between 3PM and 8:30PM UTC.

Implied volatility and greeks We now explain how implied volatility and Greeks were computed. For a given day and for a particular option, we have access to the opening bid and ask prices of that option for each five minute window. Furthermore, the bid and ask

Black-Scholes implied volatilities are computed using the bid and ask price of the option and the price of the E-mini future contract with closest maturity. Correspondingly, the usual Black-Scholes Greeks Δ and $\mathcal{V}$ are computed for both the bid and ask sides. In our analysis, we use the mid of bid and ask quantities (option prices, implied volatilities and greeks) to perform computations.

II.B.2 Estimation of covariances

This section details the estimation of the covariances Σ_{pp} and Ω_Ξ used in Section 4.

Estimation of Ω_Ξ In order to compute Ω_Ξ, we begin by computing $\Xi(p_t, t)$. Using Proposition 1, the sensitivities of the option prices are computed according to the usual Greeks adjusted for the option's sensitivity as described in Section 3.3. Thus, for a given time window $[t, t+\Delta t]$ of length $\Delta t = 5$ minutes, we measure the net traded order flow on factors δq_t and on derivatives δQ_t. We weigh order flow traded on derivatives according to sensitivities and build the aggregate net order flow on that time window: $\delta q_t + \Xi(p_t, t)\delta Q_t$. We measure this quantity over all available time windows and find that it is roughly stationary and does not depend on the value of the factors. This motivates dropping the dependence of these variables and the estimation of Ω_Ξ as the covariance of the aggregate order flow: $\Omega_\Xi \approx \mathrm{Cov}(\delta q_t + \Xi(p_t, t)\delta Q_t)$.

Estimation of Σ_{pp} After selection of the relevant factors detailed in Section 4.1, we estimate the factor return covariance matrix in the following manner. Given a time window $[t, t+\Delta t]$ of length $\Delta t = 5$ minutes, we measure the opening and closing price of all factors. The spot factor's price is taken as the price of the E-Mini future closest to expiry. For options, the factor values are computed by projecting option implied volatilities in the direction of the implied volatility surfaces: $\varsigma^q = \sum_{i=1}^M \beta^{iq}\hat{\sigma}^i$. This formula holds since, by construction, (β^{iq}) satisfies $\beta^\top\beta = I$. Thus, we compute opening and closing prices of the factors of the implied volatility surface. The factor covariance matrix Σ_{pp} is then estimated as the covariance matrix of price changes.

CHAPTER III

Dynamic cross-impact from anonymous order flow

Abstract

Trading a financial asset pushes its price as well as the prices of other assets, a phenomenon known as cross-impact. We consider a general class of kernel-based cross-impact models and investigate suitable parametrisations for trading purposes. We focus on kernels that guarantee that prices are martingales and anticipate future order flow (martingale-admissible kernels) and those that ensure there is no possible price manipulation (no-statistical-arbitrage-admissible kernels). We determine the overlap between these two classes and provide formulas for calibration of cross-impact kernels on data. We illustrate our results using SP500 futures data.

From:
A characterisation of cross-impact kernels
M. Rosenbaum, **M. Tomas**

Introduction

How do trades move prices of financial securities? It is well-known among practitioners and academics that buying a financial asset tends to push its price up while selling it tends to push its price down. This observation is one aspect of *market impact*, which describes how trades on one asset translate into its price. The many studies on market impact [BBDG18, ATHL05, Tor97] have deepened our understanding of how markets digest trades into prices. In turn, this has helped us understand key properties of dynamics of asset prices. For instance, market impact explains why price volatilities are well-modeled by rough fractional Brownian motions [JR18].

Yet classical market impact does not tell us how our trades influence prices of other assets. Thus, it ignores a potentially important aspect of price formation. As many assets are simultaneously traded in financial markets, this element is required to generate complete market dynamics and, ultimately, answer the question of how markets digest liquidity.

To solve this issue, a recent strand of works [RT21b, TMB20, WSG16, WNG17, BMEB17, AKS16, TMB21, SL19] has studied *cross-impact*, which describes how transactions on a universe of instruments drive their prices. This chapter contributes to the literature by characterising cross-impact models which lead to well-behaved market dynamics. We show how these models can be calibrated from empirical data and provide an example using SP500 futures.

We consider a stylised market made of d financial securities, continuously quoted and traded by market participants. Trading activity on financial markets is highly endogenous: statistically, trades trigger other trades [HBB13, BBDG18]. To capture this effect, we model trade dynamics in our market by Hawkes processes [Haw71a] and introduce the d-dimensional counting processes N_t^a / N_t^b denoting the number of buy/sell orders of all market participants over $[0, t]$. We classically assume that trades induce permanent impact which is linear in the traded volume, for reasons related to no-arbitrage [SL19, Gat10]. A large class of models which satisfy this property and has been well-studied when $d = 1$ is the class of propagator models [BBDG18]. Thus we restrict ourselves to this class and assume that d-dimensional price process P evolves as

$$P_t = P_0 + \int_0^t K(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b), \tag{1}$$

where $K\colon t \in \mathbb{R}_+ \mapsto K(t) \in \mathcal{M}_d(\mathbb{R})$ is a *cross-impact kernel*. The cross-impact kernel encodes all information about cross-impact in our market but, contrary to prices and trades, it is not directly observable. When it exists, the limit of the cross-impact kernel $\Lambda := \lim_{t\to\infty} K(t)$ is called the permanent cross-impact matrix since Λ_{ij} quantifies the permanent price impact of a trade on Asset j on the price of Asset i.

Within Equation (1), we examine two different classes of cross-impact kernels: those that anticipate upcoming order flow and yield martingale prices, which we dub *martingale-admissible*, and those that prevent statistical arbitrage, which we dub *no-statistical-arbitrage-admissible*, or *nsa-admissible* for short. Statistical arbitrage is meant in the sense of [Gat10]: a statistical arbitrage is a trading strategy that starts and ends with no asset holdings and has negative expected costs.

Martingale-admissible and nsa-admissible kernels enforce different aspects of price efficiency. Martingale-admissible kernels ensure that prices are not predictable and that information flow is reflected in the current price, so that no trading strategy can make a profit by forecasting prices or order flows. On the other hand, nsa-admissible kernels prevent price manipulation by large agents who could push prices to make a profit.

The main contribution of this chapter is the characterisation of the class of martingale-admissible and nsa-admissible kernels with respect to price and order flow statistics. In particular, at most one cross-impact kernel is both martingale-admissible and nsa-admissible. This characterisation can be used for calibration on real data and we provide an application on market data to illustrate our results.

We now comment on the links between our approach and the literature.

This market model using Hawkes processes in a propagator framework is a generalisation of the model from [JR18, Jai15], where only one asset is considered. In the single-asset case, this market dynamic is consistent with many empirical results concerning market impact. Therefore the multivariate generalisation of this framework will stay consistent with these findings while providing insights into cross-impact.

The chapters [SL19, BMEB17] study a class of cross-impact kernels which give rise to martingale prices. This condition is used to obtain a calibration methodology. Though the calibration methodology based on maximum likelihood is straightforward, the resulting cross-impact kernels are noisy, prone to overfitting and they have no guarantees of no-arbitrage. We show that all martingale-admissible kernels, including those presented in [SL19, BMEB17], have a certain form.

The class of nsa-admissible kernels has been described in [AKS16]. However, the provided characterisation is quite theoretical and gives no insight into which cross-impact kernels to choose in practice. This chapter extends some of the results of [AKS16]. In particular, we show that nsa-admissible kernels have constrained values at zero and infinity which are related to price and order flow statistics. Unfortunately, there are many kernels which are nsa-admissible but which lead to ill-behaved market dynamics, as pointed out in [AKS16]. We provide a methodology for obtaining a nsa-admissible kernel which is close to a martingale-admissible kernel. Doing so, we obtain a kernel which is still faithful to empirical data while preventing statistical arbitrage.

The resulting kernels can be used on market data to estimate cross-impact. Thus, the chapter adds to the literature focusing on calibrating cross-impact kernels [HS01, PV15, SL19, WSG15, BMEB17]. While some kernels are nsa-admissible such as the eigenliquidity cross-impact kernels [BMEB17], others are martingale-admissible [BMEB17, SL19]. This chapter provides kernels which can be easily calibrated in both classes.

Finally, the boundary values $K(0)$ and $\lim_{t\to\infty} K(t)$ of any cross-impact kernel K which is martingale-admissible and satisfies necessary conditions for nsa-admissability have a microscopic foundation. Indeed, both can be interpreted as solutions to the multivariate version of Kyle's insider trading problem [GdMMBB20]. Thus, although cross-impact is purely a reaction to order flow imbalance in our model, the cross-impact kernel can also be interpreted through the lens of information revelation.

The chapter is organized as follows. In Section 1, we describe our financial market. In Section 2, we characterise the classes of martingale-admissible and nsa-admissible kernels. Finally, we apply our results on market data in Section 3 before concluding in Section 4. Some proofs and additional results are relegated to an appendix.

Notation

The set of $d \times d$ real-valued square matrices is denoted by $\mathcal{M}_d(\mathbb{R})$, the set of orthogonal (also called rotation) matrices by $\mathcal{O}_d$, the set of real symmetric matrices by $\mathcal{S}_d^+(\mathbb{R})$ and the set of real symmetric positive matrices by $\mathcal{S}_d^{++}(\mathbb{R})$. Furthermore, given a matrix A in $\mathcal{M}_d(\mathbb{R})$, $A^\top$ denotes its transpose. Given A in $\mathcal{S}_d^{++}(\mathbb{R})$, we write $A^{1/2}$ for a matrix such that $A^{1/2}(A^{1/2})^\top = A$ and $\sqrt{A}$ for the square root matrix, the unique positive semi-definite symmetric matrix such that $(\sqrt{A})^2 = A$. Finally, given a vector $v \in \mathbb{R}^d$, we write $v = (v_1, \dots, v_d)$ and $\mathrm{diag}(v)$ for the diagonal matrix with entries the elements of v.

A matrix $M \in \mathcal{M}_d(\mathbb{R})$ is called non-negative if for any $x \in \mathbb{R}^d$, $x^\top M x \geq 0$. It is called non-negative definite if $z^* M z \geq 0$ for any $z \in \mathbb{C}^d$. A matrix $M \in \mathcal{M}_d(\mathbb{C})$ is called strictly positive if $x^\top M x > 0$ for any nonzero $x \in \mathbb{R}^d$ and strictly positive definite if $z^* M z > 0$ for any nonzero $z \in \mathbb{C}^d$. The conjugate transpose of a matrix $M \in \mathcal{M}_d(\mathbb{C})$ is written M^*.

A function $f\colon \mathbb{R} \to \mathbb{R}$ is called causal if $f(t) = 0$ for all negative t. The same is said of a vector-valued or matrix-valued function if all its entries are causal. Given an integrable function f, we denote its Fourier transform $\widehat{f}\colon \mathbb{C} \to \mathbb{C}$ defined, for all $\omega \in \mathbb{C}$ such that the integral converges, by

$$\widehat{f}(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} \mathrm{d}t.$$

Similarly, we define the Fourier transform of a vector-valued or matrix-valued function with integrable entries by the vector-valued or matrix-valued function Fourier transform of all its entries. For a given real-valued measure $\mu\colon \mathscr{B}(\mathbb{R}) \to \mathbb{R}$, we denote its Fourier transform by $\widehat{\mu}\colon \mathbb{C} \to \mathbb{C}$ defined, for all $\omega \in \mathbb{C}$ such that the integral converges, by

$$\widehat{\mu}(\omega) = \int_{-\infty}^{\infty} e^{-i\omega t} \mu(\mathrm{d}t).$$

Similarly, we define the Fourier transform of a vector or matrix with measure entries by the Fourier transform of all its entries. All stochastic processes in the text are defined on a probability space $(\Omega, \mathscr{F}, (\mathscr{F}_t)_{t\in\mathbb{R}}, \mathbb{P})$. Given two semi-martingale processes X and Y, we denote by $\langle X, X\rangle$ the predictable quadratic variation of X and $\langle X, Y\rangle$ the predictable quadratic covariation of X and Y.

1 Market model

This section presents the stylised market model in force throughout the chapter. Our setting extends [Jai15, JR18] to the multivariate case. For a lengthier discussion about these assumptions in the univariate case, we refer the reader to [Jai15, JR18].

We consider a market made of d different assets, quoted and traded continuously in time. We greatly simplify the market by abstracting away microstructural features and assume that agents can buy and sell Asset i at time t at the unique quoted price P_t^i. The d-dimensional

price process is denoted by $P := (P^1, \cdots, P^d)$. We assume that agents trade from time 0 onward. During this period, the cumulative traded volume by all agents at the ask (resp. bid) is denoted by V^a (resp. V^b) and the net traded volume by $V := V^a - V^b$.

A key property of the order flow in financial markets is its persistence: the sign correlation of orders is slowly decaying in time [BBDG18]. Because of this effect, a particularly successful model for order flow dynamics is the Hawkes process which can capture self-excitation and cross-excitation across time and instruments [Haw71a, BMM15]. Thus we will assume that the order flow dynamics are given by a Hawkes process.

Assumption 1 (Hawkes order flow). *The number of buy and sell market orders follows a Hawkes process (N^a, N^b), of intensity (λ^a, λ^b) and kernel $\Phi = \begin{pmatrix} \Phi^{a/a} & \Phi^{a/b} \\ \Phi^{b/a} & \Phi^{b/b} \end{pmatrix}$ such that*

$$\lambda_t^a = \mu + \int_0^t \Phi^{a/a}(t-s)\, dN_s^a + \int_0^t \Phi^{a/b}(t-s)\, dN_s^b$$
$$\lambda_t^b = \mu + \int_0^t \Phi^{b/a}(t-s)\, dN_s^a + \int_0^t \Phi^{b/b}(t-s)\, dN_s^b,$$

where in the above

- *the vector $\mu \in \mathbb{R}_+^d$ is the exogenous intensity of buy and sell market orders;*
- *the entry-wise integrable matrix function $\Phi^{a/a} \colon t \mapsto \Phi^{a/a}(t) \in \mathcal{M}_d(\mathbb{R}_+)$ (resp. $\Phi^{b/b} \colon t \mapsto \Phi^{b/b}(t) \in \mathcal{M}_d(\mathbb{R}_+)$) encodes the endogenous contribution of past buy (resp. sell) market orders on the intensity of buy (resp. sell) market orders;*
- *the entry-wise integrable matrix function $\Phi^{a/b} \colon t \mapsto \Phi^{a/b}(t) \in \mathcal{M}_d(\mathbb{R}_+)$ (resp. $\Phi^{b/a} \colon t \mapsto \Phi^{b/a}(t) \in \mathcal{M}_d(\mathbb{R}_+)$) encodes the endogenous contribution of past sell (resp. buy) market orders on the intensity of buy (resp. sell) market orders.*

We assume that the Hawkes parameters are such that $\mathbb{E}[\lambda_t^a] = \mathbb{E}[\lambda_t^b]$ for any t. Each market order on Asset i is assumed to be of constant size v_i and the spectral radius of the L^1 norm of the Hawkes kernel Φ is assumed to be below one. The latter assumption allows us to define the stationary version of the Hawkes process (in fact stationary intensity).

This framework allows for rich multivariate dynamics since we can account for self-excitation and buy/sell interactions between different assets through $\Phi_{ij}^{b/a}$ and $\Phi_{ji}^{a/b}$. We assumed that the market there are as many buy market orders than sell market orders on the small time scales of the market model.

While the order flow is persistent, the order flow at time t should not give information about the order flow at time $t' \gg t$. We formalize this in the next assumption.

Assumption 2 (Finitely predictable orderflow). *For all $t \geq 0$, $\mathbb{E}[V_s^a - V_s^b \mid \mathcal{F}_t]$ converges in probability to some finite limit as s tends to infinity.*

Note that this assumption implies additional constraints on the Hawkes parameters.

In our market, prices are driven by the transactions of all agents. To exclude trivial arbitrages and keep the model simple, it is natural to assume that the permanent component of the cross-impact is a linear function of the order flow (see for example Corollary 3.7 of [SL19]) and to consider a propagator framework as explained in the introduction [BBDG18]. This leads to the following assumption.

Assumption 3 (Price dynamics). *There exists some function* $K\colon t \in \mathbb{R}_+ \mapsto K(t) \in \mathcal{M}_d(\mathbb{R})$*, called a cross-impact kernel, such that the price process* P *satisfies, for all* $t \in \mathbb{R}_+$

$$P_t = P_0 + \int_0^t K(t-s)(dN_s^a - dN_s^b), \tag{2}$$

and $K(t) \underset{t\to\infty}{\to} \Lambda$*, where* Λ *is an invertible* $d \times d$ *matrix called the permanent cross-impact matrix.*

The diagonal functions K_{ii} of the cross-impact kernel relate past order flow on a security to its price. Off-diagonal elements K_{ij} relate past order flow of Asset j to the price of Asset i. The matrix Λ is called the permanent cross-impact matrix since Λ_{ij} quantifies how much the price of asset i is moved by the net order flow on asset j after a long period.

Finally, we make a technical assumption about the continuity of the Hawkes kernel at the origin and its decay at infinity.

Assumption 4. *The Hawkes order flow kernel* Φ *is continuously differentiable at zero and square-integrable.*

This assumption is not really constraining since Hawkes kernels for order flows are found to be square-integrable when calibrated on financial data [HBB13].

Previous hypotheses have not touched on the efficiency of prices in our market. Without imposing additional assumptions, prices may be highly predictable or agents could manipulate them through trading to generate profits. These two concepts of price efficiency are not always compatible, even in this stylised model. As such, we need to distinguish between cross-impact kernels which give martingale prices and those that prevent statistical arbitrage.

The class of cross-impact kernels which give rise to martingale prices also includes trivial examples, such as $K = 0$. To exclude these, we introduce *martingale-admissible* kernels, which anticipate the impact contribution of the order flow and lead to martingale prices. Such kernels generate non-trivial price dynamics since they incorporate the impact contribution of trades in prices. In the univariate case $d = 1$, Theorem 2.1 of [Jai15] shows that when prices are martingales and trades impact prices, we have

$$P_t - P_0 = \kappa \lim_{s\to\infty} \mathbb{E}[V_s^a - V_s^b \mid \mathcal{F}_t],$$

where $\kappa > 0$ is the permanent market impact contribution. This motivates the following definition for *martingale-admissible* kernels.

Definition 1 (Martingale-admissible kernels). *A cross-impact kernel K is said to be martingale-admissible if*

$$P_t - P_0 = \int_0^t K(t-s)(dN_s^a - dN_s^b) = \Lambda \lim_{s\to\infty} \mathbb{E}[V_s^a - V_s^b \mid \mathscr{F}_t]\,. \tag{3}$$

We later show that martingale-admissible kernels lead to martingale prices. Such kernels anticipate the market order flow to set martingale prices according to linear permanent cross-impact. This prevents agents who successfully forecast order flow to trade profitably. However, it does not forbid statistical arbitrages entirely. Before introducing relevant definitions, we define trading strategies within our market model below.

Definition 2 (Trading strategy). *The buy and sell trades sent under the trading strategy $f : \mathbb{R} \to \mathbb{R}^d$ are d-dimensional Poisson processes n^a and n^b, independent of the Hawkes process (N^a, N^b), with intensities given by $f^a := \max(f, 0)$ and $f^b := \max(-f, 0)$. The (average) cost of the trading strategy f is*

$$C(f) := \int_0^\infty \int_0^t f(t)^\top K(t-s) f(s) ds dt\,. \tag{4}$$

If $\int_0^\infty f(s)ds = 0$, the trading strategy is called a round-trip strategy and if f has finite support, it is called a finite-horizon trading strategy.

All trading strategies considered in this chapter have deterministic intensity, ignore exchange fees, bid-ask spreads and other microstructural trading costs, so that trading costs are exclusively induced by market impact. The average cost Equation (4) is derived since under the agent's trading strategy the price process becomes

$$P_t = P_0 + \int_0^t K(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b + \mathrm{d}n_s^a - \mathrm{d}n_s^b)\,,$$

so that the average trading cost of the strategy is

$$\begin{aligned}\mathbb{E}\left[\int_0^\infty (\mathrm{d}n_t^a - \mathrm{d}n_t^b)^\top (P_t - P_0)\right] &= \mathbb{E}\left[\int_0^\infty (f^a(t) - f^b(t))^\top (P_t - P_0)\mathrm{d}t\right] \\ &= \mathbb{E}\left[\int_0^\infty \int_0^t (f^a(t) - f^b(t))^\top K(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b + \mathrm{d}n_s^a - \mathrm{d}n_s^b)\mathrm{d}t\right].\end{aligned}$$

Therefore, since the counting processes n^a and n^b are independent from each other and from the Hawkes process (N^a, N^b) and $\mathbb{E}[\lambda_t^a] = \mathbb{E}[\lambda_t^b]$ for all t, we obtain

$$\begin{aligned}\mathbb{E}\left[\int_0^\infty (\mathrm{d}n_t^a - \mathrm{d}n_t^b)^\top (P_t - P_0)\right] &= \int_0^\infty \int_0^t (f^a(t) - f^b(t))^\top K(t-s)(f^a(s) - f^b(s))\mathrm{d}s\mathrm{d}t \\ &= \int_0^\infty \int_0^t f(t)^\top K(t-s) f(s)\mathrm{d}s\mathrm{d}t\,.\end{aligned}$$

This justifies Equation (4). Trading strategies which are profitable on average are called statistical arbitrages and defined below.

Definition 3 (Statistical arbitrage). *A statistical arbitrage is a finite horizon, round-trip trading strategy such that its costs are negative:*

$$C(f) < 0\,.$$

Cross-impact kernels which allow for statistical arbitrage induce important issues for applications. For example, they would bias trading strategies which seek to minimise trading costs towards trading-induced price manipulation. For such problems, we require trading costs models with theoretical guarantees of no statistical arbitrage. We call *no-statistical-arbitrage-admissible* (or *nsa-admissible* for short) cross-impact kernels that prevent statistical arbitrage.

Definition 4 (No-statistical-arbitrage-admissible kernels). *A cross-impact kernel K is said to be nsa-admissible if there are no possible statistical arbitrages, i.e. no round-trip trading strategies with average negative cost.*

Finally, we make the distinction between readily available information, such as prices and trades, and non-directly observable information, such as the cross-impact kernel K or the permanent cross-impact matrix Λ. We refer to empirical observables for information which is easily mesaurable.

Definition 5 (Empirical observables). *An empirical observable is a first or second-order moment measure of the price or order flows counting processes.*

Definition of the moment measures are given in Section III.A. Loosely speaking, they can be seen as the moments of our market variables. Empirical observables play an important role: they can be understood as key features of our stylised market, which we measure and use to derive the cross-impact kernel K. The next section shows that they constrain the class of relevant cross-impact kernels.

Our model being set, the next section presents the main results of the chapter.

2 Characterisation of cross-impact kernels

The previous section introduced the framework in force throughout the chapter. We now characterise the cross-impact kernels K which emerge from these assumptions, depending on the hypotheses on the market. We characterise martingale-admissible kernels in Section 2.1 and nsa-admissible kernels in Section 2.2. Finally, Section 2.3 concludes on the cross-impact kernels which are both martingale-admissible and nsa-admissible.

2.1 Characterisation of martingale-admissible kernels

In this section, we focus on characterising martingale-admissible kernels. We begin by characterising martingale-admissible cross-impact kernels as a function of Hawkes parameters and the permanent cross-impact matrix Λ. Then, we express these cross-impact kernels as a function of empirical observables.

2.1.1 Cross-impact kernel as a function of Hawkes parameters

The following proposition derives the martingale-admissible cross-impact kernels K as a function of Hawkes parameters and the permanent cross-impact matrix Λ.

Proposition 1. *For any martingale-admissible kernel K, the price is a martingale, $\Phi^{b/b} - \Phi^{a/b} = \Phi^{a/a} - \Phi^{b/a}$ and for all $t \in \mathbb{R}_+$*

$$K(t) = K(0)(\mathbb{I}_d - \int_0^t \varphi(s)ds),$$

where we have introduced the imbalance kernel

$$\varphi := \Phi^{b/b} - \Phi^{a/b} = \Phi^{a/a} - \Phi^{b/a}.$$

Furthermore, the immediate cross-impact matrix and permanent cross-impact matrix are related as follows:

$$K(0) = \Lambda\left(\mathbb{I}_d - \int_0^\infty \varphi(s)ds\right)^{-1} \mathrm{diag}(v)^{-1}.$$

The proof of Proposition 1 is given in Section III.C.1. Note that, by Proposition 1 and Assumption 4, any martingale-admissible cross-impact kernel K is almost-everywhere differentiable and its derivative is square-integrable. Proposition 1 provides an expression for K as a function of Hawkes parameters and the permanent cross-impact matrix Λ. However, the imbalance kernel φ is hard to estimate and we cannot measure the permanent cross-impact matrix Λ on real data. We would like to derive an analogous expression using solely empirical observables which can be easily measured on empirical data. This is the topic of the next section.

2.1.2 Cross-impact kernel as a function of empirical observables

To derive an expression for martingale-admissible cross-impact kernels, it is convenient to introduce the stationary version of the Hawkes order flow process. We write $\tilde{N}$ for the stationary version of the Hawkes process with baseline μ and kernel Φ (this process exists and is unique since the spectral radius of the L^1 norm of Φ is smaller than one, see e.g. [BM96]). Loosely speaking, $\tilde{N}$ describes the long-term behaviour of N. The stationary version allows us to define properly empirical observables but we still consider that the order flow process is given by the non-stationary process.

We write $\Omega^{\tilde{N}}$ and Ω for the reduced covariance measures, defined in Section III.A, of the multivariate stationary point processes $\tilde{N}$ and $\tilde{N}^a - \tilde{N}^b$. By construction, we have $\Omega = (\mathbb{I}_d, -\mathbb{I}_d)\Omega^{\tilde{N}}(\mathbb{I}_d, -\mathbb{I}_d)^\top$.

Using the above notations, the following proposition relates martingale-admissible kernels to empirical observables and the boundary values of the cross-impact kernel $K(0)$ and $\lim_{t\to\infty} K(t) = \Lambda$.

Proposition 2. *Any martingale-admissible cross-impact kernel K satisfies, for almost all $\omega \in \mathbb{R}$*

$$\widehat{K'}(\omega) = \frac{1}{\sqrt{2}} \mathscr{G}\mathscr{O}\mathscr{L}(\omega)^{-1} - K(0)\,, \tag{5}$$

where $\mathscr{G}$ is any matrix such that $\mathscr{G}\mathscr{G}^\top = \Lambda \int_0^\infty \Omega(ds)\Lambda^\top$, $\mathscr{L}$ is any spectral factor of Ω (see Definition 14) and $\mathscr{O}$ is the unique rotation matrix such that

$$\Lambda = \frac{1}{\sqrt{2}} \mathscr{G}\mathscr{O}\mathscr{L}(0)^{-1}\,,$$

where we recall that $\lim_{t\to\infty} K(t) = \Lambda$ and, by Proposition 1 and Assumption 4, K is almost-everywhere differentiable and its derivative is square-integrable.

The proof of Proposition 2 is given in Section III.C.2. Proposition 2 completely characterises the derivative of martingale-admissible cross-impact kernels as a function of quantities easily measurable on data through Σ, Ω and the boundary values $K(0)$ and Λ. However, these are not known a priori. Thus, for a given set of empirical observables, martingale-admissible cross-impact kernels may only differ by their boundary values.

The above characterisation for martingale-admissible cross-impact kernels is useful to calibrate martingale-admissible kernels and we make use of it in Section 3. However, to do so, we must choose values for $K(0)$ and Λ. An outstanding question is thus that of appropriate values, which we address in the next section where we find that nsa-admissible kernels have constrained boundary values.

2.2 Characterisation of nsa-admissible kernels

The previous section examined martingale-admissible cross-impact kernels. We found that such cross-impact kernels are completely constrained – except at the boundaries. In this section, we focus on nsa-admissible cross-impact kernels. Contrary to the previous section, we will find that nsa-admissible cross-impact kernels are largely unconstrained, except at the boundary values. We begin by showing the latter.

2.2.1 Constraints on the boundary values of the cross-impact kernel

This section derives the boundary values for nsa-admissible cross-impact kernels. The following proposition characterises the immediate cross-impact matrix for any nsa-admissible kernel. We introduce $\theta := (\mathbb{I}_d - \Phi^{a/a})\mu + \Phi^{a/b}\mu$ which represents the stationary average of the intensity of incoming buy or sell orders.

Proposition 3. *For any nsa-admissible kernel K, we have*

$$K(0) = \frac{1}{\sqrt{2}} (\mathscr{L}_0^{-1})^\top \sqrt{\mathscr{L}_0^\top \Sigma \mathscr{L}_0}\, \mathscr{L}_0^{-1}\,, \tag{6}$$

where

1. *the matrix $\mathscr{L}_0$ is any matrix such that $\mathscr{L}_0\mathscr{L}_0^\top = \mathrm{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$,*
2. *the matrix $\Sigma := \lim_{t\to\infty} \mathbb{E}[d\langle P, P\rangle_t]$ is loosely speaking the stationary instantaneous covariance matrix of returns. The existence of this limit is shown in the proof of the proposition.*

The proof of Proposition 3 is given in Section III.C.3. Note that $K(0)$ does not depend on the choice of $\mathscr{L}_0$ such that $\mathscr{L}_0\mathscr{L}_0^\top = \mathrm{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$. The matrix $K(0)$ has a microscopic interpretation. Indeed, within Kyle's insider trading model [Kyl85] extended to multiple assets [GdMMBB20, CK94] the market-maker adjusts his quotes according to the pricing rule Gq, where G is called the Kyle cross-impact matrix and $q = q_{IT} + q_{NT}$ is the aggregate order flow of the insider and noise traders. In this model, if the price-covariance matrix is Σ and the noise order-flow covariance matrix $\mathbb{E}[(q_{NT})^\top q_{NT}]$ is $\Omega(\{0\}) = \mathrm{diag}(\theta_1 v_1^2, \cdots, \theta_d v_d^2)$, then $G = K(0)$, where $K(0)$ is given by Proposition 3. In our model, $\Omega(\{0\})$ represents the instantaneous covariance matrix of order flow, which is diagonal since there are no simultaneous orders on different assets. Thus, although no agents in our model have information, $K(0)$ can be interpreted through the lens of information revelation.

With a quite similar proof as that of Proposition 3, we can show that the permanent cross-impact matrix for any nsa-admissible kernel is symmetric non-negative.

Proposition 4. *For any nsa-admissible kernel K, the matrix $\lim_{t\to\infty} K(t) = \Lambda$ is symmetric non-negative.*

The proof of Proposition 4 is given in Section III.C.4. Importantly, the elements of Equation (6) can be estimated quite easily on data. Thus, for any nsa-admissible kernel K, $K(0)$ can be expressed solely as a function of market observables. On the other hand, the permanent cross-impact matrix $\lim_{t\to\infty} K(t) = \Lambda$ is only constrained by symmetry and non-negativeness.

2.2.2 Constraints on the Fourier transform of the cross-impact kernel

The previous section showed that boundary values of nsa-admissible kernels are constrained and that their boundary value at zero is completely characterised. In this section, we derive necessary and sufficient conditions for nsa-admissible kernels. We begin with a result from [AKS16], which holds in a more general setting than the one in force in this chapter, given in the lemma below.

Lemma 1 (Theorem 2.10 of [AKS16]). *A continuous cross-impact kernel K is nsa-admissible if and only if there exists a matrix-valued non-negative definite Hermitian measure $\mathbb{M}$ such that for all $t \in \mathbb{R}$ we have*

$$\mathscr{Z}(t) = \int_{\mathbb{R}} e^{i\gamma t} \mathbb{M}(d\gamma),$$

where

$$\mathscr{Z}(t) := \begin{cases} K(t) \text{ if } t > 0 \\ K(0) \text{ if } t = 0 \\ K(-t)^\top \text{ if } t < 0 \end{cases}.$$

Proof. Since $K(0)$ is symmetric and K is continuous this result stems from Theorem 2.10 of [AKS16]. □

In our framework, we can extend the previous result and prove some properties concerning the smoothness of nsa-admissible kernels, which is the topic of the next proposition.

Proposition 5. *A continuous cross-impact kernel K is nsa-admissible kernel if and only if, using the notations of Lemma 1, one of the following identities holds for almost all $t \in \mathbb{R}$*

$$\mathcal{Z}'(t) = i\int_{\mathbb{R}} \gamma e^{i\gamma t}\mathbb{M}(d\gamma)$$
$$\mathcal{Z}''(t) = -\int_{\mathbb{R}} \gamma^2 e^{i\gamma t}\mathbb{M}(d\gamma)\,,$$

where $\mathbb{M}$ is a matrix-valued non-negative definite Hermitian measure such that $\mathcal{Z}(t) = \int_{\mathbb{R}} e^{i\gamma t}\mathbb{M}(d\gamma)$ and each integral converges absolutely, i.e. each integrand is absolutely integrable with respect to the measure $\mathbb{M}$. If any of these conditions is satisfied, the matrix function $\mathcal{Z}$ has twice continously differentiable entries.

The proof of Proposition 5 is given in Section III.C.5. The regularity properties of Proposition 5 enable us to check monotonicity and convexity of continuous nsa-admissible cross-impact kernels, a topic of interest as discussed in [AKS16].

2.3 Characterisation of martingale and nsa-admissible kernels

This section summarises the results from the two previous ones to characterise kernels which are martingale-admissible and nsa-admissible. We have seen that martingale-admissible kernels are constrained everywhere except at the boundaries, while nsa-admissible kernels are constrained at the boundaries but are largely unconstrained elsewhere. It is thus natural to observe that there is at most one kernel that is both martingale-admissible and nsa-admissible. In practice, this candidate kernel is not always nsa-admissible. For applications, it may be interesting to slightly relax the martingale property in order to guarantee no statistical arbitrage. Thus, we introduce a regularisation technique to obtain a kernel close to the kernel which gives martingale prices but that prevents arbitrage.

The next proposition shows that for kernels which are both nsa-admissible and martingale-admissible, the permanent cross-impact matrix is fixed. We recall that the matrix $\Sigma = \lim_{t\to\infty} \mathbb{E}[\mathrm{d}\langle P, P\rangle_t]$ is loosely speaking the stationary instantaneous covariance matrix of returns.

Proposition 6. *For any nsa-admissible, martingale-admissible kernel K, we have*

$$\lim_{t\to\infty} K(t) = \Lambda = \frac{1}{\sqrt{2}}(\mathscr{L}_\infty^{-1})^\top \sqrt{\mathscr{L}_\infty^\top \Sigma \mathscr{L}_\infty}\,\mathscr{L}_\infty^{-1}\,, \tag{7}$$

where the matrix $\mathscr{L}_\infty$ is any matrix such that $\mathscr{L}_\infty\mathscr{L}_\infty^\top = \int_0^\infty \Omega(ds)$, which is loosely speaking the stationary total autocovariance matrix of order flows.

Proof. As K is both martingale-admissible and nsa-admissible, Propositions 2 to 4 imply that Λ is a symmetric, non-negative matrix that satisfies

$$\Lambda \int_0^\infty \Omega(\mathrm{d}s)\Lambda^\top = \frac{1}{2}\Sigma.$$

The result follows. □

The previous proposition highlights that, as $K(0)$ in Proposition 3, the permanent cross-impact matrix given in Equation (7) has a microscopic interpretation. However, while $K(0)$ can be interpreted as the market-maker pricing rule in a market where the price covariance is Σ and the noise order flow covariance matrix is $\Omega(\{0\}) = \mathrm{diag}(\theta_1 \nu_1^2, \cdots, \theta_d \nu_d^2)$, the permanent cross-impact matrix Λ can be interpreted as the market-maker pricing rule in a market where the price covariance is Σ and the order flow covariance is $\int_0^\infty \Omega(\mathrm{d}s)$. The latter can be interpreted as the total order flow covariance which encapsulates instantaneous and non-instantaneous liquidity.

The next theorem summarises the results of the chapter to completely characterise cross-impact kernels which are both martingale-admissible and nsa-admissible.

Theorem 1. *There exists a unique cross-impact kernel K that is martingale-admissible and which satisfies the necessary conditions for arbitrage-admissibility outlined in Equations* (6) *and* (7). *Its expression is given by inverting Equation* (5) *and setting the boundary values given by Equations* (6) *and* (7). *Furthermore, if it satisfies Proposition 5, then K is also nsa-admissible.*

Proof. Let K be a cross-impact kernel which is both martingale-admissible and satisfies Equations (6) and (7). Then by Proposition 1, it is continuous. Furthermore, it must satisfy Propositions 1 and 3 so that it is unique and its boundary values are given by Equations (6) and (7). Since it is continuous, it is nsa-admissible if and only if it satisfies the necessary and sufficient conditions of Proposition 5 (or Lemma 1). □

The theorem shows there exists only one martingale-admissible kernel which satisfies the boundary conditions of nsa-admissible kernels. Though it may not be nsa-admissible, it is certainly closer to being nsa-admissible than other martingale-admissible kernels since it satisfies necessary conditions of nsa-admissible kernels. Given its importance, we write this cross-impact kernel K^1 in the following.

Definition 6 (K^1 kernel). *The cross-impact kernel K^1 is the unique martingale-admissible kernel that satisfies Equations* (6) *and* (7).

While K^1 is a good candidate for applications, we have no guarantee that this kernel is nsa-admissible. This naturally poses issues in certain applications. For example, in portfolio optimization, a trading cost model which allows for arbitrages induces spurious round-trip strategies, as shown in [AKS16]. Thus, we introduce a regularisation method to find a kernel close to this candidate but which is nsa-admissible, which we write K^2. This motivates to the following definition.

Definition 7 (K^2 kernel). *The cross-impact kernel K^2 is defined as*

$$K^2 = \underset{K\text{arbitrage}-\text{admissible}}{\operatorname{argmin}} \left\| \widehat{K^1} - \widehat{K} \right\|_F ,$$

where $\|\cdot\|_F$ is the Frobenius norm.

The kernel K^2 exists and is unique so that the previous definition is justified. Indeed, by Lemma 1 and Proposition 5, any nsa-admissible kernel K must be such that $\widehat{K} + \widehat{K}^*$ is non-negative. Therefore

$$\underset{K\text{arbitrage}-\text{admissible}}{\operatorname{argmin}} \left\| \widehat{K^1} - \widehat{K} \right\|_F = \underset{\widehat{K}+\widehat{K}^* \geq 0}{\operatorname{argmin}} \left\| \widehat{K^1} - \widehat{K} \right\|_F .$$

Thus, K^2 can be computed from K^1 by replacing each eigenvalue ϱ of $\widehat{K^1} + \widehat{K^1}^*$ by $\max(\varrho, 0)$. Loosely speaking, K^2 trades martingale-admissibility for arbitrage-admissibility while staying close to K^1.

The next section applies our results to market data to compute K^1 and K^2.

3 Application to financial data

This section focuses on applying the previous results to compute the kernels K^1 and K^2 on market data. Details on the methodology and data used are given in Section III.D.

The dataset used comprises of 4 years of volumes and price data for two maturities of E-Mini SP500 futures traded on the CME, so that $d = 2$. These futures are financially settled at expiry (in addition to daily settlements) according to the value of the SP500 index. The two futures selected are the lead month future, referred to as SPMINI, and the next upcoming future, referred to as SPMINI3. This data has been explored in a previous study and we refer the reader to [TMB20] for more details into the underlying data and processing methodology.[1]

We begin by reporting the relevant empirical observable of our system, namely the reduced covariance measure Ω and the price-covariance matrix Σ.

Figure III.1 reports empirical estimates of $\Omega([\tau\tau_s, (\tau+1)\tau_s[)$ for different values of τ, for a time resolution of $\tau_s = 1$ second. By a slight abuse of notation, we write $\Omega(\tau)$ for $\Omega([\tau\tau_s, (\tau+1)\tau_s[)$. With these conventions, $\Omega(0)$ represents the trade covariance, so that $\Omega_{11}(0)$ is the instantaneous variance of signed order flow on the front month future SPMINI and $\Omega_{22}(0)$ the instantaneous variance of signed order flow on SPMINI3. These quantities reflect the liquidity of the underlying assets since they increase with the daily traded volume [TMB20]. The figure shows that the front-month maturity SPMINI is approximately 10 times more liquid than the SPMINI3, which highlights that most trading occurs on the leading month contract. The order

[1]The authors thank the *Econophysics & Complex Systems* Research Chair for providing access to this data.

flow auto-covariances $\Omega_{11}(\tau)$ and $\Omega_{22}(\tau)$ are slowly decaying in τ, although $\Omega_{11}(\tau)$ exhibits faster decay than $\Omega_{22}(\tau)$. Furthermore, we observe that $\Omega_{12}(\tau) \approx \Omega_{21}(\tau)$ so that there are no lead-lag effects in the order flows. This shows that Ω cannot be easily factorized under the form $\Omega(\tau) \approx \tau^{-\beta} C$, with some $\beta < 1$ and $C \in \mathcal{M}_d(\mathbb{R})$. This hypothesis is used in certain cross-impact kernels [BMEB17].

The price-covariance matrix Σ, not shown here but reported in [TMB20], shows strong correlation ($\approx 90\%$) between the two maturities. This is natural since both futures have the same underlying.

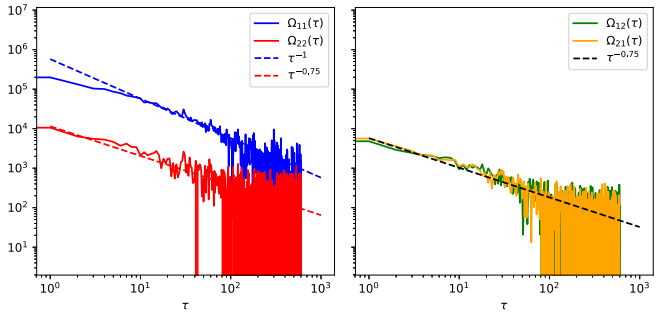


Figure III.1: **Order flow auto-covariance** Ω.
Diagonal elements of the auto-covariance measure (left) and off-diagonal elements of the auto-covariance measure (right). Dashed lines represent power-law fits of the auto-covariances. Order flow is in product units: number of contracts traded times the contract valpoint (which is 50 for these futures). Details about the estimation procedure are given in Section III.D.2.

Figure III.2 shows the estimated boundary values of the K^1 and K^2 kernels which are by definition the same for K^1 and K^2, computed using Propositions 3 and 6. The calibrated values confirm some intuitive ideas:

- price impact on the liquid future is lower than on the illiquid future: $K_{11}(0) < K_{22}(0)$ and $\Lambda_{11} < \Lambda_{22}$;

- buying a future immediately pushes the price of the other, as $K_{12}(0), K_{21}(0) > 0$, and the permanent impact contribution is positive since $\Lambda_{12}, \Lambda_{21} > 0$;

- permanent impact is lower that immediate impact, as each component of the immediate impact matrix is larger than the permanent cross-impact matrix: $K^1(0) = K^2(0) > \Lambda$.

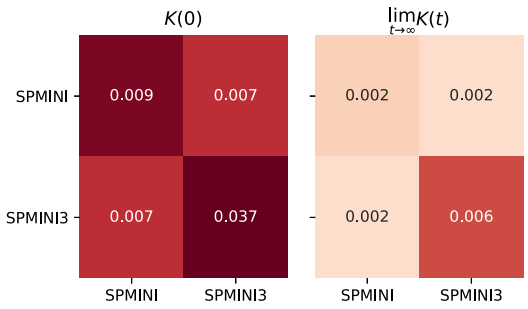


Figure III.2: **Boundary values of K^1 and K^2.**
The boundary values of the martingale-admissible kernel K^1 and the nsa-admissible kernel K^2, the immediate cross-impact matrix $K^1(0) = K^2(0)$ and the permanent cross-impact matrix $\Lambda = \lim_{t\to\infty} K^1(t) = \lim_{t\to\infty} K^2(t)$, estimated using Proposition 3 and Proposition 6. Values are reported in basis points (i.e. 10^4 of their units). Details about the estimation procedure are given in Section III.D.4.

Figures III.3 and III.4 show both cross-impact kernels. The boundary values of K^1 and K^2 are identical and given in Figure III.2. As could be checked numerically, the martingale-admissible kernel K^1 is not nsa-admissible. Thus, there is no kernel which is both martingale-admissible and nsa-admissible here. Given the very strong correlations between both assets, it is not surprising that $K^2_{21} \approx K^2_{11}$ and $K^1_{21} \approx K^1_{11}$: trading the first maturity pushes the price of each future by roughly the same amount. Finally, a somewhat surprising feature of K^2 is that it is non-monotonous. The cross-impact kernel K^2 is sensitive to numerical errors in the estimation methodology detailed in Section III.D.4, which may explain the strange value of a point of K^2_{22}. Nevertheless, it has little incidence on the kernel's fit to data (as we can see from Figure III.5).

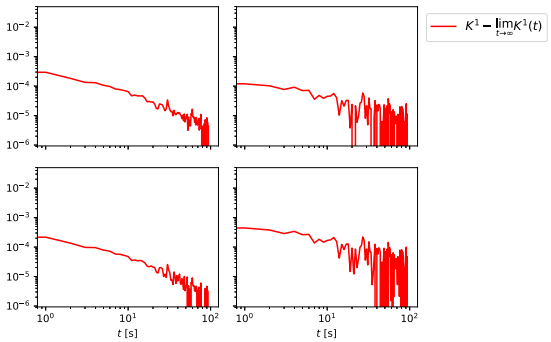


Figure III.3: **Values of the K^1 kernel.**
The values of the transient part of the martingale-admissible $K^1 - \lim_{t\to\infty} K^1(t) = K^1 - \Lambda$ (red) are reported. Each subplot shows the matrix elements of the kernels. For instance, the top left plot shows $K^1_{11} - \Lambda_{11}$ and the top right shows $K^1_{12} - \Lambda_{12}$. The permanent cross-impact matrix Λ has been removed to highlight the power-law decay of the cross-impact kernel K^1 toward its limit.

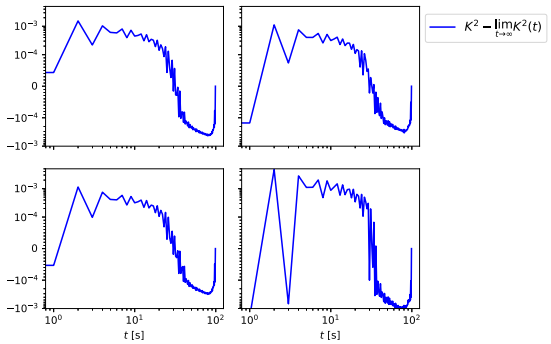


Figure III.4: **Values of the K^2 kernel.**
The values of the transient part of the nsa-admissible kernel $K^2 - \lim_{t\to\infty} K^2(t) = K^2 - \Lambda$ (blue) are reported. Each subplot shows the matrix elements of the kernels. For instance, the top left plot shows $K^2_{11} - \Lambda_{11}$ and the top right shows $K^2_{12} - \Lambda_{12}$. The permanent cross-impact matrix Λ has been removed to highlight the behaviour of the transient part of the cross-impact kernel K^2.

We illustrate the predictions of the different kernels in Figure III.5. For a given trading day taken on the 31st of January 2017, we measure the traded order flows and build the predicted price changes according to the cross-impact rule Equation (2). We complete this procedure with the martingale-admissible kernel K^1 and the nsa-admissible kernel K^2. We then compare the predicted price changes to the actual price changes. As is consistent with the literature on price impact, we see that price changes predicted from our cross-impact models are qualitatively consistent with realised price changes. However, cross-impact models are able to use trades on the liquid maturity (SPMINI) to explain price changes on the illiquid maturity (SPMINI3). This is critical since the prices of the two maturities are strongly correlated but most trades occur on the the leading month contract. Although the two kernels are quite different, their predictions are strikingly similar. This highlights that our regularisation procedure was successful in finding an nsa-admissible kernel that fits data well and prevents statistical arbitrage.

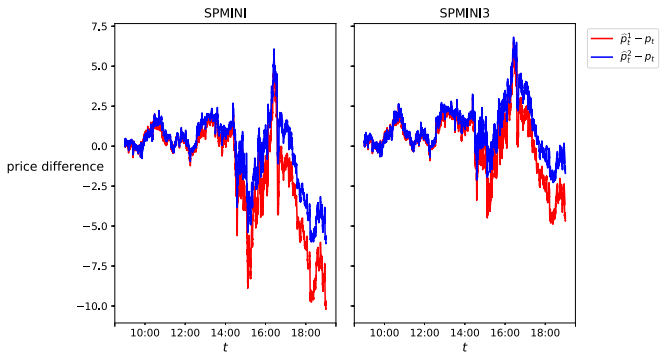


Figure III.5: **Example of predicted prices from K^1 and K^2.**
The difference between predicted prices between the realized price and the impact-induced price from Equation (1) with the martingale-admissible cross-impact kernel K^1, given by $\widehat{p}_t^1 = p_0 + \sum_{s<t} K^1(t-s)q_s$, (red) and with the nsa-admissible cross-impact kernel K^2, given by $\widehat{p}_t^2 = p_0 + \sum_{s<t} K^2(t-s)q_s$, (blue), where (q_t) are measured signed volumes of market orders. Price difference is reported in dollars per lot.

4 Conclusion

The goal of this chapter was to characterise the class of cross-impact kernels which anticipate order flows and lead to martingale prices (*martingale-admissible* kernels) and the class of cross-impact kernels which prevent statistical arbitrage (*no-statistical-arbitrage-admissible* kernels). To do so, we introduce a market where trades are self-exciting and price impact is linear in the total market order flow. In this model, we derive necessary and sufficient conditions for nsa-admissible and martingale-admissible kernels. In particular, we show that only one candidate, dubbed the K^1 kernel, could be both martingale-admissible and nsa-admissible. As there are no theoretical guarantees that the K^1 kernel prevents arbitrage, we introduce the K^2 kernel which is close to K^1 and prevents arbitrage. We find formulas for calibration of both kernels and apply them on SP500 futures.

One key result of this chapter is that, given a set of market conditions, namely the asset price covariance which encodes co-movement of assets and the auto-covariance of trades which captures the way agents trade in the market, it may not be possible to conciliate the two notions of price efficiency: martingale prices and no-arbitrage. This is specific to the multi-asset case since in the single-asset case the two notions nicely co-exist [JR18, Jai15]. Therefore, a problem we hope to address in future work is to find non-trivial market conditions where there exists a cross-impact kernel which is both nsa-admissible and martingale-admissible.

Finally, the results outlined in the chapter have applications in market-making and trading costs estimation. For market-makers, the cross-impact kernels can be calibrated in practice to better capture adverse selection or price decay after trades. For portfolio managers, these models provide better estimate of trading costs and could be used to derive more optimal

trading strategies which account for cross-impact and its decay.

III.A Moment measures for point processes

This section presents definitions of moment measures for point processes. Throughout this section, $\mathbb{X}$ denotes some Borelian subset of $\mathbb{R}^d$ and $\mathscr{X} = \mathscr{B}(\mathbb{X})$.

Definition 8 (n-th moment measure). *Given a univariate stationary point process N on $\mathbb{X}$, the first and second order moment measure of the point process N are measures on $\mathbb{X}$ and $\mathbb{X}^2$ defined for all $A, B \in \mathscr{X}$ as*

$$M_1(A) = \mathbb{E}[N(A)]$$
$$M_2(A \times B) = \mathbb{E}[N(A)N(B)]$$

whenever these expectations exist.

A key quantity is the reduced measure second order measure which we introduce below.

Definition 9 (Reduced measure (Proposition 12.6.III of [DVJ08])). *There exist a reduced measure, noted $\overset{\vee}{M}_2$, such that for any bounded measurable function f on $\mathbb{X}^2$ we have*

$$\int_{\mathbb{X}} f(x_1, x_2) M_2(dx_1, dx_2) = \int_{\mathbb{X}} \int_{\mathbb{X}} f(x, x+y) \overset{\vee}{M}_2(dy)\, dx$$

The existence result of the reduced measure satisfying the above is shown in Proposition 12.6.III of [DVJ08]. Of particular interest to us are the second-order reduced covariance measure introduced below.

Definition 10 (Reduced covariance measure). *The reduced covariance measure of a stationary point process is defined as*

$$\overset{\vee}{C}_2(du) = \overset{\vee}{M}_2(du) - m^2 l(du)\,,$$

where l is the Lebesgue measure, m is the mean intensity i.e the non-negative constant such that $M_1(du) = ml(du)$.

Extensions of the above concepts to multivariate point processes are straightforward. In particular we have the following definition.

Definition 11 (second order auto-moment measure). *Given a k-dimensional, stationary point process N on $\mathbb{X}$, the second order auto-moment measure of the point process N is a measure on $\mathbb{X}^2$ defined for all $A, B \in \mathscr{X}$ and $1 \leq i, j \leq k$ as*

$$M_{ij}(A \times B) = \mathbb{E}[N_i(A) N_j(B)]\,,$$

whenever this expectation exists.

Definition 12 (reduced covariance measure). *The reduced covariance measure of a k-dimensional, stationary point process is defined, for all $1 \leq i, j \leq k$ as*

$$\overset{\vee}{C}_{ij}(du) = \overset{\vee}{M}_{ij}(du) - m_i m_j l(du)\,,$$

where l is the Lebesgue measure, m_i is the mean intensity i.e the non-negative constant such that $M_i(du) = m_i l(du)$.

III.B Technical results

This section presents some key technical results necessary for the proofs. We begin by introducing an important functional space for our results, the Hardy space $\mathbb{H}^2$.

Definition 13 (Hardy space $\mathbb{H}^2$). *The Hardy space $\mathbb{H}^2$ is the space of functions $F\colon \mathbb{C} \to \mathbb{C}$ such that the following conditions are satisfied:*

1. *the function F is holomorphic in the upper half of the complex plane,*
2. *there exists a constant $C > 0$ such that, for all $\xi > 0$*
$$\int_{-\infty}^{\infty} |F(\omega + i\xi)|^2 d\omega < C\,,$$
3. *for almost all $\omega \in \mathbb{R}$, we have*
$$\lim_{\xi \to 0} F(\omega + i\xi) = F(\omega)\,.$$

Functions in the Hardy space $\mathbb{H}^2$ are closely related to the Fourier transform of causal functions as shown by the so-called Titchmarsh theorem below.

Theorem 2 (Titchmarsh theorem [Tit48]). *Given a real, complex-valued square-integrable function F, the following conditions are equivalent:*

1. *the inverse Fourier transform of F is a causal function,*
2. *the function F belongs to the Hardy space $\mathbb{H}^2$.*

A key theorem used in the chapter to derive the form of martingale-admissible cross-impact kernels is the matrix spectral factorization theorem.

Theorem 3 (Matrix spectral factorization theorem [WM57, WA59]). *Let $F\colon \mathbb{C} \to \mathscr{M}_d(\mathbb{R})$ be a matrix function such that F is positive definite almost everywhere on the unit circle $\mathbb{T} := \{z \in \mathbb{C}\colon |z| = 1\}$ with integrable entries on the unit circle such that the Paley-Wiener condition*

$$\log\det F \in L^1(\mathbb{T}) \tag{8}$$

is satisfied. Then F admits a factorization on $\mathbb{T}$

$$F(z) = F_+(z) F_+(z)^*\,, \tag{9}$$

where F_+ is an analytic function with entries in the Hardy space $\mathbb{H}^2$. Furthermore, the spectral factor F_+ is unique up to right multiplication by a unitary matrix (i.e. a matrix U such that $UU^* = \mathbb{I}_d$).

As spectral factors are referenced throughout the chapter, we introduce for convenience a definition below.

Definition 14 (Spectral factor). *Let $F\colon \mathbb{C} \to \mathscr{M}_d(\mathbb{R})$ be a matrix function which satisfies the hypotheses of Theorem 3. Then, a spectral factor F_+ of F is an analytic function F_+ with entries in the Hardy space $\mathbb{H}^2$ which satisfies Equation* (9).

III.C Proofs

III.C.1 Proof of Proposition 1

The proof is adapted from [Jai15]. Throughout this section we assume that there exists a martingale-admissible kernel K such that the price satisfies

$$P_t = P_0 + \int_0^t K(t-s)\mathrm{d}(N_s^a - N_s^b) = P_0 + \Lambda \lim_{s\to\infty} \mathbb{E}[V_s^a - V_s^b \mid \mathscr{F}_t]\,.$$

To compute the right hand side term, we use a classical result on Hawkes processes [BMM15, Jai15, JR18]: for all $t \geq 0$, we have

$$\begin{pmatrix}\lambda_t^a\\ \lambda_t^b\end{pmatrix} = \begin{pmatrix}\mu\\ \mu\end{pmatrix} + \int_0^t \Psi(t-s)\begin{pmatrix}\mu\\ \mu\end{pmatrix}\mathrm{d}s + \int_0^t \Psi(t-s)\mathrm{d}M_s\,, \tag{10}$$

where $\Psi := \sum_{n\geq 1}\Phi^{*n}$, Φ^{*n} being the n-th convolution product of the matrix function Φ and M is a martingale. The next lemma shows that the compuation of the conditional expectation of the Hawkes process reduces to the compuation of the conditional expectation of the intensity.

Lemma 2. *For all $t, s \geq 0$ such that $s \geq t$, we have*

$$\mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t] = N_t^a - N_t^b + \int_t^s \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]du\,.$$

Proof. Using the martingale decomposition of the Hawkes process, we have $N_t = M_t + \int_0^t \lambda_s \mathrm{d}s$ where M is a martingale. Therefore,

$$N_s^a - N_s^b = M_s^a - M_s^b + \int_0^t (\lambda_u^a - \lambda_u^b)\mathrm{d}u + \int_t^s (\lambda_u^a - \lambda_u^b)\mathrm{d}u\,.$$

Using the martingale property, we obtain

$$\mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t] = N_t^a - N_t^b + \int_t^s \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]\mathrm{d}u\,.$$

□

The next lemma computes the conditional expectation of the intensity for our Hawkes processes. It generalises Proposition 3.2 of [Jai15].

Lemma 3. *For all $t, s \geq 0$ such that $s \geq t$, we have*

$$\begin{aligned}\int_t^s \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]du = &\int_0^t \int_{t-r}^{s-r} \Xi(u)du(dN_r^a - dN_r^b)\\ &- \int_0^t \int_{t-r}^{s-r}\int_0^{t-r} \Xi(u-x)\Phi^{a/a}(x)dxdudN_r^a - \int_0^t \int_{t-r}^{s-r}\int_0^{t-r} \Xi(u-x)\Phi^{b/a}(x)dxdudN_r^a\\ &+ \int_0^t \int_{t-r}^{s-r}\int_0^{t-r} \Xi(u-x)\Phi^{a/b}(x)dxdudN_r^b + \int_0^t \int_{t-r}^{s-r}\int_0^{t-r} \Xi(u-x)\Phi^{b/b}(x)dxdudN_r^b\,,\end{aligned}$$

where $\Xi := \Psi^{a/a} - \Psi^{b/a} = \Psi^{b/b} - \Psi^{a/b}$.

Proof. We use Equation (10) to derive

$$\int_t^s \mathbb{E}[\lambda_u^a \mid \mathscr{F}_t]\mathrm{d}u = \mathbb{E}\left[\int_t^s \int_0^u \Psi^{a/a}(u-x)\mathrm{d}M_x^a \mathrm{d}u \mid \mathscr{F}_t\right] + \mathbb{E}\left[\int_t^s \int_0^u \Psi^{a/b}(u-x)\mathrm{d}M_x^b \mathrm{d}u \mid \mathscr{F}_t\right].$$

Using the fact that M is a martingale, we have

$$\begin{aligned}
\mathbb{E}\left[\int_t^s \int_0^u \Psi^{a/a}(u-x)\mathrm{d}M_x^a \mathrm{d}u \mid \mathscr{F}_t\right] &= \int_t^s \int_0^t \Psi^{a/a}(u-x)\mathrm{d}M_x^a \mathrm{d}u \\
&= \int_t^s \int_0^t \Psi^{a/a}(u-x)\left(\mathrm{d}N_x^a - (\mu + \int_0^t \Phi^{a/a}(x-r)\mathrm{d}N_r^a + \int_0^t \Phi^{a/b}(x-r)\mathrm{d}N_r^b)\mathrm{d}x\right)\mathrm{d}u \\
&= \int_t^s \int_0^t \Psi^{a/a}(u-x)\mathrm{d}N_x^a \mathrm{d}u - \int_t^s \int_0^t \Psi^{a/a}(u-x)\mu \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t \Psi^{a/a}(u-x)\int_0^t \Phi^{a/a}(x-r)\mathrm{d}N_r^a \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t \Psi^{a/a}(u-x)\int_0^t \Phi^{a/b}(x-r)\mathrm{d}N_r^b \mathrm{d}x\mathrm{d}u.
\end{aligned}$$

Consequently,

$$\begin{aligned}
\int_t^s \mathbb{E}[\lambda_u^a \mid \mathscr{F}_t]\mathrm{d}u &= \int_t^s \int_0^t \Psi^{a/a}(u-x)\mathrm{d}N_x^a \mathrm{d}u + \int_t^s \int_0^t \Psi^{a/b}(u-x)\mathrm{d}N_x^b \mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/a} + \Psi^{a/b})(u-x)\mu \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t \Psi^{a/a}(u-x)\int_0^t \Phi^{a/a}(x-r)\mathrm{d}N_r^a \mathrm{d}x\mathrm{d}u - \int_t^s \int_0^t \Psi^{a/b}(u-x)\int_0^t \Phi^{b/a}(x-r)\mathrm{d}N_r^a \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t \Psi^{a/a}(u-x)\int_0^t \Phi^{a/b}(x-r)\mathrm{d}N_r^b \mathrm{d}x\mathrm{d}u - \int_t^s \int_0^t \Psi^{a/b}(u-x)\int_0^t \Phi^{b/b}(x-r)\mathrm{d}N_r^b \mathrm{d}x\mathrm{d}u.
\end{aligned}$$

Regrouping terms from λ^a and λ^b we obtain

$$\begin{aligned}
\int_t^s \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]\mathrm{d}u &= \int_t^s \int_0^t (\Psi^{a/a} - \Psi^{b/a})(u-x)\mathrm{d}N_x^a \mathrm{d}u + \int_t^s \int_0^t (\Psi^{a/b} - \Psi^{b/b})(u-x)\mathrm{d}N_x^b \mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/a} + \Psi^{a/b} - \Psi^{b/a} - \Psi^{b/b})(u-x)\mu \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/a} - \Psi^{b/a})(u-x)\int_0^t \Phi^{a/a}(x-r)\mathrm{d}N_r^a \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/b} - \Psi^{b/b})(u-x)\int_0^t \Phi^{b/a}(x-r)\mathrm{d}N_r^a \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/a} - \Psi^{b/a})(u-x)\int_0^t \Phi^{a/b}(x-r)\mathrm{d}N_r^b \mathrm{d}x\mathrm{d}u \\
&\quad - \int_t^s \int_0^t (\Psi^{a/b} - \Psi^{b/b})(u-x)\int_0^t \Phi^{b/b}(x-r)\mathrm{d}N_r^b \mathrm{d}x\mathrm{d}u.
\end{aligned}$$

By Assumption 2, the above must converge to a finite limit as s tends to infinity. Therefore, as $\mu \neq 0$ and all elements of the matrix function Φ are non-negative we must have for all $t \geq 0$

that $\Psi^{a/a}(t)+\Psi^{a/b}(t)=\Psi^{b/a}(t)+\Psi^{b/b}(t)$. Therefore, writing $\Xi := \Psi^{a/a}-\Psi^{b/a}=\Psi^{b/b}-\Psi^{a/b}$, we have

$$\begin{aligned}\int_t^s \mathbb{E}[\lambda_u^a-\lambda_u^b\mid\mathscr{F}_t]\mathrm{d}u = &\int_t^s\int_0^t \Xi(u-x)(\mathrm{d}N_x^a-\mathrm{d}N_x^b)\mathrm{d}u\\ &-\int_t^s\int_0^t \Xi(u-x)\int_0^t \Phi^{a/a}(x-r)\mathrm{d}N_r^a\mathrm{d}x\mathrm{d}u-\int_t^s\int_0^t \Xi(u-x)\int_0^t \Phi^{b/a}(x-r)\mathrm{d}N_r^a\mathrm{d}x\mathrm{d}u\\ &+\int_t^s\int_0^t \Xi(u-x)\int_0^t \Phi^{a/b}(x-r)\mathrm{d}N_r^b\mathrm{d}x\mathrm{d}u+\int_t^s\int_0^t \Xi(u-x)\int_0^t \Phi^{b/b}(x-r)\mathrm{d}N_r^b\mathrm{d}x\mathrm{d}u.\end{aligned}$$

Using the change of variables $u'=u-r$ and $x'=x-r$ we obtain

$$\begin{aligned}\int_t^s \mathbb{E}[\lambda_u^a-\lambda_u^b\mid\mathscr{F}_t]\mathrm{d}u = &\int_0^t\int_{t-x}^{s-x} \Xi(u')\mathrm{d}u'(\mathrm{d}N_x^a-\mathrm{d}N_x^b)\\ &-\int_0^t\int_{t-r}^{s-r}\int_0^{t-r} \Xi(u'-x')\Phi^{a/a}(x')\mathrm{d}x'\mathrm{d}u'\mathrm{d}N_r^a\\ &-\int_0^t\int_{t-r}^{s-r}\int_0^{t-r} \Xi(u'-x')\Phi^{b/a}(x')\mathrm{d}x'\mathrm{d}u'\mathrm{d}N_r^a\\ &+\int_0^t\int_{t-r}^{s-r}\int_0^{t-r} \Xi(u'-x')\Phi^{a/b}(x')\mathrm{d}x'\mathrm{d}u'\mathrm{d}N_r^b\\ &+\int_0^t\int_{t-r}^{s-r}\int_0^{t-r} \Xi(u'-x')\Phi^{b/b}(x')\mathrm{d}x'\mathrm{d}u'\mathrm{d}N_r^b.\end{aligned}$$

□

The next lemma uses the previous results to compute the conditional expectation of the counting process.

Lemma 4. *We have for all $t,s\geq 0$ such that $s\geq t$*

$$\begin{aligned}\mathbb{E}[N_s^a-N_s^b\mid\mathscr{F}_t] = &N_t^a-N_t^b+\int_0^t\int_{t-r}^{s-r}\Xi(u)du(dN_r^a-dN_r^b)\\ &-\int_0^t\int_{t-r}^{s-r}\int_0^{t-r}\Xi(u-x)\Phi^{a/a}(x)dxdudN_r^a-\int_0^t\int_{t-r}^{s-r}\int_0^{t-r}\Xi(u-x)\Phi^{b/a}(x)dxdudN_r^a\\ &+\int_0^t\int_{t-r}^{s-r}\int_0^{t-r}\Xi(u-x)\Phi^{a/b}(x)dxdudN_r^b+\int_0^t\int_{t-r}^{s-r}\int_0^{t-r}\Xi(u-x)\Phi^{b/b}(x)dxdudN_r^b,\end{aligned}$$

Proof. Applying Lemma 2 along with Lemma 3, the result is straightforward. □

The next lemma derives the limit of the conditional expectation of the process $N_s^a-N_s^b$.

Lemma 5. *For all $t\geq 0$, we have*

$$\begin{aligned}\lim_{s\to\infty}\mathbb{E}[N_s^a-N_s^b\mid\mathscr{F}_t] = N_t^a-N_t^b &+\int_0^t\left(\int_{t-x}^{\infty}\Xi(u)du\right)(dN_x^a-dN_x^b)\\ &-\int_0^t\left(\int_{t-x}^{\infty}\int_0^{t-x}\Xi(u-v)(\Phi^{a/a}+\Phi^{b/a})(v)dvdu\right)dN_x^a\\ &+\int_0^t\left(\int_{t-x}^{\infty}\int_0^{t-x}\Xi(u-v)(\Phi^{a/b}+\Phi^{b/b})(v)dvdu\right)dN_x^b,\end{aligned}$$

where $\Xi := \Psi^{a/a} - \Psi^{b/a} = \Psi^{b/b} - \Psi^{a/b}$.

Proof. Taking the limit as s tends to infinity of Lemma 3 we obtain the almost sure convergence of $\int_t^s \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]\mathrm{d}u$ towards

$$\begin{aligned}\int_t^\infty \mathbb{E}[\lambda_u^a - \lambda_u^b \mid \mathscr{F}_t]\mathrm{d}u = &\int_0^t \left(\int_{t-x}^\infty \Xi(u)\mathrm{d}u\right)(\mathrm{d}N_x^a - \mathrm{d}N_x^b)\\ &- \int_0^t \left(\int_{t-x}^\infty \int_0^{t-x} \Xi(u-v)(\Phi^{a/a} + \Phi^{b/a})(v)\mathrm{d}v\mathrm{d}u\right)\mathrm{d}N_x^a\\ &+ \int_0^t \left(\int_{t-x}^\infty \int_0^{t-x} \Xi(u-v)(\Phi^{a/b} + \Phi^{b/b})(v)\mathrm{d}v\mathrm{d}u\right)\mathrm{d}N_x^b,\end{aligned}$$

where we have used that Ξ is integrable. Finally, using Lemma 2 with the above yields the result. □

In order to show that the price is a martingale, we establish some elementary results about the convergence of the conditional expectations.

Lemma 6. *For all $t, s \geq 0$ such that $s \geq t$, we write $Z_s^t := \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t]$ and $Z^t := \lim_{s\to\infty} \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t]$. Then, we have that*

1. *the random variable Z^t is integrable,*
2. *the random variables (Z_s^t) converge almost surely to Z^t,*
3. *there exists an integrable random variable Y^t such that for all $s \geq t$, $|Z_s^t| \leq Y^t$.*

Proof. By Lemma 4, we have for all $t, s \geq 0$ such that $s \geq t$

$$\begin{aligned}Z_s^t = \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t] = &N_t^a - N_t^b + \int_0^t \int_{t-r}^{s-r} \Xi(u)\mathrm{d}u(\mathrm{d}N_r^a - \mathrm{d}N_r^b)\\ &- \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} \Xi(u-x)\Phi^{a/a}(x)\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a - \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} \Xi(u-x)\Phi^{b/a}(x)\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a\\ &+ \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} \Xi(u-x)\Phi^{a/b}(x)\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b + \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} \Xi(u-x)\Phi^{b/b}(x)\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b.\end{aligned}$$

Therefore, writing $|A|$ for the matrix of absolute values of all entries of A, we have

$$\begin{aligned}|Z_s^t| \leq &N_t^a + N_t^b + \int_0^t \int_{t-r}^{s-r} |\Xi(u)|\mathrm{d}u(\mathrm{d}N_r^a + \mathrm{d}N_r^b)\\ &+ \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} |\Xi(u-x)\Phi^{a/a}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a + \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} |\Xi(u-x)\Phi^{b/a}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a\\ &+ \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} |\Xi(u-x)\Phi^{a/b}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b + \int_0^t \int_{t-r}^{s-r} \int_0^{t-r} |\Xi(u-x)\Phi^{b/b}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b\\ \leq &N_t^a + N_t^b + \int_0^t \int_{t-r}^{\infty} |\Xi(u)|\mathrm{d}u(\mathrm{d}N_r^a + \mathrm{d}N_r^b)\\ &+ \int_0^t \int_{t-r}^{\infty} \int_0^{t-r} |\Xi(u-x)\Phi^{a/a}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a + \int_0^t \int_{t-r}^{\infty} \int_0^{t-r} |\Xi(u-x)\Phi^{b/a}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^a\\ &+ \int_0^t \int_{t-r}^{\infty} \int_0^{t-r} |\Xi(u-x)\Phi^{a/b}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b + \int_0^t \int_{t-r}^{\infty} \int_0^{t-r} |\Xi(u-x)\Phi^{b/b}(x)|\mathrm{d}x\mathrm{d}u\mathrm{d}N_r^b.\end{aligned}$$

Writing Y^t for the last random variable, Y^t is independent from s, positive and we can easily check that it is of finite expectation since Ξ is integrable. This proves (iii). Furthermore, we also have $|Z^t| \leq Y^t$ so that Z^t is integrable and (i) is verified. Finally, Lemma 5 proved point (ii). □

Lemma 7. *The price is a martingale.*

Proof. Since, for all $t, \tau \geq 0$ such that $\tau \leq t$ we have

$$\begin{aligned}
\mathbb{E}[P_t \mid \mathscr{F}_\tau] &= P_0 + \Lambda \mathrm{diag}(v) \mathbb{E}[\lim_{s\to\infty} \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t] \mid \mathscr{F}_\tau] \\
&= P_0 + \Lambda \mathrm{diag}(v) \lim_{s\to\infty} \mathbb{E}[\mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t] \mid \mathscr{F}_\tau] \\
&= P_0 + \Lambda \mathrm{diag}(v) \lim_{s\to\infty} \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_\tau] \\
&= P_\tau \,,
\end{aligned}$$

where the second equality is an application of the dominated convergence theorem with the random variables $Z_s^t := \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t]$ and $Z^t := \lim_{s\to\infty} \mathbb{E}[N_s^a - N_s^b \mid \mathscr{F}_t]$, the conditions being met by Lemma 6. □

We can simplify the expression of the price process using martingale-admissibility. This is the topic of the next lemma.

Lemma 8. *The price process is of the form*

$$P_t = P_0 + \Lambda \mathrm{diag}(v) \int_0^t \varsigma(t-s)(dN_s^a - dN_s^b) \,,$$

where $\varsigma(t) = \mathbb{I}_d + \int_{t-x}^\infty \Xi(u)\,du + \int_{t-v}^\infty \int_0^{t-v} \Xi(u-v)(\Phi^{a/a} + \Phi^{b/a})(v)\,dv\,du$. *Furthermore, K is almost everywhere differentiable.*

Proof. By assumption, we know there exists some cross-impact kernel K such that for all $t \geq 0$

$$P_t = P_0 + \int_0^t K(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b) \,.$$

Therefore Lemma 5 implies that for all $t \geq 0$

$$\int_{t-v}^\infty \int_0^{t-v} \Xi(u-v)(\Phi^{a/a} + \Phi^{b/a})(v)\mathrm{d}v\mathrm{d}u = \int_{t-v}^\infty \int_0^{t-v} \Xi(u-v)(\Phi^{a/b} + \Phi^{b/b})(v)\mathrm{d}v\mathrm{d}u \,.$$

And Lemma 5 yields

$$P_t = P_0 + \Lambda \mathrm{diag}(v) \int_0^t \varsigma(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b) \,,$$

with $\varsigma(t) = \mathbb{I}_d + \int_{t-x}^\infty \Xi(u)\mathrm{d}u + \int_{t-v}^\infty \int_0^{t-v} \Xi(u-v)(\Phi^{a/a} + \Phi^{b/a})(v)\mathrm{d}v\mathrm{d}u$. In particular, we must have $\Lambda \mathrm{diag}(v)\varsigma = K$ almost everywhere and K is almost everywhere differentiable. □

To obtain a simpler expression for K and the other results of the proposition, we adapt the proof of Proposition 3.3 of [Jai15] to our structure of Hawkes processes. At time $t \geq 0$, either:

- there are no market orders and the price is differentiable with $P_t^{'} = \int_0^t K^{'}(t-s)\mathrm{d}(N^a - N^b)_s$;
- there is a buy market order on some asset, say Asset i, which happens with intensity $\lambda_t^{b,i}$ and yields a price jump of $K(0)\nu_i e_i$;
- there is a sell market order on some asset, say Asset i, which happens with intensity $\lambda_t^{b,i}$ and yields a price jump of $-K(0)\nu_i e_i$.

Therefore, we have for all $t \geq 0$

$$\begin{aligned}
\lim_{h\to 0} \frac{\mathbb{E}[P_{t+h} \mid \mathscr{F}_t] - P_t}{h} &= \int_0^t K^{'}(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b) + \sum_{i=1}^d K(0)\nu_i e_i(\lambda_t^{a,i} - \lambda_t^{b,i}) \\
&= \int_0^t K^{'}(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b) + K(0)\mathrm{diag}(\nu)(\lambda_t^a - \lambda_t^b) \\
&= \int_0^t K^{'}(t-s)(\mathrm{d}N_s^a - \mathrm{d}N_s^b) - K(0)\mathrm{diag}(\nu)\int_0^t (\Phi^{b/b} - \Phi^{a/b})(t-s)\mathrm{d}N_s^b \\
&\quad + K(0)\mathrm{diag}(\nu)\int_0^t (\Phi^{a/a} - \Phi^{b/a})(t-s)\mathrm{d}N_s^a \\
&= 0,
\end{aligned}$$

where the last equality holds since the price is a martingale by Lemma 7. Thus, we must have for all $t \geq 0$

$$\begin{aligned}
K^{'}(t) &= -K(0)\mathrm{diag}(\nu)(\Phi^{b/b}(t) - \Phi^{a/b}(t)) \\
K^{'}(t) &= -K(0)\mathrm{diag}(\nu)(\Phi^{a/a}(t) - \Phi^{b/a}(t)).
\end{aligned}$$

To conclude, we need to derive the expression of $K(0)$. Since $\Phi^{b/b} - \Phi^{a/a}$ is integrable, K converges to a finite limit and by definition of the permanent cross-impact matrix: $\lim_{t\to\infty} K(t) = \Lambda$. Thus

$$\lim_{t\to\infty} K(t) = K(0)\mathrm{diag}(\nu)\left(\mathbb{I}_d - \int_0^\infty (\Phi^{b/b}(s) - \Phi^{a/b}(s))\mathrm{d}s\right) = \Lambda.$$

Since by Assumption 1 we have $\rho(\int_0^\infty \Phi(s)\mathrm{d}s) < 1$, we get

$$K(0) = \Lambda\left(\mathbb{I}_d - \int_0^\infty (\Phi^{b/b}(s) - \Phi^{a/b}(s))\mathrm{d}s\right)^{-1}\mathrm{diag}(\nu)^{-1}.$$

Since the matrix Λ is non-singular by Assumption 3 so is the matrix $K(0)$ and we must also have

$$\Phi^{b/b} - \Phi^{a/b} = \Phi^{a/a} - \Phi^{b/a}.$$

This completes the proof of Proposition 1.

III.C.2 Proof of Proposition 2

Because of the autocovariance structure of Hawkes processes, this section makes extensive use of Fourier transforms of measures. We recall from the notation section that the Fourier transform of a measure κ at $\omega \in \mathbb{C}$ is defined, when the integral converges, as

$$\widehat{\kappa}(\omega) = \int_{-\infty}^{\infty} e^{-i\omega t} \kappa(\mathrm{d}t).$$

Writing $\theta := (\mathbb{I}_d - \Phi^{a/a})\mu + \Phi^{a/b}\mu$ for the average of the intensity of the stationary version of the Hawkes process we know from [BDM12] that

$$\widehat{\Omega}^{\tilde{N}}(\omega) = (\mathbb{I}_d - \widehat{\Phi}(\omega))^{-1} \mathrm{diag}(\theta) ((\mathbb{I}_d - \widehat{\Phi}(-\omega))^{-1})^{\top}.$$

In particular, the Fourier transform of the reduced covariance measure Ω has integrable entries.

We begin the proof of Proposition 2 with a useful lemma.

Lemma 9. *For all $\omega \in \mathbb{R}$, we have*

$$(\widehat{K'}(\omega) + K(0))\widehat{\Omega}(\omega)(\widehat{K'}(\omega) + K(0))^* = \Lambda \Omega_\infty \Lambda^{\top}.$$

Proof. Writing $\theta := (\mathbb{I}_d - \Phi^{a/a})\mu + \Phi^{a/b}\mu$ for the average of the intensity of the stationary version of the Hawkes process we know from [BDM12] that

$$\widehat{\Omega}^{\tilde{N}}(\omega) = (\mathbb{I} - \widehat{\Phi}(\omega))^{-1} \mathrm{diag}(\theta) ((\mathbb{I} - \widehat{\Phi}(-\omega))^{-1})^{\top}.$$

Therefore we have

$$\begin{aligned}
\widehat{\Omega}^{\tilde{N}}(\omega) &= (\mathbb{I}_d - \widehat{\Phi}(\omega))^{-1} \mathrm{diag}(\theta) ((\mathbb{I}_d - \widehat{\Phi}(\omega))^{-1})^* \\
\widehat{\Omega}(\omega) &= (\mathbb{I}_d, -\mathbb{I}_d) \widehat{\Omega}^{\tilde{N}}(\omega) (\mathbb{I}_d, -\mathbb{I}_d)^{\top}.
\end{aligned}$$

Using pseudo-inverses we obtain

$$\begin{aligned}
\widehat{\Omega}(\omega)^{-1} &= \frac{1}{4} (\mathbb{I}_d, -\mathbb{I}_d) (\widehat{\Omega}^{\tilde{N}}(\omega))^{-1} (\mathbb{I}_d, -\mathbb{I}_d)^{\top} \\
&= (\mathbb{I}_d - \widehat{\varphi}(\omega))^* \mathrm{diag}(\theta)^{-1} (\mathbb{I}_d - \widehat{\varphi}(\omega)) \\
&= (\mathbb{I}_d - \widehat{\varphi}(\omega))^* ((\mathbb{I}_d - \widehat{\varphi}(0))^{-1})^* \Omega_\infty^{-1} (\mathbb{I}_d - \widehat{\varphi}(\omega))^{-1} (\mathbb{I}_d - \widehat{\varphi}(0)),
\end{aligned}$$

where we have introduced $\Omega_\infty := \widehat{\Omega}(0)$. Therefore, since Λ is real, by Proposition 1 we get

$$(\widehat{K'}(\omega) + K(0))\widehat{\Omega}(\omega)(\widehat{K'}(\omega) + K(0))^* = \Lambda \Omega_\infty \Lambda^{\top}.$$

□

We now prove Proposition 2. First, note that $\widehat{\Omega}$ admits a spectral factor. since it is positive-definite almost everywhere on the unit circle. Furthermore, by Theorem 2 and Assumption 4, φ is square-integrable and causal. Thus, Lemma 9 shows that $(\mathbb{I}_d - \widehat{\varphi})^{-1}\mathrm{diag}(\sqrt{\theta})$ is a spectral factor of $\widehat{\Omega}$. Since there exists a spectral factor, the Paley-Wiener condition Equation (8) is necessarily satisfied.

As $\widehat{\Omega}$ satisfies the conditions of Theorem 3, let $\mathscr{L}$ be any spectral factor of $\widehat{\Omega}$. Then, since the products of two functions of causal inverse Fourier transform also has a causal inverse Fourier transform, the inverse Fourier transform of $\widehat{K'}\mathscr{L}$ is causal. Furthermore, since $\widehat{K'}$ and $\mathscr{L}$ are both integrable functions, their product is square-integrable. Thus, the product satisfies the first condition of Theorem 2 and therefore the product belongs to the Hardy space $\mathbb{H}^2$. Therefore, by Lemma 9, $\widehat{K'}\mathscr{L}$ is a spectral factor of the positive-definite matrix $\Lambda\Omega_\infty\Lambda^\top$. It follows by the uniqueness of spectral factors up to a unitary matrix that there exists some unitary matrix $\mathscr{O}$ such that for almost all $\omega \in \mathbb{R}$

$$\widehat{K'}(\omega) = \mathscr{G}\mathscr{O}\mathscr{L}(\omega)^{-1} - K(0)\,,$$

where $\mathscr{G}\mathscr{G}^\top = \Lambda\Omega_\infty\Lambda^\top$. This concludes the proof.

III.C.3 Proof of Proposition 3

The immediate cross-impact matrix $K(0)$ describes how trades push prices on very short time scales. As such, we intuitively expect that it must be constrained to prevent pair-trading arbitrage. In fact, we show in this section that it can be completely characterised. The first lemma shows that the immediate cross-impact matrix must be symmetric non-negative.

Lemma 10. *The immediate cross-impact matrix $K(0)$ is symmetric and non-negative.*

Proof. This proof uses no-arbitrage arguments. It is inspired from [SL19] but adapted here since assumptions are slightly different. We consider a deterministic trading strategy ending at time T given by the function $f\colon [0,T] \to \mathbb{R}^d$ which determines the buy and sell market orders according to Definition 2. We re-write the cross-impact contributions as $K(t) = M + H(t)$, where M is the immediate impact matrix and $H(0) = 0$. From Proposition 1 we know that H is continuous at zero. With these conventions, the average cost $C(f)$ of the trading strategy f given in Definition 2 is written as

$$C(f) = \int_0^T f(t)^\top \int_0^t M f(s)\mathrm{d}s\mathrm{d}t + \int_0^T f(t)^\top \int_0^t H(t-s) f(s)\mathrm{d}s\mathrm{d}t =: C_i(f) + C_s(f),$$

where we have split costs into two parts. The first, $C_i(f)$, represents immediate impact costs and the second, $C_s(f)$, represents the rest. Then

$$C_i(f) = \frac{1}{2}\sum_{i=1}^d M_{ii}\left\|f_i\right\|_1^2 + \sum_{i\neq j}^d M_{ij}\int_0^T\int_0^t f_i(t) f_j(s)\mathrm{d}s\mathrm{d}t.$$

Similarly, for the other impact costs, we have

$$C_s(f) = \sum_{i=1}^{d} \int_0^T \int_0^t f_i(t) H_{ii}(t-s) f_i(s) \mathrm{d}s\mathrm{d}t + \sum_{i \neq j}^{d} \int_0^T \int_0^t f_i(t) H_{ij}(t-s) f_j(s) \mathrm{d}s\mathrm{d}t.$$

We choose two distinct assets, Asset a and Asset b and consider, as in [SL19], a round-trip pair-trading strategy of the following form, where $v_p, v_q \in \mathbb{R}$:

$$f_p(t) := \begin{cases} v_p & \text{for } 0 \leq t \leq T/3 \\ 0 & \text{for } T/3 \leq t \leq 2T/3 \\ -v_p & \text{for } 2T/3 \leq t \leq T \end{cases}, \qquad f_q(t) := \begin{cases} v_q & \text{for } 0 \leq t \leq T/3 \\ -v_q & \text{for } T/3 \leq t \leq 2T/3 \\ 0 & \text{for } 2T/3 \leq t \leq T \end{cases}.$$

This strategy only trades Asset p and Asset q, so that for all other Asset i, $f_i = 0$. This is a round-trip strategy since $\int_0^T f = 0$. Then, the immediate impact costs contribution is

$$C_i(f) = \frac{T^2}{18}(M_{pq} - M_{qp}) v_p v_q.$$

For the other impact costs, we have

$$\begin{aligned} C_s(f) = &\int_0^T \int_0^t f_a(t) H_{pp}(t-s) f_p(s) \mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_q(t) H_{qq}(t-s) f_q(s) \mathrm{d}s\mathrm{d}t \\ &+ \int_0^T \int_0^t f_p(t) H_{pq}(t-s) f_q(s) \mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_q(t) H_{qp}(t-s) f_q(s) \mathrm{d}s\mathrm{d}t. \end{aligned}$$

Therefore, since $H(0) = 0$ and H is continous at $t = 0$, for small enough execution times T we have that for all $t \in [0, T]$, $| H_{ij}(t) | \leq \epsilon$ for all $(i, j) \in \{p, q\}^2$. Thus, combining both impact terms, we get

$$\frac{C(f)}{T^2} \leq \frac{v_a v_b}{18}(M_{qp} - M_{pq}) + \epsilon.$$

So, unless $M_{pq} = M_{qp}$, the volumes v_p and v_q can be chosen so that trading costs of this round trip strategy are negative. Thus, if $M_{pq} \neq M_{qp}$, arbitrages are possible. Therefore the immediate cross-impact matrix $M = K(0)$ is necessarily symmetric.

Since $K(0)$ is symmetric and K is continuous by Proposition 1, Lemma 2.8 of [AKS16] implies that $K(0)$ is non-negative definite and hence non-negative. This completes the proof. □

The previous results show that the immediate cross-impact matrix is symmetric non-negative. By further using the price dynamics, we are able to relate it to the instantaneous covariance matrix of prices and order flows.

We have

$$\mathrm{d}P_t = K(0)\mathrm{diag}(v)(\mathrm{d}N_t^b - \mathrm{d}N_t^a).$$

Taking the predictable quadratic variation of the processes, we obtain

$$\mathrm{d}\langle P, P \rangle_t = K(0)\mathrm{diag}(v^2)\mathrm{diag}(\lambda_t^a + \lambda_t^b)K(0)^\top \mathrm{d}t$$

Taking expectations on both sides and writing the return covariance matrix $\Sigma_t \mathrm{d}t := \mathbb{E}[\mathrm{d}\langle P, P\rangle_t]$ and the average intensity $\theta_t := \mathbb{E}[\lambda_t^b] = \mathbb{E}[\lambda_t^a]$ we get

$$\frac{1}{2}\Sigma_t = K(0)\mathrm{diag}(v^2\theta_t)K(0)^\top.$$

Since this holds for all t, and $\theta_t \underset{t\to\infty}{\to} \theta = (\mathbb{I}_d - \Phi^{a/a})\mu + \Phi^{a/b}\mu$, it must hold as t tends to infinity and we get

$$\frac{1}{2}\Sigma = K(0)\mathrm{diag}(\theta v^2)K(0)^\top,$$

where $\Sigma = \lim_{t\to\infty} \mathbb{E}[\mathrm{d}\langle P_t, P_t\rangle]$, which is well-defined by passing to the limit in the above. Since, by Lemma 10, $K(0)$ is symmetric non-negative and satisfies the above, it must be of the form (see for example Proposition 3 of [TMB20])

$$K(0) = \frac{1}{\sqrt{2}}(\mathscr{L}_0^{-1})^\top \sqrt{\mathscr{L}_0^\top \Sigma \mathscr{L}_0}\, \mathscr{L}_0^{-1},$$

where $\mathscr{L}$ is given in the proposition. This completes the proof of Proposition 3.

III.C.4 Proof of Proposition 4

As in the proof of Section III.C.3, we consider a trading strategy ending at time T given by the function $f\colon [0,T] \to \mathbb{R}^d$. We re-write the cross-impact contributions as $K(t) = \Lambda + \Gamma(t)$, where $\Gamma(t) \underset{t\to\infty}{\to} 0$. Then, average cost $C(f)$ of the trading strategy f given by Definition 2 can be re-written as

$$C(f) := \int_0^T f(t)^\top \int_0^t \Lambda f(s)\mathrm{d}s\mathrm{d}t + \int_0^T f(t)^\top \int_0^t \Gamma(t-s)f(s)\mathrm{d}s\mathrm{d}t = C_p(f) + C_t(f),$$

where we have split into permanent and temporary impact costs. Then

$$C_p(f) = \frac{1}{2}\sum_{i=1}^d \Lambda_{ii}\left\|f_i\right\|_1^2 + \sum_{i\neq j}^d \Lambda_{ij}\int_0^T \int_0^t f_i(t)f_j(s)\mathrm{d}s\mathrm{d}t.$$

Similarly, for the temporary impact costs, we have

$$C_t(f) = \sum_{i=1}^d \int_0^T \int_0^t f_i(t)\Gamma_{ii}(t-s)f_i(s)\mathrm{d}s\mathrm{d}t + \sum_{i\neq j}^d \int_0^T \int_0^t f_i(t)\Gamma_{ij}(t-s)f_j(s)\mathrm{d}s\mathrm{d}t.$$

Consider two distinct assets, Asset a and Asset b. Consider, as in [SL19], a round-trip trading strategy of the following form, here $v_p, v_q \in \mathbb{R}$:

$$f_a(t) := \begin{cases} v_p & \text{for } 0 \le t \le T/3 \\ 0 & \text{for } T/3 \le t \le 2T/3, \\ -v_p & \text{for } 2T/3 \le t \le T \end{cases} \qquad f_q(t) := \begin{cases} v_q & \text{for } 0 \le t \le T/3 \\ -v_q & \text{for } T/3 \le t \le 2T/3. \\ 0 & \text{for } 2T/3 \le t \le T \end{cases}$$

Then, the permanent impact costs contribution is of the form

$$C_p(f) = (\Lambda_{pq} - \Lambda_{qp}) \frac{T^2}{18} \nu_p \nu_q.$$

Similarly, for the temporary impact costs, we have

$$\begin{aligned} C_t(f) = &\int_0^T \int_0^t f_p(t) \Gamma_{pp}(t-s) f_p(s) \mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_q(t) \Gamma_{qq}(t-s) f_q(s) \mathrm{d}s\mathrm{d}t \\ &+ \int_0^T \int_0^t f_p(t) \Gamma_{pq}(t-s) f_q(s) \mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_q(t) \Gamma_{qp}(t-s) f_p(s) \mathrm{d}s\mathrm{d}t. \end{aligned}$$

Therefore, since Γ is power-law with all exponents strictly below 1, we have

$$\frac{C(f)}{T^2} \underset{T\to\infty}{=} \frac{\nu_p \nu_q}{18} (\Lambda_{qp} - \Lambda_{pq}) + o(1),$$

so that if $\Lambda_{qp} \neq \Lambda_{pq}$, arbitrages are possible. Therefore Λ is necessarily symmetric.

We now show that Λ is non-negative. Let $\tau \in \mathbb{R}$ and $\eta \in \mathbb{R}^d$ and consider a trading strategy f which buys portfolio η and waits τ units of time to sell it. Then, the cost of this trading strategy is

$$C(f) = \eta^\top (\Lambda + \Gamma(\tau)) \eta,$$

and by no-arbitrage, $C(f) \geq 0$. Therefore, for all $\tau \in \mathbb{R}$ and $\eta \in \mathbb{R}^d$, we have

$$\eta^\top \Lambda \eta \geq -\eta^\top \Gamma(\tau) \eta,$$

so that using the fact that $\Gamma(\tau) \underset{\tau\to\infty}{\to} 0$, Λ is non-negative. This concludes the proof of Proposition 4.

III.C.5 Proof of Proposition 5

To prove Proposition 5, we proceed in three steps. First, we use polarization to show that it suffices to prove the result for $\xi^* \mathcal{Z} \xi$ for any $\xi \in \mathbb{C}^d$. Second, we use sufficient conditions for smoothness on characteristic functions. Finally, we show that these conditions are satisfied in our setting.

We begin with a polarization identity. Since $\mathcal{Z}$ is a continuous positive definite function, Theorem 2.10 of [AKS16] shows that for every $\xi \in \mathbb{C}^d$ the continuous function $\mathcal{Z}_\xi \colon t \mapsto \xi^* \mathcal{Z}(t) \xi$ is positive definite. Furthermore, for all $1 \leq a, b \leq d$, the component $e_b^\top \mathcal{Z}(t) e_a$ is equal to

$$\frac{1}{2} \left(\mathcal{Z}_{e_a + e_b}(t) - i \mathcal{Z}_{e_a - i e_b}(t) - (1-i) \mathcal{Z}_{e_a}(t) - (1-i) \mathcal{Z}_{e_b}(t) \right).$$

Thus it suffices to prove the result for $\mathcal{Z}_\xi$ for $\xi \in \mathbb{C}^d$, which we show below.

For all $\xi \in \mathbb{C}^d$, we introduce the measure $\mathbb{Q}_\xi := \dfrac{\xi^* \mathbb{M} \xi}{\xi^* \mathbb{M}(\mathbb{R}) \xi}$. For any $\xi \in \mathbb{C}^d$, $\mathbb{Q}_\xi$ is indeed a probability measure since the matrix-valued measure $\mathbb{M}$ is non-negative definite and of finite total variation. Thus, the function $\mathcal{Z}_\xi$ is the characteristic function associated to the probability measure $\mathbb{Q}_\xi = \dfrac{\xi^* \mathbb{M} \xi}{\xi^* \mathbb{M}(\mathbb{R}) \xi}$. We now use Theorem 2.3.1 of [Luk70] which gives a sufficient condition for the smoothness of a characteristic function. To apply the theorem, we must show that for all $\xi \in \mathbb{C}^d$ we have

$$\liminf_{t \to 0} \frac{| \mathcal{Z}_\xi(2t) - \mathcal{Z}_\xi(0) + \mathcal{Z}_\xi(-2t) - \mathcal{Z}_\xi(0) |}{4t^2} < \infty.$$

Using the definition of $\mathcal{Z}_\xi$, the above condition is equivalent to

$$\liminf_{t \to 0} \frac{| \xi^* \left(K(2t) - K(0) + K(2t)^\top - K(0)^\top \right) \xi |}{4t^2} < \infty.$$

However, by Assumption 4, K and its derivative are continuously differentiable at zero and Proposition 3 shows that $K(0)$ is symmetric, so that

$$\frac{K(2t) - K(0) + K(2t)^\top - K(0)^\top}{4t^2} \underset{t \to 0}{=} \frac{1}{2t} (K'(0) + K'(0)^\top) + K''(0) + K''(0)^\top + o(1).$$

Therefore the condition will be satisfied and the proposition proven if $K'(0) = -K'(0)^\top$, that is if $K'(0)$ is antisymetric. Then Theorem 2.3.1 of [Luk70] yields that $\mathcal{Z}_\xi$ is twice differentiable and the integrals converge absolutely. To prove this result, we will use the no-arbitrage condition and the smoothness of the cross-impact kernel around zero.

As in the proofs of Sections III.C.3 and III.C.4, we consider a trading strategy ending at time T given by the function $f \colon [0, T] \to \mathbb{R}^d$. We re-write the cross-impact contributions as $K(t) = K(0) + tK'(0) + \mathcal{R}(t)$, where $\mathcal{R}$ is such that $\mathcal{R}(0) = 0$, $\mathcal{R}$ is continuously differentiable at zero and $\mathcal{R}(t) \underset{t \to 0}{=} o(t)$. For convenience, we write $M(t) := K(0) + tK(0)$. Then the trading cost of the trading strategy is

$$C(f) := \int_0^T f(t)^\top \int_0^t M(t-s) f(s) \mathrm{d}s \mathrm{d}t + \int_0^T f(t)^\top \int_0^t \mathcal{R}(t-s) f(s) \mathrm{d}s \mathrm{d}t = C_1(f) + C_2(f).$$

Where we have split contributions. Consider two distinct assets, Asset a and Asset b and the round-trip trading strategy of the following form, where $v_a, v_b \in \mathbb{R}$:

$$f_a(t) := \begin{cases} v_a & \text{for } 0 \le t \le T/3 \\ 0 & \text{for } T/3 \le t \le 2T/3 \\ -v_a & \text{for } 2T/3 \le t \le T \end{cases}, \qquad f_b(t) := \begin{cases} v_b & \text{for } 0 \le t \le T/3 \\ -v_b & \text{for } T/3 \le t \le 2T/3 \\ 0 & \text{for } 2T/3 \le t \le T \end{cases}.$$

This strategy only trades Asset a and Asset b, i.e. for all other Asset i, $f_i = 0$. This is a round-trip strategy since $\int_0^T f = 0$. Then, computing contributions as in Lemma 10 and using the fact that $K(0)$ is symmetric, we obtain

$$C_1(f) = \frac{-5T^3}{162} v_a v_b (K'(0)_{ab} + K'(0)_{ba}).$$

For the other contribution, we have

$$C_2(f) = \int_0^T \int_0^t f_a(t)\mathscr{R}_{aa}(t-s)f_a(s)\mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_a(t)\mathscr{R}_{bb}(t-s)f_a(s)\mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_a(t)\mathscr{R}_{ab}(t-s)f_b(s)\mathrm{d}s\mathrm{d}t + \int_0^T \int_0^t f_b(t)\mathscr{R}_{ba}(t-s)f_a(s)\mathrm{d}s\mathrm{d}t.$$

Therefore, since $\mathscr{R}(0) = 0$ and $\mathscr{R}(t) \underset{t\to 0}{=} o(t)$, for small enough execution times T we have that for all $t \in [0,T]$, $|\mathscr{R}_{ij}(t)| \leq \epsilon t$ for all $(i,j) \in \{a,b\}^2$. Thus, we then have, where $c_1 > 0$ is a constant independent of f, T and ϵ

$$C_2(f) \leq \epsilon c_1 T^3.$$

Thus, combining both terms we obtain

$$\frac{C(f)}{T^3} \leq c_2 \nu_a \nu_b (K'(0)_{ab} + K'(0)_{ba}) + \epsilon,$$

where $c_2 \neq 0$ is a constant independent of f, T and ϵ. Therefore, unless $K'(0)_{ab} = -K'(0)_{ba}$, the round-trip strategy yields negative costs and arbitrages are possible. Therefore $K'(0)$ is necessarily antisymmetric. The proposition then follows.

III.D Calibration methodology details

This section details the calibration methodology. The objective is to calibrate the cross-impact kernels K^1 and K^2 on empirical data. Section III.D.1 gives additional details on the dataset. Section III.D.2 details the methodology for the estimation of the empirical observables Σ and Ω. Section III.D.3 explains how spectral factors of the reduced covariance measure are computed. Finally, Section III.D.4 explains the construction of both kernels.

III.D.1 Data preparation and processing

The data is processed according to the procedure outlined in [TMB20]. For the reader's convenience, we recall some key elements here. The two instruments selected are the leading and third month E-Mini Futures. Prices and trades are gathered from anonymous trades and quotes data from the CME and prices are taken as the mid-price of the best bid and ask prices of each instrument. To avoid stationarity issues, we remove data outside the commonly traded hours of both instruments and the first and last 30 minutes of the trading period. Data ranges from January 2015 to December 2018.

III.D.2 Estimation of empirical observables

Using the data outlined in Section III.D.1, we detail here the estimation methodology for Σ and Ω. To simplify computations and reduce noise, we aggregate order flows and prices by bins of 1 seconds. Each bin contains the opening and closing price p_t^o and p_t^c as well as

the total signed order flow q_t. Both of these random variables have zero mean. Using these conventions, we use naive statistical estimators to compute the price-covariance and the order flow auto-covariance, namely, for a daily timeseries of prices and signed order flows $\{p_t\}_{t=1}^T$ and $\{q_t\}_{t=1}^T$ at bins of one second:

$$\Sigma = \frac{1}{T-1}\sum_{t=1}^{T}(p_t^c - p_t^o)(p_t^c - p_t^o)^\top$$
$$\Omega(\tau) = \frac{1}{T}\sum_{t=1}^{T} q_{t+\tau}q_t^\top .$$

This procedure is averaged across days to obtain an empirical estimate of Σ and Ω using approximately 800 days of available data. For stationary ergodic point processes, these averages do converge towards the theoretical price-covariance and reduced covariance measure by Proposition 8.3.1 of [DVJ08].

III.D.3 Estimation of a spectral factor of Ω

Once Σ and Ω have been estimated, the cross-impact kernel K^1 can be computed by solving numerically Equation (5) and setting the boundary conditions imposed by arbitrage-admissibility. To do so, we need to compute a spectral factor of Ω.

To solve Equation (9), we use the SBR2 algorithm [MBC+07, WMW15], a polynomial eigenvalue decomposition method, to compute a numerical approximation of a spectral factor of Ω, $\mathscr{L}$. The implementation details follow the simple version of this algorithm (and not its subsequent improvements) outlined in Equation (9). The algorithm was tested on the examples provided in both papers and results were similar to those reported in the papers. To estimate the accuracy of the spectral decomposition, we computed the Frobenius norm of the error matrix $\frac{\|\Omega(z) - \mathscr{L}(z)\mathscr{L}(1/z)^*\|_F}{\|\Omega(z)\|_F} \approx 6\cdot 10^{-8}$. Thus, numerically, $\mathscr{L}$ is a good approximation for a spectral factor of Ω.

A key property is that this spectral factor has the convenient form $\mathscr{L}(z) = D(z)H(z)$ where H is a para-unitary polynomial matrix and D is (close to) a diagonal polynomial matrix. Both D and H are outputs of the SBR2 algorithm. To check how close D is to a diagonal matrix, we compute the Frobenius norm of its off-diagonal elements relative to its diagonal elements. The numerical results yield 10^{-15}, which shows that the decomposition was successful. Once this spectral factor has been obtained, we need to compute the matrix polynomial $\mathscr{L}^{-1}$ to obtain the martingale-admissible kernel K^1.

III.D.4 Computation of the cross-impact kernels

Because of the structure of $\mathscr{L}(z)$, computing its inverse is straightforward. One the one hand, the inverse of the para-unitary polynomial matrix $H(z)$ is $H(1/z)^*$. On the other hand, the inverse of $D(z)$ is obtained by taking the inverse of its diagonal elements, which is straightforward using pole decomposition. This allows us to compute numerically the polynomial matrix

$\mathscr{L}^{-1}$ and, using Equation (5), the martingale-admissible kernel K^1.

The numerical computation of the poles of polynomials is done using the `residuez` function of the `signal` library of the `scipy` module [VGO+20] of the `python` programming language [VRD09]. All poles found were strictly inside the unit circle. However, they were numerically close to modulus one, which is consistent with the long-range auto-correlations of Ω reported in Figure III.1.

Once the martingale-admissible kernel K^1 has been obtained, we derive the nsa-admissible kernel K^2 in the following manner. First, we compute the (numerical) Fourier transform of K^1. The symmetric part of this Fourier transform is then modified so that all its eigenvalues are non-negative. Finally, this clipped Fourier transform is added to the asymmetric part of the Fourier transform of K^1. The (numerical) inverse Fourier transform then yields the nsa-admissible kernel K^2.

CHAPTER IV

Measuring cross-impact from labeled order flow

Abstract

Using two databases of meta-orders on stocks and futures, we measure cross-impact: the influence trades on one asset have on the prices of other assets. We propose generalisations of the square-root law which account for return and order flow correlations and provide significantly better fit than the single-asset square-root law. Finally, we provide a recipe for predicting transaction costs of portfolios from daily price and volume data on stocks and futures.

From:
Cross-impact on meta-orders
M. Tomas, I. Mastromatteo, M. Benzaquen

Introduction

A key goal of market microstructure is to understand how markets digest trades to form prices. Thanks to growing amounts of financial data, studies have been able to precisely measure subtle effects. Many focused on *market impact*: the average price change induced after a meta-order (a sequence of trades from the same investor in the same direction) [BBDG18, ATHL05, Tor97, ZTFL15, BBLB18, BBLB19, BR13, BILL15]. These studies find that, when an investor buys or sells Q dollars of risk (where dollars of risk := number of shares × volatility × asset price) of an asset on a given trading day, he pushes on average the price of the asset according to the "square-root law":

$$\mathbb{E}[s \mid Q] \approx Y \left(\frac{Q}{V}\right)^{\bullet\delta}, \tag{1}$$

where $s := \frac{S_c - S_o}{\sigma S_o}$ is the normalized daily asset return, S_c is the price at the end of the trading day, S_o the price at the beginning of the trading day, $\sigma := (S_{\text{high}} - S_{\text{low}})/S_o$ is the daily return volatility of the asset, V is the daily traded dollars of risk, $\delta \approx 0.5$ captures the concavity of

market impact, $x^{\bullet\delta} = |x|^{\delta}\mathrm{sgn}(x)$ and Y is a prefactor of unit scale. This finding is robust across asset classes and time periods, holding even on Bitcoin [DB15] and options [TEB16, Sai19]. Refinements of the square-root law suggest that the relationship is linear for small values of the traded volume [BBLB19, ZTFL15].

Yet market impact and the square-root law do not tell us how trading portfolios affects market prices. Thus, it provides an incomplete picture of the price formation process. For practitioners, estimating transaction costs of portfolios using Equation (1) may severely misrepresent actual costs. This led to some recent work which measures *cross-impact* [RT21b, TMB20, WSG16, WNG17, BMEB17, TMB21, SL19], the multivariate version of market impact. These studies use plentiful anonymous trades data to calibrate models. While they may make model-founded predictions, to our knowledge there is no study which has provided empirical measures of cross-impact from meta-orders.

To address this, this chapter measures cross-impact using two databases of meta-orders. The first database contains trades executed by multiple investors in the U.S. Equity market. The second covers meta-orders executed by a single asset manager in the futures markets.

The main contribution of this chapter is a generalisation of the square-root law which accounts for the cross-section of returns and order flows. To do so, we propose models for the average normalized return conditioned on a traded amount of risk $Q := (Q_1, \cdots, Q_d)$ where Q_i is the net traded dollars of risk on Asset i. Using meta-order data and previous studies on linear cross-impact models [CK94, GdMMBB20, BMEB17, MBEB17, TMB20, TMB21] we find that there exist a set of portfolio directions $\Pi = (\pi_1, \cdots, \pi_d)$ such that, for a portfolio π in Π, the projected normalized returns $s_\pi = \pi^\top s$ have a concave dependence on the projected traded dollars of risk $Q_\pi = \pi^\top Q$ similar to the square-root law:

$$\mathbb{E}[s_\pi \mid Q_\pi] \approx Y\sigma_\pi \left(\frac{Q_\pi}{V_\pi}\right)^{\bullet\delta}, \tag{2}$$

where $\sigma_\pi^2 := \pi^\top \varrho \pi$ is the daily volatility of the portfolio, $\varrho := \mathrm{Corr}(s)$ is the daily normalized return correlation matrix, $V_\pi^2 := \pi^\top \Omega \pi$ is a proxy for the liquidity of the portfolio where $\Omega := \mathrm{Cov}(q)$ is the daily traded risk covariance matrix (where $q = (q_1, \cdots, q_d)$ is the daily traded risk on all assets), $\delta \approx 0.5$ and Y is a prefactor of unit scale.

Equation (2) gives predictions for portfolio returns according to the traded meta-order dollars of risk. For a given basis Π, we propose the following impact law for the average normalized return of Asset i conditioned on the net traded risk on each asset:

$$\mathbb{E}[s_i \mid Q_1, \cdots, Q_d] \approx Y \sum_{k=1}^{d} \pi_k^i \sigma_{\pi_k} \left(\frac{Q_{\pi_k}}{V_{\pi_k}}\right)^{\bullet\delta} = Y \sum_{k=1}^{d} \pi_k \sqrt{\pi_k^\top \varrho \pi_k} \left(\frac{\pi_k^\top Q}{\sqrt{\pi_k^\top \Omega \pi_k}}\right)^{\bullet\delta}. \tag{3}$$

We find that on both datasets, Equation (3) has significantly better fit with π which depends on the correlation structure of asset returns and liquidity than for the basis $\Pi = (e_1, \cdots, e_d)$,

which ignores cross-sectional effects. As such, we can view Equation (3) as a more precise generalisation of the square-root law which accounts for the structure of returns and liquidity.

We now comment on the links between our chapter and the literature.

The proposed impact law gives predictions of cross-impact from meta-orders. We can use it to compare the predictions of different cross-impact models studied in empirical studies [SL19, BMEB17, MBEB17, TMB21, TMB20, RT21a, WSG16, WSG15] and inform modeling decisions. In particular, our findings motivate the choice of particular propagator models for cross-impact, favoring those introduced in [RT21a, BMEB17, MBEB17].

Furthermore, while the proposed impact law was only examined on stocks and futures, it could be extended to deal with more general cases, such as derivatives, thanks to results discussed in [TMB21]. It would then yield predictions consistent with what has been observed in meta-order studies on options [TEB16, Sai19]. This suggests the chapters' results hold in more generality than simply on stocks and futures.

The proposed impact law gives predictions for transaction costs. These could be incorporated in portfolio construction problems and prescribe specific choices for cross-impact modeling in these frameworks, where the cross-impact model is often left unspecified [EMK19, GP16]. Furthermore, they could help regulators better assess the liquidation risk of portfolios.

The chapter is organized as follows. Section 1 proposes a formula for cross-impact associated to a given orthonormal basis and introduces the different bases examined in the chapter. Section 2 applies our framework on the meta-order datasets to measure goodness-of-fit of our predictions. Finally, we conclude on the contributions of the chapter, open questions, and directions for future work.

1 Cross-impact

This section proposes a formula to estimate cross-impact. We begin with its definition.

Definition 1 (Cross-impact). *An asset manager decides to buy or sell a portfolio $Q = (Q_1, \cdots, Q_d)$ of d different assets, each made up of Q_i dollar-risks of Asset i. He, or his broker, splits his order in smaller orders and executes them on the market. The cross-impact of a meta-order of the portfolio Q is*

$$\mathscr{CI}(Q) := \mathbb{E}[s \mid Q], \tag{4}$$

where $s = (s_1, \cdots, s_d)$ are the normalized daily returns of the assets.

Directly measuring cross-impact as defined in Equation (4) is arduous since we need to measure the average of a random variable conditioned with d random variables, where d is large. Since precise measures of market impact require on the scale of 10^5 meta-orders [ZTFL15, BBLB18, BBLB19, ATHL05, TEB16], precise measures of cross-impact would require

on the scale of 10^{5d} meta-orders. Thus, we are unable to directly measure cross-impact. To tackle this problem, the next section introduces the concept of diagonalisation of cross-impact.

1.1 Diagonalisation of cross-impact and proposed cross-impact law

Before introducing the concept of diagonalisation of cross-impact, we note that we can apply a change of basis to simplify the problem. Let $\Pi := (\pi_1, \cdots, \pi_d)$ be an orthonormal basis of $\mathbb{R}^d$. Then, since Π is a basis of $\mathbb{R}^d$, there is a one to one correspondence between the distribution of the family of N random variables $\mathbb{E}[s_i \mid Q_1, \cdots, Q_d]$ and the distribution of the family of N random variables

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}],$$

where $s_{\pi_i} := \pi_i^\top s$ is the portfolio return and $Q_{\pi_i} := \pi_i^\top Q$ is the traded volume of dollar-risk of the portfolio. This reformulation does not yield a straightforward solution since it is similarly difficult to estimate the above d random variables. However, it suggests an alternative approach. If we had

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}] = \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}],$$

then the price impact on portfolio π_i only depends on what is traded on portfolio π_i. This leads to the following definition.

Definition 2 (Diagonal basis for cross-impact). *An orthonormal basis $\Pi = (\pi_1, \cdots, \pi_d)$ of $\mathbb{R}^d$ is said to diagonalise cross-impact if, for all $1 \leq i \leq d$, we have*

$$\mathbb{E}[s_{\pi_i} \mid Q_{\pi_1}, \cdots, Q_{\pi_d}] = \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}], \tag{5}$$

where $s_{\pi_i} := \pi_i^\top s$ is the daily normalized return of the portfolio and $Q_{\pi_i} := \pi_i^\top Q$ is the traded volume of dollar-risk of the portfolio.

Naturally, the challenge is then to find a basis such that Equation (5) holds. Inspired by the previous definition, for a given orthonormal basis Π, we associate the following prediction for cross-impact:

$$\widehat{\mathscr{CI}}_\Pi(Q) := Y \sum_{k=1}^{d} \mathscr{I}_{\pi_k}(Q_{\pi_k})\pi_k, \tag{6}$$

where $\mathscr{I}_{\pi_k}(Q_{\pi_k}) := \mathbb{E}[s_{\pi_k} \mid Q_{\pi_k}]$ are functions to be estimated. This prediction is motivated by the fact that when a basis respects Equation (5), cross-impact can be reduced to Equation (6). For this chapter, we have restricted our study to orthonormal bases. The portfolios of these bases have unit risk and no overlap, i.e. $\pi_i^\top \pi_j = 0$ for two different portfolios π_i and π_j. Their components can be interpreted as positions in each asset expressed in units of risk.

Depending on the choice of basis Π, Equation (6) may yield sensible predictions or not. The next sections examine relevant choices of bases.

1.2 Bases of interest

1.2.1 Canonical basis

The first basis examined is the canonical basis which ignores cross-sectional effects and we thus dub the `direct` basis:

$$\Pi_{\texttt{direct}} := (e_1, \cdots, e_d)\,,$$

where the e_i are the canonical vectors of $\mathbb{R}^d$. This basis uses the predictions for market impact on each asset to predict cross-impact. As underscored in the introduction, the functions $\mathscr{I}_{e_i}$ are well described by the square-root law of Equation (1). As a basis which ignores cross-sectional effects, we expect this basis of being far from satisfying the diagonalisation condition of Equation (5). Overall, this basis should work best when asset prices and order flows are not correlated and break down when there is strong structure in prices and order flows.

1.2.2 Random basis

The second basis of interest is the `random` basis defined as

$$\Pi_{\texttt{random}} := (o_1, \cdots, o_d)\,,$$

where the basis $(o_1, \cdots, o_d)$ has been randomly sampled in the orthogonal group. This basis serves as a zero-intelligence basis which accounts for cross-sectional effects. Since this choice randomly mixes the information of all assets, it should be closer to satisfying the diagonalisation condition of Equation (5) than the `direct` basis. However, the cross-impact predictions associated to this basis of Equation (6) may have worse fit than those associated to the `direct` basis because the portfolios o_i have little economic interpretation. Contrary to the `direct` basis, we have no prior knowledge of the functions $\mathscr{I}_{o_i}$, so that we will need to estimate it using meta-order data.

1.2.3 Eigenliquidity model basis

The third selected basis accounts for return correlations. It is taken using the eigenvectors of the normalized return correlation matrix $\rho := \mathrm{Corr}(s)$:

$$\Pi_{\texttt{el}} := \mathrm{eigenvectors}(\rho)\,.$$

This basis is dubbed eigenliquidity model basis (`el` for short) basis since it is inspired by the eigenliquidity cross-impact model [TMB20, BMEB17, MTB14] which has the same eigenvectors as ρ. Given the goodness-of-fit of the linear eigenliquidity model on anonymous data, the cross-impact predictions associated to this basis may fit empirical data well. As in the `random` basis, we will have to estimate empirically the functions $\mathscr{I}_\pi$ where π is in the basis $\Pi_{\texttt{el}}$.

1.2.4 Kyle model basis

The final basis chosen uses both return and order flow structure, encoded in the normalized return correlation matrix ρ and the co-trading effects encoded in the traded risk covariance

matrix $\Omega := \mathrm{Cov}(q)$, where $q := (q_1, \cdots, q_d)$ is the daily signed traded risk of each asset. The `kyle` basis is defined as

$$\Pi_{\texttt{kyle}} := \mathrm{eigenvectors}((\mathscr{L}^{-1})^\top \sqrt{\mathscr{L}^\top \rho \mathscr{L}}\, \mathscr{L}^{-1}),$$

where $\mathscr{L}$ is such that $\mathscr{L}\mathscr{L}^\top = \Omega$. We refer to this basis as the `kyle` basis since it is inspired by the multivariate Kyle model for cross-impact [TMB20, GdMMBB20, CK94], which has also shown good fit on empirical data [TMB20]. Once again, the functions $\mathscr{I}_\pi$ where π is in the basis $\Pi_{\texttt{kyle}}$ have to be estimated.

2 Empirical results

This section assesses the quality of the predictions of our setup on meta-order datasets.

2.1 Meta-order datasets

This study uses two databases of proprietary meta-orders to analyse cross-impact. We present the characteristics of each dataset below.

2.1.1 ANcerno dataset

The first database is made available by ANcerno, formerly Abel Noser Corporation, which is one of leading consulting firm that works with institutional investors to monitor their equity trading costs. This database contains orders executed by different institutional investors in the U.S. equity market. Our sample of this database contains stocks and orders from 2000 to 2014. For previous studies on the ANcerno database, we refer the reader to [BBLB18, BBLB19, ZTFL15].

In the following we will identify a meta-order as a series of consecutive orders performed by a single investor, through a single broker within a single day, on a given stock and with a fixed direction (buy/sell). Thus each meta-order is characterised by a broker label, a stock symbol, the signed number of exchanged shares (counted positively for buy orders and negatively for sell orders), executed during a physical time interval $[t_s, t_e]$ with t_s the start time and t_e the end time of the execution.

There are many entries in the ANcerno database and some filtering is needed to remove possibly erroneous data. To do so, we follow the cleaning procedure introduced in [ZTFL15], which we recall below:

(i) remove meta-orders which last for more than one day;
(ii) remove meta-orders which end after 4pm;
(iii) remove meta-orders which last for less than 2 minutes;
(iv) remove meta-orders with participation ratio larger than 30%.

The statistical properties of the remaining meta-orders are in line with the previous studies [ZTFL15, BBLB19, BBLB18].

To avoid an elaborate analysis of when precisely each meta-order starts and ends, how they overlap and which reference price to take, we consider impact at the daily scale. Doing so, we aggregate the meta-orders of different asset managers on a given stock. For a given trading day and asset, the signed daily volume is the sum of the signed volume of all meta-orders on that day. We assume that the daily meta-order starts when the market opens and finishes when the market closes.

The resulting dataset is combined with daily open/high/low/close price data and sector classification. To do so, we match symbols from the ANcerno database using available market data. Around 40% of stocks do not match our market data and are removed in this procedure.

The final dataset contains 2708 stocks, distributed among different sectors. With our daily aggregation methodology, we are left with around 2.8 million meta-orders. To keep the number of distinct stocks tractable, we select 1000 random stocks among our dataset. We are left with 1.6 million meta-orders.

2.1.2 CFM Futures dataset

The second dataset employed in this study contains data on the proprietary meta-orders executed by the asset manager Capital Fund Management (CFM) on the futures market. After cleaning, the dataset covers around 250 different futures, with approximately 200,000 meta-orders from 2012 to 2019.

Contrary to the ANcerno dataset, meta-orders of this dataset are only those executed by CFM. Thus, a meta-order is identified with a future code, the signed number of exchanged lots (counted positively for buy orders and negatively for sell orders), executed during a physical time interval $[t_s, t_e]$ with t_s the start time and t_e the end time of the execution. To apply a consistent methodology with the ANcerno dataset, we aggregate orders at the daily scale.

2.2 Estimates of ρ and Ω

This section details the estimation of the normalized return correlation matrix ρ and traded risk covariance matrix Ω. In this chapter, we only have access to daily price and volume data. Therefore, we are unable to directly compute Ω which requires access to signed trades. Instead, we propose to use an estimate of the liquidity corelation matrix based on the return correlation matrix. From previous studies [BMEB17, TMB20, TMB21], we know that the order flow correlation matrix shares the first eigenvectors of the return correlation matrix, although the order flow correlation matrix is much more diagonal. Thus, we set $\rho^{\Omega} := \alpha I + (1-\alpha)\rho$ for some choice of α which captures the scale of ρ. For the ANcerno dataset, previous studies [TMB20, BMEB17] motivate the choice $\alpha = 0.9$ as the traded risk correlation matrix was found to have some of the structure of ρ - although more diagonal than ρ. For the CFM futures dataset, the previous study [TMB20] suggests $\alpha = 0.1$ as the traded risk correlation matrix was found to be close to the return correlation matrix, in particular because the liquidity of

strongly correlated instruments is strongly correlated. For example, two futures with the same underlying have significant co-trading.

Using the above estimate for ϱ^{Ω}, the daily traded risk covariance matrix on a given day is approximated by $\Omega_t := \mathrm{diag}(V_t)\varrho^{\Omega}\mathrm{diag}(V_t)$, where V_t is the vector of daily traded risk of each asset. To compute the eigenvectors of the `kyle` basis, we compute the average daily traded risk covariance matrix as $\bar{\Omega} := \mathrm{diag}(\bar{V})\varrho^{\Omega}\mathrm{diag}(\bar{V})$, where $\bar{V}$ is the vector of average daily traded risk of each asset.

Figure IV.1 shows the structure of the daily normalized return correlation matrix ϱ on both datasets.

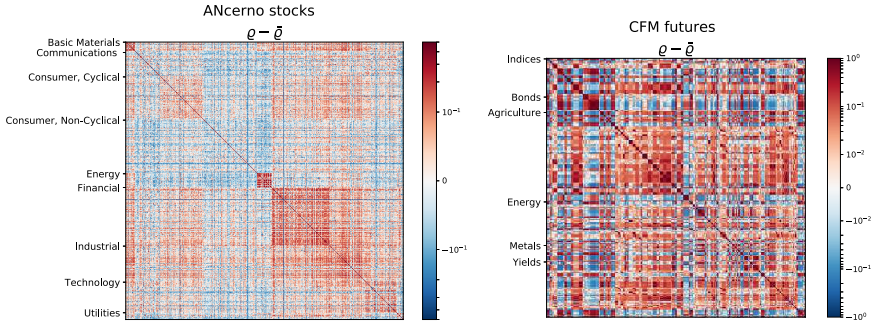


Figure IV.1: **Price correlation matrix ϱ on both datasets.**
Left: ANcerno dataset. **Right**: CFM dataset. Correlation matrices are grouped by sector to highlight the block-wise structure. For the ANcerno dataset, the average pair-wise correlation value of stocks is removed to remove the influence of the market mode. Both correlation matrices show strong structure, with the correlation matrices on future highlighting the strong correlation within futures of the same sector. We can also distinguish sub-blocks within a particular sector, which correspond to futures with the same underlying.

2.3 Diagonalisation in bases of linear cross-impact models

Given the estimates for ϱ and Ω of the previous section, we can now stress-test different bases. This section examines how close different bases are to satisfying the diagonalisation condition of Equation (5). To do so, for a given orthonormal basis Π, we introduce the residual differences

$$Z_{ij}^{\Pi} := \left(\mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}, Q_{\pi_j}] - \mathbb{E}[s_{\pi_i} \mid Q_{\pi_i}]\right)^2, \tag{7}$$

which measure the error of the approximation. A global metric for our diagonalisation approximation error in the basis Π is then

$$Z^{\Pi} = \frac{1}{d} \sum_{\substack{1 \le i,j \le d \\ i \neq j}} Z_{ij}^{\Pi}, \tag{8}$$

which measures the average error made with the approximation.

Table IV.1 reports the separation metric for the different bases introduced in Section 1.2. All models from bases which account for cross-sectional effects have better scores than the `direct` basis model. As previously mentioned, the `random` model is expected to perform best since both elements of Equation (7) are close to zero. It is noteworthy then that, in both instances, the `el` and `kyle` bases reach comparable scores to the `random` basis, which suggests they are efficient in separating the information and somewhat close to satisfying the diagonalisation condition.

Dataset	Basis			
	`direct`	`random`	`el`	`kyle`
ANcerno stocks	11.0 ± 0.9	0.61 ± 0.01	0.50 ± 0.05	0.54 ± 0.05
CFM futures	2.02 ± 0.01	0.977 ± 0.003	0.55 ± 0.02	0.68 ± 0.02

Table IV.1: **Separation metric Z^Π in the bases of different impact models.**
The table reports the separation metric Z^Π on both datasets. For a given pair of portfolios π_i, π_j from a given basis, Z^Π_{ij} is computed by estimating both sides of Equation (7) using empirical averages. Then, due to the large number of portfolio pairs to examine to compute Equation (8), Z^Π is estimated by averaging Z^Π_{ij} on a smaller sample of portfolio pairs and the error bars correspond to the 95% confidence interval for our subsampled estimate of Z^Π.

2.4 Fit of the functions $\mathscr{I}_\pi$

The previous section showed that the `random`, `el` and `kyle` bases were much closer to satisfying the diagonalisation condition than the `direct` basis. For a given basis $\Pi = (\pi_1, \cdots, \pi_d)$, the associated predicted cross-impact of Equation (6) requires estimating the functions $\mathscr{I}_{\pi_k}(Q_{\pi_k}) := \mathbb{E}[s_{\pi_k} \mid Q_{\pi_k}]$. These functions are already known for the `direct` basis, since they correspond to market impact and are described by the square-root law. However, they need to be estimated for the `random`, `el` and `kyle` bases.

Figure IV.2 shows that, for each portfolio π in these bases, the functions $\mathscr{I}_\pi$ is well-described by a square-root law like fit:

$$\mathbb{E}[s_\pi \mid Q_\pi] \approx Y \sigma_\pi \left(\frac{Q_\pi}{V_\pi} \right)^{\bullet \delta}, \tag{9}$$

where $\delta \approx 0.4 - 0.6$, Y is a dimensionless prefactor (where $Y \approx 0.3$ on stocks and $Y \approx 0.5$ on futures), $V_\pi^2 := \pi^\top \Omega \pi$ is a proxy for the liquidity in the direction of the portfolio, $\sigma_\pi^2 := \pi^\top \varrho \pi$ is a proxy for the volatility of the portfolio. A key difference with the square-root law is the scale of the dimensionless prefactor Y. Although it is independent of the portfolio, it changes depending on the dataset and, in particular, the number of assets d (see [TMB20] for more details).

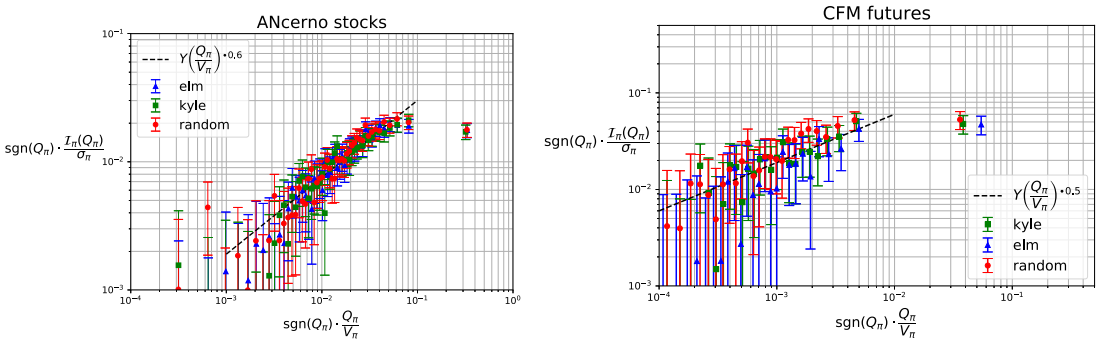


Figure IV.2: **Fit of $\mathscr{I}_\pi$ in different bases on both datasets.**
Left: ANcerno stocks dataset. **Right**: CFM futures dataset. Each point is computed by averaging normalized returns on binned traded risk. Error bars shown are computed using 95% confidence intervals. The ANcerno dataset has significantly more data, which allows us to obtain more precision. On both datasets, for participation ratios $\eta_\pi := \frac{Q_\pi}{V_\pi}$ in the range $10^{-3} \lesssim \eta_\pi \lesssim 10^{-1}$, the square-root law fit is a good approximation.

2.5 Goodness-of-fit

Using the results of Section 2.4 we can plug the functional form of Equation (9) in the prediction of cross-impact of Equation (6) to associate with a given basis Π among the `direct`, `random`, `el` and `kyle` bases the following prediction for cross-impact:

$$\widehat{\mathscr{C}\mathscr{I}}_\Pi(Q) \approx Y \sum_{k=1}^{d} \pi_k \sigma_{\pi_k} \left(\frac{Q_{\pi_k}}{V_{\pi_k}}\right)^{\bullet\delta} = Y \sum_{k=1}^{d} \pi_k \sqrt{\pi_k^\top \varrho \pi_k} \left(\frac{\pi_k^\top Q}{\sqrt{\pi_k^\top \Omega \pi_k}}\right)^{\bullet\delta}. \tag{10}$$

Equation (10) accounts for all portfolios, many of which may have small volatility and provide negligible contributions. To assess the quality of fit of Equation (10) as a function of the number of directions considered, we consider the truncated predictions $\widehat{\mathscr{C}\mathscr{I}}_{\Pi,n}(Q)$ defined as

$$\widehat{\mathscr{C}\mathscr{I}}_{\Pi,n}(Q) := Y \sum_{k=1}^{n} \pi_k \sigma_{\pi_k} \left(\frac{Q_{\pi_k}}{V_{\pi_k}}\right)^{\bullet\delta} = Y \sum_{k=1}^{n} \pi_k \sqrt{\pi_k^\top \varrho \pi_k} \left(\frac{\pi_k^\top Q}{\sqrt{\pi_k^\top \Omega \pi_k}}\right)^{\bullet\delta}, \tag{11}$$

where the portfolios π_k are sorted in decreasing order of volatility $\pi_k^\top \varrho \pi_k$. To evaluate the goodness-of-fit of the predictions obtained with Equation (11), we compute the r^2 between realized returns and the predictions given by $\widehat{\mathscr{C}\mathscr{I}}_{\Pi,n}(Q)$ as

$$r^2 = 1 - \frac{1}{d} \sum_{i=1}^{d} \frac{\mathrm{Var}(s_i - e_i^\top \widehat{\mathscr{C}\mathscr{I}}_{\Pi,n}(Q))}{\mathrm{Var}(s_i)},$$

where $\mathrm{Var}(X)$ is the variance of the random variable X. To favour the `direct` basis, the i-th variance in the sum is estimated on empirical data only using days where a meta-order on

Asset i was executed.

Figure IV.3 reports the estimated r^2 as a function of the fraction of explained variance $\frac{\sum_{i=1}^{n} \pi_i^\top \varrho \pi_i}{\sum_{i=1}^{N} \pi_i^\top \varrho \pi_i}$ associated to the chosen basis. On both datasets, the goodness-of-fit of cross-impact predictions given in the bases of the `kyle` and `el` models is higher than for the `direct` model. This suggests that, in these bases, Equation (6) gives a more precise generalisation of the square-root law which accounts for the structure of returns and liquidity.

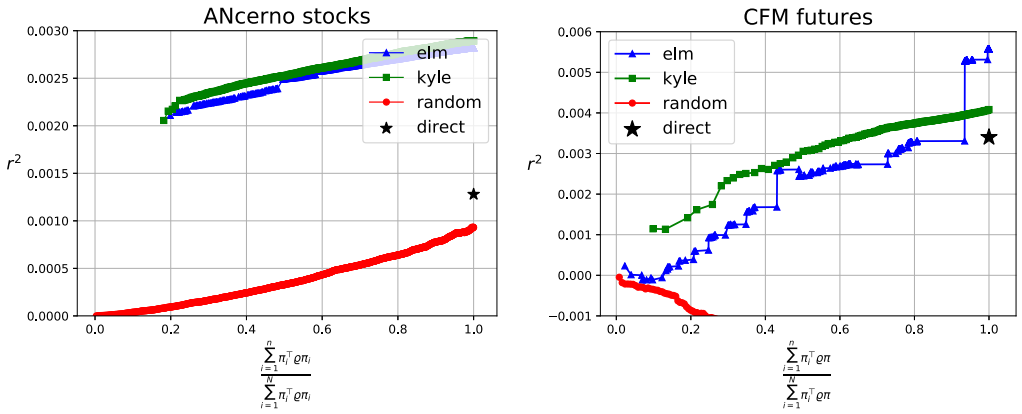


Figure IV.3: **Fit of different impact models on both datasets.**
Left: ANcerno stocks dataset. **Right**: CFM futures dataset. The r^2-squared measure of fit is computed for the `el` (blue triangles), `kyle` (green squares), `random` (red circles) and `direct` (black star) bases as a function of the cumulative explained variance. As expected, the random model shows poor performance on both datasets. Strikingly, both cross-impact bases outperform the `direct` basis. The difference is more pronounced on stocks than on futures. The quality of fit improves for both models as the explained variance grows.

Conclusion

The goal of this chapter was to use two meta-order databases to measure the cross-impact of investor orders. To do so, we proposed a generalisation of the square-root law which accounts for the structure of returns and liquidity. Section 2.5 showed that this generalisation gives more precise predictions than the square-root law.

A key result of this chapter is a recipe to compute the cross-impact of meta-orders from daily data. As a first step, this chapter has opens the door to other relevant questions concerning the cross-impact of meta-orders. For example, one could study how cross-impact decays after the meta-order. Similarly, one could try to relate the microscopic price formation process to the observed cross-impact. One possible approach would be to extend limit order book models of the type of [MTB14, TLD+11, DBMB15] to multiple assets. This study would give benchmarks for the predictions of such models.

The setup presented in this chapter is limited to the case of orthonormal bases and it would be interesting to examine non-orthonormal bases. For example, one could generalise the linear Kyle cross-impact model to a non-linear setting. Such a model would yield a recipe for aggregating trades and prices which differs from the models presented in this chapter.

Finally, an area of interest which we omitted in this chapter are the properties satisfied by the proposed cross-impact law. It would be relevant to examine which desirable properties outlined in [TMB20] are satisfied by the presented models. This could help us understand how to further improve the proposed cross-impact laws.

CHAPTER V

Microfounding multivariate price dynamics without order flows

Abstract

Rough volatility is a well-established statistical stylised fact of financial assets. This property has lead to the design and analysis of various new rough stochastic volatility models. However, most of these developments have been carried out in the mono-asset case. In this work, we show that some specific multivariate rough volatility models arise naturally from microstructural properties of the joint dynamics of asset prices. To do so, we use Hawkes processes to build microscopic models that reproduce accurately high frequency cross-asset interactions and investigate their long term scaling limits. We emphasize the relevance of our approach by providing insights on the role of microscopic features such as momentum and mean-reversion on the multidimensional price formation process. We in particular recover classical properties of high-dimensional stock correlation matrices.

From:
From microscopic price dynamics to multidimensional rough volatility models
M. Rosenbaum, **M. Tomas**

Introduction

A microstructural viewpoint on rough volatility

It is now widely accepted that volatility is rough (see [GJR18] and among others [DFZ19, LMPR18]): the log-volatility process is well-approximated by a fractional Brownian motion with small Hurst parameter $H \approx 0.1$, which corresponds to Hölder regularity of order $H - \epsilon$, $\epsilon > 0$. Furthermore, rough volatility models capture key features of the implied volatility surface and its dynamics (see [BFG16, EEGR19, HMT21]).

The macroscopic phenomenon of rough volatility is seemingly universal: it is observed for a large class of financial assets and across time periods. This universality may stem from

fundamental properties such as market microstructure or no arbitrage. This raised interest in building *microscopic* models for market dynamics which reproduce rough volatility at a *macroscopic* scale. For us, the microscopic time scale is of the order of milliseconds, where asset prices are jump processes, while the macroscopic scale is approximately of the order of days, where asset prices appear essentially continuous.

Hawkes processes, first introduced in [Haw71a, Haw71b, HO74] to model earthquake aftershocks, are nowadays very popular to model the high frequency dynamics of prices of financial assets (see [BMM15] for an overview of applications). In particular, the papers [EEFR18, JR15, JR16] successfully establish a link between rough volatility and history dependent Hawkes-type point processes which reproduce:

1. the no statistical arbitrage property: it is very hard to design strategies which are on average profitable at the high frequency scale;
2. the long memory property of order flow due to the splitting of large orders (meta-orders) into smaller orders;
3. the high degree of endogeneity of financial markets: the large majority of market activity (including price moves, cancellations and market and limit orders) occurs in response to previous market activity (as opposed to exogenous information such as news).

We refer to [EEFR18, HBB13] for details about these three stylised facts. This Hawkes-based microscopic framework can easily account for other features of markets: for example [JR18] examines the issue of permanent market impact, [EEGRR20] studies how a bid/ask asymmetry creates a negative price/volatility correlation, while the so-called Zumbach effect is considered in [DJR19].

Inspired by [EEFR18, JR15, JR16], the goal of this chapter is to use Hawkes processes to find a micro-founded setting of multivariate rough volatility which:

1. enforces no statistical arbitrage between multiple assets;
2. is consistent with the long memory property of the order flow and the high degree of endogeneity of financial markets;
3. explains stylised facts from the microscopic price formation process, with a focus on the structure of high-dimensional stock correlation matrices.

This approach enables us to characterise the type of price dynamics arising from those constraints. Readers interested in multivariate rough volatility may consult [CT19] for general construction of a class of affine multivariate rough covariance models. Our goal is more modest here: we are interested in finding macroscopic dynamics originating from microscopic insights, not in a full mathematical analysis of the class of possible models for multivariate rough volatility. Note also that in the concomitant work [JCLP19], the authors study weak solutions of stochastic Volterra equations in a very comprehensive framework. Some of our technical results can be derived from their general approach. In our setting, we rather provide simple

and natural proofs inspired from [EEFR18, JR15, JR16], allowing us to emphasize financial interpretations of the results, which is the core of this work.

Modeling endogeneity of financial markets

We first introduce the asymptotic framework which models the high endogeneity of financial markets in the mono-asset case (as in [BDHM13, EEFR18, JR15, JR16]) for clarity purposes before moving to the multivariate setting of interest. At the high frequency scale, the price is a piecewise constant process with upward and downward jumps captured by a bi-dimensional counting process $N = (N^{1+}, N^{1-})$, with N^{1+} counting the number of upward price moves and N^{1-} the number of downward price moves. Assuming that all jumps are of the same size, the microscopic price of the asset is the difference of the number of upward and downward jumps (where the initial price is set to zero for simplicity) and therefore can be written

$$P_t = N_t^{1+} - N_t^{1-}.$$

Our assumption is that N is a Hawkes process with intensity $\lambda = (\lambda^{1+}, \lambda^{1-})$ such that

$$\begin{aligned}\lambda_t^{1+} &= \mu_t^{1+} + \int_0^t \phi_{1+,1+}(t-s)dN_s^{1+} + \int_0^t \phi_{1+,1-}(t-s)dN_s^{1-} \\ \lambda_t^{1-} &= \mu_t^{1-} + \int_0^t \phi_{1-,1+}(t-s)dN_s^{1+} + \int_0^t \phi_{1-,1-}(t-s)dN_s^{1-},\end{aligned}$$

where the $\mu\colon \mathbb{R}_+ \to \in \mathbb{R}_+^2$ is called the ***baseline*** and $\phi\colon \mathbb{R}_+ \to \mathcal{M}_2(\mathbb{R}_+)$ is called the ***kernel***, where we write vectors and matrices in bold and $\mathcal{M}_{n,m}(X)$ (resp. $\mathcal{M}_n(X)$) for the set of X-valued $n \times m$ (resp. $n \times n$) matrices. From a financial perspective, we can easily interpret the different terms above:

- on the one hand, μ_1^+ (resp. μ_1^-) is an exogenous source of upward (resp. downward) price moves;
- on the other hand, ϕ is an endogenous source of price moves. For example, $\phi_{1+,1-}$ increases the intensity of upward price jumps after a downward price jump, creating a mean-reversion effect (while $\phi_{1+,1+}$ creates a trending effect).

To further encode the long-memory property of the order flow, [EEFR18, JR15] consider heavy-tailed kernels where, writing $\rho(M)$ for the spectral radius of a matrix M, for some $c > 0$ and $\alpha \in (1/2, 1)$ we have

$$\rho\Big(\int_t^\infty \phi(s)ds\Big) \underset{t\to\infty}{\sim} ct^{-\alpha}.$$

Such a model satisfies the stability property of Hawkes processes (see for example [JR15]) as long as $\rho(\|\phi\|_1) < 1$ (writing $\|\cdot\|_1$ for the L^1 norm). In fact, calibration of Hawkes processes on financial data suggests that this stability condition is almost violated. To account for this effect, the authors of [EEFR18, JR15] model the market up to time T with a Hawkes process N^T of baseline μ^T and kernel ϕ^T. The microscopic price until time T is then

$$P_t^{T,1} = N_t^{T,1+} - N_t^{T,1-}.$$

In order to obtain macroscopic dynamics, the time horizon must be large, thus the sequence T_n tends towards infinity (from now on, we write T for T_n). As T tends to infinity, ϕ^T almost saturates the stability condition: $\lim_{n\to\infty} \rho(\|\phi^T\|_1) = 1$. A macroscopic limit then requires scaling the processes appropriately to obtain a non-trivial limit. Details on the proper rescaling of the processes are given in the introduction.

Multivariate setting

Having described the asymptotic setting in the mono-asset case, we now model m different assets. The associated counting process is now a $2m$-dimensional process $N^T = (N^{T,1+}, N^{T,1-}, N^{T,2+}, \dots, N^{T,m-})$ and its intensity satisfies

$$\lambda_t^T = \mu^T{}_t + \int_0^t \phi(t-s)^T dN_s^T.$$

The counting process N includes the upward and downward price jumps of m different assets and the microscopic price of Asset i, where $1 \le i \le m$, is simply

$$P_t^{T,i} = N_t^{T,i+} - N_t^{T,i-}.$$

This allows us to capture correlations between assets since, focusing for example on Asset 1, we have

$$\lambda_t^{T,1+} = \mu_t^{T,1+} + \int_0^t \phi_{1+,2+}^T(t-s) dN_s^{T,2+} + \int_0^t \phi_{1-,2+}^T(t-s) dN_s^{T,2+} + \cdots.$$

Therefore $\phi_{1+,2+}^T$ increases the intensity of upward jumps on Asset 1 after an upward jump of Asset 2 while $\phi_{1-,2+}^T$ increases the intensity of downward jumps, etc.

We now need to adapt the nearly-unstable setting to the multidimensional case. Thus we have to find how to saturate the stability condition and to translate the long memory property of the order flow.

In [EEFR18], $\phi^T(t)$ is taken diagonalisable (in a basis independent of T and t) with a maximum eigenvalue $\xi^T(t)$ such that $\lim_{T\to\infty} \|\xi^T\|_1 = 1$. However this structure leads to the same volatility for all assets and thus cannot be a satisfying solution for realistic market dynamics. We take here a sequence of trigonalisable (in a basis O independent of T and t) kernels $\phi^T(t)$ with $n_c > 0$ eigenvalues almost saturating the stability condition. Thus the Hawkes kernel is taken of the form (using block matrix notation in force throughout the chapter)

$$\phi^T(t) = O \begin{pmatrix} A^T(t) & 0 \\ B^T(t) & C^T(t) \end{pmatrix} O^{-1},$$

where $A^T : \mathbb{R}_+ \to \mathcal{M}_{n_c}(\mathbb{R})$, $B^T : \mathbb{R}_+ \to \mathcal{M}_{2m-n_c,n_c}(\mathbb{R})$ and $C^T : \mathbb{R}_+ \to \mathcal{M}_{2m-n_c}(\mathbb{R})$. Note that we will see that in the limit, macroscopic volatilities and prices are independent of the chosen

basis. We assume that the stability condition is saturated at the speed $T^{-\alpha}$ where $\alpha \in (1/2, 1)$ is again related to the tail of the matrix kernel (see below). The saturation condition translates to

$$T^{\alpha}\big(I - \int_0^{\infty} A^T\big) \underset{T\to\infty}{\to} K,$$

where K is an invertible matrix.

We now need to encode the long memory property of the order flow. We can expect orders to be sent jointly on different assets (this can be due, for example, to portfolio rebalancing, risk management or optimal trading) and split under different time scales depending on idiosyncratic components (such as daily traded volume or volatility). Empirically the approximation that despite idiosyncrasies a common time scale for order splitting exists is partially justified: for example [BMEB17] shows that market impact, which is directly related to the order flow, is well-approximated by a single time scale for many stocks. Finally, this property is encoded by imposing a heavy-tail condition for $A := \lim_{T\to\infty} A^T$ with the previous exponent α:

$$\alpha x^{\alpha} \int_x^{\infty} A(s) ds \underset{x\to\infty}{\to} M,$$

with M an invertible matrix.

Main results and organization of the chapter

In the framework described above, we show that the macroscopic limit of prices is a multivariate version of the rough Heston model introduced in [EEGRR20, EEGR19], where the volatility process is a solution of a multivariate rough stochastic Volterra equation. Thus we derive a natural multivariate setting for rough volatility using nearly-unstable Hawkes processes.

More precisely, define the rescaled processes (see [JR15] for details), for $t \in [0, 1]$:

$$X_t^T := \frac{1}{T^{2\alpha}} N_{tT}^T \tag{1}$$

$$Y_t^T := \frac{1}{T^{2\alpha}} \int_0^{tT} \lambda_s ds \tag{2}$$

$$Z_t^T := T^{\alpha}(X_t^T - Y_t^T) = \frac{1}{T^{\alpha}} M_{tT}^T \tag{3}$$

$$P_t^T = \frac{1}{T^{2\alpha}} (N_{tT}^{T,1+} - N_{tT}^{T,1-}, \cdots, N_{tT}^{T,m+} - N_{tT}^{T,m-}). \tag{4}$$

We refer to P^T as the (rescaled) microscopic price process. Under some additional technical and no statistical arbitrage assumptions, there exists an n_c dimensional process $\tilde{V}$, matrices $\Theta^1 \in \mathcal{M}_{n_c}(\mathbb{R}), \Theta^2 \in \mathcal{M}_{n-n_c}(\mathbb{R}), \Lambda_0 \in \mathcal{M}_{n_c}(\mathbb{R}), \Lambda_1 \in \mathcal{M}_{n_c}(\mathbb{R}), \Lambda_2 \in \mathcal{M}_{n_c, n-n_c}(\mathbb{R}), \theta_0 \in \mathbb{R}^{n_c}$ and a Brownian motion B such that

- Any macroscopic limit point P of the sequence P^T satisfies

$$P_t = (I + \Delta)^{\dagger} Q \int_0^t \mathrm{diag}(\sqrt{V_s}) dB_s,$$

where $Q := (e_1 - e_2 \mid \cdots \mid e_{2m-1} - e_{2m})$, writing ${}^{\dagger}Q$ for the transpose of Q, $(e_i)_{1\leq i\leq 2m}$ for the canonical basis of $\mathbb{R}^{2m}$ and $\Delta = (\Delta_{ij})_{1\leq i,j\leq m} \in \mathscr{M}_m(\mathbb{R})$ is defined in Section 2 while V is defined below.

- $\Theta^1 \tilde{V} = (V^1, \cdots, V^{n_c})$ and $\Theta^2 \tilde{V} = (V^{n_c+1}, \cdots, V^n)$.
- $\tilde{V}$ has Hölder regularity $\alpha - 1/2 - \epsilon$ for any $\epsilon > 0$.
- For any t in $[0,1]$, $\tilde{V}$ satisfies

$$\tilde{V}_t = \int_0^t (t-s)^{\alpha-1}(\theta_0 - \Lambda_0 \tilde{V}_s)ds + \int_0^t (t-s)^{\alpha-1}\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1 \tilde{V}_s})dW_s + \int_0^t (t-s)^{\alpha-1}\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2 \tilde{V}_s})dZ_s,$$

where $W := (B^1, \cdots, B^{n_c})$, $Z := (B^{n_c+1}, \cdots, B^n)$ and we write $\sqrt{x}$ for the component-wise square root of vectors of non-negative entries.

Thus the volatility process V is driven by $\tilde{V}$, which represents volatility factors, of which there are as many as there are critical directions.

We can use this result to provide microstructural foundations for some empirical properties of correlation matrices. Informally, considering that our assets have similar self-exciting features in their microscopic dynamics, we show that any macroscopic limit point P of the sequence P^T satisfies P

$$P_t = \Sigma \int_0^t \mathrm{diag}(\sqrt{V_s})dW_s,$$

where W is a Brownian motion, V satisfies a stochastic Volterra equation and Σ has one very large eigenvalue followed by smaller eigenvalues that we can interpret as due to the presence of sectors and a bulk of eigenvalues much smaller than the others. This is typical of actual stock correlation matrices (see for example [LCBP99] for an empirical study).

The chapter is organised as follows. Section 1 rigorously introduces the technical framework sketched in the introduction. We present and discuss the main results in Section 2 which are then applied in examples developed in Section 3. Proofs and technical results are available in an appendix.

1 Assumptions

Before presenting the main results, we make precise the framework sketched out in the introduction. Different examples of Hawkes processes satisfying our assumptions are given in Section 3.

Consider a sequence of measurable functions $\phi^T : \mathbb{R}_+ \to \mathscr{M}_{2m}(\mathbb{R}_+)$ and $\mu^T : \mathbb{R}_+ \to \mathbb{R}_+^{2m}$, where the pair (μ^T, ϕ^T) will be used to model the market dynamics until time T via a Hawkes process N^T of baseline μ^T and kernel ϕ^T. Each kernel ϕ^T is stable ($\rho(\|\phi^T\|_1) < 1$).

Assumption 1. *There exists O an invertible matrix such that each ϕ^T can be written as*

$$\phi^T = O\begin{pmatrix} A^T & 0 \\ B^T & C^T \end{pmatrix} O^{-1},$$

where $A^T : \mathbb{R}_+ \to \mathcal{M}_{n_c}(\mathbb{R})$, $B^T : \mathbb{R}_+ \to \mathcal{M}_{2m-n_c,n_c}(\mathbb{R})$, $C^T : \mathbb{R}_+ \to \mathcal{M}_{2m-n_c}(\mathbb{R})$. Furthermore, the sequence ϕ^T converges towards $\phi : \mathbb{R}_+ \to \mathcal{M}_{2m}(\mathbb{R}_+)$ as T tends to infinity and, writing A, B, C for the limits of A^T, B^T, C^T as T tends to infinity, $\rho(\int_0^\infty C) < 1$.

Additionally, there exists $\alpha \in (1/2, 1)$, K, M invertible matrices and $\mu : [0,1] \to \mathbb{R}_+$ such that

$$T^\alpha (I - \|A^T\|_1) \underset{T\to\infty}{\to} K \tag{5}$$

$$\alpha x^\alpha \int_x^\infty A(s)ds \underset{x\to\infty}{\to} M \tag{6}$$

$$T^{1-\alpha}\mu_{tT}^T \underset{T\to\infty}{\to} \mu_t, \tag{7}$$

where KM^{-1} has strictly positive eigenvalues.

Realistic market dynamics require enforcing no statistical arbitrage conditions on the kernels, as in the spirit of [JR15]. To determine which conditions need to be satisfied to prevent such arbitrage, we write the intensity of the counting process λ^T using the compensator process $M_t^T := N_t^T - \lambda_t^T$ and $\psi^T = \sum_{k\geq 1} (\phi^T)^{*k}$ (see for example Proposition 2.1 in [JR15]). We have

$$\lambda_t^T = \mu^T + \int_0^t \psi^T(t-s)\mu_s^T ds + \int_0^t \psi^T(t-s)dM_s^T. \tag{8}$$

Thus, the expected intensities of upward and downward price jumps of Asset i are

$$\mathbb{E}[\lambda_t^{T,i+}] = \mu_t^{T,i+} + \sum_{1\leq j\leq 2m} \int_0^t \psi_{i+,j-}^T(t-s)\mu_s^{T,j-}ds + \sum_{1\leq j\leq 2m} \int_0^t \psi_{i+,j+}^T(t-s)\mu_s^{T,j+}ds$$

$$\mathbb{E}[\lambda_t^{T,i-}] = \mu_t^{T,i-} + \sum_{1\leq j\leq 2m} \int_0^t \psi_{i-,j-}^T(t-s)\mu_s^{T,j-}ds + \sum_{1\leq j\leq 2m} \int_0^t \psi_{i-,j+}^T(t-s)\mu_s^{T,j+}ds.$$

The above leads us to the following assumption.

Assumption 2. *For any $1 \leq i, j \leq m$:*

1. $\psi_{i+,j+}^T + \psi_{i+,j-}^T = \psi_{i-,j+}^T + \psi_{i-,j-}^T$ *(no pair-trading arbitrage)*
2. $\lim_{T\to\infty} \left(\int_0^\infty \psi_{i+,j}^T - \int_0^\infty \psi_{i+,j+}^T \right) < \infty$ *(suitable asymptotic behaviour of the intensities)*

Under the above conditions and if $\mu^{T,i+} = \mu^{T,i-}$ for all $1 \leq i \leq m$, then $\mathbb{E}[\lambda_t^{T,i+}] = \mathbb{E}[\lambda_t^{T,i-}]$ and there are on average as many upward than downward jumps, which we interpret as a no statistical arbitrage property.

Define, for any $1 \le i, j \le m$,

$$\delta^T_{ij} := \psi^T_{j+,i+} - \psi^T_{j-,i+} \tag{9}$$

$$\Delta_{ij} := \lim_{T\to\infty} \left\| \psi^T_{j+,i+} \right\|_1 - \left\| \psi^T_{j-,i+} \right\|_1 . \tag{10}$$

We can make the following remark.

Remark 1. *Note that for any* $1 \le k \le m$*, defining* $v_k := e_{k+} - e_{k-}$ *and using* (1) *of Assumption 2, we have*

$$\begin{aligned}
{}^{\dagger}\psi^T v_k &= {}^{\dagger}\psi^T (e_{k+} - e_{k-}) \\
&= (\psi^T_{k+,1+} - \psi^T_{k-,1+}) e_{1+} + (\psi^T_{k+,1-} - \psi^T_{k-,1-}) e_{1-} + \cdots + (\psi^T_{m+,1+} - \psi^T_{m-,1+}) e_{m-} \\
&= (\psi^T_{k+,1+} - \psi^T_{k-,1+}) e_{1+} - (\psi^T_{k+,1+} - \psi^T_{k-,1+}) e_{1-} + \cdots + (\psi^T_{m+,1+} - \psi^T_{m-,1+}) e_{m-} \\
&= (\psi^T_{k+,1+} - \psi^T_{k-,1+}) v_1 + \cdots + (\psi^T_{k+,m+} - \psi^T_{k-,m+}) v_m \\
&= \delta^T_{k1} v_1 + \cdots + \delta^T_{kn} v_m .
\end{aligned}$$

A sufficient condition for the no pair-trading arbitrage Equation (1) *of Assumption 2 to hold is that, for all* $1 \le i \le m$*,*

$${}^{\dagger}\phi^T v_i = \sum_{1\le j\le m} ({}^{\dagger}\phi^T v_i \cdot v_j) v_j ,$$

since then we have, for any $1 \le k \le m$*,*

$$\sum_{1\le l\le m} (\psi^T_{k+,l+} - \psi^T_{k-,l+}) e_{l+} - (\psi^T_{k+,l+} - \psi^T_{k-,l+}) e_{l-} = \sum_{1\le l\le m} (\psi^T_{k+,l+} - \psi^T_{k-,l+}) e_{l+} - (\psi^T_{k+,l-} - \psi^T_{k-,l-}) e_{l-} .$$

In our applications in Section 3 we will use this condition as it is easier to check assumptions on ϕ *than on* ψ*.*

2 Main results

We are now in the position to rigorously state the main results of this chapter. We use the processes X^T, Y^T and Z^T defined in the introduction (see Equations (1), (2) (3)) and write

$$O^{-1} = \begin{pmatrix} O^{(-1)}_{11} & O^{(-1)}_{12} \\ O^{(-1)}_{21} & O^{(-1)}_{22} \end{pmatrix}, \quad O = \begin{pmatrix} O_{11} & O_{12} \\ O_{21} & O_{22} \end{pmatrix}.$$

We set

$$\begin{aligned}
\Theta^1 &:= \big(O_{11} + O_{12}(I - \int_0^\infty C)^{-1} \int_0^\infty B\big) K^{-1} \\
\Theta^2 &:= \big(O_{21} + O_{22}(I - \int_0^\infty C)^{-1} \int_0^\infty B\big) K^{-1} \\
\theta_0 &:= \begin{pmatrix} O^{(-1)}_{11} & 0 \\ 0 & O^{(-1)}_{12} \end{pmatrix} \mu \\
\Lambda &:= \frac{\alpha}{\Gamma(1-\alpha)} K M^{-1} .
\end{aligned}$$

We have the following theorem.

Theorem 1. *The sequence (X^T, Y^T, Z^T) is C-tight for the Skorokhod topology. Furthermore, for every limit point (X, Y, Z) of the sequence, there exists a positive process V and an $2m$-dimensional Brownian motion B such that*

1. $X_t = \int_0^t V_s ds$, $Z_t = \int_0^t \mathrm{diag}(\sqrt{V_s}) dB_s$.
2. *There exists $\tilde{V}$ a process of Hölder regularity $\alpha - 1/2 - \varepsilon$ for any $\varepsilon > 0$ such that $\Theta^1 \tilde{V} = (V^1, \cdots, V^{n_c})$, $\Theta^2 \tilde{V} = (V^{n_c+1}, \cdots, V^{2m})$ and $\tilde{V}$ is solution of the following stochastic Volterra equation:*

$$\begin{aligned} \forall t \in [0,1], \tilde{V}_t = &\frac{1}{\Gamma(\alpha)} \Lambda \int_0^t (t-s)^{\alpha-1} (\theta_0 - \tilde{V}_s) ds \\ &+ \frac{1}{\Gamma(\alpha)} \Lambda \int_0^t (t-s)^{\alpha-1} O_{11}^{(-1)} \mathrm{diag}(\sqrt{\Theta^1 \tilde{V}_s}) dW_s^1 \\ &+ \frac{1}{\Gamma(\alpha)} \Lambda \int_0^t (t-s)^{\alpha-1} O_{12}^{(-1)} \mathrm{diag}(\sqrt{\Theta^2 \tilde{V}_s}) dW_s^2, \end{aligned} \tag{11}$$

where $W^1 := (B^1, \cdots, B^{n_c})$, $W^2 := (B^{n_c+1}, \cdots, B^{2m})$, Θ^1, Θ^2, $O_{11}^{(-1)}$, $O_{12}^{(-1)}$, θ_0 do not depend on the chosen basis.

Finally, any limit point P of the rescaled price processes P^T satisfies

$$P_t = (I + \Delta)^\dagger Q \left(\int_0^t \mathrm{diag}(\sqrt{V_s}) dB_s + \int_0^t \mu_s ds \right),$$

where Δ is defined in Equation (10).

Theorem 1 links multivariate nearly unstable Hawkes processes and multivariate rough volatility. We note that:

- The resulting stochastic Volterra equation has non-trivial solutions, as the examples in Section 3 will show.
- From a financial perspective, Theorem 1 shows that the limiting volatility process for a given asset is a sum of different factors. The matrix Δ mixes them and is therefore responsible for correlations between asset prices. Remarks and comments on $I + \Delta$ are developed in Section 3.
- The theorem implies that adding/removing an asset to/from a market has an impact on the individual volatility of other assets. We can estimate the magnitude of such volatility modifications by calibrating Hawkes processes on price changes.
- Since there is a one to one correspondence between the Hurst exponent H and the long memory parameter of the order flow α, our model yields the same roughness for all assets. Extensions to allow for different exponents to coexist, for example by introducing an asset-dependent scaling through $D = (\alpha_1, \cdots, \alpha_m)$ and studying $T^{-D} \lambda_{tT}^T$, are more intricate. In particular, one needs to use a special function extending the Mittag-Leffler matrix function such that its Laplace transform is of the form $(I + \Lambda t^D)^{-1}$.

3 Applications

In this section, we give examples of processes obtained through Theorem 1 under different assumptions on the microscopic parameters. The first example highlights the flexibility of our framework and shows that the obtained limit in Theorem 1 is non-trivial. We then study the influence of microscopic parameters on the limiting price and volatility processes when modeling two assets. Finally, we model many different assets to reproduce realistic high-dimensional correlation matrices.

3.1 An example of non-trivial volatility process obtained through Theorem 1

Before presenting some truly relevant results for finance, we develop an example demonstrating that the solutions to the Volterra equations of the form of Equation (11) are non-trivial. The structure of our Volterra equations is close to those studied in [AJLP19], which proves existence and uniqueness of affine Volterra equations. In particular, this paper covers Volterra equations of the following type, for $\alpha \in (1/2, 1)$:

$$X_t = X_0 + \int_0^t (t-s)^{\alpha-1} b(X_s) ds + \int_0^t (t-s)^{\alpha-1} \sigma(X_s) dB_s,$$

where $b\colon \mathbb{R} \to \mathbb{R}^n$ and $\sigma\colon \mathbb{R} \to \mathscr{M}_n(\mathbb{R})$ are continuous functions. A key condition required for existence and uniqueness is sublinear growth condition on b and σ, that is

$$\|b(x)\|_2 \vee \|\sigma(x)\|_2 \leq c(1 + \|x\|_2), \tag{12}$$

for some constant $c > 0$ where $\|\cdot\|_2$ is the usual Euclidian norm for vectors and matrices. Thus, this setting covers equations of the type

$$X_t = X_0 + \int_0^t (t-s)^{\alpha-1} b(X_s) ds + \int_0^t (t-s)^{\alpha-1} \mathrm{diag}(\sqrt{X_s}) dB_s,$$

which are a particular case of Theorem 1. However, note that Condition 12 fails when $\sigma(x) = \Sigma \mathrm{diag}(\sqrt{x})$ for some non-diagonal matrix Σ. Interestingly, this setting is covered in our approach as illustrated by the following corollary.

Corollary 1. *We can find a microscopic process satisfying the assumptions of Theorem 1 such that V is a non-negative process which satisfies, for any t in [0,1],*

$$V_t = \int_0^t (t-s)^{\alpha-1} (\theta - G V_s) ds + \int_0^t (t-s)^{\alpha-1} \Sigma \mathrm{diag}(\sqrt{V_s}) dB_s,$$

where θ is a 4-dimensional vector, G, Σ are 4×4 non-diagonal matrices and B is a 4-dimensional Brownian motion.

Thus, our framework yields non-trivial solutions and leads to interesting new examples of processes. We now focus on building realistic models to discuss the correspondence between the microscopic parameters of the Hawkes kernel and macroscopic quantities such as correlations and volatility.

3.2 Influence of microscopic properties on the price dynamics of two correlated assets

Our first model to understand the price formation process focuses on two assets. Let $\mu^1, \mu^2 > 0$, $\alpha \in (1/2, 1), \gamma_1, \gamma_2$ in $[0,1]$, $H^c_{21}, H^a_{21}, H^c_{12}, H^a_{12}$ in $[0,1]$ [1] such that (here $\sqrt{\cdot}$ is the principal square root, so that if $x<0$, $\sqrt{x} = i\sqrt{-x}$):

$$
\begin{aligned}
&0 \le (H^c_{12} + H^a_{12})(H^c_{21} + H^a_{21}) < 1 \\
&0 \le |\, 1 - (\gamma_1+\gamma_2) - \sqrt{(H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21}) + (\gamma_1 - \gamma_2)^2}\,| < 1 \\
&0 \le |\, 1 - (\gamma_1+\gamma_2) + \sqrt{(H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21}) + (\gamma_1 - \gamma_2)^2}\,| < 1.
\end{aligned}
$$

We now have to choose a kernel which satisfies the different assumptions of Section 1 to model the interactions between our two assets. Theorem 1 states that the only relevant parameters for the macroscopic price are K and M. For simplicity we choose the kernel such that $M = \alpha I$. This leads us to define, for $t \ge 0$,

$$
\begin{aligned}
\phi_1^T(t) &= (1-\gamma_1)\alpha(1-T^{-\alpha})\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} & \phi_{21}^{T,c}(t) &= \alpha T^{-\alpha} H^c_{21}\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} \\
\phi_2^T(t) &= \gamma_1\alpha(1-T^{-\alpha})\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} & \phi_{21}^{T,a}(t) &= \alpha T^{-\alpha} H^a_{21}\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} \\
\tilde\phi_1^T(t) &= (1-\gamma_2)\alpha(1-T^{-\alpha})\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} & \phi_{12}^{T,c}(t) &= \alpha T^{-\alpha} H^c_{12}\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} \\
\tilde\phi_2^T(t) &= \gamma_2\alpha(1-T^{-\alpha})\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)} & \phi_{12}^{T,a}(t) &= \alpha T^{-\alpha} H^a_{12}\mathbb{1}_{t\ge 1}\, t^{-(\alpha+1)}.
\end{aligned}
$$

For a realistic model, we impose the exogenous source of upward and downward price moves to be equal: $\mu^{1+} = \mu^{1-}$ and $\mu^{2+} = \mu^{2-}$. Thus, the sequence of baselines and kernels are chosen as

$$
\mu^T = T^{\alpha-1}\begin{pmatrix}\mu^1\\ \mu^1\\ \mu^2\\ \mu^2\end{pmatrix}, \quad \phi^T = \begin{pmatrix} \phi_1^T & \phi_2^T & \phi_{12}^{T,c} & \phi_{12}^{T,a} \\ \phi_2^T & \phi_1^T & \phi_{12}^{T,a} & \phi_{12}^{T,c} \\ \phi_{21}^{T,c} & \phi_{21}^{T,a} & \tilde\phi_1^T & \tilde\phi_2^T \\ \phi_{21}^{T,a} & \phi_{21}^{T,c} & \tilde\phi_2^T & \tilde\phi_1^T \end{pmatrix}.
$$

Applying theorem 1 yields the following result.

Corollary 2. *Consider any limit point P of P^T. Under the above assumptions, it satisfies*

$$
P_t = \frac{\sqrt{2}}{4\gamma_1\gamma_2 - (H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21})}\begin{pmatrix} 2\gamma_2 & H^c_{21} - H^a_{21} \\ H^c_{12} - H^a_{12} & 2\gamma_1 \end{pmatrix}\int_0^t \begin{pmatrix} \sqrt{V_s^1}\, dW_s^1 \\ \sqrt{V_s^2}\, dW_s^2 \end{pmatrix}, \tag{13}
$$

with

$$
\begin{aligned}
\begin{pmatrix} V_t^1 \\ V_t^2 \end{pmatrix} = {} & \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)}\int_0^t (t-s)^{\alpha-1}\left(\begin{pmatrix}\mu^1\\ \mu^2\end{pmatrix} - \frac{1}{1-(H^c_{12}+H^a_{12})(H^c_{21}+H^a_{21})}\begin{pmatrix} 1 & H^c_{21}+H^a_{21} \\ H^c_{12}+H^a_{12} & 1\end{pmatrix}\begin{pmatrix} V_s^1 \\ V_s^2\end{pmatrix}\right) ds \\
& + \sqrt{2}\frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)}\int_0^t (t-s)^{\alpha-1}\begin{pmatrix} \sqrt{V_s^1}\, dZ_s^1 \\ \sqrt{V_s^2}\, dZ_s^2\end{pmatrix},
\end{aligned} \tag{14}
$$

[1] The superscripts c (resp. a) stand for continuation (resp. alternation) to describe that after a price move in a given direction, H^c (resp. H^a) encodes the tendency to trigger other price moves in the same (resp. opposite) direction will follow.

where W and Z are bi-dimensional independent Brownian motions. This model helps us understand how microscopic parameters drive the price formation process to generate a macroscopic price and volatility. We begin our remarks with some definitions.

We call *momentum* the trend (i.e., the imbalance between the number of upward and downward jumps) created by jumps of one asset on itself . The opposite effect is referred to as *mean-reversion.* For example, the parameter γ_1 controls the intensity of self-induced bid-ask bounce on Asset 1: when γ_1 close to zero corresponds to a strong momentum while γ_1 close to one corresponds to a strong mean-reversion.

We call *cross-asset momentum* the trend created by jumps of one asset on another. For example, cross-asset momentum from Asset 2 to Asset 1 (resp. Asset 1 to Asset 2) appears via $H^c_{21} - H^a_{21}$ (resp. $H^c_{12} - H^a_{12}$): when both $H^c_{21} - H^a_{21}$ and $H^c_{12} - H^a_{12}$ are nill, the prices of Asset 1 and Asset 2 are uncorrelated. We now turn to comments on the volatility process.

Because of its role in the single-asset case, we refer to V as the *fundamental variance*: for example V^1 is the fundamental variance of Asset 1. The equation satisfied by V only depends on the sum of the feedback effects between each asset through $H^c_{12} + H^a_{12}$: from a volatility viewpoint, upward and downward jumps have the same impact. Furthermore, we can compute the expected fundamental variance using Mittag-Leffler functions (see Section V.A).

Mean-reversion drives down volatility while cross-asset momentum increases it. Indeed, computing $\mathbb{E}[(P^1_t)^2]$ for example we get:

$$\mathbb{E}[(P^1_t)^2] = 2\frac{4\gamma_2^2 \int_0^t \mathbb{E}[V^1_s]ds + (H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21}) \int_0^t \mathbb{E}[V^2_s]ds}{[4\gamma_1\gamma_2 - (H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21})]^2}.$$

In particular, increasing γ_1 or γ_2 does not change V but reduces $\mathbb{E}[(P^1_t)^2]$. This example may be particularly relevant to understand the contribution of Asset 2 to the volatility of Asset 1 through calibration to market data since if Asset 2 were removed from the market, we would have $\mathbb{E}[(P^1_t)^2] = \dfrac{1}{2\gamma_1}$. Focusing now on the price formation process, we see that it results from a combination of momentum, mean-reversion and cross-asset momentum. We illustrate this in two extreme cases: when there is no cross-asset momentum and when cross-asset momentum is strong.

- When there is no cross-asset momentum (i.e. $H^c_{12} = H^a_{12}$ and $H^c_{21} = H^a_{21}$) at the microscopic scale a price move on Asset 2 has the same impact on the intensity of upward and downward price moves of Asset 1. Thus the difference between the expected number of upward and downward jumps does not change after a price move on Asset 2: the expected microscopic price of Asset 1 is unaffected and price moves of Asset 2 generate no trend on Asset 1. This results in macroscopic prices being uncorrelated (see Equation (13)).
- On the other hand, when cross-asset momentum is strong (i.e. $(H^c_{12} - H^a_{12})(H^c_{21} - H^a_{21}) \approx 4\gamma_1\gamma_2$, for example if $H^c_{12} - H^a_{12} = 2\gamma_1\sqrt{1-\epsilon}, H^c_{12} - H^a_{12} = 2\gamma_2\sqrt{1-\epsilon}$ for some small $\epsilon > 0$),

at the microscopic scale, a price move on Asset 2 significantly increases the probability of a future price move of Asset 1 in the same direction (and vice-versa). In this context we have

$$\Delta + I = \frac{1}{2\gamma_1\gamma_2\epsilon}\begin{pmatrix}\gamma_2 & \gamma_2\sqrt{1-\epsilon}\\ \gamma_1\sqrt{1-\epsilon} & \gamma_1\end{pmatrix}.$$

Using Equation (13) we can check that $\frac{\mathbb{E}[P_t^1 P_t^2]}{\sqrt{\mathbb{E}[(P_t^1)^2]\mathbb{E}[(P_t^2)^2]}} \underset{\epsilon\to 0}{\to} 1$ and prices evolve in unison.

This example underlines that in our approach (thanks to our no-arbitrage constraint) microscopic features transfer to macroscopic properties in an intuitive way.

3.3 Reproducing realistic correlation matrices of large number of assets using microscopic properties

It is well-known that the correlation matrix of stocks has few large eigenvalues outside of a "bulk" of eigenvalues attributable to noise (see for example [LCBP99]). The largest eigenvalue is referred to as the market mode (because the associated eigenvector places a roughly equal weight on each asset) and is much larger than other eigenvalues. Other significant eigenvalues can be related to the presence of sectors: groups of companies with similar characteristics.

How can we provide microstructural foundations for this stylised fact? The large eigenvalue associated to the market mode implies that, in a first approximation, stock prices move together: a price increase on one asset is likely followed by a price increase on all other assets. Translating this in our framework, an upward (resp. downward) jump on a given asset increases the probability of an upward (resp. downward) jump on all other assets. We further expect that an upward price move on an asset increases this probability much more on an asset from the same sector than on an unrelated one.

The above remarks lead us to consider a model where:

- All stocks share some fundamental high-frequency properties by having similar self-excitement parameters in the kernel.
- Stocks have a stronger influence on price changes of stocks within the same sector.
- Within the same sector, all stocks have the same microscopic parameters.

The technical details of our setting are presented in Appendix V.C.3 and we only provide here essential elements to understand the framework. Let $\mu^1, \dots, \mu^m > 0$ be the baselines of each asset. Using the same notations as before, take γ in $[0,1]$, α in $(1/2,1)$ and $H^c, H^a > 0$. We consider $R > 0$ different sectors, Sector r having m_r stocks. For a pair of stocks which we dub $1,2$ to make an analogy with the previous example, we have that:

- The self excitement parameters are equal: $\gamma_1 = \gamma_2 = \gamma$ where γ is the same for all stocks.

- If Stock 1 and Stock 2 do not belong to the same sector, $H^c_{21} = H^c_{12} = H^c$, $H^a_{21} = H^a_{12} = H^a$ where H^c, H^a are the same for all stocks.
- If Stock 1 and Stock 2 belong to the same sector r, $H^c_{21} = H^c_{12} = H^c + H^c_r$, $H^a_{21} = H^a_{12} = H^a + H^a_r$ where H^c_r, H^a_r are the same for all stocks belonging to sector r.

The asymptotic framework is built as in the previous example, with the details given in the proof of Corollary 3 in Appendix V.C.3. We write $i_r := m_0 + m_1 + \cdots + m_{r-1}$ for $1 \leq r \leq R$ (with convention $m_0 = 1$) so that stocks from sector r are indexed between i_r and i_{r+1} excluded and define the following vectors

$$\begin{aligned} w &:= \frac{1}{\sqrt{m}}(e_1 + \cdots + e_m) \\ w_r &:= \frac{1}{\sqrt{m_r}} \sum_{i_r \leq i < i_{r+1}} e_i \\ \theta &:= \sum_{1 \leq i \leq m} \mu^i e_i. \end{aligned}$$

We consider an asymptotic framework where the number of assets will eventually grow to infinity. As will become clear in the proof, the only non-trivial regime appears when $H^c, H^a, H^c_r, H^a_r \underset{m\to\infty}{=} \mathcal{O}(m^{-1})$. Thus we assume that $mH^c, mH^a, mH^c_r, mH^a_r$ converge to $\bar{H}^c, \bar{H}^a, \bar{H}^c_r, \bar{H}^a_r$ as m tends to infinity. We also assume that the proportion of stocks in a given sector relative to the total number of stocks does not vanish: for each $1 \leq r \leq R$, $\frac{m_r}{m} \underset{m\to\infty}{\to} \eta_r > 0$. Define the following constants which will appear in the price and volatility processes: $\lambda^+ := \bar{H}^c + \bar{H}^a, \lambda^+_r := \bar{H}^c_r + \bar{H}^c_r$, $\lambda^- := \bar{H}^c - \bar{H}^a, \lambda^-_r := \bar{H}^c_r - \bar{H}^a_r$. Applying Theorem 1 yields the following result.

Corollary 3. *Consider any limit point P of P^T. Under the above assumptions, it satisfies:*

$$P_t = \sqrt{2}\Sigma_\varepsilon \int_0^t \mathrm{diag}(\sqrt{V_s}) dW_s,$$

where W is a Brownian motion, $\Sigma_\varepsilon := (2\gamma I - \lambda^- v^\dagger v - \sum_{1\leq r\leq R} \eta_r \lambda^-_r {v_r}^\dagger v_r + \varepsilon)^{-1}$ with ϵ a deterministic $m \times m$ matrix such that $\rho(\epsilon) \underset{m\to\infty}{=} o(m^{-1})$ and V satisfies the stochastic Volterra equation

$$V_t = \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1}(\theta - \mathcal{V}_\epsilon V_s) ds + \frac{\sqrt{2}\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \mathrm{diag}(\sqrt{V_s}) dZ_s,$$

with Z a Brownian motion independent from W and $\mathcal{V}_\epsilon := \left(I - \lambda^+ v^\dagger v - \sum_{1\leq r\leq R} \eta_r \lambda^+_r {v_r}^\dagger v_r + \epsilon\right)^{-1}$ where ε is a deterministic $m \times m$ matrix such that $\rho(\varepsilon) \underset{m\to\infty}{=} o(m^{-1})$.

Under the previous corollary, writing $\propto$ for equality up to a multiplicative constant, the expected fundamental variance can be written using the cumulative Mittag-Leffler function

$$\mathbb{E}[V_t] \propto F^{\alpha, \mathcal{V}_\epsilon}(t)\theta.$$

Since $\rho(\epsilon) \underset{m\to\infty}{=} o(m^{-1})$, we neglect it in further comments and use the approximation $\mathcal{V}_\epsilon \approx \mathcal{V}_0$. Writing ξ for the largest eigenvalue of $\mathcal{V}_0$ and neglecting other eigenvalues (which is reasonable if $\lambda^+ + \sum_{1\le r\le R}\eta_r\lambda_r^+ \approx 1$) and z for the associated eigenvector, using the definition of the Mittag-Leffler function (see Definition 4 in Appendix V.B.2), we have

$$\mathbb{E}[V_t] \propto F^{\alpha,\xi}(t)({}^\dagger\theta z)z.$$

In the further approximation that $\eta_r\lambda_r^+$ is independent r, we have $z \propto (1,\cdots,1)$ and

$$\begin{aligned}\mathbb{E}[{P_t}^\dagger P_t] &\propto \Sigma_\varepsilon \mathrm{diag}(\mathbb{E}[V_t])^\dagger \Sigma_\varepsilon \\ &\propto \Sigma_\varepsilon \mathrm{diag}(z)^\dagger \Sigma_\varepsilon \\ &\propto {\Sigma_\varepsilon}^\dagger \Sigma_\varepsilon \propto {\Sigma_\varepsilon}^2.\end{aligned}$$

Therefore the eigenvectors of $\mathbb{E}[{P_t}^\dagger P_t]$ are those of Σ_ε. As $\rho(\varepsilon) \underset{m\to\infty}{=} o(m^{-1})$, we neglect it in further comments and use the approximation $\Sigma_\varepsilon \approx \Sigma_0$. When $\lambda^- + \sum_{1\le r\le R}\eta_r\lambda_r^- \approx 2\gamma$, Σ_0 has one large eigenvalue followed by $R-1$ smaller eigenvalues and much smaller eigenvalues. This is consistent with stylised facts of high-dimensional stock correlation matrices and we have thus built a microscopic model to explain the macroscopic structure of correlation matrices.

The conditions $\lambda^- + \sum_{1\le r\le R}\eta_r\lambda_r^- \approx 1$ and $\lambda^+ + \sum_{1\le r\le R}\eta_r\lambda_r^+ \approx 1$ correspond to the parameters being close to the point where all directions are critical: when $\lambda^- + \sum_{1\le r\le R}\eta_r\lambda_r^- \approx 2\gamma$ or $\lambda^- + \sum_{1\le r\le R}\eta_r\lambda_r^- \approx 1$, the spectral radius of $\int_0^\infty C$ is equal to one and we cannot split the kernel into a critical and a non-critical component.

It would be interesting to study other implications of this model. In particular, we believe that encoding a negative price/volatility correlation into the microscopic parameters could explain the so-called index leverage effect (see [RAB11] for a definition and empirical analysis of this stylised fact).

V.A Proof of Theorem 1

We split the proof into four steps. Our approach is inspired by [EEFR18, JR15, JR16]. First, we show that the sequence (X^T, Y^T, Z^T) is C-tight. Second, we use tightness and representation theorems to find equations satisfied by any limit point (X, Y, Z) of (X^T, Y^T, Z^T). Third, properties of the Mittag-Leffler function enable us to prove Equation (11). Fourth and finally, we derive the equation satisfied by any limit point P of P^T.

Preliminary lemmas

We start with lemmas that will be useful in the proofs. Lemma A.1 from [EEFR18] yields

$$\frac{1}{T^\alpha}\lambda^T_{tT} = \frac{\mu^T_{tT}}{T^\alpha} + \frac{1}{T^\alpha}\int_0^{tT} \psi^T(tT-s)\mu^T_s\,ds + \frac{1}{T^\alpha}\int_0^{tT} \psi^T(tT-s)\,dM^T_s. \tag{15}$$

Thus to investigate the limit of $\frac{1}{T^\alpha}\lambda^T_{\cdot T}$ we need to study $\frac{1}{T^\alpha}\psi^T(T\cdot)$, which we will do through its Laplace transform. Given a $L^1(\mathbb{R}_+)$ function f, we write its Laplace transform $\hat{f}(t) := \int_0^\infty f(x)e^{-tx}dx$, for $t \geq 0$ (and similarly for matrix-valued functions $F = (F_{ij})$ where each $F_{ij} \in L^1(\mathbb{R}_+)$). Remark that $\widehat{f^{*k}} = \hat{f}^k$, where $*k$ is the convolution product iterated k times. The following lemma holds.

Lemma 1. *We have the following convergence for any $t \geq 0$:*

$$T^{-\alpha}\widehat{\psi^T(T\cdot)}(t) \underset{T\to\infty}{\to} O\begin{pmatrix} \left[\frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K\right]^{-1} & 0 \\ (I - \int_0^\infty C)^{-1}\int_0^\infty B\left[\frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K\right]^{-1} & 0 \end{pmatrix} O^{-1}, \tag{16}$$

where K and M are defined in Equation (5) *and* (6).

Proof. Define $\varphi^T := O^{-1}\hat{\phi}^T O$. Then

$$\hat{\psi}^T(t) = \sum_{k\geq 1}\hat{\phi}^{T,*k} = O(I - \hat{\varphi}^T)^{-1}\hat{\varphi}^T O^{-1}.$$

We can use the shape of φ^T and matrix block inversion to rewrite this expression. Doing so, we find

$$\hat{\psi}^T(t) = O\begin{pmatrix} (I - \hat{A}^T(t))^{-1}\hat{A}^T(t) & 0 \\ (I - \hat{C}^T(t))^{-1}\hat{B}^T(t)(I - \hat{A}^T(t))^{-1}\hat{A}^T(t) - (I - \hat{C}^T(t))^{-1}\hat{B}^T(t) & (I - \hat{C}^T(t))^{-1}\hat{C}^T(t) \end{pmatrix} O^{-1}.$$

To derive the limiting process, we use Equations (5) and (6). Using integration by parts and a Tauberian theorem as in [EEFR18, JR16], we have

$$\int_0^\infty A^T - \hat{A}^T(t/T) \underset{T\to\infty}{=} \frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M T^{-\alpha} + o(T^{-\alpha})$$
$$I - \int_0^\infty A^T \underset{T\to\infty}{=} KT^{-\alpha} + o(T^{-\alpha}).$$

Therefore

$$T(I-\hat{A}^T(t/T)) = T(\int_0^\infty A^T - \hat{A}^T(t/T)) + T(I-\int_0^\infty A^T)$$
$$\underset{T\to\infty}{=} \left[\frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K\right] T^{1-\alpha} + o(T^{1-\alpha}).$$

Consequently

$$T^{\alpha-1}T(I-\hat{A}^T(t/T)) \underset{T\to\infty}{=} \frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K + o(1).$$

By Assumption 1 M is invertible and KM^{-1} has strictly positive eigenvalues. Thus $Mt+K = (KM^{-1}+tI)M$ is invertible for any $t \geq 0$. The Laplace transform of $T^{-\alpha}\psi^T(T\cdot)$ being $T^{1-\alpha}\widehat{\psi}^T(\cdot/T)$, we have proved for any $t \geq 0$,

$$\widehat{T^{-\alpha}\psi^T(T\cdot)}(t) \underset{T\to\infty}{\to} O\begin{pmatrix} \left[\frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K\right]^{-1} & 0 \\ (I-\int_0^\infty C)^{-1}\int_0^\infty B\left[\frac{\Gamma(1-\alpha)}{\alpha}t^\alpha M + K\right]^{-1} & 0 \end{pmatrix} O^{-1}.$$

□

We show in the technical appendix that the inverse Laplace transform of $\Lambda(t^\alpha I + \Lambda)^{-1}$, where $\Lambda \in \mathcal{M}_n(\mathbb{R})$ has positive eigenvalues, is a simple extension of the Mittag-Leffler density function to matrices (see Definition 4 in the appendix) denoted by $f^{\alpha,\Lambda}$. Thus we define for any $t \in [0,1]$

$$f(t) := O\begin{pmatrix} K^{-1}f^{\alpha,\frac{\alpha}{\Gamma(1-\alpha)}KM^{-1}} & 0 \\ (I-\int_0^\infty C)^{-1}\int_0^\infty BK^{-1}f^{\alpha,\frac{\alpha}{\Gamma(1-\alpha)}KM^{-1}} & 0 \end{pmatrix} O^{-1}. \tag{17}$$

The following lemma shows the weak convergence of ψ^T towards f.

Lemma 2. *For any bounded measurable function g and $1 \leq i,j \leq n$*

$$\int_{[0,1]} g(x)T^{-\alpha}\psi_{ij}^T(Tx)dx \underset{T\to\infty}{\to} \int_{[0,1]} g(x)f_{ij}(x)dx.$$

Proof. First note that when $\|f_{ij}\|_1 = 0$ (which implies $f_{ij} = 0$), using Equation (16) with $t = 0$ we have

$$\left\|T^{1-\alpha}\psi_{ij}^T\right\|_1 \underset{T\to\infty}{\to} \|f_{ij}\|_1 = 0,$$

which implies, since $1-\alpha \geq 0$,

$$\left\|\psi_{ij}^T\right\|_1 \underset{T\to\infty}{\to} 0.$$

Therefore, as $\psi_{ij}^T \geq 0$, for any bounded measurable function g

$$\left|\int_{[0,1]} g(x)T^{-\alpha}\psi_{ij}^T(Tx)dx\right| \leq c\int_{[0,1]} T^{-\alpha}\psi_{ij}^T(Tx)dx \leq c\left\|T^{1-\alpha}\psi_{ij}^T\right\|_1,$$

and the result holds. Assume now that $\|f_{ij}\|_1 > 0$. It will be convenient for us to proceed with random variables, so define

$$\rho_{ij}^T := \frac{T^{-\alpha}\psi_{ij}^T(T\cdot)}{\left\|T^{1-\alpha}\psi_{ij}^T\right\|_1}.$$

We can view ρ_{ij}^T as the density of a random variable taking values in $[0,1]$, say S. Lemma 1 gives the convergence of the characteristic functions of S towards

$$\hat{\rho}_{ij} := \frac{\hat{f}_{ij}}{\|f_{ij}\|_1}.$$

Since ρ_{ij} is continuous (as ψ_{ij}^T is continuous), Levy's continuity theorem guarantees that ρ_{ij}^T converges weakly towards ρ_{ij}. Therefore for any bounded measurable function g

$$\int_{[0,1]} g(x)\rho_{ij}^T(x)dx \underset{T\to\infty}{\to} \int_{[0,1]} g(x)\rho_{ij}(x)dx$$
$$\int_{[0,1]} g(x)\frac{T^{-\alpha}\psi_{ij}^T(Tx)}{\left\|T^{1-\alpha}\psi_{ij}^T\right\|_1}dx \underset{T\to\infty}{\to} \int_{[0,1]} g(x)\frac{f_{ij}(x)}{\|f_{ij}\|_1}dx.$$

Equation (16) implies $\left\|T^{1-\alpha}\psi_{ij}^T\right\|_1 \underset{T\to\infty}{\to} \|f_{ij}\|_1$, so that together with the above we have

$$\int_{[0,1]} g(x)T^{-\alpha}\psi_{ij}^T(Tx)dx \underset{T\to\infty}{\to} \int_{[0,1]} g(x)f_{ij}(x)dx.$$

□

We introduce the cumulative functions

$$F^T(t) = \int_0^t T^{-\alpha}\psi^T(Ts)ds$$
$$F(t) = \int_0^t f(s)ds.$$

We have just shown in particular that F^T converges pointwise towards F and therefore, by Dini's theorem, converges uniformly towards F.

V.A.1 Step 1: C-tightness of (X^T, Y^T, Z^T)

Recall the definition of the rescaled processes:

$$X_t^T := \frac{1}{T^{2\alpha}}N_{tT}^T$$
$$Y_t^T := \frac{1}{T^{2\alpha}}\int_0^{tT}\lambda_s ds$$
$$Z_t^T := T^\alpha(X_t^T - Y_t^T) = \frac{1}{T^\alpha}M_{tT}^T.$$

As in [EEFR18] and [JR16] we show that the limiting processes of X^T and Y^T are the same and that the limiting process of Z^T is the quadratic variation of the limiting process of X^T. We have the following proposition:

Proposition 1 (C-tightness of (X^T, Y^T, Z^T)). *The sequence (X^T, Y^T, Z^T) is C-tight and if (X, Z) is a possible limit point of (X^T, Z^T), then Z is a continuous martingale with $[Z, Z] = \mathrm{diag}(X)$. Furthermore, we have the convergence in probability*

$$\sup_{t\in[0,1]} \left\| Y_t^T - X_t^T \right\|_2 \xrightarrow[T\to\infty]{\mathbb{P}} 0.$$

Proof. The proof is esentially the same as in [EEFR18], adapting for our structure of Hawkes processes. We have

$$\lambda_t^T = \mu_t^T + \int_0^t \psi^T(t-s)\mu_s^T ds + \int_0^t \psi^T(t-s) dM_s^T,$$

and therefore

$$\begin{aligned}\mathbb{E}[N_T^T] &= \mathbb{E}[\int_0^T \lambda_s^T ds] \\ &= \int_0^T \mu_t^T dt + \int_0^T \int_0^t \psi^T(t-s)\mu_s^T ds dt \le cT^{2\alpha} \left\|\mu\right\|_\infty,\end{aligned}$$

where we used the convergence of $T^{1-\alpha}\mu_{T\cdot}^T$ (see Equation (7)) together with the weak convergence of $T^{-\alpha}\psi^T(T\cdot)$ (see Lemma 2). It follows then that

$$\mathbb{E}[X_1^T] = \mathbb{E}[Y_1^T] \le c,$$

and since the processes are increasing, X^T and Y^T are tight. As the maximum jump size of X^T and Y^T tends to 0, we have the C-tightness of (X^T, Y^T). Since N^T is the quadratic variation of M^T, $(M^{T,i})^2 - N^{T,i}$ is an L^2 martingale starting at 0 and Doob's inequality yields

$$\begin{aligned}\sum_{1\le i\le n} \mathbb{E}[\sup_{t\in[0,1]} (X_t^{T,i} - Y_t^{T,i})^2] &\le 4 \sum_{1\le i\le n} \mathbb{E}[(X_1^{T,i} - Y_1^{T,i})^2] \\ &\le 4T^{-4\alpha} \sum_{1\le i\le n} \mathbb{E}[(M_T^{T,i})^2] \\ &\le 4T^{-4\alpha} \sum_{1\le i\le n} \mathbb{E}[N_T^{T,i}] \\ &\le cT^{-2\alpha}.\end{aligned}$$

Using the same approach as in [EEFR18] we conclude that Z is a continuous martingale and $[Z, Z]$ is the limit of $[Z^T, Z^T]$.

□

V.A.2 Step 2: Rewriting of limit points of (X^T, Y^T, Z^T)

By Proposition 1, for any limit point (X, Y) of (X^T, Y^T), we have $X = Y$ almost surely. We use Y^T to derive an equation for $Y = X$. As $Y^T = \frac{1}{T^{2\alpha}} \int_0^{tT} \lambda_s^T ds$, we first study λ_{sT}^T. Using Equation (15) we get

$$\begin{aligned}\int_0^t \lambda_s^T ds &= \int_0^t \mu_s^T ds + \int_0^t \int_0^u \psi^T(s-u)\mu_u^T du ds + \int_0^t \psi^T(t-s) M_s^T ds \\ &= \int_0^t \mu_s^T ds + \int_0^t \psi^T(t-s) \int_0^s \mu_u^T du ds + \int_0^t \psi^T(t-s) M_s^T ds.\end{aligned}$$

A change variables of leads to

$$\begin{aligned}\int_0^{tT} \lambda_s^T ds &= \int_0^{tT} \mu_s^T ds + \int_0^{tT} \psi^T(tT-s) \int_0^s \mu_u^T du ds + \int_0^{tT} \psi^T(tT-s) M_s^T ds \\ &= \int_0^{tT} \mu_s^T ds + T \int_0^t \psi^T(tT-sT) \int_0^{sT} \mu_u^T du ds + \int_0^t \psi^T(tT-sT) M_{sT}^T T ds \\ &= T \int_0^t \mu_{sT}^T ds + T \int_0^t \psi^T(T(t-s)) \int_0^{sT} \mu_u^T du ds + T \int_0^t \psi^T(T(t-s)) M_{sT}^T ds.\end{aligned}$$

Therefore

$$T^{2\alpha} Y_t^T = T \int_0^t \mu_{sT}^T ds + T \int_0^t \psi^T(T(t-s)) \int_0^{sT} \mu_u^T du ds + T \int_0^t \psi^T(T(t-s)) M_{sT}^T ds \tag{18}$$

$$=: T^{2\alpha}(Y_t^{T,1} + Y_t^{T,2} + Y_t^{T,3}), \tag{19}$$

with obvious notations. Thus, to obtain our limit we use the convergence properties of F^T which we derived previously. We have the following proposition.

Proposition 2. *Consider (X, Z) a limit point of (X^T, Z^T). Then,*

$$X_t = \int_0^t F(t-s)\mu_s ds + \int_0^t F(t-s) dZ_s.$$

Proof. Let (X, Y, Z) be a limit point of (X^T, Y^T, Z^T). First, since $T^{1-\alpha}\mu_{tT}^T \underset{T\to\infty}{\to} \mu_t$ (see Equation (7)), $Y_t^{T,1}$ converges to 0 as T tends to infinity. Moving on to $Y^{T,2}$, by integration by parts we have

$$\begin{aligned}Y_t^{T,2} &= \int_0^t T^{1-\alpha} \psi^T(T(t-s)) T^{-\alpha} \int_0^{sT} \mu_u^T du ds \\ &= \left[F^T(t-s) T^{-\alpha} \int_0^{sT} \mu^T{}_u du \right]_0^t + \int_0^t F^T(t-s) T^{1-\alpha} \mu^T{}_{sT} ds \\ &= \int_0^t F^T(t-s) T^{1-\alpha} \mu^T{}_{sT} ds.\end{aligned}$$

Using Equation (7) again together with the uniform convergence of F^T (see Lemma 2) we have the convergence

$$Y_t^{T,2} \underset{T\to\infty}{\to} \int_0^t F(t-s)\mu_s ds.$$

Finally, $Y_t^{T,3}$ can be written as

$$\begin{aligned} Y_t^{T,3} &= T^{1-2\alpha}\int_0^t \psi^T(T(t-s))M_{sT}^T ds = \int_0^t F^T(t-s)dZ_s^T \\ &= \int_0^t F(t-s)dZ_s + \int_0^t F(t-s)(dZ_s^T - dZ_s) + \int_0^t (F^T(t-s)-F(t-s))dZ_s^T. \end{aligned}$$

The Skorokhod representation theorem applied to (Z^T, Z) yields the existence of copies in law $(\tilde{Z}^T, \tilde{Z})$, $\tilde{Z}^T$ converging almost surely to $\tilde{Z}$. We proceed with $(\tilde{Z}^T, \tilde{Z})$ and keep previous notations. The stochastic Fubini theorem [Ver12] gives, almost surely

$$\int_0^t F(t-s)(dZ_s^T - dZ_s) = \int_0^t f(s)(Z_{t-s}^T - Z_{t-s})ds.$$

From the dominated convergence theorem we obtain the almost sure convergence

$$\int_0^t f(s)(Z_{t-s}^T - Z_{t-s})ds \underset{T\to\infty}{\to} 0.$$

Furthermore, since $[Z^T, Z^T] = \mathrm{diag}(X^T)$ we have

$$\sum_{1\le i\le n}\mathbb{E}\left[\left(\int_0^t (F^T(t-s)-F(t-s))dZ_s^T\right)_i^2\right] \le \sum_{1\le i,j\le n}\int_0^t (F_{ij}^T(t-s)-F_{ij}(t-s))^2 T^{1-\alpha}\mathbb{E}[\lambda_{sT}^{T,j}]ds.$$

Using Equation (15) together with Lemma 1 we can bound $\mathbb{E}[\lambda_{sT}^{T,j}]$ independently of T and

$$\sum_{1\le i\le n}\mathbb{E}\left[\left(\int_0^t (F^T(t-s)-F(t-s))dZ_s^T\right)_i^2\right] \le c\sum_{1\le i,j\le n}\int_0^t (F_{ij}^T(t-s)-F_{ij}(t-s))^2 ds.$$

The right hand side converges to 0 by the dominated convergence theorem together with the uniform convergence of F^T towards F (see Lemma 2). From Proposition 1 we know that $Y = X$ almost surely. Putting everything together, almost surely,

$$X_t = \int_0^t F(t-s)\mu_s ds + \int_0^t F(t-s)dZ_s.$$

This is valid for any limit point (X, Z) of (X^T, Z^T), which concludes the proof. □

The previous proposition gives suitable martingale properties of limit points of Z^T to apply the martingale representation theorem, which is the topic of the following proposition.

Proposition 3. *Let (X,Z) be a limit point of (X^T,Z^T). There exists, up to an extension of the original probability space, an n-dimensional Brownian motion B and a non-negative process V such that*

$$X_t = \int_0^t V_s ds$$
$$Z_t = \int_0^t \mathrm{diag}(\sqrt{V_s}) dB_s$$
$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s)\mathrm{diag}(\sqrt{V_s}) dB_s.$$

Proof. This proof relies on the martingale representation theorem applied to Z. Consider (X,Z) a limit point of (X^T,Z^T). Following the proof of Theorem 3.2 in [JR16], X can be written as the integral of a process V

$$X_t = \int_0^t V_s ds,$$

with V satisfying the equation

$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s) dZ_s.$$

Therefore, as $[Z,Z]_t = \mathrm{diag}(X_t) = \mathrm{diag}(\int_0^t V_s ds)$ and Z is a continuous martingale, by the martingale representation theorem (see for example Theorem 3.9 from [RY13]), there exists (up to an enlargement of the probability space) a multivariate Brownian motion B and a predictable square integrable process H such that

$$Z_t = \int_0^t H_s dB_s.$$

Furthermore, note that as V is a non-negative process as X is a non-decreasing process and we have

$$Z_t = \int_0^t \mathrm{diag}(\sqrt{V_s})\mathrm{diag}(\sqrt{V_s})^{-1} H_s dB_s.$$

A simple computation shows that, since $[Z,Z]_t = \int_0^t H_s{}^\dagger H_s ds = X_t = \int_0^t V_s ds$, the process $\tilde{B}_t := \int_0^t \mathrm{diag}(\sqrt{V_s})^{-1} H_s dB_s$ is a Brownian motion. Finally,

$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s)\mathrm{diag}(\sqrt{V_s}) d\tilde{B}_s.$$

□

A straightforward application of Lemma 4.4 and Lemma 4.5 in [JR16] yields the following lemma.

Lemma 3. *Consider a (weak) non-negative solution V of the stochastic Volterra equation*

$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s)\mathrm{diag}(\sqrt{V_s}) dB_s,$$

where B is a Brownian motion. Then every component of V has pathwise Hölder regularity $\alpha - 1/2 - \epsilon$ for any $\epsilon > 0$.

V.A.3 Step 3: proof of Equation (11)

Properties of the Mittag-Leffler function (as in [EEFR18]) enable us to rewrite the previous stochastic differential equation using power-law kernels, which is the subject of the next proposition. Let $\Theta^1 := (O_{11}+O_{12}(I-\int_0^\infty C)^{-1}\int_0^\infty B)K^{-1}$, $\Theta^2 := (O_{21}+O_{22}(I-\int_0^\infty C)^{-1}\int_0^\infty B)K^{-1}$ and $\Lambda := \dfrac{\alpha}{\Gamma(1-\alpha)}KM^{-1}$.

Proposition 4. *Given an m-dimensional Brownian motion B, a non-negative process V is solution of the following stochastic differential equation*

$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s)\mathrm{diag}(\sqrt{V_s})dB_s,$$

if and only if there exists a process $\tilde{V}$ of Hölder regularity $\alpha-1/2-\epsilon$ for any $\epsilon>0$ such that $\Theta^1\tilde{V}_t = (V^1,\cdots,V^{n_c})$ and $\Theta^2\tilde{V}_t = (V^{n_c+1},\cdots,V^{2m})$ are non-negative processes and $\tilde{V}$ is solution of the following stochastic Volterra equation

$$\begin{aligned}\tilde{V}_t &= \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}(O_{11}^{(-1)}\mu^1 + O_{12}^{(-1)}\mu^2 - \tilde{V}_s)ds \\ &+ \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{11}^{(-1)}\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW_s^1 + \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{12}^{(-1)}\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW_s^2,\end{aligned}$$

where $W^1 := (B^1,\cdots,B^{n_c})$ and $W^2 := (B^{n_c+1},\cdots,B^{2m})$.

Proof. We begin by showing the first implication. Starting from Proposition 3 we have

$$V_t = \int_0^t f(t-s)\mu_s ds + \int_0^t f(t-s)\mathrm{diag}(\sqrt{V_s})dB_s.$$

Developing from the definition of f in Equation (17), for any $t\in[0,1]$, f can be written

$$f(t) = \begin{pmatrix}(O_{11}+O_{12}(I-\int_0^\infty C)^{-1}\int_0^\infty B)K^{-1}f^{\alpha,\Lambda}(t) & 0\\ (O_{21}+O_{22}(I-\int_0^\infty C)^{-1}\int_0^\infty B)K^{-1}f^{\alpha,\Lambda}(t) & 0\end{pmatrix}\begin{pmatrix}O_{11}^{(-1)} & O_{12}^{(-1)}\\ O_{21}^{(-1)} & O_{22}^{(-1)}\end{pmatrix}.$$

Defining $V^1 := (V^1,\cdots,V^{n_c})$ and $V^2 := (V^{n_c+1},\cdots,V^{2m})$, we have

$$\begin{aligned}V_t^1 &= \Theta^1\int_0^t f^{\alpha,\Lambda}(t-s)O_{11}^{(-1)}\mu_s^1 ds + \Theta^1\int_0^t f^{\alpha,\Lambda}(t-s)O_{12}^{(-1)}\mu_s^2 ds \\ &+ \Theta^1\int_0^t f^{\alpha,\Lambda}(t-s)O_{11}^{(-1)}\mathrm{diag}(\sqrt{V_s^1})dW_s^1 + \Theta^1\int_0^t f^{\alpha,\Lambda}(t-s)O_{12}^{(-1)}\mathrm{diag}(\sqrt{V_s^2})dW_s^2.\end{aligned}$$

If Θ^1 were non-singular, we could express V^1 with power-law kernels thanks to the same approach as in [EEFR18]. In general we define

$$\begin{aligned}\tilde{V}_t &:= \int_0^t f^{\alpha,\Lambda}(t-s)(O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2)ds \\ &+ \int_0^t f^{\alpha,\Lambda}(t-s)O_{11}^{(-1)}\mathrm{diag}(\sqrt{V_s^1})dW_s^1 + \int_0^t f^{\alpha,\Lambda}(t-s)O_{12}^{(-1)}\mathrm{diag}(\sqrt{V_s^2})dW_s^2.\end{aligned}$$

From the same arguments as in Lemma 3, Hölder regularity of V carries to $\tilde{V}$, and the components of $\tilde{V}$ are of Hölder regularity $\alpha - 1/2 - \epsilon$ for any $\epsilon > 0$, hence Lemma 3 shows $\mathcal{K} := I^{1-\alpha}\tilde{V}$ is well-defined, where $I^{1-\alpha}$ is the fractional integration operator of order $1-\alpha$ (see Definition 1 in Appendix V.B.2). Note that for any t in $[0,1]$, using Lemma 4 of Appendix V.B.2, we have

$$\begin{aligned}
\mathcal{K}_t &= \int_0^t \Lambda(I - F^{\alpha,\Lambda}(t-s))(O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2)ds \\
&\quad + \int_0^t \Lambda(I - F^{\alpha,\Lambda}(t-s))O_{11}^{(-1)}\mathrm{diag}(\sqrt{V_s^1})dW_s^1 + \int_0^t \Lambda(I - F^{\alpha,\Lambda}(t-s))O_{12}^{(-1)}\mathrm{diag}(\sqrt{V_s^2})dW_s^2 \\
&= \Lambda\int_0^t (O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2)ds + \int_0^t \Lambda O_{11}\mathrm{diag}(\sqrt{V_s^1})dW_s^1 + \int_0^t \Lambda O_{12}^{(-1)}\mathrm{diag}(\sqrt{V_s^2})dW_s^2 \\
&\quad - \Lambda\int_0^t \left[F^{\alpha,\Lambda}(t-s)O_{11}^{(-1)}\mu_s^1 + \int_0^s f^{\alpha,\Lambda}(s-u)O_{11}^{(-1)}\mathrm{diag}(\sqrt{V_u^1})dW_u^1\right]ds \\
&\quad - \Lambda\int_0^t \left[F^{\alpha,\Lambda}(t-s)O_{12}^{(-1)}\mu_s^2 + \int_0^s f^{\alpha,\Lambda}(s-u)O_{12}^{(-1)}\mathrm{diag}(\sqrt{V_u^2})dW_u^2\right]ds.
\end{aligned}$$

The last two terms can be rewritten using the definition of $\tilde{V}$, so that

$$\mathcal{K}_t = \Lambda\int_0^t (O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2 - \tilde{V}_s)ds + \Lambda\int_0^t O_{11}^{(-1)}\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW_s^1 + \Lambda\int_0^t O_{12}^{(-1)}\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW_s^2.$$

Thanks to the Hölder regularity of $\tilde{V}$, we can now apply the fractional differentiation operator of order $1-\alpha$ (see Definition 1 in Appendix V.B.2) together with the stochastic Fubini Theorem to deduce

$$\begin{aligned}
\tilde{V}_t &= \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}(O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2 - \tilde{V}_s)ds \\
&\quad + \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{11}^{(-1)}\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW_s^1 + \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{12}^{(-1)}\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW_s^2.
\end{aligned}$$

This concludes the proof of the first implication. We now show the second implication. Suppose there exists $\tilde{V}$ of Hölder regularity $\alpha - 1/2 - \epsilon$ for any $\epsilon > 0$ such that $\Theta^1\tilde{V}$ and $\Theta^2\tilde{V}$ are positive, solution of the following stochastic Volterra equation:

$$\begin{aligned}
\tilde{V}_t &= \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}(O_{11}^{(-1)}\mu_s^1 + O_{12}^{(-1)}\mu_s^2 - \tilde{V}_s)ds \\
&\quad + \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{11}^{(-1)}\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW_s^1 + \frac{1}{\Gamma(\alpha)}\Lambda\int_0^t (t-s)^{\alpha-1}O_{12}^{(-1)}\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW_s^2.
\end{aligned}$$

Let us write for this proof $\theta := \Lambda O_{11}^{(-1)}\mu^1 + \Lambda O_{12}^{(-1)}\mu^2, \Lambda_1 := \Lambda O_{11}^{(-1)}, \Lambda_2 := \Lambda O_{12}^{(-1)}$ so that, for any t in $[0,1]$,

$$\begin{aligned}
\tilde{V}_t &= \frac{1}{\Gamma(\alpha)}\int_0^t (t-s)^{\alpha-1}(\theta_s - \Lambda\tilde{V}_s)ds \\
&\quad + \frac{1}{\Gamma(\alpha)}\int_0^t (t-s)^{\alpha-1}\Lambda_1\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW_s^1 + \frac{1}{\Gamma(\alpha)}\int_0^t (t-s)^{\alpha-1}\Lambda_2\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW_s^2.
\end{aligned}$$

Remark that the above can be written

$$\tilde{V}_t = I^\alpha(\theta - \Lambda\tilde{V})_t + I^\alpha_{B^1}(\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}}))_t + I^\alpha_{B^2}(\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}}))_t,$$

where I^α_B is the fractional integration operator with respect to B (see Definition 2 in Appendix V.B.2). Iterating the application of I^α we find that, for any $N \geq 1$, $\tilde{V}$ satisfies

$$\tilde{V} = \sum_{1\leq k\leq N} \Lambda^{k-1}(-1)^{k-1} I^{(k-1)\alpha}[I^\alpha\theta + I^\alpha_{B^1}(\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}})) + I^\alpha_{B^2}(\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}}))] + \Lambda^N(-1)^N I^{(N+1)\alpha}\tilde{V}.$$

Now, note that θ, $\mathrm{diag}(\sqrt{\Theta^1\tilde{V}})$, $\mathrm{diag}(\sqrt{\Theta^2\tilde{V}})$ and $\tilde{V}$ are square-integrable processes and Lemma 8 in Appendix V.B.2 shows that the sum converges almost surely to the series while $\Lambda^N(-1)^N I^{(N+1)\alpha}\tilde{V}$ converges almost surely to zero as N tends to infinity. Thus we have

$$\begin{aligned}
\tilde{V} &= \sum_{k\geq 0} \Lambda^k(-1)^k I^{k\alpha}[I^\alpha\theta + I^\alpha_{B^1}(\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}})) + I^\alpha_{B^2}(\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}}))] \\
&= \sum_{k\geq 0} \Lambda^k(-1)^k I^{k\alpha} I^\alpha\theta + \sum_{k\geq 0} \Lambda^k(-1)^k I^{k\alpha} I^\alpha_{B^1}(\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}})) + I^\alpha_{B^2}(\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}}))] \\
&= \Lambda^{-1}\sum_{k\geq 0} \Lambda^{k+1}(-1)^k I^{(k+1)\alpha}\theta + \sum_{k\geq 0} \Lambda^k(-1)^k I^{k\alpha} I^\alpha_{B^1}(\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}})) + I^\alpha_{B^2}(\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}}))].
\end{aligned}$$

Lemmas 5 and 7 shown in Appendix V.B.2 enable us to rewrite the above using the matrix Mittag-Leffler function. This yields, for any t in $[0,1]$ and almost surely,

$$\tilde{V}_t = \Lambda^{-1}\int_0^t f^{\alpha,\Lambda}(t-s)\theta_s ds + \Lambda^{-1}\int_0^t f^{\alpha,\Lambda}(t-s)\Lambda_1 \mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dW^1_s + \Lambda^{-1}\int_0^t f^{\alpha,\Lambda}(t-s)\Lambda_2 \mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dW^2_s.$$

Replacing $\theta, \Lambda_1, \Lambda_2$ by their expressions, almost surely and for any t in $[0,1]$,

$$\begin{aligned}
\tilde{V}_t = &\int_0^t f^{\alpha,\Lambda}(t-s)(O^{(-1)}_{11}\mu^1_s + \Lambda O^{(-1)}_{12}\mu^2_s)ds \\
&+ \int_0^t f^{\alpha,\Lambda}(t-s)O^{(-1)}_{11}\mathrm{diag}(\sqrt{\Theta^1\tilde{V}_s})dB_s{}^1 + \int_0^t f^{\alpha,\Lambda}(t-s)O^{(-1)}_{12}\mathrm{diag}(\sqrt{\Theta^2\tilde{V}_s})dB_s{}^2.
\end{aligned}$$

This concludes the second implication and the proof. □

V.A.4 Step 4: Equation satisfied by the limiting price process

The previous results on the convergence of the intensity process enable us to now turn to the question of the limiting price dynamics. Recall that the sequence of rescaled price processes P^T is defined as

$$P^T := {}^\dagger Q X^T,$$

where $Q = \begin{pmatrix} e_1 - e_2 & | & \cdots & | & e_{2m-1} - e_{2m} \end{pmatrix}$. We have the following result.

Proposition 5. *Let (X, Z) be a limit point of (X^T, Z^T) and $P = {}^\dagger Q X$. Then*

$$P_t = (I + \Delta)^\dagger Q(Z_t + \int_0^t \mu_s ds).$$

where $\Delta = (\int_0^\infty \delta^T_{ij})_{1\leq i,j\leq m}$.

Proof. Let (X, Z) be a limit point from (X^T, Z^T). For any $1 \leq i \leq m$ we can compute the difference between upward and downard jumps on Asset i

$$v_i \cdot N^T{}_t = v_i \cdot M_t^T + v_i \cdot \int_0^t \lambda_s ds,$$

with the following expression for the integrated intensity:

$$\int_0^{tT} \lambda^T{}_s ds = T \int_0^t \mu_{sT}^T ds + T \int_0^t \int_0^{T(t-s)} \psi^T(u) du \mu_{Ts}^T ds + \left\|\psi^T\right\|_1 M_{tT}^T - \int_0^{tT} \int_{tT-s}^{\infty} \psi^T(u) du dM_s^T.$$

Thus the microscopic price for the Asset i satisfies

$$\begin{aligned}
T^{-\alpha} v_i \cdot N^T{}_{tT} &= T^{1-\alpha} \int_0^t v_i \cdot \mu_{sT}^T ds + T^{1-\alpha\,\dagger} \left\|\psi^T\right\|_1 v_i \cdot \int_0^t \mu_{Ts}^T ds + v_i \cdot Z_t^T + {}^{\dagger} \left\|\psi^T\right\|_1 v_i \cdot Z_t^T \\
&\quad - T^{-\alpha} \int_0^t \int_{T(t-s)}^{\infty} {}^{\dagger}\psi^T(u) v_i \cdot \mu_{Ts}^T du ds - T^{-\alpha} \int_0^{tT} \int_{tT-s}^{\infty} \psi^T(u) du dM_s^T \\
&= \sum_{1 \leq k \leq m} (\mathbb{1}_{ik} + \int_0^{\infty} \delta_{ik}^T), v_k \cdot Z_t^T + \sum_{1 \leq k \leq m} (\mathbb{1}_{ik} + \int_0^{\infty} \delta_{ik}^T) T^{1-\alpha} \int_0^t v_k \cdot \mu_{sT}^T ds \\
&\quad - \int_0^t \int_{tT-s}^{\infty} {}^{\dagger}\psi^T(u) v_i du \cdot dZ_s^T - T^{-\alpha} \int_0^t \int_{T(t-s)}^{\infty} {}^{\dagger}\psi^T(u) v_i \cdot \mu_{Ts}^T du ds.
\end{aligned}$$

It is straightforward to show that the last two terms converge to zero and thus, any limit point P of $P^T = {}^{\dagger}QX^T$ is such that

$$P_t = (I + \Delta)^{\dagger} Q (Z_t + \int_0^t \mu_s ds).$$

□

Replacing Z by the expression obtained in Proposition 3 concludes the proof of Theorem 1 since

$$P_t = (I + \Delta)^{\dagger} Q \big(\int_0^t \operatorname{diag}(\sqrt{V_s}) dB_s + \int_0^t \mu_s ds \big).$$

V.B Technical results

V.B.1 Independence of Equation (11) from chosen basis

We consider two representations which satisfy Assumption 1. Let $P, \tilde{P}$ be invertible matrices, $0 \leq n_c, n_{c'} \leq n$ and $A^T \in \mathscr{F}(\mathscr{M}_{n_c}(\mathbb{R}))$, $C^T \in \mathscr{F}(\mathscr{M}_{n-n_c}(\mathbb{R}))$, $B^T \in \mathscr{F}(\mathscr{M}_{n-n_c,n_c}(\mathbb{R}))$ and $\tilde{A}^T \in \mathscr{F}(\mathscr{M}_{n_{c'}}(\mathbb{R}))$, $\tilde{C}^T \in \mathscr{F}(\mathscr{M}_{n-n_{c'}}(\mathbb{R}))$, $\tilde{B}^T \in \mathscr{F}(\mathscr{M}_{n-n_{c'},n_{c'}}(\mathbb{R}))$ such that

$$\phi^T = P \begin{pmatrix} A^T & 0 \\ B^T & C^T \end{pmatrix} P^{-1} = \tilde{P} \begin{pmatrix} \tilde{A}^T & 0 \\ \tilde{B}^T & \tilde{C}^T \end{pmatrix} \tilde{P}^{-1}.$$

We write A for the limit of A^T (and similarly for B^T, C^T, etc.). First, remark that we must have $n_c = n_{c'}$. Indeed, since $\rho(\int_0^{\infty} C) < 1$ and $\rho(\int_0^{\infty} \tilde{C}) < 1$, 1 is neither an eigenvalue of $\int_0^{\infty} C$

nor of $\int_0^\infty \tilde{C}$. Yet, since $A = I$ and $\tilde{A} = I$, 1 is an eigenvalue of ϕ with multiplicity n_c and $n_{c'}$. Therefore $n_c = n_{c'}$.

We have, writing $L = P^{-1}\tilde{P}$,

$$\begin{pmatrix} A & 0 \\ B & C \end{pmatrix} = L \begin{pmatrix} \tilde{A} & 0 \\ \tilde{B} & \tilde{C} \end{pmatrix} L^{-1}.$$

Since $A = \tilde{A} = I$ because of Equation (5), developing and using the assumption that $I - C$ is invertible, we get

$$\begin{aligned} L_{12} &= 0 \\ (I-C)L_{21} &= BL_{11} - L_{22}\tilde{B} \\ CL_{22} &= L_{22}\tilde{C}. \end{aligned}$$

Since $LL^{-1} = I$, $L_{11} = I$, $L_{22} = I$, $L_{21} = -L_{21}^{(-1)}$, we deduce

$$L_{11} = I, \quad L_{22} = I, \quad L_{12} = 0, \quad (I-C)L_{21} = B - \tilde{B}, \quad C = \tilde{C}.$$

As $L = P^{-1}\tilde{P}$, we have

$$P^{-1} = \begin{pmatrix} I & 0 \\ (I-C)^{-1}(B-\tilde{B}) & I \end{pmatrix} \tilde{P}^{-1} = \begin{pmatrix} \tilde{P}_{11}^{(-1)} & \tilde{P}_{12}^{(-1)} \\ (I-C)^{-1}(B-\tilde{B})\tilde{P}_{11}^{(-1)} + \tilde{P}_{21}^{(-1)} & (I-C)^{-1}(B-\tilde{B})\tilde{P}_{12}^{(-1)} + \tilde{P}_{22}^{(-1)} \end{pmatrix}.$$

Developing $\tilde{P} = PL$ together with the above, we find

$$\begin{aligned} &\tilde{P}_{11}^{(-1)} = P_{11}^{(-1)}, \tilde{P}_{12}^{(-1)} = P_{12}^{(-1)}, \tilde{P}_{12} = P_{12}, \tilde{P}_{22} = P_{22} \\ &\tilde{P}_{11} = P_{11} + P_{12}(I-C)^{-1}(B-\tilde{B}) \\ &\tilde{P}_{21} = P_{21} + P_{22}(I-C)^{-1}(B-\tilde{B}). \end{aligned}$$

Thus

$$\begin{aligned} &\tilde{P}_{11}^{(-1)} = P_{11}^{(-1)}, \tilde{P}_{12}^{(-1)} = P_{12}^{(-1)} \\ &\tilde{P}_{11} + \tilde{P}_{12}(I-C)^{-1}\tilde{B} = P_{11} + P_{12}(I-C)^{-1}B \\ &\tilde{P}_{21} + \tilde{P}_{22}(I-C)^{-1}\tilde{B} = P_{21} + P_{22}(I-C)^{-1}B. \end{aligned}$$

Therefore regardless of the chosen basis, Equation (11) is the same, which concludes the proof.

V.B.2 Fractional operators

This section is a brief reminder on fractional operators which are used in proofs. We also introduce the matrix extended Mittag-Leffler function.

Definition 1 (Fractional differentiation and integration operators). *For $\alpha \in (0,1)$, the fractional differentiation (resp. integration) operator denoted by D^α is defined as*

$$D^\alpha f(t) := \frac{1}{\Gamma(1-\alpha)} \frac{d}{dt} \int_0^t (t-s)^{-\alpha} f(s) ds,$$

where f is a measurable, Hölder continuous function of order strictly greater than α. The fractional integration operator denoted by I^α is defined as

$$I^\alpha f(t) := \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} f(s) ds.$$

where f is a measurable function.

It will be convenient for us to define fractional integration with respect to a Brownian motion.

Definition 2 (Fractional integration operator with respect to a Brownian motion). *Given a Brownian motion B and $\alpha \in (1/2,1)$, the fractional integration operator with respect to B, denoted by I_B^α, is defined as*

$$I_B^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{1-\alpha} f(s) dB_s.$$

for f a measurable, square integrable stochastic process.

Remark 2. *The fractional integration of a matrix-valued stochastic process f with respect to a multivariate Brownian motion B is:*

$$I_B^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{1-\alpha} f(s) dB_s.$$

We now extend the Mittag-Leffler function to matrices (for a theory of matrix-valued functions, see for example [Hig08]). We have the following definition.

Definition 3 (Matrix-extended Mittag-Leffler function). *Let $\alpha, \beta \in \mathbb{C}$ such that $\mathrm{Re}(\alpha), \mathrm{Re}(\beta) > 0$, $\Lambda \in \mathscr{M}_n(\mathbb{R})$. Then the matrix Mittag-Leffler function is defined as*

$$E_{\alpha,\beta}(\Lambda) := \sum_{n \geq 0} \frac{\Lambda^n}{\Gamma(\alpha n + \beta)}.$$

We also extend the Mittag-Leffler density function for matrices.

Definition 4 (Mittag-Leffler density for matrices). *Let $\alpha \in \mathbb{C}$ such that $\mathrm{Re}(\alpha) > 0$, $\Lambda \in \mathscr{M}_n(\mathbb{R})$. Then, the matrix Mittag-Leffler density function $f^{\alpha,\Lambda}$ is defined as*

$$f^{\alpha,\Lambda}(t) := \Lambda t^{\alpha-1} E_{\alpha,\alpha}(-\Lambda t^\alpha)$$

We write $F^{\alpha,\Lambda}$ for the cumulative matrix Mittag-Leffler density function

$$F^{\alpha,\Lambda}(t) := \int_0^t f^{\alpha,\Lambda}(s) ds$$

Using Definition 3, it is easy to show the following lemma.

Lemma 4. *Let* $\alpha \in \mathbb{C}$ *such that* $\mathrm{Re}(\alpha) > 0$, $\Lambda \in \mathscr{M}_n(\mathbb{R})$. *Then,*

$$I^{1-\alpha} f^{\alpha,\Lambda} = \Lambda(I - F^{\alpha,\Lambda}).$$

Furthermore, if $\alpha \in (1/2, 1)$

$$\widehat{f^{\alpha,\Lambda}}(z) = \Lambda(Iz^\alpha + \Lambda)^{-1}.$$

We need another important property relating Mittag-Leffler functions with fractional integration operators.

Lemma 5. *Let* $\alpha > 0$ *and* $\Lambda \in \mathscr{M}_m(\mathbb{R})$. *Then*

$$I^1 f^{\alpha,\Lambda} = \sum_{n\geq 1} (-1)^{n-1} \Lambda^n I^{n\alpha}(1)$$

Proof. Using Lemma 4 and repeated applications of I^α, for all $N \geq 1$ we have

$$If^{\alpha,\Lambda} = \sum_{1\leq n\leq N} (-1)^{n-1} \Lambda^n I^{n\alpha}(1) + (-1)^{N-1} \Lambda^N I^{N\alpha} If^{\alpha,\Lambda}.$$

Therefore, if we show that

$$(-1)^{N-1} \Lambda^N I^{N\alpha} If^{\alpha,\Lambda} \underset{N\to\infty}{\to} 0,$$

the result will follow. To prove this we make use of the series expansion of $I^{N\alpha} f^{\alpha,\Lambda}$ to deduce bounds which will converge to zero. Writing C a constant independent of t and N which may change from line to line, $N_\alpha = \lfloor \frac{1}{\alpha} \rfloor$ and $\|\cdot\|_{\mathrm{op}}$ for the operator norm, we have

$$\begin{aligned}
\left\|\Lambda^N f^{\alpha,\Lambda}(t)\right\|_{\mathrm{op}} &= \left\| \Lambda^{N+1} \sum_{n\geq 0} (-1)^n \frac{t^{(n+1)\alpha-1}}{\Gamma((n+1)\alpha)} \right\|_{\mathrm{op}} \\
&\leq \left\| \Lambda^{N+1} \sum_{0\leq n\leq N_\alpha} (-1)^n \frac{t^{(n+1)\alpha-1}}{\Gamma((n+1)\alpha)} + \Lambda^{N+1} C \right\|_{\mathrm{op}}.
\end{aligned}$$

Therefore, when applying the fractional integration operator of order $N\alpha$ we have, writing $g_n : t \mapsto t^{(n+1)\alpha-1}$

$$\begin{aligned}
I^{N\alpha} \left\|\Lambda^N f^{\alpha,\Lambda}(t)\right\|_{\mathrm{op}} &\leq \left\| \Lambda^{N+1} I^{N\alpha} \Big(\sum_{0\leq n\leq N_\alpha} (-1)^n \frac{g_n}{\Gamma((n+1)\alpha)} \Big) + \Lambda^{N+1} I^{N\alpha}(C) \right\|_{\mathrm{op}} \\
&\leq \sum_{0\leq n\leq N_\alpha} \frac{1}{\Gamma((n+1)\alpha)} \left\|\Lambda^{N+1} I^{N\alpha}(g_n)\right\|_{\mathrm{op}} + \left\|\Lambda^{N+1} I^{N\alpha}(C)\right\|_{\mathrm{op}}.
\end{aligned}$$

An explicit computation of $I^{N\alpha}(g_n)$ shows the convergence to zero of the right hand side as N tends to infinity, which concludes the proof. □

Finally, we need to combine fractional integration I^α with I_B^α. We have the following lemma.

Lemma 6. *Let $m \geq 1$, B an m-dimensional Brownian motion, X a $m \times m$ matrix valued adapted square-integrable stochastic process and $\alpha, \beta > 0$. Then we have:*

$$I^{\alpha} I_B^{\beta}(X) = I_B^{\alpha+\beta}(X).$$

Proof. The proof is a straightforward application of the definition of the operators together with the stochastic Fubini theorem. □

The next lemma is useful to transform stochastic convolutions of stochastic processes with the Mittag-Leffler density function into series of repeated applications of I_B^{α}.

Lemma 7. *Let $m \geq 1$, B an m-dimensional Brownian motion, X a $m \times m$ matrix valued adapted and square-integrable stochastic process, $\alpha > 0$ and $\Lambda \in \mathcal{M}_m(\mathbb{R})$. Then, for all $t \geq 0$ and almost surely*

$$\int_0^t f^{\alpha,\Lambda}(t-s) X_s dB_s = \sum_{n \geq 1} (-1)^{n-1} \Lambda^n I_B^{n\alpha}(X),$$

where the series converges almost surely.

Proof. Using Lemma 5, we can write the integral using a series of fractional integration operators and apply the stochastic Fubini theorem (as X is square-integrable) to obtain

$$\begin{aligned}
\int_0^t f^{\alpha,\Lambda}(t-s) X_s dB_s &= \int_0^t \sum_{n \geq 1} (-1)^{n-1} \Lambda^n I^{n\alpha-1}(1)_{t-s} X_s dB_s \\
&= \sum_{n \geq 1} \int_0^t (-1)^{n-1} \Lambda^n I^{n\alpha-1}(1)_{t-s} X_s dB_s \\
&= \sum_{n \geq 1} (-1)^{n-1} \Lambda^n \int_0^t I^{n\alpha-1}(1)_{t-s} X_s dB_s \\
&= \sum_{n \geq 1} \frac{(-1)^{n-1}}{\Gamma(n\alpha-1)} \Lambda^n \int_0^t \int_0^{t-s} (t-s-\tau)^{n\alpha-2} d\tau X_s dB_s.
\end{aligned}$$

After a change of variables and using the stochastic Fubini theorem (see for example [Ver12]), we deduce the simpler expression

$$\int_0^t f^{\alpha,\Lambda}(t-s) X_s dB_s = \sum_{n \geq 1} \frac{(-1)^{n-1}}{\Gamma(n\alpha-1)} \Lambda^n \int_0^t (t-\tau)^{n\alpha-2} \int_0^{\tau} X_s dB_s d\tau.$$

Integrating by parts, we finally obtain the result:

$$\begin{aligned}
\int_0^t f^{\alpha,\Lambda}(t-s) X_s dB_s &= \sum_{n \geq 1} \frac{(-1)^{n-1}}{\Gamma(n\alpha-1)(n\alpha-1)} \Lambda^n \int_0^t (t-\tau)^{n\alpha-1} X_\tau dB_\tau, \\
&= \sum_{n \geq 1} \frac{(-1)^{n-1}}{\Gamma(n\alpha)} \Lambda^n \int_0^t (t-\tau)^{n\alpha-1} X_\tau dB_\tau, \\
&= \sum_{n \geq 1} (-1)^{n-1} \Lambda^n I_B^{n\alpha}(X).
\end{aligned}$$

□

The last lemma gives convergence for terms of a series of repeated iterations of I^α.

Lemma 8. *Let $\alpha > 0$, $\Lambda \in \mathcal{M}_m(\mathbb{R})$, B an m-dimensional Brownian motion and X a m-dimensional vector valued square-integrable stochastic process. Then, almost surely and for all $t \in [0,1]$*

$$(-1)^{N-1}\Lambda^N I^{N\alpha}(X)_t \underset{N\to\infty}{\to} 0$$
$$\sum_{n\geq N}(-1)^{n-1}\Lambda^n I_B^{n\alpha}(\mathrm{diag}(X))_t \underset{N\to\infty}{\to} 0.$$

Proof. Let $N^* > N_\alpha := \lfloor \frac{1}{\alpha} \rfloor$. Since X is square-integrable, we have

$$\mathbb{E}\Big[\Big\|\sum_{N>N_*}\Lambda^N I_B^{(N+1)\alpha}(\mathrm{diag}(X))_t\Big\|^2\Big] \leq \sum_{N_1,N_2>N_*}\mathbb{E}[^\dagger(\Lambda^{N_1} I_B^{(N_1+1)\alpha}(\mathrm{diag}(X))_t)(\Lambda^{N_2} I_B^{(N_2+1)\alpha}(\mathrm{diag}(X))_t)].$$

Using the Cauchy-Schwartz inequality and writing $\|\cdot\|_{\mathrm{op}}$ for the operator norm associated to the Euclidian norm, we find

$$\begin{aligned}
\mathbb{E}\Big[\Big\|\sum_{N>N_*}\Lambda^N I_B^{(N+1)\alpha}(\mathrm{diag}(X))_t\Big\|^2\Big] &\leq \sum_{N_1,N_2>N_*}\|\Lambda\|_{\mathrm{op}}^{N_1+N_2}\sum_{1\leq k,l\leq m}\mathbb{E}[I_{B^k}^{(N_1+1)\alpha}(X^k)_t I_{B^l}^{(N_2+1)\alpha}(X^l)_t] \\
&\leq \sum_{N_1,N_2>N_*}\|\Lambda\|_{\mathrm{op}}^{N_1+N_2}\frac{1}{\Gamma((N_1+1)\alpha)\Gamma((N_2+1)\alpha)}\sum_{1\leq i\leq m}\int_0^t (t-s)^{(N_1+N_2)\alpha-2}\mathbb{E}[(X_s^i)^2]ds \\
&\leq c\sum_{N_1,N_2>N_*}\frac{\|\Lambda\|_{\mathrm{op}}^{N_1+N_2}}{\Gamma((N_1+1)\alpha)\Gamma((N_2+1)\alpha)} \\
&\leq c\Big(\sum_{N>N_*}\frac{\|\Lambda\|_{\mathrm{op}}^N}{\Gamma((N+1)\alpha)}\Big)^2.
\end{aligned}$$

Thus by comparison of functions (for example by application of Stirling's formula), for all $\epsilon > 0$,

$$\sum_{N>N_\alpha}\mathbb{P}\Big(\Big\|\sum_{N>N_*}\Lambda^N I_B^{(N+1)\alpha}(\mathrm{diag}(X))_t\Big\| > \epsilon \leq \frac{1}{\epsilon^2}\sum_{N_*\geq N_\alpha}\mathbb{E}\Big[\Big\|\sum_{N>N_*}\Lambda^N I_B^{(N+1)\alpha}(\mathrm{diag}(X))_t\Big\|^2\Big] < \infty.$$

The Borel-Cantelli lemma yields the almost sure convergence to zero of $\Lambda^N I_B^{(N+1)\alpha}(\mathrm{diag}(X))$ as $N\to\infty$. The same approach yields the almost sure convergence to zero of $(-1)^{N-1}\Lambda^N I^{N\alpha}(X)$ as $N\to\infty$. □

V.C Proof of Corollaries

V.C.1 Proof of Corollary 1

Take $\mu^1, \mu^2 > 0$, $\alpha \in (1/2, 1)$, $\kappa \in [0,1]$, $H_{21}^b, H_{21}^s, H_{12}^b, H_{12}^s \in [0,1]$ such that (here $\sqrt{\cdot}$ is the principal square root, so that if $x < 0$, $\sqrt{x} = i\sqrt{-x}$):

$$0 \leq (H_{12}^b + H_{12}^s)(H_{21}^b + H_{21}^s) < 1$$
$$0 \leq |\kappa - \sqrt{(H_{12}^b - H_{12}^s)(H_{21}^b - H_{21}^s)}| < 1$$
$$0 \leq |\kappa + \sqrt{(H_{12}^b - H_{12}^s)(H_{21}^b - H_{21}^s)}| < 1.$$

Define now the following functions, for $t \geq 0$, which will appear in the structure of the kernel:

$$\begin{aligned}
\phi_1^T(t) &:= \alpha(1+\kappa/2)\mathbb{1}_{t\geq 1} t^{-(\alpha+1)} & \phi_3^{b,T}(t) &= \alpha T^{-\alpha} H_{21}^b \mathbb{1}_{t\geq 1} t^{-(\alpha+1)} \\
\phi_2^T(t) &:= \alpha(1+\kappa/2)\mathbb{1}_{t\geq 1} t^{-(\alpha+1)} & \phi_3^{s,T}(t) &= \alpha T^{-\alpha} H_{21}^s \mathbb{1}_{t\geq 1} t^{-(\alpha+1)} \\
\lambda^T(t) &:= \alpha(\kappa-\kappa T^{-\alpha})\mathbb{1}_{t\geq 1} t^{-(\alpha+1)} & \phi_4^{b,T}(t) &:= \alpha T^{-\alpha} H_{12}^b \mathbb{1}_{t\geq 1} t^{-(\alpha+1)} \\
\tilde{\lambda}^T(t) &:= \alpha(\kappa-\kappa T^{-\alpha})\mathbb{1}_{t\geq 1} t^{-(\alpha+1)} & \phi_4^{s,T}(t) &= \alpha T^{-\alpha} H_{12}^s \mathbb{1}_{t\geq 1} t^{-(\alpha+1)}.
\end{aligned}$$

The sequence of baselines and kernels are chosen as:

$$\mu^T = T^{\alpha-1}\begin{pmatrix}\mu^1\\ \mu^1\\ \mu^2\\ \mu^2\end{pmatrix}, \quad \phi^T = \begin{pmatrix} \phi_1^T & \phi_1^T-\lambda^T & \phi_3^{T,b} & \phi_3^{T,s} \\ \phi_2^T-\lambda^T & \phi_2^T & \phi_3^{T,s} & \phi_3^{T,b} \\ \phi_4^{T,b} & \phi_4^{T,s} & \phi_1^T & \phi_1^T-\tilde{\lambda}^T \\ \phi_4^{T,s} & \phi_4^{T,b} & \phi_2^T-\tilde{\lambda}^T & \phi_2^T \end{pmatrix}.$$

The above sequence naturally satisfies the different assumptions outlined in Section 1. Indeed, using the following change of basis

$$O = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix},$$

we have, with notations from Section 1,

$$\begin{aligned}
A &= \begin{pmatrix} \phi_1+\phi_2-\lambda & \phi_3^b+\phi_3^s \\ \phi_4^b+\phi_4^s & \phi_1+\phi_2-\tilde{\lambda} \end{pmatrix} \\
B &= (\phi_1-\phi_2)I \\
C &= \begin{pmatrix} \lambda & \phi_3^b-\phi_3^s \\ \phi_4^b-\phi_4^s & \tilde{\lambda} \end{pmatrix} \\
M &= \alpha I \\
K &= \begin{pmatrix} \kappa & H_{21}^b+H_{21}^s \\ H_{12}^b+H_{12}^s & \kappa \end{pmatrix}.
\end{aligned}$$

Furthermore, we can check that the assumptions of Section 1 are satisfied if

$$\begin{aligned}
&0 \leq H_{21}H_{12} < 1 \\
&0 \leq |\, \kappa - \sqrt{(H_{12}^b-H_{12}^s)(H_{21}^b-H_{21}^s)}\,| < 1 \\
&0 \leq |\, \kappa + \sqrt{(H_{12}^b-H_{12}^s)(H_{21}^b-H_{21}^s)}\,| < 1.
\end{aligned}$$

Under those conditions, we can apply Theorem 1 and compute the relevant quantities which appear in the limiting stochastic differential equation of volatility. We note for convenience

$$\begin{pmatrix} x & y \\ z & w \end{pmatrix} := \left(I - \int_0^\infty C\right)^{-1} \int_0^\infty B.$$

Then, straightforward linear algebra yields

$$O_{11} + O_{12}\Big(I - \int_0^\infty C\Big)^{-1} \int_0^\infty B = \begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix}$$
$$\Big(O_{11} + O_{12}\Big(I - \int_0^\infty C\Big)^{-1} \int_0^\infty B\Big) O_{11}^{(-1)} = \frac{1}{2}\begin{pmatrix} 1+x & 1+x \\ 1-x & 1-x \end{pmatrix}$$
$$\Big(O_{11} + O_{12}\Big(I - \int_0^\infty C\Big)^{-1} \int_0^\infty B\Big) O_{12}^{(-1)} = \frac{1}{2}\begin{pmatrix} y & y \\ -y & -y \end{pmatrix},$$

so that, using the notations of Theorem 1 for the Brownian motion B, W^1 and W^2, we have

$$\Big(O_{11} + O_{12}\Big(I - \int_0^\infty C\Big)^{-1} \int_0^\infty B\Big) O_{11}^{(-1)} \int_0^t \mathrm{diag}\Big(\sqrt{\Theta^1 \tilde{V}_s}\Big) dW_s^1 = \frac{1}{2}\int_0^t \begin{pmatrix} (1+x)\big(\sqrt{V_t^1} dB_s^1 + \sqrt{V_t^2} dB_s^2\big) \\ (1-x)\big(\sqrt{V_t^1} dB_s^1 + \sqrt{V_t^2} dB_s^2\big) \end{pmatrix}$$
$$\Big(O_{11} + O_{12}\Big(I - \int_0^\infty C\Big)^{-1} \int_0^\infty B\Big) O_{12}^{(-1)} \mathrm{diag}\Big(\sqrt{\Theta^2 \tilde{V}_s}\Big) dW_t^2 = \frac{1}{2}\int_0^t \begin{pmatrix} y\big(\sqrt{V_t^3} dB_t^3 + \sqrt{V_t^4} dB_t^4\big) \\ -y\big(\sqrt{V_t^3} dB_t^3 + \sqrt{V_t^4} dB_t^4\big) \end{pmatrix}.$$

Therefore, writing

$$\Sigma_1 := \frac{1}{2}\begin{pmatrix} 1+x & 1+x & y & y \\ 1-x & 1-x & -y & -y \end{pmatrix},$$

we have the following equation for the fundamental variance of Asset 1

$$\frac{\Gamma(1-\alpha)\Gamma(\alpha)}{\alpha}\begin{pmatrix} V_t^1 \\ V_t^2 \end{pmatrix} = \int_0^t (t-s)^{\alpha-1}\Big[\begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix}\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - \begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix} K^{-1} \begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix}^{-1} \begin{pmatrix} V_s^1 \\ V_s^2 \end{pmatrix}\Big] ds$$
$$+ \int_0^t (t-s)^{\alpha-1} \Sigma_1 \mathrm{diag}(\sqrt{V_s}) dB_s.$$

By symmetry, we can find the analogue to the above on the second asset. Using the following notations

$$\Sigma := \frac{\alpha}{\Gamma(1-\alpha)\Gamma(\alpha)}\frac{1}{2}\begin{pmatrix} 1+x & 1+x & y & y \\ 1-x & 1-x & -y & -y \\ z & z & 1+w & 1+w \\ -z & -z & 1-w & 1-w \end{pmatrix}, \quad D := \frac{\alpha}{\Gamma(1-\alpha)\Gamma(\alpha)}\begin{pmatrix} 1+x & y \\ 1-x & -y \\ 1+w & z \\ 1-w & -z \end{pmatrix},$$
$$G := \frac{\alpha}{\Gamma(1-\alpha)\Gamma(\alpha)}\begin{pmatrix} \begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix} K^{-1} \begin{pmatrix} 1+x & y \\ 1-x & -y \end{pmatrix}^{-1} & 0 \\ 0 & \begin{pmatrix} z & 1+w \\ -z & 1-w \end{pmatrix} K^{-1} \begin{pmatrix} z & 1+w \\ -z & 1-w \end{pmatrix}^{-1} \end{pmatrix},$$

where we have written for convenience

$$\begin{pmatrix} x & y \\ z & w \end{pmatrix} := \frac{\int_0^\infty \phi_1 - \int_0^\infty \phi_2}{\int_0^\infty \lambda \int_0^\infty \tilde{\lambda} - (\int_0^\infty \phi_4^b - \int_0^\infty \phi_4^s)(\int_0^\infty \phi_3^b - \int_0^\infty \phi_3^s)} \begin{pmatrix} \int_0^\infty \tilde{\lambda} & -(\int_0^\infty \phi_3^b - \int_0^\infty \phi_3^s) \\ -(\int_0^\infty \phi_4^b - \int_0^\infty \phi_4^s) & \int_0^\infty \lambda \end{pmatrix}.$$

Therefore V satisfies the following stochastic Volterra equation

$$\frac{\Gamma(1-\alpha)\Gamma(\alpha)}{\alpha}V_t = \int_0^t (t-s)^{\alpha-1}\left[D\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix} - GV_s\right]ds + \int_0^t (t-s)^{\alpha-1}\Sigma\mathrm{diag}(\sqrt{V_s})dB_s.$$

This concludes the proof of Corollary 1.

V.C.2 Proof of Corollary 2

We split the proof into two steps. First, we show that the structure of the kernel satisfies the assumptions of Section 1. Then we compute the equations satisfied by variance and prices.

Checking for the assumptions of Theorem 1

We write

$$O_1 := \begin{pmatrix}1\\1\\0\\0\end{pmatrix} O_2 := \begin{pmatrix}0\\0\\1\\1\end{pmatrix} O_3 := \begin{pmatrix}1\\-1\\0\\0\end{pmatrix} O_4 := \begin{pmatrix}0\\0\\1\\-1\end{pmatrix}.$$

Then, setting $O := \begin{pmatrix}O_1 \mid O_2 \mid O_3 \mid O_4\end{pmatrix}$, we have

$$\phi^T = O\begin{pmatrix} \phi_1^T + \phi_2^T & \phi_{12}^{T,c} + \phi_{12}^{T,a} & 0 & 0\\ \phi_{21}^b + \phi_{21}^s & \tilde{\phi}_1^T + \tilde{\phi}_2^T & 0 & 0\\ 0 & 0 & \phi_1^T - \phi_2^T & \phi_{12}^{T,c} - \phi_{12}^{T,a}\\ 0 & 0 & \phi_{21}^b - \phi_{21}^s & \tilde{\phi}_1^T - \tilde{\phi}_2^T \end{pmatrix} O^{-1}.$$

It is straightforward to check that the assumptions are satisfied if

$$\begin{aligned} &0 \le (H_{12}^c + H_{12}^a)(H_{21}^c + H_{21}^a) < 1\\ &0 \le |\, 1 - (\gamma_1+\gamma_2) - \sqrt{(H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a) + (\gamma_1-\gamma_2)^2}\,| < 1\\ &0 \le |\, 1 - (\gamma_1+\gamma_2) + \sqrt{(H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a) + (\gamma_1-\gamma_2)^2}\,| < 1. \end{aligned}$$

Under those conditions $K = I - H$ has positive eigenvalues and therefore $KM^{-1} = \frac{1}{\alpha}K$ has positive eigenvalues. Therefore all the assumptions of Theorem 1 are satisfied.

Limiting variance process

Since we can apply Theorem 1, we now compute the relevant quantities. As $B = 0$, writing $H^{12} := H_{12}^a + H_{12}^c$ and $H^{21} := H_{21}^a + H_{21}^c$, we have

$$O^{-1} = \frac{1}{2}\begin{pmatrix}1 & 1 & 0 & 0\\ 0 & 0 & 1 & 1\\ 1 & -1 & 0 & 0\\ 0 & 0 & 1 & -1\end{pmatrix} \quad K^{-1} = \frac{1}{1-H_{12}H_{21}}\begin{pmatrix}1 & H_{12}\\ H_{21} & 1\end{pmatrix}$$

$$\Theta^1 = \frac{1}{1-H_{12}H_{21}}\begin{pmatrix}1 & H_{12}\\ 1 & H_{12}\end{pmatrix} \quad \Theta^2 = \frac{1}{1-H_{12}H_{21}}\begin{pmatrix}H_{21} & 1\\ H_{21} & 1\end{pmatrix}.$$

One can check that the equations satisfied by $\Theta^1 \tilde{V}$ and $\Theta^2 \tilde{V}$ are, where B is a Brownian motion,

$$\begin{aligned}
\Theta^1 \tilde{V}_t &= \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \left[\begin{pmatrix} \mu_1 \\ \mu_1 \end{pmatrix} - \begin{pmatrix} \tilde{V}_s^1 \\ \tilde{V}_s^1 \end{pmatrix} \right] ds \\
&\quad + \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \sqrt{\tilde{V}_s^1 + H_{12} \tilde{V}_s^2} \begin{pmatrix} dB_s^1 + dB_s^2 \\ dB_s^1 + dB_s^2 \end{pmatrix} \\
\Theta^2 \tilde{V}_t &= \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \left[\begin{pmatrix} \mu_2 \\ \mu_2 \end{pmatrix} - \begin{pmatrix} \tilde{V}_s^2 \\ \tilde{V}_s^2 \end{pmatrix} \right] ds \\
&\quad + \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \sqrt{\tilde{V}_s^2 + H_{21} \tilde{V}_s^1} \begin{pmatrix} dB_s^3 + dB_s^4 \\ dB_s^3 + dB_s^4 \end{pmatrix}.
\end{aligned}$$

Note that the above implies that $V^{1+} = V^{1-}$ and $V^{2+} = V^{2-}$. This property is due to the the symmetric structure of the baselines and kernels. Therefore, the joint dynamics can be fully captured by considering the joint dynamics of (V^{1+}, V^{2+}). Thus, writing $V^1 := V^{1+} = V^{1-}$ and $V^2 := V^{2+} = V^{2-}$, we have

$$\begin{aligned}
\Gamma(\alpha) \frac{\Gamma(1-\alpha)}{\alpha} V_t^1 &= \int_0^t (t-s)^{\alpha-1} (\mu_1 - \tilde{V}_s^1) ds + \int_0^t \sqrt{V_t^1} (dB_s^1 + dB_s^2) \\
\Gamma(\alpha) \frac{\Gamma(1-\alpha)}{\alpha} V_t^2 &= \int_0^t (t-s)^{\alpha-1} (\mu_2 - \tilde{V}_s^2) ds + \int_0^t \sqrt{V_t^2} (dB_s^3 + dB_s^4).
\end{aligned}$$

We can write the above without $\tilde{V}$ as

$$\Gamma(\alpha) \frac{\Gamma(1-\alpha)}{\alpha} \begin{pmatrix} V_t^1 \\ V_t^2 \end{pmatrix} = \int_0^t (t-s)^{\alpha-1} \Big(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - K^{-1} \begin{pmatrix} V_s^1 \\ V_s^2 \end{pmatrix} \Big) ds + \int_0^t (t-s)^{\alpha-1} \begin{pmatrix} \sqrt{V_s^1} (dB_s^{11} + dB_s^{12}) \\ \sqrt{V_s^2} (dB_s^{21} + dB_s^{22}) \end{pmatrix}.$$

Limiting price process

Turning now to the price process, it remains to compute Δ (see Equation (10)) using the definition. We have

$$\begin{aligned}
{}^{\dagger}\|\psi\|_1 O_3 &= \sum_{k \geq 1} {}^{\dagger}\|\phi\|_1^k O_3 \\
&= O \sum_{k \geq 1} \Big[\Big(\int_0^\infty C \Big)_{11}^k e_3 + \Big(\int_0^\infty C \Big)_{12}^k e_4 \Big] \\
&= \sum_{k \geq 1} \Big[\Big(\int_0^\infty C \Big)_{11}^k O_3 + \Big(\int_0^\infty C \Big)_{12}^k O_4 \Big] \\
&= \Big[\Big(I - \int_0^\infty C \Big)^{-1} - I \Big]_{11} O_3 + \Big[\Big(I - \int_0^\infty C \Big)^{-1} - I \Big]_{12} O_4,
\end{aligned}$$

which, by definition of Δ, yields

$$\begin{aligned}
\Delta_{11} &= \Big[\Big(I - \int_0^\infty C \Big)^{-1} - I \Big]_{11} = \frac{2\gamma_2}{4\gamma_1\gamma_2 - (H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a)} - 1 \\
\Delta_{12} &= \Big[\Big(I - \int_0^\infty C \Big)^{-1} - I \Big]_{12} = \frac{H_{21}^c - H_{21}^a}{4\gamma_1\gamma_2 - (H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a)}.
\end{aligned}$$

Therefore,

$$\Delta = \frac{1}{4\gamma_1\gamma_2 - (H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a)} \begin{pmatrix} 2\gamma_2 & H_{21}^c - H_{21}^a \\ H_{12}^c - H_{12}^a & 2\gamma_1 \end{pmatrix} - I.$$

Finally, any limit point P of the sequence of microscopic price processes satisfies the following equation

$$P_t = \frac{1}{4\gamma_1\gamma_2 - (H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a)} \begin{pmatrix} 2\gamma_2 & H_{21}^c - H_{21}^a \\ H_{12}^c - H_{12}^a & 2\gamma_1 \end{pmatrix} \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} \int_0^t \begin{pmatrix} \sqrt{V_s^1}\, dB_s^1 \\ \sqrt{V_s^1}\, dB_s^2 \\ \sqrt{V_s^2}\, dB_s^3 \\ \sqrt{V_s^2}\, dB_s^4 \end{pmatrix}$$

$$= \frac{1}{4\gamma_1\gamma_2 - (H_{12}^c - H_{12}^a)(H_{21}^c - H_{21}^a)} \begin{pmatrix} 2\gamma_2 & H_{21}^c - H_{21}^a \\ H_{12}^c - H_{12}^a & 2\gamma_1 \end{pmatrix} \int_0^t \begin{pmatrix} \sqrt{V_s^1}(dB_s^1 - dB_s^2) \\ \sqrt{V_s^2}(dB_s^3 - dB_s^4) \end{pmatrix}.$$

This concludes the proof of Corollary 2.

V.C.3 Proof of Corollary 3

We define the interaction kernel between Asset i and Asset j, for $1 \le i, j \le m$, define

$$\phi_{ij}^T(t) := \begin{cases} \alpha(1 - T^{-\alpha}) \mathbb{1}_{t \ge 1} t^{-(\alpha+1)} \begin{pmatrix} (1-\gamma) & \gamma \\ \gamma & (1-\gamma) \end{pmatrix} & \text{if } i = j, \\ \alpha T^{-\alpha} \mathbb{1}_{t \ge 1} t^{-(\alpha+1)} \begin{pmatrix} H^c & H^a \\ H^a & H^c \end{pmatrix} & \text{if Asset } i \text{ and Asset } j \text{ belong to the same sector,} \\ \alpha T^{-\alpha} \mathbb{1}_{t \ge 1} t^{-(\alpha+1)} \begin{pmatrix} H^c + H_r^c & H^a + H_r^a \\ H^a + H_r^a & H^c + H_r^c \end{pmatrix} & \text{otherwise.} \end{cases}$$

Finally, the complete Hawkes baseline and kernel structure is

$$\mu^T = T^{\alpha-1} \begin{pmatrix} \mu^1 \\ \mu^1 \\ \vdots \\ \mu^m \\ \mu^m \end{pmatrix}, \quad \phi^T = \begin{pmatrix} \phi_{11}^T & \phi_{12}^T & \cdots & \phi_{1m}^T \\ \phi_{21}^T & \phi_{22}^T & \cdots & \phi_{2m}^T \\ \vdots & \cdots & \ddots & \vdots \\ \phi_{m1}^T & \cdots & \cdots & \phi_{mm}^T \end{pmatrix}.$$

As in the previous example, the proof is split into three steps. First, we show that the structure of the kernel satisfies the assumptions required to apply Theorem 1. Then, we compute the equation satisfied by the variance and finally the limiting price process.

Checking assumptions of Theorem 1

We can examine the structure of the kernel as in the two-asset example. Define the following basis:

$$O_i := \begin{cases} e_{2i} + e_{2i+1} & \text{if } 1 \le i \le m, \\ e_{2i} - e_{2i} & \text{if } m+1 \le i \le 2m. \end{cases}$$

Using the notations of Section 1, straightforward computations allow us to write

$$\phi^T = O\begin{pmatrix} A^T & 0 \\ B^T & C^T \end{pmatrix} O^{-1} = O\begin{pmatrix} A^T & 0 \\ 0 & C^T \end{pmatrix} O^{-1},$$

where we can compute A^T and C^T. Checking the assumptions is done as in the two-asset case, though the conditions have changed here due to the new structure of the kernel. For example, since

$$\lim_{T\to\infty} \int_0^\infty \phi^T O_{m+i} = (1-2\gamma)O_{n+i} + (H^c - H^a) \sum_{1\le j\ne i\le m} O_{m+j} + \sum_{1\le j\ne i\le m} \sum_{1\le r\le R} (H_r^c - H_r^a) O_{m+j},$$

we have, writing $J := {e_1}^\dagger e_1 + \cdots + {e_m}^\dagger e_m$ and for any $1 \le r \le R$, $J_r := {e_{i_r}}^\dagger e_{i_r} + \cdots + {e_{i_r+m_r}}^\dagger e_{i_r+m_r}$,

$$\int_0^\infty C = (1-2\gamma)I + (H^c - H^a)J + \sum_{1\le r\le R} (H_r^c - H_r^a) J_r.$$

Therefore, as the eigenvalues of $\int_0^\infty C$ can be made explicit, if

$$|\lambda^- + \sum_{1\le r\le R} \lambda_r^- | < 2\gamma,$$

then $\rho(\int_0^\infty C^T) < 1$ and $\rho(\int_0^\infty C) < 1$. Similarly, we can easily check that a necessary condition for $\rho(\int_0^\infty A^T) < 1$ for T large enough is

$$|H^c + H^a + \sum_{1\le r\le R} \frac{m_r - 1}{m-1} (H_r^c + H_r^a)| < \frac{1}{m-1}.$$

Since we are interested in the limit where the number of assets grows to infinity, we also impose

$$|\lambda^- + \sum_{1\le r\le R} \eta_r \lambda_r^- | < 2\gamma$$
$$|\lambda^+ + \sum_{1\le r\le R} \eta_r \lambda_r^+ | < 1.$$

Combined, we have verified all the assumptions on the structure of the kernel. We thus move to assumptions on K and $\Lambda := KM^{-1}$. As in the two-asset example, we have here $M = \alpha I$. Since $K = I - (H^c + H^a)J - \sum_{1\le r\le R}(H_r^c + H_r^a)J_r$, the eigenvalues of K (and therefore those of Λ) are all strictly positive. Thus we have checked all necessary conditions to apply Theorem 1. We can thus state the equation satisfied by the variance process.

Limiting variance process

As in the previous example, we have $V^{i+} = V^{i-}$. Thus, we write the underlying variance of asset i V^i and use the (slight) abuse of notation and define $V := (V^1, V^2, \cdots, V^m)$. Then V satisfies

$$V_t = \frac{\alpha}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \left[\theta - K^{-1} V_s\right] ds + \frac{\alpha\sqrt{2}}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_0^t (t-s)^{\alpha-1} \mathrm{diag}(\sqrt{V_s}) dB_s,$$

where B is a Brownian motion. We can rewrite K^{-1} as

$$K^{-1} = \left(I - (H^c + H^a)J - \sum_{1 \leq r \leq R} (H_r^c + H_r^a) J_r \right)^{-1},$$
$$= \left(I - (H^c + H^a)(m-1) w^\dagger w - \sum_{1 \leq r \leq R} (H_r^c + H_r^a)(m_r - 1) {w_r}^\dagger w_r - \epsilon \right)^{-1},$$

with the small term ϵ

$$\epsilon := (H^c + H^a)(J - (m-1) w^\dagger w) + \sum_{1 \leq r \leq R} (H_r^c + H_r^a)(J_r - (m_r - 1) {w_r}^\dagger w_r).$$

It is easy to check that $\rho(\epsilon) \underset{m \to \infty}{=} o(\frac{1}{m})$, which concludes our study of the variance process. We now turn to the equation satisfied by the limiting price process.

Limiting price process

Using the same approach as in the two-asset case, computing Δ boils down to computing $(I - \int_0^\infty C)^{-1}$. Using the expression for $\int_0^\infty C$ derived previously, we have

$$(I - C)^{-1} = \frac{1}{2\gamma} (I - \frac{H^c - H^a}{2\gamma} J - \sum_{1 \leq r \leq R} \frac{H_r^c - H_r^a}{2\gamma} J_r)^{-1}.$$

Therefore, repeating the same approach we used for K^{-1} yields

$$(I - C)^{-1} = (2\gamma I - \lambda^- w^\dagger w - \sum_{1 \leq r \leq R} \eta_r \lambda_r^- {w_r}^\dagger w_r - \epsilon)^{-1},$$

with $\rho(\epsilon) = o(\frac{1}{m})$. Thus, we have the expression of Δ

$$\Delta = (2\gamma I - \lambda^- w^\dagger w - \sum_{1 \leq r \leq R} \eta_r \lambda_r^- {w_r}^\dagger w_r - \epsilon)^{-1} - I.$$

Plugging this into Theorem 1, we have the equation satisfied by macroscopic prices, which concludes the proof of Corollary 3.

CHAPTER VIA

Deep learning volatility

Abstract

We present a neural network based calibration method that performs the calibration task within a few milliseconds for the full implied volatility surface. The framework is consistently applicable throughout a range of volatility models—including the rough volatility family—and a range of derivative contracts. The aim of neural networks in this work is an off-line approximation of complex pricing functions, which are difficult to represent or time-consuming to evaluate by other means. We highlight how this perspective opens new horizons for quantitative modelling: The calibration bottleneck posed by a slow pricing of derivative contracts is lifted. This brings several numerical pricers and model families (such as rough volatility models) within the scope of applicability in industry practice. The form in which information from available data is extracted and stored influences network performance. This approach is inspired by representing the implied volatility and option prices as a collection of pixels. In a number of applications we demonstrate the prowess of this modelling approach regarding accuracy, speed, robustness and generality and also its potentials towards model recognition.

From:
Deep learning volatility: a deep neural network perspective on pricing and calibration in (rough) volatility models
B. Horvath, A. Muguruza, **M. Tomas**

Introduction

Approximation methods for option prices came in all shapes and forms in the past decades and they have been extensively studied in the literature and well-understood by risk managers. Clearly, the applicability of any given option pricing method (Fourier pricing, PDE methods, asymptotic methods, Monte Carlo, ... etc.) depends on the regularity properties of the particular stochastic model at hand. Therefore, tractability of stochastic models has been one of the most decisive qualities in determining their popularity. In fact it is often a more important quality than the modelling accuracy itself: It was the (almost instantaneous) SABR asymptotic formula that helped SABR become the benchmark model in fixed income desks, and similarly

the convenience of Fourier pricing is largely responsible for the popularity of the Heston model, despite the well-known hiccups of these models. Needless to say that it is the very same reason (the concise Black Scholes formula) that still makes the Black-Scholes model attractive for calculations even after many generations of more realistic and more accurate stochastic market models have been developed. On the other end of the spectrum are rough volatility models, for which (despite a plethora of modelling advantages, see [BFG16, EER18, GJR18] to name a few) the necessity to rely on relatively slow Monte Carlo based pricing methods creates a major bottleneck in calibration, which has proven to be a main limiting factor with respect to industrial applications. This dichotomy can become a headache in situations when we have to weigh up the objectives of accurate pricing vs. fast calibration against one another in the choice of our pricing model. In this work we explore the possibilities provided by the availability of an algorithm that –for a choice of model parameters– directly outputs the corresponding vanilla option prices (as the Black-Scholes formula does) for a large range of maturities and strikes of a given model.

In fact, the idea of mapping model parameters to shapes of the implied volatility surface directly is not new. The stochastic volatility inspired SSVI, eSSVI surfaces (see [Gat04, GJ14, HM17]) do just that: A given set of parameters is translated directly to different shapes of (arbitrage-free) implied volatility surfaces, bypassing the step of specifying any stochastic dynamics for the underlying asset. For stochastic models that admit asymptotic expansions, such direct mappings from model parameters to (approximations of) the implied volatility surface in certain asymptotic regimes can be obtained (one example is the famous SABR formula). Such asymptotic formulae are typically limited to certain asymptotic regimes along the surface by their very nature. Complementary to asymptotic expansions we explore here a direct (approximative) mapping from different parameter combinations of stochastic models to different shapes of implied volatility surface for intermediate regimes. Its appeal is that it combines the advantages of direct parametric volatility surfaces (of the SSVI family) with the possibility to link volatility surfaces to the stochastic dynamics of the underlying asset.

In this chapter we apply deep neural networks (merely) as powerful high-dimensional functional approximators to approximate the multidimensional pricing functionals from model parameters to option prices. The advantage of doing so via deep neural networks over standard (fixed-basis) functional approximations is that deep neural networks are agnostic to the approximation basis [GBC16]. This makes them robustly applicable to several stochastic models consistently. Our objective in doing so is to move the (often time-consuming) numerical approximation of the pricing functional into an off-line preprocessing step. This preprocessing amounts to storing the approximative direct pricing functional in form of the network weights after a supervised training procedure. Using available numerical approximations of option prices as ground truth (in a stochastic model of our choice), we train a neural network to learn an accurate approximation of the pricing functional. After training, the network outputs–for any choice of model parameters–the corresponding implied volatilities within milliseconds for a large range of maturities and strikes along the whole surface. Furthermore, we show

that this procedure generalises well for unseen parameter combinations: the accuracy of price approximation of our neural network pricing functional on out-of-sample data is within the same range as the accuracy of the original numerical approximation used for training. The accuracy of this direct pricing map is demonstrated in our numerical experiments.

One of the striking advantages of this approach is that it speeds up the (on-line) calibration Rough Volatility models to the realm of just a few milliseconds. There have been several recent contributions on neural network calibrations of stochastic models [BHM+19, BGTW19, Her16, Kon18, DSMRS18]. Clearly, much depends on the finesse of the particular network design with respect to the performance of these networks. One contribution of this chapter is to achieve a fast and accurate calibration of the rough Bergomi model of [BFG16] with a general forward variance curve (approximated by piecewise constant function). To demonstrate this, we first perform calibration experiments on simulated data and show calibration accuracy in controlled experiments. To demonstrate the speed and prowess of the approach we then calibrate the rough Bergomi model to historical data and display the evolution of parameters on a dataset consisting of 10 years of SPX data. Another advantage of our modelling choice is that by its very design it can be applied to portfolios including multiple strikes and maturities at the same time which is the first step towards their application as hedging instruments. See for example Buehler et al. [BGTW19] for a motivation.

The chapter is organised as follows: In Section 1 we present a neural network perspective on model calibration and recall stochastic models that are considered in later sections. Section 2 reviews different possible setup and motivates our own approach. In Section 4 we present numerical experiments of price approximations of vanilla and some exotic options, calibration to synthetic data and to historical data. We conclude with further potential applications and outlook to future work.

Numerical experiments and codes are provided on GitHub: NN-StochVol-Calibrations, where an accessible code demo of our results can be downloaded. We also created a library of stochastic models where this approach is demonstrated to work well.

1 A neural network perspective on model calibration

In plain words, any calibration procedure is meant to fix the model parameters such that the model is as close as possible to the observed reality. In a financial context, our model represents the underlying (stocks, indices, volatility, etc.) and we are interested in calibrating the model to available market prices of financial contracts based on this underlying.

Let us first formalise this by setting the notation $\mathscr{M} := \mathscr{M}(\theta)_{\theta\in\Theta}$ which represents an abstract model with parameters θ in the set $\Theta \subset \mathbb{R}^n$, for some $n \in \mathbb{N}$. Thus the model $\mathscr{M}(\theta)$ (stochastic or parametric) and the corresponding prices of financial contracts are fully specified by the choice of the parameter combination $\theta \in \Theta$. Furthermore, we introduce a pricing map $P : \mathscr{M}(\theta, \zeta) \to R^m$, where $\zeta : (C(\mathbb{R}) \to \mathbb{R}^m)$, $m \in \mathbb{N}$ denote the financial products we aim to price, such as vanilla options for (a set of) given maturities and strikes. Let us denote the observed market data corresponding to the contracts by $\mathscr{P}^{MKT}(\zeta) \in \mathbb{R}^m, \ \ m \in \mathbb{N}$.

Parameter Calibration: The parameter configuration $\hat{\theta}$ solves an (ideal) δ-calibration problem for a model $\mathscr{M}(\Theta)$ for the conditions $\mathscr{P}^{MKT}(\zeta)$ if

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(P(\mathscr{M}(\theta), \zeta), \mathscr{P}^{MKT}(\zeta)) \tag{1}$$

where $\delta(\cdot,\cdot)$ is a suitable choice of metric for the financial contract ζ at hand.

For most financial models however (1) represents an idealised form of the calibration problem as in practice there rarely exists an analytical formula for the option price $P(\mathscr{M}(\theta), \zeta)$ and for the vast majority of financial models it needs to be computed by some numerical approximation scheme.

Approximate Parameter Calibration We say that the parameter configuration $\hat{\theta} \in \Theta$ solves an *approximate δ-calibration* problem for the model $\mathscr{M}(\Theta)$ for the conditions $\mathscr{P}^{MKT}(\zeta)$ if

$$\hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(\widetilde{P}(\mathscr{M}(\theta), \zeta), \mathscr{P}^{MKT}(\zeta)) \tag{2}$$

where $\delta(\cdot,\cdot)$ is a suitably chosen metric and $\widetilde{P}$ is a numerical approximation of the pricing map P.

In the remainder of this chapter it is this second type of calibration problem that we will be concerned with: In our numerical experiments (Section 4) we consider the numerical approximation $\widetilde{P}$ of the pricing map P as the benchmark (available truth) for generating synthetic training samples in the training a neural network to approximate pricing maps. Clearly, the better the original numerical approximations, the better the network approximation will be. In a separate work we will illuminate this perspective with a Bayesian analysis of the calibration procedure.

1.1 A brief reminder of some (rough) models considered

We would like to emphasize that our methodology can in principle be applied to any (classical or rough) volatility model. From the classical Black Scholes or Heston models to the rough Bergomi model of [BFG16], also to large class of rough volatility models (see Horvath, Jacquier and Muguruza [HJM17] for a general setup). In fact the methodology is not limited to stochastic models, also parametric models of implied volatility could be used for generating training samples of abstract models, but we have not pursued this direction further.

The Rough Bergomi model
In the abstract model framework, the rough Bergomi model is represented by $\mathcal{M}^{rBergomi}(\Theta^{rBergomi})$, with parameters $\theta = (\xi_0, \nu, \rho, H) \in \Theta^{rBergomi}$. On a given filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mathbb{P})$ the model corresponds to the following system

$$\begin{aligned} dX_t &= -\frac{1}{2} V_t dt + \sqrt{V_t} dW_t, \quad \text{for } t > 0, \quad X_0 = 0, \\ V_t &= \xi_0(t) \mathcal{E}\left(\sqrt{2H}\nu \int_0^t (t-s)^{H-1/2} dZ_s\right), \quad \text{for } t > 0, \quad V_0 = v_0 > 0 \end{aligned} \tag{3}$$

where $H \in (0,1)$ denotes the Hurst parameter, $\nu > 0$, $\mathcal{E}(\cdot)$ the stochastic exponential [DD70], and $\xi_0(\cdot) > 0$ denotes the initial forward variance curve (see [Ber15, Section 6]), and W and Z are correlated standard Brownian motions with correlation parameter $\rho \in [-1,1]$. To fit the model parameters into our abstract model framework $\Theta^{rBergomi} \subset \mathbb{R}^n$ for some $n \in \mathbb{N}$, the initial forward variance curve $\xi_0(\cdot) > 0$ is approximated by a piecewise constant function in our numerical experiments in Sections 3.1.1 and 3.2.1. We refer the reader to Horvath, Jacquier and Muguruza [HJM17] for one general setting of rough volatility models and their numerical simulation.

The Heston model
The Heston model is described by the system

$$\begin{aligned} dS_t &= \sqrt{V_t} S_t dW_t \quad \text{for } t > 0, \quad S_0 = s_0 \\ dV_t &= a(b - V_t)dt + v\sqrt{V_t} dZ_t \quad \text{for } t > 0, \quad V_0 = v_0 \end{aligned} \tag{4}$$

with W and Z Brownian motions with correlation parameter $\rho \in [-1,1]$, $a, b, v > 0$ and $2ab > v^2$. In our framework it is denoted by $\mathcal{M}^{Heston}(\theta)$ with $\theta = (a, b, v, \rho) \in \Theta^{Heston} \subset \mathbb{R}^4$. The Heston model was considered by [BHM+19, DRF18] in different neural network contexts.

The Bergomi model
In the general n-factor Bergomi model, the volatility is expressed as

$$V_t = \xi_0(t) \mathcal{E}\left(\eta_i \sum_{i=1}^n \int_0^t \exp\left(-\kappa_i (t-s) dW_s^i\right)\right) \quad \text{for } t > 0, \quad V_0 = v_0 > 0, \tag{5}$$

where $\eta_1, \ldots, \eta_n > 0$ and $(W^1, \ldots, W^n)$ is an n-dimensional correlated Brownian motion, $\mathcal{E}(\cdot)$ the stochastic exponential [DD70], and $\xi_0(\cdot) > 0$ denotes the initial forward variance curve, see

[Ber15, Section 6] for details. In this work we consider the Bergomi model for $n = 1, 2$ in Section 4. Henceforth, $\mathscr{M}^{1FBergomi}(\xi_0, \beta, \eta, \rho)$ represents the 1 Factor Bergomi model, corresponding to the following dynamics:

$$\begin{aligned} dX_t &= -\frac{1}{2} V_t dt + \sqrt{V_t} dW_t \quad \text{for } t > 0,\ X_0 = 0 \\ V_t &= \xi_0(t) \mathscr{E}\left(\eta \int_0^t \exp(-\beta(t-s)) dZ_s\right) \quad \text{for } t > 0, \quad V_0 = v_0 > 0, \end{aligned} \tag{6}$$

where $\nu > 0$, and W and Z are correlated standard Brownian motions with correlation parameter $\rho \in [-1, 1]$. To fit the model parameters into our abstract model framework $\Theta^{1FBergomi} \subset \mathbb{R}^n$, for some $n \in \mathbb{N}$, the initial forward variance curve $\xi_0(\cdot) > 0$ is approximated in our numerical experiments by a piecewise constant function in Sections 3.1.1, and 3.2.1.

The SABR model
The stochastic alpha beta rho model of Hagan et al. [HKLW02, HLW15] is denoted in our setting as $\mathscr{M}^{SABR}(\alpha, \beta, \rho)$ and is defined as

$$\begin{aligned} dS_t &= V_t S_t^{\beta} dW_t \quad \text{for } t > 0, \quad S_0 = s_0. \\ dV_t &= \alpha V_t dZ_t \quad \text{for } t > 0, \quad V_0 = v_0 \end{aligned} \tag{7}$$

where $v_0, s_0, \alpha > 0$ and $\beta \in [0, 1]$. The SABR model is considered by McGhee in [McG18] in a neural network context (see also Section 1.3).

1.2 Calibration bottlenecks in volatility modelling and deep calibration

Whenever for a stochastic volatility model the numerical approximate calibration procedures (2) are computationally slow, a bottleneck in calibration time can deem the model of limited applicability for industrial production irrespective of other desirable features the model might have. This is the case in particular for the family rough volatility models, where the rough fractional Brownian motion in the volatility dynamics rules out usual Markovian pricing methods such as finite differences. So far such calibration bottlenecks have been a major limiting factor for the class of rough volatility models, whose overwhelming modelling advantages have been explored and highlighted in rapidly expanding number of academic articles [AGLM18, ALV07, BFG16, BFG+20, BFG+19, BLP17, EER19, Fuk11, GJR18, JMM18, HJL19, JPS18] in the past years. Other examples include models with delicate degeneracies (such as the SABR model around zero forward) which for a precise computation of arbitrage-free prices require time consuming numerical pricing methods such as Finite Element Methods [HR18], Monte Carlo [COVDW11, LGO17b] or the evaluation of multiple integrals [AKS13].

Contrary to Hernandez's [Her16] pioneering work, where he develops a direct calibation via NN, we set up and advocate a two setp calibration approach.

Two Step Approach (i) Learn a model and (ii) Calibrate to data: One separates the calibration procedure described in (2) into two parts: **(i)** We first learn (approximate) the

pricing map by a neural network that maps parameters of a stochastic model to pricing functions (or implied volatilities (cf. section (1.1) and we store this map during an off-line training procedure. In a second step **(ii)** we calibrate (on-line) the now deterministic approximative learned price map, which speeds up the on-line calibration by orders of magnitude. To formalise the two step approach, we write for a payoff ζ and a model $\mathscr{M}$ with parameters $\theta \in \Theta$

$$\textbf{(i) Learn: } \widetilde{F}(\Theta,\zeta) = \widetilde{P}(\mathscr{M}(\Theta,\zeta)) \qquad \textbf{(ii) Calibrate: } \hat{\theta} = \underset{\theta\in\Theta}{\operatorname{argmin}}\, \delta(\widetilde{F}(\theta,\zeta), \mathscr{P}^{MKT}(\zeta)). \tag{8}$$

Note that in part **(ii)** of (8) we essentially replaced $\widetilde{P}(\mathscr{M}(\Theta,\zeta))$ in equation (2) by its learned (deterministic) counterpart $\widetilde{F}(\Theta,\zeta)$ (which will be a Neural Network see Section 3.1) from **(i)**. Therefore, this second calibration is–by its deterministic nature–considerably faster than calibration of all those traditional stochastic models, which involve numerical simulation of the expected payoff $P(\mathscr{M}(\theta,\zeta)) = \mathbb{E}[\zeta(X(\theta))]$ for some underlying stochastic process X^θ. The first part **(i)** in (8) denotes an approximation of the pricing map through a neural network, which is calibrated in a supervised training procedure using the original (possibly slow) numerical pricing maps for training (see sections 3.1 and 4 for details in specific examples).

In the following sections we elaborate on the objectives and advantages of this two step calibration approach and present examples of neural network architectures, precise numerical recipes and training procedures to apply the two step calibration approach to a family of stochastic volatility models. We also present some numerical experiments (corresponding codes are available on GitHub: NN-StochVol-Calibrations) and report on learning errors and on calibration times.

1.3 Challenges in neural network approximations of pricing functionals

In general problem (1) and henceforth (2) is solved using suitable numerical optimisation techniques such as gradient descent [GBC16], specific methods for certain metrics (such as Lavenberg-Marquadnt [Lev44] for L^2), neural networks, or tailor-made methods to the complexity of the optimisation problem and objective function at hand[1]. But irrespective of their level of sophistication all optimisers for calibration share a common property: repeated (iterative) evaluation of the pricing map $\theta \mapsto P(\mathscr{M}(\theta),\zeta)$ (resp. an approximation $\widetilde{P}$ thereof) on each instance θ of consecutive parameter combinations until a sufficiently small distance $\delta(\widetilde{P}(\mathscr{M}(\theta),\zeta), \mathscr{P}^{MKT}(\zeta)$ between model prices and observed prices is achieved. Consequently, the pricing map is arguably the computational cornerstone of a calibration algorithm. Main differences between specific calibration algorithms effectively lie in the way the specific choice of evaluated parameter combinations $\{\theta_1,\theta_2\ldots\}$ are determined, which hence determines the total number N of functional evaluations of the pricing function $\big(P(\mathscr{M}(\theta_i),\zeta)\big)_{i=1\ldots N}$ used in the calibration until the desired precision $\delta(\widetilde{P}(\mathscr{M}(\hat{\theta}),\zeta), \mathscr{P}^{MKT}(\zeta))$ is achieved. In case the pricing map

$$\begin{aligned} P(\mathscr{M}(\cdot),\zeta) \;:\; \Theta &\longrightarrow P(\mathscr{M}) \\ \theta &\mapsto P(\mathscr{M}(\theta),\zeta) \end{aligned}$$

[1] For details and an overview on calibration methods see [GBC16].

involved in (1) is available in closed form, and can be evaluated instantaneously, the calibration (2) is fast even if a high number N of functional evaluations is used. If the pricing map is approximated numerically, calibration time depends strongly on the time needed to generate a functional evaluation of the numerical approximation

$$\theta_i \mapsto \widetilde{P}(\mathscr{M}(\theta_i),\zeta), \qquad \theta_i \in \{\theta_1,\dots\theta_N\} \tag{9}$$

at each iteration $i = 1,\dots,N$ of the calibration procedure. Slow functional evaluations potentially cause substantial bottlenecks in calibration time. This is where we see the most powerful use of the prowess of neural network approximation:

A neural network is constructed to replace in **(i)** of (8) the pricing map, that is to approximate (for a given financial contract ζ) the pricing map from the full set[2] of model parameters Θ of the model to the corresponding prices $P(\mathscr{M}(\theta,\zeta))$. The ***first challenge*** for the neural network approximator of pricing functionals is to speed up this process and enable us to obtain ***faster functional evaluations*** and thereby lift the bottleneck of calibration. The ***second challenge*** is to do so with an accuracy that remains within the error bounds of the original numerical pricing discretisation:

$$\begin{aligned} \widetilde{F} : \Theta &\longrightarrow \widetilde{P}(\mathscr{M}) \\ \theta &\mapsto \widetilde{F}(\theta,\zeta) \end{aligned} \tag{10}$$

More precisely (motivated by (2)), for any parameter combination $\theta \in \Theta$ we aim to approximate the numerical approximation $\widetilde{P}$ of the true option price P with the neural network $\widetilde{F}$ up to the same order of precision $\epsilon > 0$ up to which $\widetilde{P}$ approximates P. That is, for any $\theta \in \Theta$

$$\widetilde{F}(\theta) = P(\mathscr{M}(\theta),\zeta) + \mathscr{O}(\epsilon) \quad \text{whenever} \quad \widetilde{P}(\mathscr{M}(\theta),\zeta) = P(\mathscr{M}(\theta),\zeta) + \mathscr{O}(\epsilon).$$

Therefore, our training objective is

$$\widetilde{F}(\theta) = \widetilde{P}(\mathscr{M}(\theta),\zeta) + \mathscr{O}(\epsilon). \tag{11}$$

where $\widetilde{P}$ is the available numerical approximation of the pricing function, which is considered as ground truth. In our numerical experiments in Section 4 we demonstrate that our approximation network achieves this approximation accuracy and yields a substantial speedup in terms of functional evaluations.

1.4 Motivations for our choice of training setup and features of neural networks as approximators of pricing functionals

There are several advantages of separating the tasks of pricing and calibration which we address in full detail in a separate work. Here we recall some of the most convincing reasons to do so. Above all, the most appealing reason is that it allows us to build upon the knowledge

[2]Note that the set $\theta_1,\dots,\theta_N$ in (9) is extended to the full set of possible parameter combinations Θ in (10).

we have gained about the models in the past decades, which is of crucial importance from a risk management perspective. By its very design, deep learning the *price approximation* **(i)** combined with **(ii)** deterministic calibration does not cause more headache to risk managers and regulators than the corresponding stochastic models do. Designing the training as described above demonstrates how deep learning techniques can successfully extend the toolbox of financial engineering, without making compromises on any of our objectives.

1. The knowledge gathered in many years of experience with traditional models remains useful and risk management libraries of models remain valid. The neural network is only used as a computational enhancement of models.

2. The availability of training data for training the deep neural network does not cause any constraints as it is synthetically generated by traditional numerical methods.

3. This can be extended beyond the models presented in this work: Whenever a consistent numerical pricer exists for a model, it can be approximated and replaced by a deep neural network that provides fast numerical evaluations of the pricing map.

Here, we identify the grid-based apporach as our choice of training. Though a thorough analysis of the best training approaches is subject to further research, we have good reason to believe that the grid-based approach provides a powerful and robust methodology for training:

1.4.1 Reasons for the choice of grid-based implicit training

In the grid-based approach we evaluate the values of implied volatility surface along 8×11 gridpoints with $80,000$ different parameter combinations we effectively evaluate the "fit" of the surface to numerically generated ones across the same number of points. By moving the evaluation of the implied volatilities into the objective function we improve the learning in many aspects:

- The first advantage of implicit training is that it efficiently exploits the structure of the data. Updates in neighbouring volatility points σ_{n-1} and σ_n can be incorporated in the learning process. If the output is a full grid as in (16) this effect is further enhanced. Updates of the network on each gridpoint also imply additional information for updates of the network on neighbouring gridpoints. One can say that we regard the implied volatility surface as an image with a given number of pixels.

- A further advantage of the image based implicit training is, that by evaluating the objective function on a larger set of (grid) points, injectivity of the mapping can be more easily guaranteed than in the pointwise training: Two distinct parameter combinations are less likely to yield the same value across a set of gridpoints, then if evaluated only on a single point.

- We do not limit ourselves to one specific grid on the implied volatility surface. We store the generated 60,000 sample paths for the training data and chose a set of maturities (here 8) and strikes (here 11) to evaluate prices corresponding to these paths. But we can easily add and evaluate additional maturities and strikes to the same set of paths. Note in particular that in this training design we can refine the grid on the implied volatility surface without increasing the number of training samples needed and without significantly increasing the computational time for training as the portfolio of vanilla options on the same underlying grows with different strikes and maturities.

1.4.2 Some relevant properties of deep neural networks as functional approximators

Deep feed forward[3] neural networks are the most basic deep neural networks, originally designed to approximate some function F^*, which is not available in closed form but only through sample pairs of given input data x and output data $y = F^*(x)$. In a nutshell, a feed forward network defines a mapping $y = F(x, w)$ and the training determines (calibrates) the optimal values of network parameters $\widehat{w}$ that result in the best function approximation[4] $F^*(\cdot) \approx F(\cdot, \widehat{w})$ of the unknown function $F^*(\cdot)$ for the given pairs of input and output data (x, y), cf. [GBC16, Chapter 6].
To formalise this, we introduce some notation and recall some basic definitions and principles of function approximation via (feedforward) neural networks:

Definition 1 (Neural network). *Let $L \in \mathbb{N}$ and the tuple $(N_1, N_2 \cdots, N_L) \in \mathbb{N}^L$ denote the number of layers (depth) and the number of nodes (neurons) on each layer respectively. Furthermore, we introduce the affine functions*

$$\begin{aligned} w^l :\ \mathbb{R}^{N_l} &\longrightarrow \mathbb{R}^{N_{l+1}} \text{ for } 1 \le l \le L-1 \\ x &\mapsto A^{l+1}x + b^{l+1} \end{aligned} \tag{12}$$

acting between layers for some $A^{l+1} \in \mathbb{R}^{N_{l+1} \times N_l}$. The vector $b^{l+1} \in \mathbb{R}^{N_{l+1}}$ denotes the bias term *and each entry $A^{l+1}_{(i,j)}$ denotes the* weight *connecting node $i \in N_l$ of layer l with node $j \in N_{l+1}$ of layer $l+1$. For the the collection of affine functions of the form* (12) *on each layer we fix the notation $w = (w^1, \ldots, w^L)$. We call the tuple w the* network weights *for any such collection of affine functions. Then a Neural Network $F(w, \cdot) : \mathbb{R}^{N_0} \to \mathbb{R}^{N_L}$ is defined as the composition:*

$$F := F_L \circ \cdots \circ F_1 \tag{13}$$

where each component is of the form $F_l := \sigma_l \circ W^l$. The function $\sigma_l : \mathbb{R} \to \mathbb{R}$ is referred to as the activation function. It is typically nonlinear and applied component wise on the outputs of the affine function W^l. The first and last layers, F_1 and F_L, are the input and output layers. Layers in between, $F_2 \cdots F_{L-1}$, are called hidden layers.

[3] The network is called feed forward if there are no feedback connections in which outputs of the model are fed back into itself.

[4] In our case y is a 8×11-point grid on the implied volatility surface and x are model parameters $\theta \in \Theta$, for details see Section 2.

The following central result of Hornik justifies the use of neural networks as approximators for multivariate functions and their derivatives.

Theorem 1 (Universal approximation theorem (Hornik, Stinchcombe and White [HSW89])). *Let $\mathcal{NN}^{\sigma}_{d_0,d_1}$ be the set of neural networks with activation function $\sigma : \mathbb{R} \mapsto \mathbb{R}$, input dimension $d_0 \in \mathbb{N}$ and output dimension $d_1 \in \mathbb{N}$. Then, if σ is continuous and non-constant, $\mathcal{NN}^{\sigma}_{d_0,1}$ is dense in $L^p(\mu)$ for all finite measures μ.*

There is a rapidly growing literature on approximation results with neural networks, see [Hor91, HSW90, Mha93, SCC18] and the references therein. Among these we would like to single out one particular result:

Theorem 2 (Universal approximation theorem for derivatives (Hornik, Stinchcombe and White [HSW90])). *Let $F^* \in \mathscr{C}^n$ and $F : \mathbb{R}^{d_0} \to \mathbb{R}$ and $\mathcal{NN}^{\sigma}_{d_0,1}$ be the set of single-layer neural networks with activation function $\sigma : \mathbb{R} \mapsto \mathbb{R}$, input dimension $d_0 \in \mathbb{N}$ and output dimension 1. Then, if the (non-constant) activation function is $\sigma \in \mathscr{C}^n(\mathbb{R})$, then $\mathcal{NN}^{\sigma}_{d_0,1}$ arbitrarily approximates f and all its derivatives up to order n.*

Remark 1. *Theorem 2 highlights that the smoothness properties of the activation function are of significant importance in the approximation of derivatives of the target function F^*. In particular, to guarantee the convergence of l-th order derivatives of the target function, we choose an activation function $\sigma \in C^l(\mathbb{R})$. Note that the ReLu activation function , $\sigma_{ReLu}(x) = (x)^+$ is not in $\mathscr{C}^l(\mathbb{R})$ for any $l > 0$, while $\sigma_{Elu}(x) = \alpha(e^x - 1)$ is smooth.*

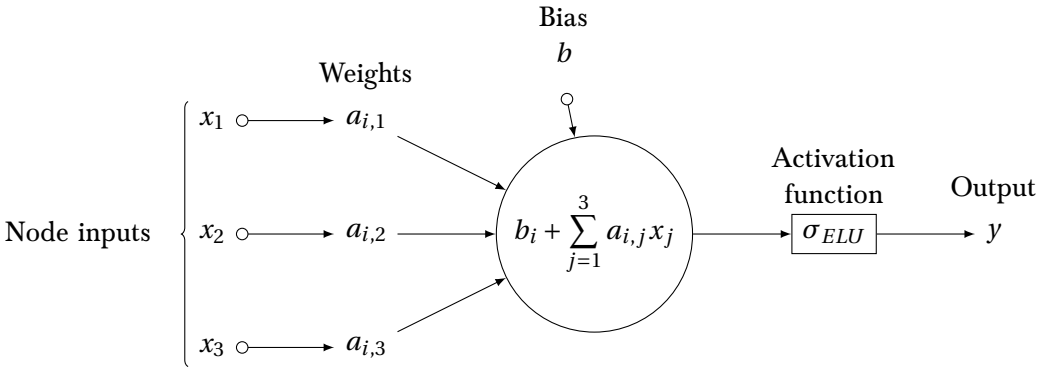


Figure VIA.1: **In detail neuron behaviour**.

The following Theorem provides theoretical bounds for the above rule of thumb and establishes a connection between the number of nodes in a network and the number of training samples needed to train it.

Theorem 3 (Estimation bounds for Neural Networks (Barron [Bar94])). *Let $\mathcal{NN}^{\sigma}_{d_0,d_1}$ be the set of single-layer neural networks with Sigmoid activation function $\sigma(x) = \frac{e^x}{e^x+1}$, input dimension*

$d_0 \in \mathbb{N}$ *and output dimension* $d_1 \in \mathbb{N}$. *Then:*

$$\mathbb{E}\|F^* - \hat{F}\|_2^2 \leq \mathcal{O}\left(\frac{C_f^2}{n}\right) + \mathcal{O}\left(\frac{nd_0}{N}\log N\right)$$

where n *is the number of nodes,* N *is the training set size and* C_{F^*} *is the first absolute moment of the Fourier magnitude distribution of* F^*.

Remark 2. *Barron's [Bar94] insightful result gives a rather explicit decomposition of the error in terms of bias (model complexity) and variance:*

- $\mathcal{O}\left(\frac{C_{F^*}^2}{n}\right)$ *represents the model complexity, i.e. the larger* n *(number of nodes) the smaller the error*
- $\mathcal{O}\left(\frac{nd_0}{N}\log N\right)$ *represents the variance, i.e. a large* n *must be compensated with a large training set* N *in order to avoid overfitting.*

Finally, we motivate the use of multi layer networks and the choice of network depth. Even though a single layer might theoretically suffice to arbitrarily approximate any continuous function,in practice the use of multiple layers dramatically improves the approximation capacities of network. We informally recall the following Theorem due to Eldan and Shamir [ES16] and refer the reader to the original paper for details.

Theorem 4 (Power of depth of Neural Networks (Eldan and Shamir [ES16])). *There exists a simple (approximately radial) function on* $\mathbb{R}^d$*, expressible by a small 3-layer feedforward neural networks, which cannot be approximated by any 2-layer network, to more than a certain constant accuracy, unless its width is exponential in the dimension.*

Remark 3. *In spite of the specific framework by Eldan and Shamir [ES16] being restrictive, it provides a theoretical justification to the power of "deep" neural networks (multiple layers) against "shallower" networks (i.e. few layers) as in [McG18] with a larger number of neurons. On the other hand, multiple findings indicate [BHM+19, IS15] that adding hidden layers beyond 4 hidden layers does not significantly improve network performance.*

2 Pricing and calibration with neural networks: Optimising network and training

In this section we compare different objective functions (direct calibration to data to an image-based implicit learning approach) and motivate our choice of image-based objective function. We give details about network architectures for the approximation network and compare different optimisers for the calibration step.

2.1 The objective function

1. Learn the map $F^*(\theta) = \{P^{\mathscr{M}(\theta)}(\zeta_i)\}_{i=1}^n$ via neral network, where $\{\zeta_i\}_{i=1,\dots,n}$ represents the exotic product attributes (such as maturity, strike or barrier level) on a prespecified grid with size n.
$$\hat{w} = \underset{w\in\mathbb{R}^n}{\operatorname{argmin}} \sum_{u=1}^{N_{Train}} \sum_{i=1}^{n} \left(F(\theta_u, w)_i - F^*(\theta_u)_i\right)^2.$$
2. Solve
$$\hat{\theta} := \underset{\theta\in\Theta}{\operatorname{argmin}} \sum_{i=1}^{n} (\widetilde{F}(\theta)_i - P^{MKT}(\zeta_i))^2. \tag{14}$$

2.1.1 For vanillas

As in many academic and industry research papers, we pursue the calibration of vanilla contracts via approximation of the implied volatility surface[5].

We take this idea further and design an implicit form of the pricing map that is based on storing the implied volatility surface as an image given by a grid of "pixels". This image-based representation has a formative contribution in the performance of the network we present in Section 4. We present our contribution here; Let us denote by $\Delta := \{k_i, T_j\}_{i=1,\ j=1}^{n,\ m}$ a fixed grid of strikes and maturities, then we propose the following two step approach:

1. Learn the map $F^*(\theta) = \{\sigma_{BS}^{\mathscr{M}(\theta)}(T_i, k_j)\}_{i=1,\ j=1}^{n,\ m}$ via neural network $\widetilde{F}(\theta) := F(\theta, \hat{w})$ where
$$\begin{aligned} F^* : \Theta &\longrightarrow \mathbb{R}^{n\times m} \\ \theta &\mapsto F^*(\theta) \end{aligned} \tag{16}$$
where the input is a parameter combination $\theta \in \Theta$ of the stochastic model $\mathscr{M}(\Theta)$ and the output is a $n \times m$ grid on the implied volatility surface $\{\sigma_{BS}^{\mathscr{M}(\theta)}(T_i, k_j)\}_{i=1,\ j=1}^{n,\ m}$ where

[5]For sake of completeness we introduce the Black-Scholes Call pricing function in terms of log-strike k, initial spot S_0, maturity T and volatility σ:
$$BS(\sigma, S_0, k, T) := S_0\mathscr{N}(d_+) - K\mathscr{N}(d_-), \quad d_\pm := \frac{\log(S_0) - k}{\sqrt{T}\sigma} \pm \frac{\sqrt{T}\sigma}{2},$$
where $\mathscr{N}(\cdot)$ denotes the Gaussian cumulative distribution function. The implied volatility induced by a Call option pricing function $P(K, T)$ is then given by the unique solution $\sigma_{BS}(k, T)$ of the following equation
$$BS(\sigma_{BS}(k, T), S_0, k, T) = P(k, T).$$
Precisely, we seek to solve the following calibration problem
$$\hat{\theta} := \underset{\theta\in\Theta}{\operatorname{argmin}}\, d(\Sigma_{BS}^{\mathscr{M}(\theta)}, \Sigma_{BS}^{MKT}) \tag{15}$$
where $\Sigma_{BS}^{\mathscr{M}(\theta)} := \{\sigma_{BS}^{\mathscr{M}(\theta)}(k_i, T_j)\}_{i=1,..,n,\ j=1,\dots,m}$ represents the set of implied volatilities generated by the model pricing function $P(\mathscr{M}(\theta), k, T)$ and $\Sigma_{BS}^{MKT} := \{\sigma_{BS}^{MKT}(k_i, T_j)\}_{i=1,..,n,\ j=1,\dots,m}$ are the corresponding market implied volatilities, for some metric $d : \mathbb{R}^{n\times m} \times \mathbb{R}^{n\times m} \to \mathbb{R}^+$.

$n, m \in \mathbb{N}$ are chosen appropriately (see Section 3.1). Then,

$$\hat{w} = \underset{w \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{u=1}^{N_{Train}} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(F(\theta_u, w)_{ij} - F^*(\theta_u)_{ij} \right)^2 .$$

2. Solve

$$\hat{\theta} := \underset{\theta \in \Theta}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{m} (\widetilde{F}(\theta)_{ij} - \sigma_{BS}^{MKT}(T_i, k_j))^2.$$

Remark 4. *Notice that $\hat{w}(\Delta)$ depends on Δ implicitly, consequently so does $\widetilde{F}(\theta) = F(\theta, \hat{w}(\Delta))$ (hence the name implicit learning). This setting is similar to that of image recognition and exploits the structure of the data to reduce the complexity of the Network (see Section 4 for details).*

Remark 5. *In our experiments we chose $n = 8$ and $m = 11$. At first, a criticism of mapping* (16) *might be the inability to extrapolate/interpolate between maturities/strikes outside the grid Δ. However, one is free to choose the grids Δ as fine as needed. In addition, one may use standard (arbitrage free) uni/bi-variate splines techniques to extrapolate/interpolate across strikes and maturities, as with traditional market data observable only at discrete points.*

Figure VIA.2: Volatility surface generated by the neural network approximator and the corresponding original counterpart on a grid given by 8 maturities and 11 strikes.

2.1.2 Some exotic payoffs

Our framework extends to a number of exotic products such as: Digital barriers, no-touch (or double no-touch) barrier, cliquets or autocallables.

We present some numerical experiments in Section 3.3, to demonstrate the pricing of digital barrier options. More precisely, in Section 3.3 we consider down-and-in such as down-and-out digital barrier options, the main building blocks of many Autocallable products. For a barrier level $B < S_0$ and maturity T the payoff is given by:

$$P^{Down-and-In}(B, T) = \mathbb{E}\left[\mathbf{1}_{\{\tau_B \leq T\}}\right] \tag{17}$$

$$P^{Down-and-Out}(B, T) = \mathbb{E}\left[\mathbf{1}_{\{\tau_B \geq T\}}\right] \tag{18}$$

where $\tau_B = \inf_t \{S_t = B\}$. In this setting, we may easily generate a grid for barrier levels and maturities $\Delta^{Barrier} := \{B_i, T_j\}_{i=1,\, j=1}^{n,\ m}$ that we can fit in the objective function specified in (14)

2.2 Network architecture and training

Motivated by the above analysis, we choose to set up the calibration in the implicit two-step approach. This involves a separation of the calibration procedure into (i) “Deep approximation" an approximation network with an implicit training and (ii) “Calibration" a calibration layer on top. We first start by describing the approximation network in the implicit image-based training and discuss the calibration in Section 3.2 below. In addition, we will highlight specific techniques that contribute to the robustness and efficiency of our design.

2.2.1 Network architecture of the implied volatility map approximation

Here we motivate our choice of network architecture for the following numerical experiments which were inspired by the analysis in the previous sections. Our network architecture is summarised in the graph VIA.3 below.

1. A fully connected feed forward neural network with 4 hidden layers (due to Theorem 4) and 30 nodes on each layers (see Figure VIA.3 for a detailed representation)

2. Input dimension = n, number of model parameters

3. Output dimension = 11 strikes× 8 maturities for this experiment, but this choice of grid can be enriched or modified.

4. The four inner layers have 30 nodes each, which adding the corresponding biases results on a number

$$(n+1)\times 30 + 4\times(1+30)\times 30 + (30+1)\times 88 = 30n + 6478$$

   of network parameters to calibrate (see Section 1.4.2 for details).

5. Motivated by Theorem 2 we choose the Elu $\sigma_{Elu} = \alpha(e^x - 1)$ activation function for the network.

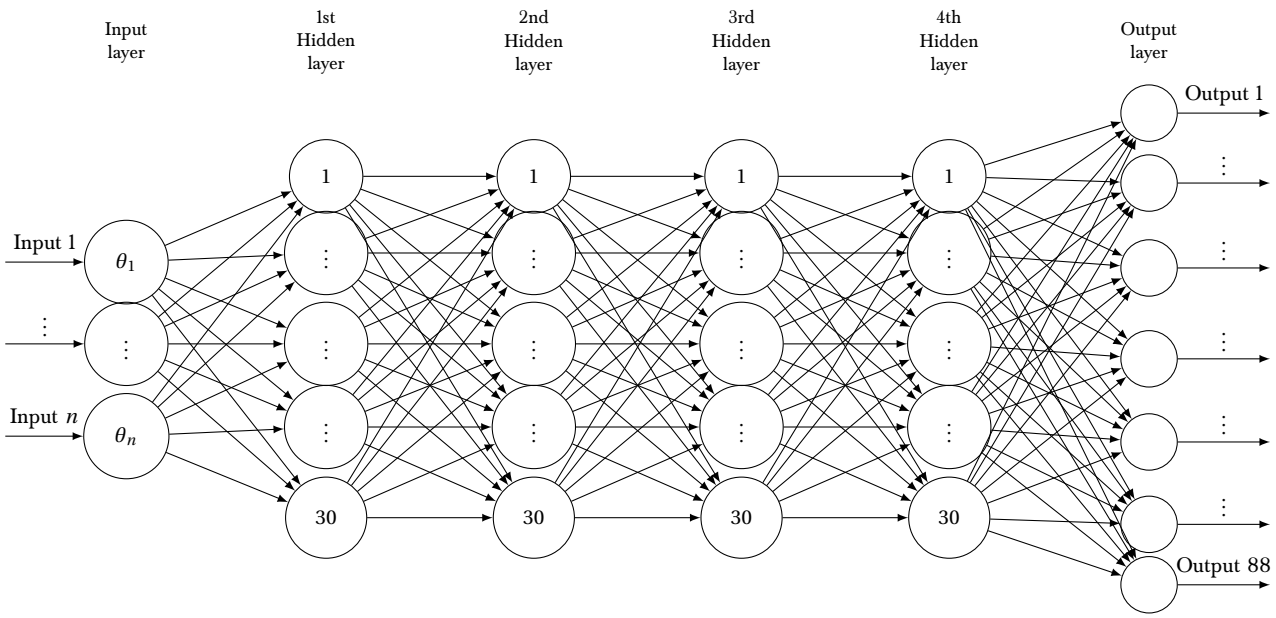


Figure VIA.3: **Neural network architecture.**
Our neural network architecture with 4 hidden layers and 30 neurons on each hidden layer, with the model parameters of the respective model on the input layer and with the 8×11 implied volatility grid on the output layer.

2.2.2 Training of the approximation network

We follow the common features of optimization techniques and choose mini-batches, as described in Goodfellow, Bengio and Courville [GBC16]. Typical batch size values range from around 10 to 100. In our case we started with small batch sizes and increased the batch size until training performance consistently reached a plateau. Finally, we chose batch sizes of 32, as performance is similar for batch sizes above this level, and larger batch sizes increase computation time by computing a larger number of gradients at a time.

In our training design, we use a number of regularisation techniques to speed up convergence of the training, to avoid overfitting and improve the network performance.

1) Early stopping: We choose the number of epochs as 200 and stop updating network parameters if the error has not improved in the test set for 25 steps.

2) Normalisation of model parameters: Usually, model parameters are restricted to a given domain i.e. $\theta \in [\theta_{min}, \theta_{max}]$. Then, we perform the following normalisation transform:

$$\frac{2\theta - (\theta_{max} + \theta_{min})}{\theta_{max} - \theta_{min}} \in [-1, 1].$$

3) Normalisation of implied volatilities: The normalisation of implied volatilities is a more delicate matter, since $\sigma_{BS}(T,k,\theta^{train}) \in [0,\infty)$, for each T and k. Therefore, we choose to normalise the surface subtracting the sample empirical mean and dividing by the sample standard deviation.

2.3 The calibration step

Once the pricing map approximation $\widetilde{F}$ for the implied volatility is found, only the calibration step in (2) is left to solve. In general, for financial models the pricing map F^* is assumed to be smooth (at least C^1 differentiable) with respect to all its input parameters θ.

Gradient-based optimizers

A standard necessary first order condition for optimality in (2) is that

$$\nabla^{\theta}\delta\left(\widetilde{F}(\mathscr{M}(\theta),\zeta),\mathscr{P}^{MKT}(\zeta)\right)=0, \tag{19}$$

provided that the objective function is smooth. Then, a natural update rule is to move along the gradient via Gradient Descent i.e.

$$\theta_{i+1}=\theta_i-\lambda\nabla^{\theta}\delta\left(\widetilde{F}(\mathscr{M}(\theta_i),\zeta),\mathscr{P}^{MKT}(\zeta)\right),\quad \lambda>0. \tag{20}$$

A common feature of gradient based optimization methods building on (20) is the use of the gradient $\nabla^{\theta}\delta\left(\widetilde{F}(\mathscr{M}(\theta),\zeta),\mathscr{P}^{MKT}(\zeta)\right)$, hence its correct and precise computation is crucial for subsequent success. Examples of such algorithms, are Levenberg-Marquardt [Lev44, Mar63], Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [NW06], L-BFGS-B [ZBLN97] and SLSQP [Kra88]. The main advantage of the aforementioned methods is the quick convergence towards condition (19). However, (19) only gives necessary and not sufficient conditions for optimality, hence special care must be taken with non-convex problems.

Remark 6. *Notably, making use of Theorem 2 we use a smooth activation functions in order to guarantee* $\nabla^{\theta}\widetilde{P}\approx\nabla^{\theta}\widetilde{F}$

Gradient-free optimizers

Gradient-free optimization algorithms are gaining popularity due to the increasing number of high dimensional nonlinear, non-differentiable and/or non-convex problems flourishing in many scientific fields such as biology, physics or engineering. As the name suggests, gradient-free algorithms make no C^1 assumption on the objective function. Perhaps, the most well known example is the Simplex based Nelder-Mead [NM65] algorithm. However, there are many other methods such as COBYLA [Pow94] or Differential Evolution [SP97] and we refer the reader to [RS13] for an excellent review on gradient-free methods. The main advantage of these methods is the ability to find global solutions in (2) regardless of the objective function.

In contrast, the main drawback is a higher computational cost compared to gradient methods.

To conclude, we summarise the advantages of each approach in Table VIA.1.

	Gradient-based	**Gradient-free**
Convergence Speed	Very Fast	Slow
Global Solution	Depends on problem	Always
Smooth activation function needed	Yes to apply Theorem 2	No
Accurate gradient approximation needed	Yes	No

Table VIA.1: **Comparison of Gradient vs. Gradient-free methods**.

3 Numerical experiments

In our numerical experiments we demonstrate that the accuracy of the approximation network indeed remains within the accuracy of the Monte Carlo error bounds and proclaimed in the introductory sections' objectives. For this we first compute the benchmark Monte Carlo errors in Figures VIA.4-VIA.5 and compare this with the neural network approximation errors in Figures VIA.6 and VIA.7. For this separation into steps (i) and (ii) to be computationally meaningful, the neural network approximation has to be a reasonably accurate approximation of the true pricing functionals and each functional evaluation (i.e. evaluation an option price for a given price and maturity) should have a considerable speed-up in comparison to the original numerical method. In this section we demonstrate that our network achieves both of these goals.

3.1 Numerical accuracy and speed of the price approximation for vanillas

As mentioned in Section 1 one crucial difference that sets apart this work from direct neural network approaches, as pioneered by Hernandez [Her16], is the separation of (i) the implied volatility approximation function, mapping from parameters of the stochastic volatility model to the implied volatility surface–thereby bypassing the need for expensive Monte-Carlo simulations—and (ii) the calibration procedure, which (after this separation) becomes a simple deterministic optimisation problem. As outlined in Section 1.3 our aim for the Step (**i**) in the two-step training approach is to achieve a considerable speedup per functional evaluation of option prices while maintaining the numerical accuracy of the original pricer. Here we demonstrate how our NN training for Step (**i**) achieves these goals outlined in Section 1.3:

1. Approximation accuracy: here we compare the error of the approximation network error to the error of Monte Carlo evaluations. We compute Monte Carlo prices with $60,000$ paths as reference at the nodes where we compute the implied volatility grid using Algorithm 3.5 in Horvath, Jacquier and Muguruza [HJM17]. In Figures VIA.4 and VIA.5 the approximation accuracy of the Monte Carlo method for the full implied volatility

surface is computed using pointwise relative error with respect to the 95% Monte Carlo confidence interval. Figures VIA.6 and VIA.7 demonstrate that the same approximation accuracy for the neural network is achieved as for the Monte Carlo approximation (i.e. within a few basis points). For reference, the spread on options is around 0.2% in implied volatility terms for the most liquid and those below a year. This translates into 1% relative error for a implied volatility of 20%.

2. Approximation speed: Table VIB.1 shows the CPU computation time per functional evaluation of a full surface under two different models; rBergomi 3 and 1 Factor Bergomi 6 (for a reminder see Section 3.1.1 for details).

	MC Pricing 1F Bergomi Full Surface	MC Pricing rBergomi Full Surface	NN Pricing Full Surface	NN Gradient Full Surface	Speed up NN vs. MC
Piecewise constant forward variance	300,000 μs	500,000 μs	30.9 μs	113 μs	9,000 – 16,000

Table VIA.2: Computational time of pricing map (entire implied volatility surface) and gradients via Neural Network approximation and Monte Carlo (MC). If the forward variance curve is a constant value, then the speed-up is even more pronounced

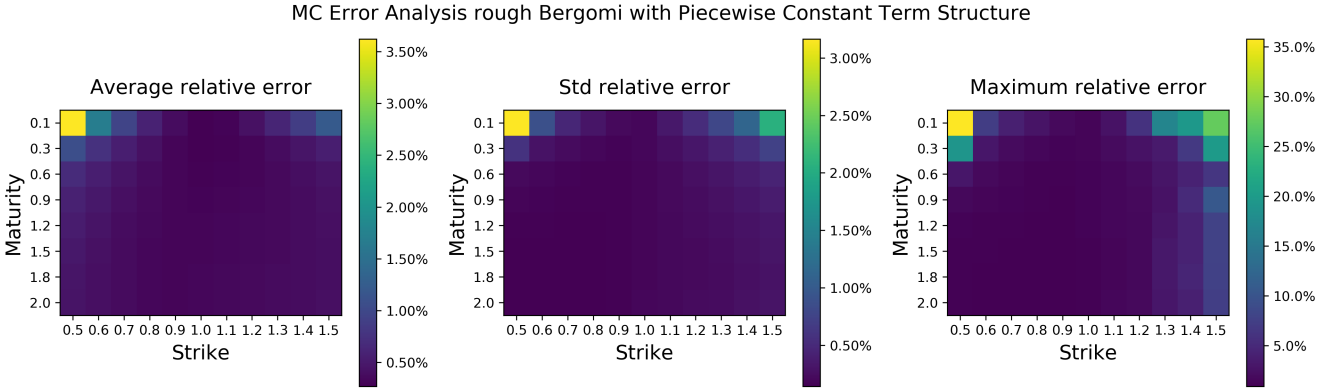


Figure VIA.4: **Error metrics for rough Bergomi prices obtained by Monte-Carlo evaluations.**
As benchmark we recall average relative errors of Monte Carlo prices computed across 80,000 random parameter combinations of the Rough Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right) on implied volatility surfaces in the Rough Bergomi model, computed using 95% confidence intervals.

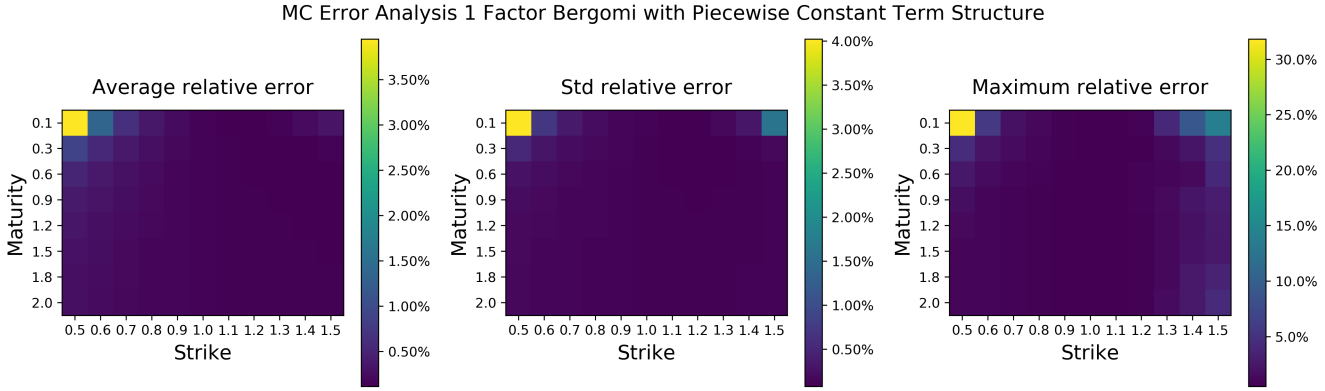


Figure VIA.5: **Error metrics for 1 Factor Bergomi prices obtained by Monte-Carlo evaluations.**
As benchmark we recall average relative errors of Monte Carlo prices computed across 80,000 random parameter combinations of the 1 Factor Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right) on implied volatility surfaces in the 1 Factor Bergomi model, computed using 95% confidence intervals.

3.1.1 Neural network price approximation in (rough) Bergomi models with piecewise constant forward variance curve

We consider a piecewise constant forward variance curve $\xi_0(t) = \sum_{i-1}^{n} \xi_i \mathbf{1}_{\{t_{i-1}<t<t_i\}}$ where $t_0 = 0 < t_1 < ... < t_n$ and $\{t_i\}_i = 1, .., n$ are the option maturity dates ($n = 8$ in our case). This is the modelling approach suggested by Bergomi [Ber15]. We will consider again the rough Bergomi 3 and 1 Factor Bergomi models 6

- Normalized parameters as input and normalised implied volatilities as output
- 4 hidden layers with 30 neurons and *Elu* activation function
- Output layer with *Linear* activation function
- Total number of parameters: 6808
- Train Set: 68,000 and Test Set: 12,000
- Rough Bergomi sample:$(\xi_0, \nu, \rho, H) \in \mathcal{U}[0.01, 0.16]^8 \times \mathcal{U}[0.5, 4.0] \times \mathcal{U}[-0.95, -0.1] \times \mathcal{U}[0.025, 0.5]$
- 1 Factor Bergomi sample:$(\xi_0, \nu, \rho, \beta) \in \mathcal{U}[0.01, 0.16]^8 \times \mathcal{U}[0.5, 4.0] \times \mathcal{U}[-0.95, -0.1] \times \mathcal{U}[0, 10]$
- strikes=$\{0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5\}$
- maturities=$\{0.1, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.0\}$

- Training data samples of Input-Output pares are computed using Algorithm 3.5 in Horvath, Jacquier and Muguruza [HJM17] with 60,000 sample paths and the spot martingale condition i.e. $\mathbb{E}[S_t] = S_0,\ t \geq 0$ as control variate.

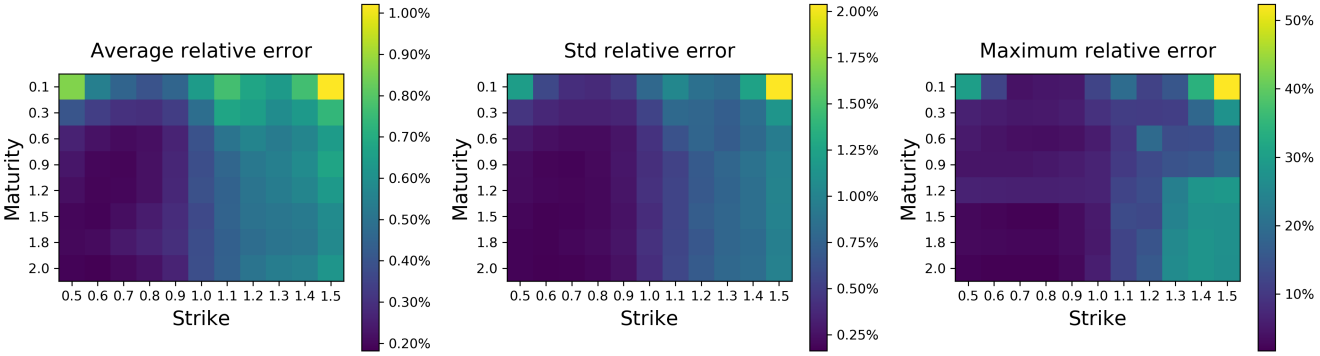


Figure VIA.6: **Error metrics for rough Bergomi prices obtained by neural network approximation.**
We compare surface relative errors of the neural network approximator against the Monte Carlo benchmark across all training data (68,000 random parameter combinations)in the rough Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right).

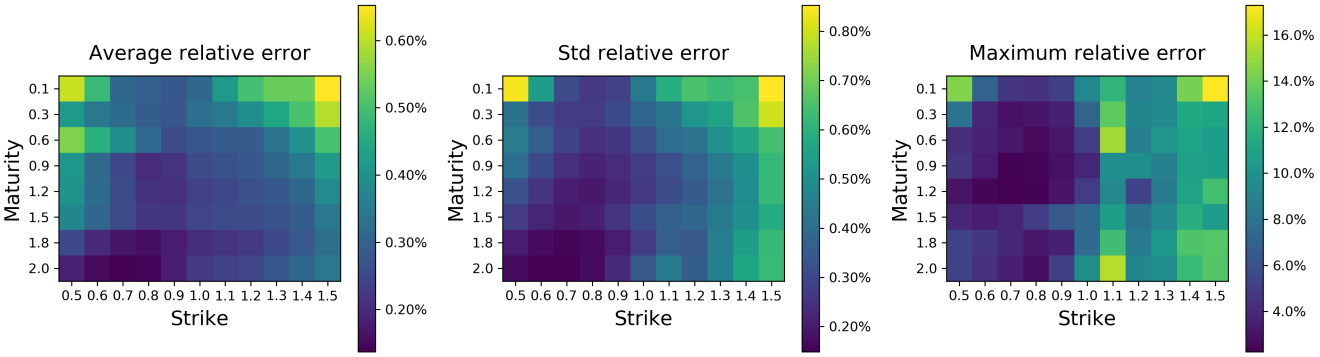


Figure VIA.7: **Error metrics for 1 Factor Bergomi prices obtained by neural network approximation.**
We compare surface relative errors of the neural network approximator against the Monte Carlo benchmark across all training data (68,000 random parameter combinations)in the 1 Factor Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right).

Figures VIA.6 and VIA.7 show that the average (across all parameter combinations) relative error between neural network and Monte Carlo approximations is far less than 0.5% consistently

(left image in Figures VIA.6 and VIA.7) with a standard deviation of less than 1% (middle image in Figures VIA.6 and VIA.7). The maximum relative error goes as far as 25%. We conclude that the methodology generalises adequately to the case of non-constant forward variances, by showing the same error behaviour.

3.2 Calibration speed and accuracy for implied volatility surfaces

Figure VIA.8 reports average calibration times on test set for different parameter combinations on each of the models analysed. We conclude that gradient-based optimizers outperform (in terms of speed) gradient-free ones. Moreover, in Figure VIA.8 one observes that computational times in gradient-free methods are heavily affected by the dimension of the parameter space, i.e. flat forward variances are much quicker to calibrate than piecewise constant ones. We find that Lavenberg-Marquardt is the most balanced optimizer in terms of speed/convergence and we choose to perform further experiments with this optimizer. The reader is encouraged to keep in mind that a wide range of optimizers is available for the calibration and the optimal selection of one is left for future research.

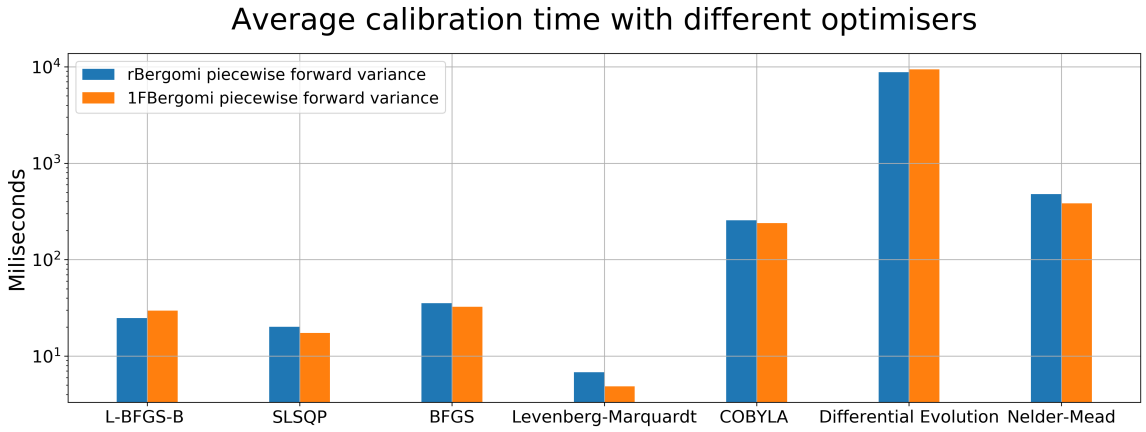


Figure VIA.8: **Average calibrations times for all models using a range of optimizers.**

In order to assess the accuracy, we report the calibrated model parameters $\widehat{\theta}$ compared to the synthetically generated data with the set of parameters $\overline{\theta}$ that was chosen for the generation of our synthetic data. We measure the accuracy of the calibration via parameter relative error i.e.

$$E_R(\widehat{\theta}) = \frac{|\widehat{\theta} - \overline{\theta}|}{|\overline{\theta}|}$$

as well as the root mean square error (RMSE) with respect to the original surface i.e.

$$\text{RMSE}(\widehat{\theta}) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(\widetilde{F}(\widehat{\theta})_{ij} - \sigma_{BS}^{MKT}(T_i, k_j))^2}.$$

Therefore, on one hand a measure of good calibration is a small RMSE. On the other hand, a measure of parameter sensitivity on a given model is the combined result of RMSE and parameter relative error.

3.2.1 A calibration experiment with simulated data in (rough) Bergomi models with piecewise constant forward variances

We consider the rough Bergomi model (3) and the Bergomi model (6) with a piecewise constant term-structure of forward variances. Figures VIA.9 and VIA.10 show that the 99% quantile of the RMSE is below 1% and shows that the Neural Network approach generalises properly to the piecewise constant forward variance. Again, we find that the largest relative errors per parameter are concentrated around 0, consequence of using the relative error as measure. This suggests a successful generalisation to general forward variances, which to our knowledge has not been addressed before by means of neural networks or machine learning techniques.

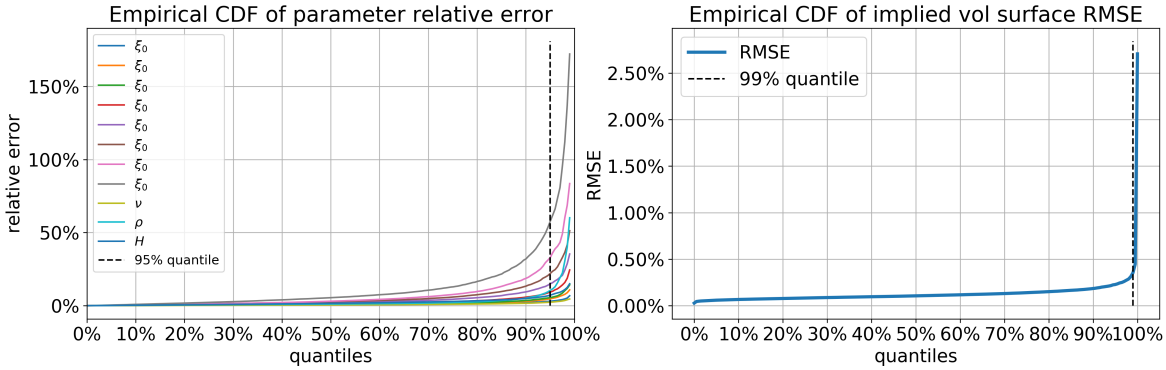


Figure VIA.9: **Synthetic parameter and implied volatility surface error of the neural network based calibration for the rough Bergomi model.**
Cumulative Distribution Function (CDF) of Rough Bergomi parameter relative errors (left) and RMSE (right) after Levengerg-Marquardt calibration across test set random parameter combinations.

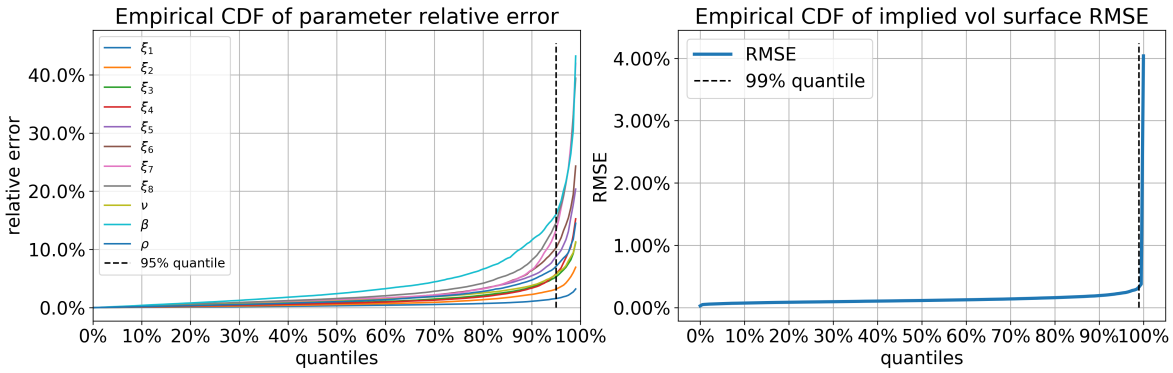


Figure VIA.10: **Synthetic parameter and implied volatility surface error of the neural network based calibration for the 1 Factor Bergomi model.**
Cumulative Distribution Function (CDF) of 1 Factor Bergomi parameter relative errors (left) and RMSE (right) after Levengerg-Marquardt calibration across test set random parameter combinations.

3.2.2 Calibration in the rough Bergomi model with historical data

As previously mentioned, the natural use of neural network approximators is the model calibration to historical data. We discussed that as along as the approximation is accurate, the calibration task should be performed within the given tolerance. Furthermore, one should expect such tolerance to be aligned with the neural network accuracy obtained in both training and test sets.

In this section we will perform a historical calibration using the neural network approximation and compare it with that of the brute force monte carlo calibration. Precisely we seek to solve the following optimisation problem for the rough Bergomi model

$$\hat{\theta^{rBergomi}} := \underset{\theta^{rBergomi} \in \Theta^{rBergomi}}{\operatorname{argmin}} \sum_{i=1}^{5} \sum_{j=1}^{9} (\widetilde{F}(\theta)_{ij} - \sigma_{BS}^{MKT}(T_i, k_j))^2.$$

where $\theta^{rBergomi} = (\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \nu, \rho, H)$ and $\Theta^{rBergomi} = [0.01, 0.25]^5 \times [0.5, 4] \times [-1, 0] \times [0.025, 0.5]$. As for the time grid we choose

$$(T_1, T_2, T_3, T_4, T_5) := \frac{1}{12} \times (1, 3, 6, 9, 12)$$

and for the strike grid

$$k_i := 0.85 + (i-1) \times 0.05 \quad \text{for } i = 1, ..., 9.$$

We consider SPX market smiles between 01/01/2010 and 18/03/2019 on the pre-specified time and strike grid. Figure VIA.11 shows the historical evolution of rough Bergomi parameters calibrated to SPX using the neural network price. In particular we note that $H < \frac{1}{2}$ as

previously discussed in many academic papers [AGLM18, ALV07, BFG16, BFG$^+$20, BFG$^+$19, BLP17, EER19, Fuk11, GJR18, JMM18, HJL19, JPS18], moreover we may confirm that under $\mathbb{Q}$, $H \in [0.1, 0.15]$ as found in Gatheral, Jaisson and Rosenbaum [GJR18] under $\mathbb{P}$. Figure VIA.12, benchmarks the NN optimal fit using Levenberg-Marquardt and Differential Evolution against a brute force MC calibration via Levenberg-Marquardt. Again, we find that the discrepancy between both is below 0.2% most of the time and conclude that the Differential Evolution algorithm does outperform the Levenberg-Marquardt. This in turn, suggests that the neural network might not be precise enough on first order derivatives. This observation, is left as an open question for further research. Perhaps surprisingly, we sometimes obtain a better fit using the neural network than the MC pricing itself. This could be caused by the fact that gradients in the neural network are exact, whereas when using MC brute force calibration we resort to finite differences to approximate gradients.

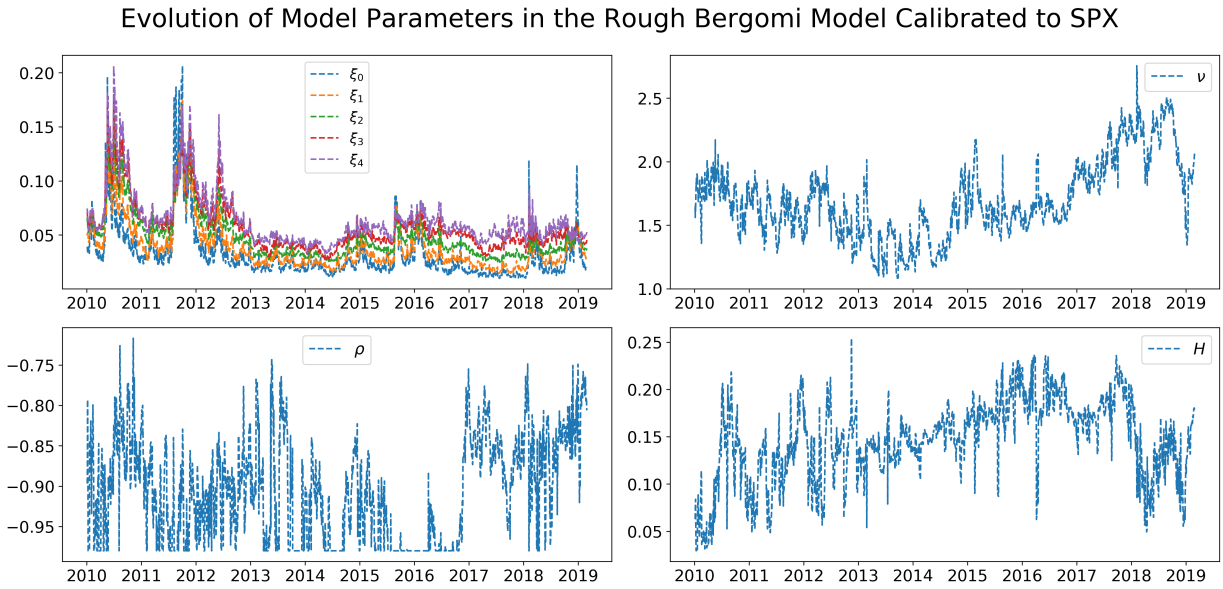


Figure VIA.11: **Historical Evolution of parameters in the rough Bergomi model with a piecewise constant forward variance term structure calibrated on SPX.**

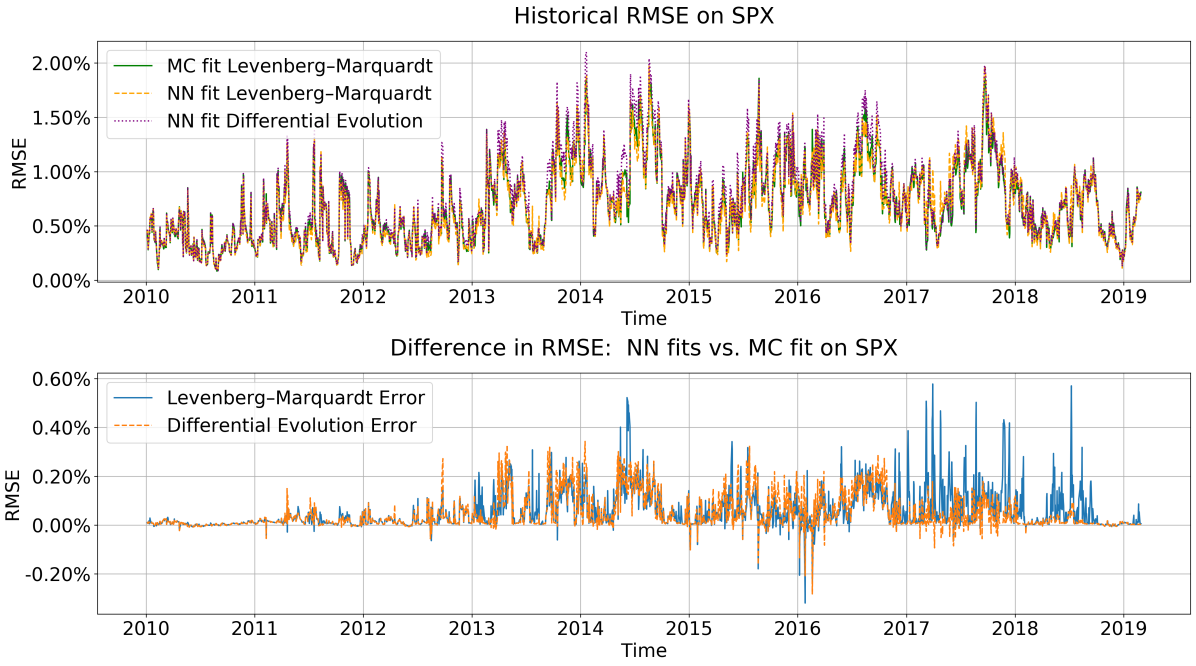


Figure VIA.12: **Comparison of quality of fit of different calibration approaches.** The image above compares historical RMSE obtained by the neural network best fit via Levenberg-Marquardt (dashed orange line) and Differential Evolution (dotted purple line) against the brute force MC calibration (green line) via Levenberg-Marquardt. Picture below shows the difference against MC brute force calibration.

3.3 Numerical experiments with barrier options in the rough Bergomi model

In this section we show that our methodology can be easily extended to exotic options. To do so we test our image-based approach on digital barrier options. We follow the same architecture and experimental design described in Section 3.1 for the rough Bergomi model. As described in Section 2.1.2 we adapt the objective function to the payoffs given in (17) and (18) and replace the strike grid by a barrier level grid. Figure VIA.13 confirm the accuracy of the neural network approximation with average absolute errors of less than 10bps with standard deviation of 10bps.

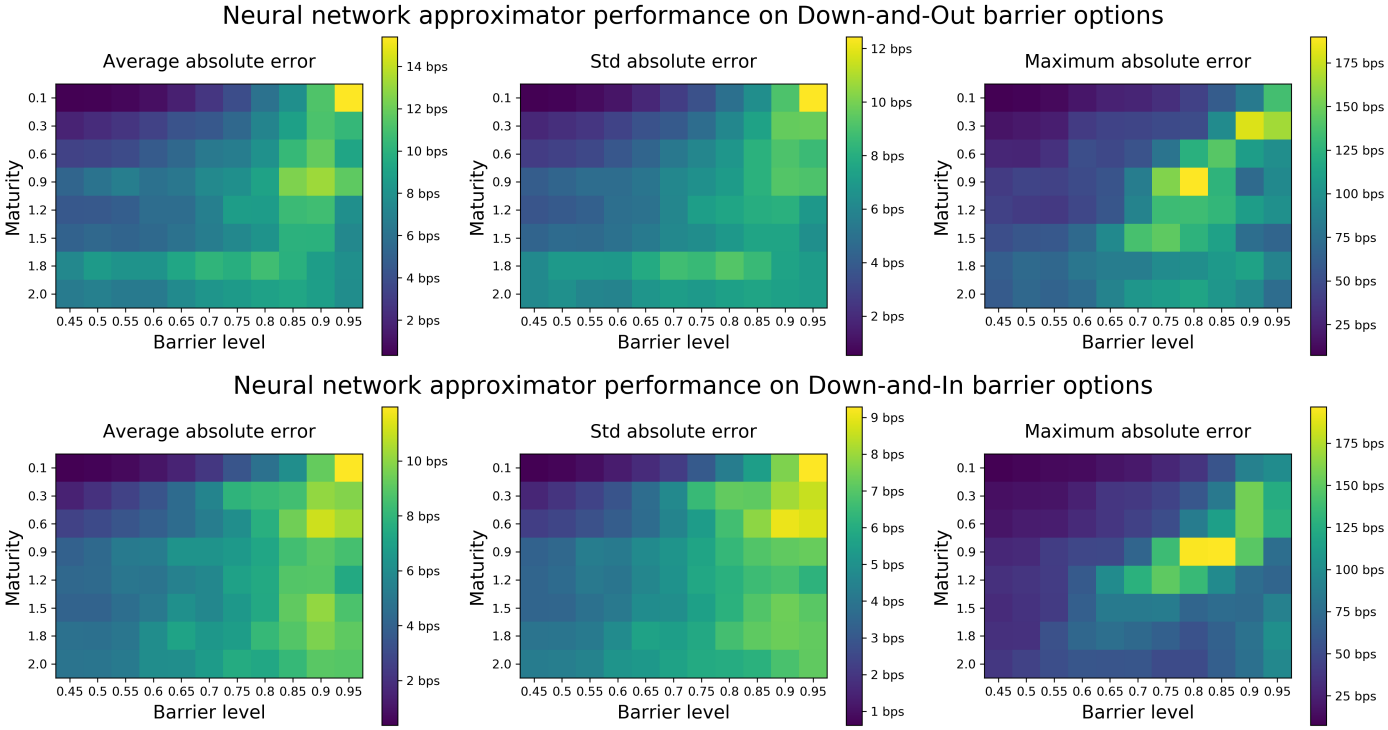


Figure VIA.13: **Error metrics for neural network approximation of exotic derivatives' prices.**
Picture above: Down-and-Out neural network absolute error analysis on test set. Picture below: Down-and-In neural network absolute error analysis on test set.

Conclusions and outlook: "best-fit" models

To sum up, neural networks have the potential to efficiently approximate complex functions, which are difficult to represent and time-consuming to evaluate by other means. Using deep neural networks, as we will do here, to approximate the pricing map (or equivalently the implied volatility mapping) from parameters of traditional models to shapes of the implied

volatility surface represented by grid of implied volatility values speeds up each functional evaluation, while maintaining control over reliability and interpretability of network outputs. The implicit grid based approach that we advocate here, also allows further applications that opens up further landscapes for financial modelling.

Potential applications and outlook towards mixture of "expert" models: In the previous sections we set up a powerful approximation method to closely approximate implied volatilities under different stochastic models and highlighted that the choice of the objective function (evaluation of the surface on a grid, inspired by pixels of an image) was crucial for the performance of the network. Now we are interested in the inverse task and ask whether a neural network—trained by this objective function to multiple stochastic models simultaneously—can identify which stochastic model a given set of data comes from. By doing so, potential applications we have in mind are twofold:

- Ultimately we are interested in which model (or what mixture of existing stochastic models) best describes the market.
- From a more academic and less practical perspective, we are interested whether and to what extent is it possible to"translate" parameters of one stochastic model to parameters of another.

We conduct a further, preliminary experiment as a proof of concept in the classification setting. We train a further neural network to identify which of three given stochastic volatility model generated a given implied volatility surface.

Training procedure: Implied volatility surfaces in this experiment were generated by the Heston, Bergomi and rough Bergomi models (see Section 1.1 for a reminder). For each volatility surface, a "flag" was assigned corresponding to the model (eg: 1 for Heston, 2 for Bergomi and 3 for rough Bergomi). The training set thus consists of surfaces of the form: $(\Sigma_{BS}^{\mathcal{M}(\theta)}, I)$, where $\mathcal{M}$ is one of the three models $\mathcal{M}^{\text{Heston}}$, $\mathcal{M}^{\text{Bergomi}}$, $\mathcal{M}^{\text{rBergomi}}$, θ an admissible combination of parameters for that model (thus in Θ^{Heston}, Θ^{Bergomi} or Θ^{rBergomi}) and I the flag identifying the model which generated the surface ($I = 1$ if $\mathcal{M} = \mathcal{M}^{\text{Heston}}$, $I = 2$ if $\mathcal{M} = \mathcal{M}^{\text{Bergomi}}$ and $I = 3$ if $\mathcal{M} = \mathcal{M}^{\text{rBergomi}}$).

We define a mixture of these surfaces as $\Sigma^{\mathcal{M}^{\text{Mixture}((a,b,c))}} := a\Sigma^{\mathcal{M}^{\text{Heston}}} + b\Sigma^{\mathcal{M}^{\text{Bergomi}}} + c\Sigma^{\mathcal{M}^{\text{rough Bergomi}}}$, where $a, b, c \geq 0$ and $a + b + c = 1$. So far the training is suitable for recognition of a single model surface (either $a = 0, b = 0, c = 1$, $a = 0, b = 1, c = 0$ or $a = 1, b = 0, c = 0$). To generalise this to mixtures, we randomly select surfaces (one from each model) and compute the mixture surface $\Sigma^{\mathcal{M}^{\text{Mixture}((a,b,c))}} = a\Sigma^{\mathcal{M}^{\text{Heston}}} + b\Sigma^{\mathcal{M}^{\text{Bergomi}}} + c\Sigma^{\mathcal{M}^{\text{rough Bergomi}}}$. The corresponding probabilities are $(a, b, c = 1 - a - b)$.

Network Architecture: The classifier is a small, fully connected feedforward network for the same reasons as those outlined in section 3.1. The network is composed of 2 hidden layers (of 100 and 50 output nodes respectively) with exponentially linear activation functions

and an output layer with a softmax activation function. Thus, the output of the network represents the probabilities of a given surface belonging to a particular model. We used stochastic gradient descent with 20 epochs to minimize cross-entropy (the cross-entropy of two discrete distributions (p, q) with K possible distinct values is $H(p, q) := -\sum_{1 \le i \le K} p_i \log q_i$.).

A numerical experiment on model recognition: We report on one of many experiments here as a proof of concept: We test the method on mixtures of rough Bergomi and Heston surfaces (hence setting $b = 0$ in the training). To vary the type of mixtures generated, we chose $a \in \{0, 0.1, \cdots, 0.9, 1\}$. For each a, the mixture surface is computed as the convex combination of a randomly chosen surface from the rough Bergomi and the Heston model, and repeated 20 times. The training set has 320,000 surfaces. To further test the robustness of the model, validation surfaces were generated using a finer grid of mixture parameters: $a \in \{0, 0.05, \cdots, 0.95, 1\}$. In total, the validation set is made up of 105,000 surfaces.

We report the classifiers' effectiveness and comment on the results in Figure VIA.14.

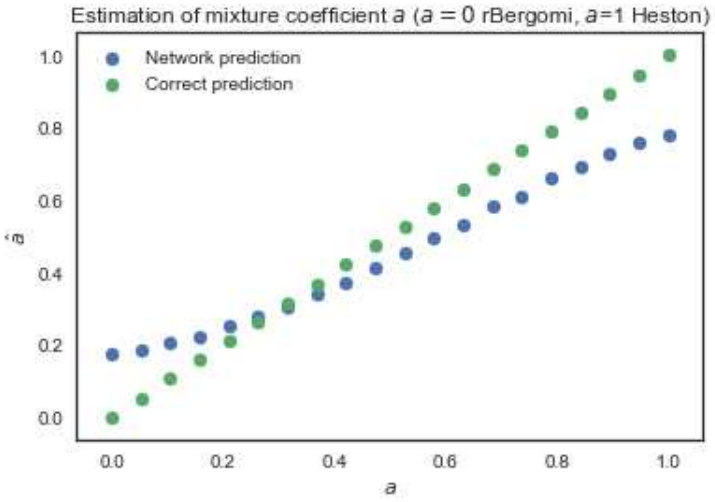


Figure VIA.14: **Error of the neural network classifier depending on the mixture coefficient** a.
Each point of the plot corresponds to the average estimated coefficient by the neural network for all mixture surfaces with a given a. For example, for $a = 0$, the surfaces are generated from the rough Bergomi model. For each parameter combination from those surfaces, we compute the predicted mixture coefficient and average all of them over the validation set to report $\hat{a}$. The network never sets the mixture coefficient very close to 1 or 0, attributing the surface to one specific model. This may be explained using Bayesian reasoning.

CHAPTER VIB

On deep calibration of rough volatility models

Abstract

Techniques from deep learning play a more and more important role for the important task of calibration of financial models. The pioneering paper by Hernandez [Risk, 2017] was a catalyst for resurfacing interest in research in this area. In this chapter we advocate an alternative (two-step) approach using deep learning techniques solely to learn the pricing map – from model parameters to prices or implied volatilities. Having a fast and accurate neural-network-based approximating pricing map (first step), we can then (second step) use traditional model calibration algorithms. In this work we showcase a direct comparison of different potential approaches to the learning stage and present algorithms that provide a sufficient accuracy for practical use. In particular provide the first and simplest neural network-based calibration methods for rough volatility models with a constant intitial volatility for which calibration can be done on the fly. We demonstrate the method via a hands-on calibration engine on the rough Bergomi model, for which classical calibration techniques are difficult to apply due to the high cost of all known numerical pricing methods. Furthermore, we display and compare different types of sampling and training methods and elaborate on their advantages under different objectives. As a further application we use the fast pricing method for a Bayesian analysis of the calibrated model.

From:
On deep calibration of (rough) stochastic volatility models
C. Bayer, B. Horvath, A. Muguruza, B. Stemper, **M. Tomas**

Introduction

Almost half a century after its publication, the option pricing model by Black, Scholes and Merton remains one of the most popular analytical frameworks for pricing and hedging European options in financial markets. A part of its success stems from the availability of explicit and hence instantaneously computable closed formulas for both theoretical option prices and option price sensitivities to input parameters (*Greeks*), albeit at the expense of

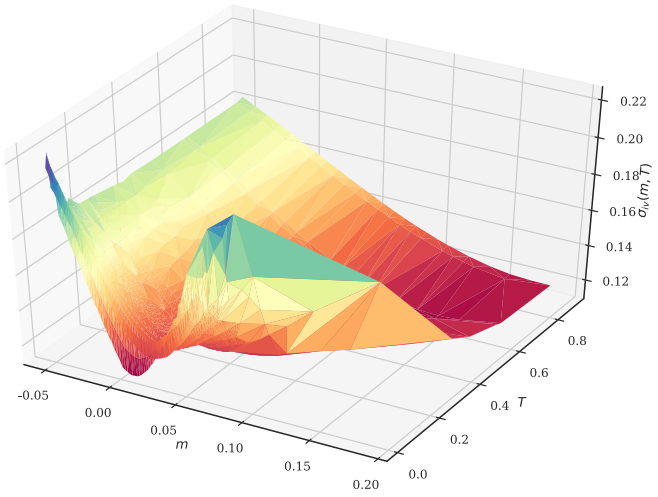


Figure VIB.1: **SPX Market Implied Volatility surface on 15th February 2018.**
IVs have been inverted from SPX Weekly European plain vanilla call mid prices and the interpolation is a (non-arbitrage-free) Delaunay triangulation. Axes denote log-moneyness $m = \log(K/S_0)$ for strike K and spot S_0, time to maturity T in years and market implied volatility $\sigma_{\text{iv}}(m, T)$.

assuming that *volatility* – the standard deviation of log returns of the underlying asset price – is deterministic and constant. Still, in financial practice, the Black-Scholes model is considered as the sophisticated transform between option prices and Black-Scholes (BS) *implied volatility (IV)* σ_{iv} where the latter is defined as the constant volatility input needed in the BS formula to match a given (market) price. It is a well-known fact that in empirical IV surfaces obtained by transforming market prices of European options to IVs, it can be observed that IVs vary across moneyness and maturities, exhibiting well-known smiles and at-the-money (ATM) skews and thereby contradicting the flat surface predicted by Black-Scholes (Figure VIB.1). In particular, Bayer, Friz, and Gatheral [BFG16] report empirical at-the-money volatility skews of the form

$$\left| \frac{\partial}{\partial m} \sigma_{\text{iv}}(m, T) \right| \sim T^{-0.4}, \quad T \to 0, \tag{1}$$

for log moneyness m and time to maturity T.
While plain vanilla European call and put options often show enough liquidity to be marked-to-market, pricing and hedging path-dependent options (so-called *exotics*) necessitates an option pricing model that prices European options *consistently* with respect to observed market IVs across moneyness and maturities. In other words, it should parsimoniously capture stylized facts of empirical IV surfaces. In the past, in order to address the shortcomings of standard models such as Black-Scholes and to incorporate the stochastic nature of volatility itself, popular bivariate diffusion models such as SABR [HKLW02] or Heston [Hes93] have been developed to capture *some* important stylized facts. However, according to Gatheral [Gat11], diffusive stochastic volatility models in general fail to recover the exploding power-law nature

(1) of the volatility skew as time to maturity goes to 0 and instead predict a constant behaviour.

Sparked by the seminal work of [ALV07, Fuk11, GJR18], we have since seen a shift from classical diffusive modeling towards so-called rough stochastic volatility models. They may be defined as a class of *continuous-path* stochastic volatility models where the instantaneous volatility is driven by a stochastic process with Hölder regularity smaller than Brownian Motion, typically modeled by a fractional Brownian Motion with Hurst parameter $H < \frac{1}{2}$. The evidence for this paradigm shift is by now overwhelming, both under the physical measure where time series analysis suggests that log realized volatility has Hölder regularity in the order of ≈ 0.1 [BLP17, GJR18] and also under the pricing measure where the empirically observed power-law behaviour of the volatility skew near zero may be reproduced in the model [ALV07, BFG16, BFG+19, Fuk11]. Serious computational and mathematical challenges arise from the non-Markovianity of fractional Brownian motion, effectively forcing researchers to resort to asymptotic expansions [BFG+19, FZ17] in limiting regimes or (variance-reduced) Monte Carlo schemes [BFG+20, BFG16, HJM17, MP18] to compute fair option prices. This poses considerable bottlenecks for calibration of rough volatility models for practical purposes. One contribution of this work is to provide and explore different neural network based solutions to the task of fast calibration of rough volatility models.

Remark 1. *State of the art models usually super-impose a local volatility over a stochastic volatility backbone model, in order to combine realistic dynamics – as achieved with stochastic volatility – with exact fit of traded vanilla options. In this framework, calibration of a rough volatility backbone model can be seen as the first step of the total calibration procedure, which is then complemented by a particle method, see [Mug19] or [GHL12].*

The solution we provide here is demonstrated on the rough Bergomi model but due to the nature of neural network approximations (as opposed to static polynomial approximations) it is fundamentally model agnostic and it consistently[1] carries over to other rough volatility models (of the same complexity) and to classical stochastic volatility models, which are by nature simpler to approximate.

The “need for speed" is by no means limited to rough volatility models, although our initial motivation was indeed the rough Bergomi model. Parallel to this work, Ferguson and Green address in [FG18, Section 1.1] the ongoing struggle for faster pricing algorithms for more and more complex products and propose a deep learning approach to pricing basket options in a lognormal setting to achieve considerable speed-ups over Monte Carlo pricers. High dimensional problems as in [FG18] are one useful applications of the speedup resulting from this methodology. But it can also enable us to speed up more involved numerical methods for benchmark stochastic volatility models: multiple integrals [AKS19], Monte Carlo-type methods

[1] By consistency we mean here that the proposed network (with the same architecture) can be trained on different models consistently without further modifications and yield satisfactory results irrespective of the chosen model for training. Our numerical experiments show that for the calibration of classical stochastic volatility models (SABR, Heston) a simpler network architecture is sufficient, while rough volatility models require a more nuanced network design.

[LGO17a] or Finite Element Methods [HR18] for the SABR model can thus compete in speed with the original SABR expansion formula [HKLW02], by pre-learning them through the DNN.

In related contexts deep BSDE solvers have been used to replace Monte Carlo methods for solving Backward Stochastic Differential Equations in high dimension [HJW18, HL17, VSS18] which can arise from pricing problem. Other authors used computational speedups provided by neural networks in the context of computationally expensive valuation adjustments [Gre15, HL17].

The work we present here is very much in the spirit of the pioneering work of Avellaneda, Carelli and Stella [MA99]. Our focus in this work is on model calibration of stochastic volatiliy models and we propose computationally efficient and ready-to-use algorithms that can be applied to a variety of settings. Bearing in mind that deep neural network solutions are often challenged by concerns of generalisation and “black-box-solutions", our goal is to limit the application of neural networks to parts of the calibration process that we can control and validate. As a first step, we identify the parts of the calibration process that are mainly responsible for the prevailing calibration bottlenecks, which we will replace by a deep neural network.

In this context, we distinguish two kinds of approaches. The first, pioneered by Hernandez [Her16], seeks to learn the mapping from implied volatility surfaces to model parameters (inverse problem) directly. In [Her16], Hernandez proposes to use a neural network to learn the complete calibration routine taking market data as inputs and returning calibrated model parameters, and calibrates the popular short rate model of Hull and White [HW90] to market data in numerical experiments. In Section 2.1 we describe this approach in more detail and perform a similar calibration experiment with the Heston Model.

In the rest of this chapter, we will refer to it as the *one-step* approach. In a second strand of research neural networks have been applied not directly to calibration problems, but simply to obtain an approximated representation of derivative valuations, i.e. of option pricing maps: For example Hutchinson, Lo and Poggio [HLP94] such as Culkin and Das [CD17] applied neural networks to learn the Black-Scholes formula and McGhee demonstrates in [McG18] a neural networks representation of the lognormal SABR model. In this chapter we explore the advantages of shaping this second strand of research into a building block of a single *two-step approach*[2].

The *two-step* approach, which we highlight in this chapter, first approximates the pricing map, (denoted, by φ from model parameters to option prices) by a neural network (Step **(i)**) before calibrating the model, (via traditional calibration algorithms applied to the approximate pricing map φ_{NN}) to market data (Step **(ii)**). Thereby we optimally leverage the capability of neural networks to approximate functions which are only implicitly available through

[2]Later works of Liu et al [LBGO19] further develop this line of research by developing sophisticated methods to speed up the training process for Markovian stochastic volatility models.

input-output pairs $\{(x_i, \phi(x_i))\}_{i=1}^N$, by training a fully-connected neural network on specifically tailored, synthetically generated training data to learn an approximative representation φ_{NN} of the pricing functional φ. Details of this approach and its benefits are further explained in Section 2.

In particular, in Section 2.2 we compare two network architectures and sampling methods according to different modelling objectives. Among these, the grid-based approach is particularly designed for applicability and efficiency in every day calibration practice. The novelty of our grid-based approach will allow us to tackle the calibration problem with a remarkably small neural network (3 layers and 30 neurons each), which to the best of our knowledge is the smallest network in the literature to successfully solve the calibration/pricing task. Moreover, the architecture of the problem easily permits that the proposed method can run on a standard PC (with or without GPU) and it is not necessary to use server side resources for computations. This in turn, opens the door to its practical implementation in the financial industry without the need to update current hardware systems.

The overall benefits of the *two-step approach* are plentiful:

- First, evaluations of φ_{NN} amount to cheap and almost instantaneous forward runs of a pre-trained network. Second, automatic differentiation of φ_{NN} with respect to the model parameters returns fast and accurate approximations of the Jacobians needed for the LM calibration routine. Used together, they allow for the efficient calibration of *any* (rough) stochastic volatility model including *rough Bergomi*.

- The *two-step* approach also has overwhelming risk management benefits. Firstly, we can understand and interpret the output of our neural network and therefore test the output as a function of model parameters against traditional numerical methods. (Indeed, the output values correspond to option prices in the model under consideration.) The second overwhelming advantage is that existing risk management libraries of models remain valid with minimal modification. The neural network is only used as a computational enhancement of models, and therefore, the knowledge and intuition gathered in many years of experience with traditional models remains useful.

- The training becomes more robust (with respect to generalisation errors on unseen data). Additionally, the trained network is independent from market data, and, in particular, from changing market environments.

- We can train the network to synthetic data – model prices or implied volatilities computed by any adequate numerical method. In particular, we can easily provide as large training sets as desired.

Both generating the synthetic data set as well as the actual neural network training are expensive in time and computing resource requirements, yet they only have to be performed a single time. Trained networks may then be quickly and efficiently saved, moved and deployed. We demonstrate this first advantage in a further application: a Bayesian calibration experiment,

which is facilitated by our ability to nearly instantaneously call functional evaluations of option prices in a given model. To quantify the uncertainty about model parameter estimates obtained by calibrating with φ_{NN}, we infer model parameters in a Bayesian spirit from (i) a synthetically generated IV surface and (ii) SPX market IV data. In both experiments, a simple (weighted) Bayesian nonlinear regression returns a (joint) posterior distribution over model parameters that (1) correctly identifies sensible model parameter regions and (2) places its peak at or close to the true (in the case of the synthetic IV) or previously reported [BFG16] (in the case of the SPX surface) model parameter values. Both experiments thus confirm the idea that φ_{NN} is sufficiently accurate for calibration.

The chapter is organised as follows: In Section 1 we present an abstract point of view on model calibration in finance. In Section 2 we give an overview of applications of techniques from deep learning to model calibration. We also introduce our own framework and discuss possible advantages and disadvantages as compared to other approaches. In Section 3 we focus on the concrete implementation of our methods, both for the learning and for the actual calibration stage. Numerical experiments are then presented in Section 4. In addition, we also apply the network in a Bayesian approach. The Appendix VIB.A contains a numerical comparison with an alternative deep learning approach to calibration.

1 Model calibration revisited

Calibration describes the procedure of tuning model parameters to fit a model surface to an empirical implied volatility surface obtained by transforming liquid European option market prices to Black-Scholes implied volatilities. A mathematically convenient approach consists of minimizing the weighted squared differences between market and model implied volatlities of $N \in \mathbb{N}$ plain vanilla European options.

Suppose that a model is parametrized by a set of parameters Θ, i.e., by $\theta \in \Theta$. We refer to Example VIB.1 for a concrete example. Furthermore, we consider options parametrized by a parameter $\zeta \in Z$. E.g., for put and call options we generally have $\zeta = (T, k)$, the option's maturity and log-moneyness. There might be further parameters which are needed to compute prices but can be observed on the market and, hence, do not need to be calibrated. For instance, the spot price of the underlying, the interest rate, or the forward variance curve in Bergomi-type models (see [Ber15]) falls under this type. For this quick overview, we ignore this category. We introduce the ***pricing map***

$$(\theta, \zeta) \mapsto P(\theta, \zeta), \tag{2}$$

the price of an option with parameters ζ in the model with parameters θ. It is this map (2) that we will learn by a neural network in the following sections either directly, for some (simple or exotic) payoff functions ζ, or indirectly, by learning the implied volatility map

$$\theta \mapsto \sigma(\theta, K, T), \tag{3}$$

of vanilla payoffs $\zeta(\cdot, K, T) \equiv (\,\cdot\,|_T - K)_+$ for some $T, K > 0$. Financial practice often prefers to work with implied volatilities rather than option prices, and we will also do so in the numerical parts of this chapter containing vanilla contracts. For the purpose of this introduction, any mentioning of a *price* may be, mutatis mutandis, replaced by the corresponding implied volatility.

Observations of market prices $\mathscr{P}(\zeta)$ for options are parametrized by ζ for a (finite) subset $\zeta \in Z' \subset Z$ of all possible option parameters.
When the model is *calibrated*, a model parameter θ is identified which minimizes a distance δ between model prices $(P(\theta,\zeta))_{\zeta\in Z'}$ and observed market prices $(\mathscr{P}(\zeta))_{\zeta\in Z'}$, i.e.,

$$\widehat{\theta} = \operatorname*{argmin}_{\theta\in\Theta} \delta\left((P(\theta,\zeta))_{\zeta\in Z'}, (\mathscr{P}(\zeta))_{\zeta\in Z'}\right). \tag{4}$$

Hence, the faster each model price $(P(\theta,\zeta))$ can be computed, the faster the calibration routine.

The most common choice of a distance function δ is a suitably weighted least squares function, i.e.,

$$\widehat{\theta} = \operatorname*{argmin}_{\theta\in\Theta} \sum_{\zeta\in Z'} w_\zeta \left(P(\theta,\zeta) - \mathscr{P}(\zeta)\right)^2.$$

Here, the weights w_ζ can be chosen in order to reflect importance of an option at ζ and the reliability of the market observation $\mathscr{P}(\zeta)$. For instance, a reasonable choice might be the inverse of the bid-ask spread (see [Con10] for a motivation), which puts low weight on prices of illiquid options.

As long as the number of model parameters is smaller than the number $|Z'|$ of calibration instruments, the calibration problem is an example of an overdetermined non-linear least squares problem, usually solved numerically using iterative solvers such as the de-facto standard Levenberg-Marquardt (LM) algorithm [Lev44, Mar63]. Let $J = J(\theta)$ denote the Jacobian of the map $\theta \mapsto (P(\theta,\zeta)_{\zeta\in Z'}$ and let

$$R(\theta) := (P(\theta,\zeta) - \mathscr{P}(\zeta))_{\zeta\in Z'}$$

denote the residual, then the Levenberg-Marquart algorithm iteratively computes increments $\Delta\theta_k := \theta_{k+1} - \theta_k$ by solving

$$\left[J^T W J + \lambda I\right] \Delta\theta_k = J^T W R \tag{5}$$

where I denotes the identity matrix, $W = \operatorname{diag}\left(w_\zeta\right)$, and $\lambda \in \mathbb{R}$.
It is hence necessary that the *normal equations* (5) be quickly and accurately solved for the iterative step $\Delta\theta_k$. In a general (rough) stochastic volatility setting this is problematic: The true implied volatility map as well as its Jacobian J are unknown in analytical form. In the absence of an analytical expression for $\Delta\theta_k$, an immediate remedy is:

(I) Replace the (theoretical) true pricing (or implied volatility) map P by an efficient numerical approximation $\tilde{P}$ such as Monte Carlo, Fourier pricing.

(II) Apply finite-differences to $\tilde{P}$ to compute an approximate Jacobian $\tilde{J}$.

Algorithm 1: Levenberg-Marquart calibration

Input: Implied vol map $\widetilde{P}$ and its Jacobian $\widetilde{J}$, market quotes $\mathscr{P}$

Parameters: Initialise Lagrange multiplier $\lambda > 0$, maximum number of iterations $n_{\max}$, minimum tolerance of step norm $\varepsilon_{\min}$, bounds $0 < \beta_0 < \beta_1 < 1$

Result: Calibrated model parameters $\theta^\star$

initialize model parameters $\theta = \theta_0$ and step counter $n = 0$;
compute $\widetilde{R}(\theta) = \widetilde{P}(\theta) - \mathscr{P}$ and $\widetilde{J}(\theta)$ and solve normal equations (5) for $\Delta\theta$;
while $n < n_{max}$ *and* $\int_0^\infty \Delta\theta_2 > \varepsilon$ **do**
  compute relative improvement $c_\theta = \frac{\int_0^\infty \widetilde{R}(\theta)_2 - \int_0^\infty \widetilde{R}(\theta+\Delta\theta)_2}{\int_0^\infty \widetilde{R}(\theta)_2 - \int_0^\infty \widetilde{R}(\theta)+\widetilde{J}(\mu)\Delta\theta_2}$ with respect to predicted improvement under linear model;
  if $c_\theta \le \beta_0$ **then** reject $\Delta\theta$, set $\lambda = 2\lambda$;
  if $c_\theta \ge \beta_1$ **then** accept $\Delta\theta$, set $\theta = \theta + \Delta\theta$ and $\lambda = \frac{1}{2}\lambda$;
  compute $\widetilde{R}(\theta)$ and $\widetilde{J}(\theta)$ and solve normal equations (5) for $\Delta\theta$;
  set $n = n + 1$;
end

In particular, in many (rough) stochastic volatility models such as the *rough Bergomi model* (see Example VIB.1), expensive Monte Carlo simulations have to be used to approximate the pricing map (see [HJM17] or [BLP17] for details). In a common calibration scenario where the normal equations (5) have to be solved frequently, the approach outlined above thus renders calibration prohibitively expensive.

Remark 2. *We note that many modern tensor-based machine learning frameworks are ideally suited for calibration tasks because they directly provide gradients of the output variable by use of automatic differentiation.*

We would like to emphasize that our methodology can in principle be applied to any model with finitely many parameters, from the classical Black Scholes or Heston models to the rough Bergomi model of [BFG16], also to large class of rough volatility models (see Horvath, Jacquier and Muguruza [HJM17] for a general setup). In fact the methodology is not limited to stochastic models, also parametric models of implied volatility could be used for generating training samples of abstract models, but we have not pursued this direction further. For the sake of concreteness, we give an example of one rough volatility model, since computational costs of available numerical methods are especially limiting for this model class.

Example VIB.1. *In the abstract model framework, the rough Bergomi model [BFG16] is represented by* $\mathscr{M}^{\text{rBergomi}}(\Theta^{\text{rBergomi}})$*, with parameters* $\theta = (\xi_0, \eta, \rho, H) \in \Theta^{\text{rBergomi}}$*. For instance, we may choose*

$$\Theta^{\text{rBergomi}} = \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times [-1, 1] \times]0, 1/2[,$$

to stay in a truly rough setting. The model corresponds to the following system for the log price X

and the instantaneous variance V:

$$dX_t = -\frac{1}{2}V_t dt + \sqrt{V_t}dW_t, \quad \text{for } t > 0, \quad X_0 = 0, \tag{6a}$$

$$V_t = \xi_0(t)\mathcal{E}\left(\sqrt{2H}\eta \int_0^t (t-s)^{H-1/2} dZ_s\right), \quad \text{for } t > 0, \quad V_0 = v_0 > 0, \tag{6b}$$

where H denotes the Hurst parameter, $\eta > 0$, $\mathcal{E}(\cdot)$ the Wick exponential, and $\xi_0(\cdot) > 0$ denotes the initial forward variance curve (see [Ber15, Section 6]), and W and Z are correlated standard Brownian motions with correlation parameter $\rho \in [-1, 1]$. In this work, we shall consider the constant forward variance case $\xi_0(t) := \xi_0 \geq 0$ and address more general forward variance setups in a separate work.

2 Deep calibration

In the following sections we elaborate on the objectives and advantages of this two step calibration approach and present examples of neural network architectures, precise numerical recipes and training procedures to apply the two step calibration approach to a family of stochastic volatility models. We also present some numerical experiments and report the learning errors compared to chosen parameters of the synthetic data.
There are several advantages of separating the tasks of pricing and calibration. Above all, the most appealing reason is that it allows us to build upon the knowledge we have gained about the models in the past decades, which is of crucial importance from a risk management perspective. By its very design, **(i)** deep learning the *price approximation* combined with **(ii)** deterministic calibration does not cause more headache to risk managers and regulators than the corresponding stochastic models do. Designing the training as described above demonstrates how deep learning techniques can successfully extend the toolbox of financial engineering, without imposing the need for substantial changes in our risk management libraries.

2.1 One-step approach: Deep calibration by the inverse map

An increasingly popular approach in quantitative finance (and many other fields of engineering) is to develop purely data-driven frameworks, without relying on formal models. While this approach bears many advantages, this kind of modelling is still relatively new and several risk-management considerations are not yet addressed. In its current state this approach leaves for example the meaning of calibrated network parameters unexplained, not to mention the ambiguity about the choice of the number of network parameters and network design. This can cause major challenges towards today's regulatory requirements. In addition, issues of generalisation – how can one price exotic options in a network trained with vanilla option data, to give a simple example – are difficult to analyse, and traditional paradigms of finance – such as no arbitrage – are hard to guarantee in the absence of a model. We refer to the works [BHL$^+$20, JL19] for examples of data driven modelling frameworks.
A second, more model based approach was proposed in the pioneering work of Hernandez [Her16], followed by several other authors such as Stone [Sto20], Dimitroff, Röder and

Fries [DRF18] and many others. A main characteristic of the neural network proposed by [Her16] is that option price approximation and parameter calibration are done in one step within the same network. Indeed, the idea is to directly learn the whole calibration problem, i.e., to learn the model parameters as a function of the market prices (typically parametrized as implied volatilities). In the formulation of Section 1, this means that we learn the mapping

$$\Pi^{-1} : (\mathscr{P}(\zeta))_{\zeta \in Z'} \mapsto \widehat{\theta}.$$

More precisely, [Her16] trains a deep neural network based on labelled data (x_i, y_i), $i = 1, \dots, N$, with

$$x_i = (\mathscr{P}(\zeta))_{\zeta \in Z'_i}$$

for day t_i (in the past) and the corresponding labels

$$y_i = \widehat{\theta}_i,$$

obtained from calibrating the model to the market data y_i using traditional calibration routines. The number of labelled data points N is, of course, limited to the amount of (reliable) historical market price data available.
In spite of the promising results by Hernandez [Her16] the main drawback of this approach, as Hernandez observes, is the lack of control on the function Π^{-1}. Furthermore, from a risk management perspective one has no guarantee how well the learned mapping of Π^{-1} will solve the calibration problem when exposed to unseen data. In fact, this is the behaviour observed in Hernandez [Her16], since the out of sample performance tends to differ from the in sample one, suggesting a not fully satisfactory generalisation of the learned map. We recover the same behaviour of the inverse map in our own experiments, which we included in Appendix VIB.A.

2.2 Two-step approach: Learning the pricing map (or implied volatility map) of models

The two step approach is somewhere mid-way between a sole reliance on traditional pricing methods (Monte Carlo, finite elements, finite differences, Fourier methods, asymptotic methods etc.) and the direct approach described above that calibrate directly to the price data. Here, one separates the calibration procedure as described in Section 1 **(i)** We first learn (approximate) the pricing map by a neural network that maps parameters of a stochastic model to prices or implied volatilities. In other words, we set up and train (off-line) a neural network to learn the pricing map P. In a second step **(ii)** we calibrate (on-line) the model – as approximated by the neural network trained in step **(i)** – to market data using a standard calibration routine. To formalise the two step approach, for an option parametrized by ζ and a model $\mathscr{M}$ with parameters $\theta \in \Theta$ we write $\widetilde{P}(\theta, \zeta) \approx P(\theta, \zeta)$ for the approximation $\widetilde{P}$ of the true pricing map P based on a neural network. Then, in the second step, for a properly chosen distance function δ (and a properly chosen optimization algorithm) we calibrate the model by computing

$$\widehat{\theta} = \operatorname*{argmin}_{\theta \in \Theta} \delta\left(\left(\widetilde{P}(\theta, \zeta)\right)_{\zeta \in Z'}, (\mathscr{P}(\zeta))_{\zeta \in Z'}\right). \tag{7}$$

In principle, this method is not unlike traditional calibration routines, as the true option price has to be numerically approximated for all but the most simple models. This particular approximation method tends to be orders of magnitudes faster compared to other numerical approximation methods for all tested models. In particular, note that the (slow) training stage of the neural network itself only has to be done once. We will come back to comparisons of actual computational times in the numerical section of this chapter.
At this stage, we note that the deep calibration routine is not yet specified in any details: apart from purely numerical details such as the choice of the architecture of the neural networks, the loss functions and optimization algorithms of both the training of the neural networks in stage **(i)** and the actual calibration in stage **(ii)**, one particularly important choice is whether the neural network learns implied volatilities of individual options or rather a full implied volatility surface. Before discussing these details, let us already highlight some of the differences to the one-step approach of [Her16]. In principle, the one-step approach is orders of magnitude faster by construction, however we will demonstrate in Section 4 that the two-step approach calibrates within milliseconds making the speed difference irrelevant in practice. Moreover, we see the main benefit of the two-step approach in the increased stability, which is influenced by two key differences:

- As the neural network is only responsible for option pricing in the model, synthetic data is used for training. Hence, we can easily increase the number of training data, and the training data are completely unpolluted from market imperfections.

- The two-step approach induces a natural decomposition of the overall calibration error into a pricing error (from the neural network) and a model misfit to the market data. Hence, the performance of the neural network itself is generally independent of changing market regimes – which might, of course, change the suitability of the model under consideration.

These points, in particular, imply that frequent re-training of the neural network is not needed in the two-step approach.

2.2.1 The two step approach: Pointwise training and implicit and grid-based training

The underlying principle of the two-step approach appears in one way or another in a number of related contributions De Spiegeleer, Madan, Reyners and Schoutens [DSMRS18] and McGhee [McG18]. In fact, the early works of Hutchinson, Lo and Poggio [HLP94] and the more recent work of Culkin and Das [CD17]–where Deep Neural Networks are applied to learn the Black-Scholes formula–can be recognised as Step **(i)** of the two-step approach in a Black-Scholes context. Also Ferguson and Green [FG18] examine Step **(i)** of the two-step approach in [FG18] for basket options in a lognormal context and observe that the network even has a smoothing effect and increased accuracy in comparison to the underlying Monte Carlo prices. In this section, we examine its advantages and present an analysis of the objective function with the goal to enhance learning performance. Within this framework, the pointwise approach has the ability to asses the quality of $\widetilde{P}$ using Monte Carlo or PDE methods, and

indeed it is superior in terms of robustness.

Pointwise learning

Step (i): Learn the map $\widetilde{P}(\theta, T, k) = \widetilde{\sigma}^{\mathscr{M}(\theta)}(T, k)$ – that is in equation (7) above we have $\zeta = (T, k)$. In the case of vanilla options ($\zeta = (T, k)$) one can rephrase this learning objective as an implied volatility problem: In the implied volatility problem the more informative implied volatility map $\widetilde{\sigma}^{\mathscr{M}(\theta)}(T, k)$ is learned, rather than call- or put option prices $\widetilde{P}(\theta, T, k)$. We denote the artificial neural network by $\widetilde{F}(w; \theta, \zeta)$ as a function of the weights w of the neural network, the model parameters θ and the option parameters ζ. The optimisation problem to solve is the following:

$$\widehat{\omega} := \underset{w \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{Train}}} \eta_i (\widetilde{F}(w; \theta_i, T_i, k_i) - \widetilde{\sigma}^{\mathscr{M}}(\theta_i, T_i, k_i))^2. \tag{8}$$

where $\eta_i \in \mathbb{R}_{>0}$ is a weight vector.

Step (ii): Solve the classical model calibration problem for the market quotes $\{\sigma_{\text{BS}}^{\text{MKT}}(k_j, T_j)\}_{j=1}^m$

$$\widehat{\theta} := \underset{\theta \in \Theta}{\operatorname{argmin}} \sum_{j=1}^{m} \beta_j (\widetilde{F}(\widehat{w}; \theta_i, T_i, k_i) - \sigma_{\text{BS}}^{\text{MKT}}(k_j, T_j))^2.$$

for some user specified weights $\beta_j \in \mathbb{R}_{>0}$, where now the (numerical approximation of the) option price $\widetilde{P}(\theta, T, k)$ resp. implied volatility $\widetilde{\sigma}^{\mathscr{M}(\theta)}(T, k)$ is replaced by the DNN approximation $\widetilde{F}(\widehat{\omega}; \theta, T, k)$ obtained in Step **(i)**. We note here that $\widetilde{F}(\widehat{w}; \theta_i, T_i, k_i)$ being a Neural Network, all gradients with respect to (θ, T, k) are available in closed-form and are fast to evaluate.

The critical part is, of course, the first step, as the second one merely corresponds to classical calibration against liquid options. For the first step, key issues are the choice of training data and the architecture of the neural network. Regarding the training data, the general idea is as follows:

1. Choose realistic *"prior"* distributions for both model parameters θ and option parameters ζ (= (T, k) in the above notation). The point is that many theoretically possible parameters are very unlikely to ever occur in real markets, for both model and option parameters. Hence, it is wasteful to spend resources to learn the pricing map for, say, maturities in the range of hundreds of years. The simplest choice is to simply impose uniform distributions on truncated parameter ranges, but nothing prevents more "informed" possibilities, for instance taking into account historical distributions of estimated model parameter values or observed option parameter values.

2. Simulate model and option parameters according to the distribution chosen before and compute the corresponding option price or implied volatility, which serves as label for the respective parameter vector. The computation can be done for any available

numerical method, for instance Monte Carlo simulation. As an aside, this mechanism can, of course, be used to produce training, testing and validation data in the sense of the machine learning literature.

Remark 3. *Note that the above mentioned "informed" parameter distributions could also be encoded as weights into the loss function for the training of the neural network.*

Remark 4. *Instead of simulation of parameter values, we could also consider deterministic grids in the parameter space. In very high dimensional parameter spaces this probably becomes unfeasible due to the curse of dimensionality, but in the current context this approach may very well improve training of the neural network. We leave a comparison to future work.*

Implicit & grid-based learning

We take this idea further and design an implicit form of the pricing map that is based on storing the implied volatility surface as an image given by a grid of "pixels". This image-based representation has a formative contribution in the performance of the network we present in Section 4. Let us denote by $\Delta := \{k_i, T_j\}_{i=1,\,j=1}^{n,\,m}$ a fixed grid of strikes and maturities, then we propose the following two step approach:

Step (i): Learn the map $\widetilde{F}(w,\theta) = \{\sigma^{\mathscr{M}(\theta)}(T_i,k_j)\}_{i=1,\,j=1}^{n,\,m}$ via neural network where the input is a parameter combination $\theta \in \Theta$ of the stochastic model $\mathscr{M}(\theta)$ and the output is a $n \times m$ grid on the implied volatility surface $\{\sigma^{\mathscr{M}(\theta)}(T_i,k_j)\}_{i=1,\,j=1}^{n,\,m}$ where $n,m \in \mathbb{N}$ are chosen appropriately (see Section 3.1) on a predefined fixed grid of maturities and strikes. $\widetilde{F}$ takes values in $\mathbb{R}^L$ where $L = \text{strikes} \times \text{maturities} = nm$. The optimisation problem in the image-based implicit learning approach is:

$$\widehat{\omega} := \underset{w\in\mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{Train}}^{\text{reduced}}} \sum_{j=1}^{L} \eta_j (\widetilde{F}(w,\theta_i)_j - \widetilde{\sigma}^{\mathscr{M}}(\theta_i,T_j,k_j))^2, \tag{9}$$

where $N_{\text{Train}} = N_{\text{Train}}^{\text{reduced}} \times L$ and $\eta_i \in \mathbb{R}_{>0}$ is a weight vector.

Step (ii): Solve the minimisation problem

$$\widehat{\theta} := \underset{\theta\in\Theta}{\operatorname{argmin}} \sum_{i=1}^{L} \beta_j (\widetilde{F}(\widehat{\omega},\theta)_i - \sigma_{\text{BS}}^{\text{MKT}}(T_i,k_i))^2,$$

for some user specified weights $\beta_j \in \mathbb{R}_{>0}$. We note here that $\widetilde{F}(\widehat{\omega},\theta)$ being a Neural Network, all gradients with respect to θ are available in closed-form and are fast to evaluate.

The data generation stage for the image-based approach works as in the pointwise approach, except that the option parameters $\zeta = (T,k)$ are, fixed and are no longer part of the learning algorithms – except implicitly in the output/labels of the neural network. This is why they appear in the general objective function of pointwise learning (8) but no longer appear in the

objective function (9) of the grid-based learning above. In practice, we choose a grid Δ of size 8×11.

Clearly, the neural network does depend on the grid Δ of option parameters ζ. Hence, we need to interpolate between gridpoints in order to be able to calibrate (in the calibration Step **(ii)**) also to such options, whose maturity and strike do not exactly lie on the grid Δ. This indirect dependence of the trained network on Δ is alluded to by the name "implicit learning".

Implicit smile-based learning:
–And outlook towards an implicit learning with more elaborate grids and tessalations of the IV surface–
We note that McGhee [McG18] follows an *implicit* approach for the lognormal SABR model, which lies somewhere between the pointwise and the image-based approaches of Step **(i)**: There, the inputs are $(\theta^{\text{SABR}}, T, k_1, \dots, k_{10})$, and there are ten volatility outputs $\sigma_1, \dots, \sigma_{10}$ per maturity T. Since between the reference points of the smile McGhee [McG18] also interpolates (by splines) based on a smoothness assumption of implied volatilities, we also refer to this approach as *implicit* training. The reference points $k_1, \dots, k_{10}$ on the volatility surface are determined as a direct functional of the model parameters θ^{SABR} and of the maturity T, that is the learning is done slice-by slice. This sampling technique showcases an excellent working example of a *representative functional sampling* on the surface, where more samples are taken in certain regions of the surface, to ensure a good accuracy of the training in those regions (e.g. regions with higher liquidity). Though the sampling of the strikes in [McG18] is bespoke to the SABR model, it motivates the idea of *representative sampling grid (or tessalation net)*, which would be desirable to achieve also in a model agnostic context. We note that the introduction of the weight vectors $\eta_i \in \mathbb{R}_{>0}$ in the objective function (9) of the grid-wise approach has a similar effect as a higher sampling frequency of a neighbourhood/point.

2.2.2 The role of the objective function: Pointwise training versus implicit and grid-based training

Comparing the pointwise approach (characterised by the general objective function (8)) and the image-based approach (characterised by the objective function (9)), we find that both of them can be advantageous in certain situations. We highlight the connection between the two below, and elaborate on some of the respective advantages of each approach.

Equation (9) can be brought to the form of (8) equation by inserting (into (8)) the specification values $\theta = \theta'$, with

$$\theta'_1 = \theta_1, \dots, \theta'_L = \theta_1, \theta'_{L+1} = \theta_2, \dots,$$

and recalling that $L = \text{strikes} \times \text{maturities}$ and $N_{\text{Train}} = N_{\text{Train}}^{\text{reduced}} \times L$. Hence, the pointwise approach is more general than the image-based one.

With this in mind we make the general note, many of the various advantages and disadvantages of both approaches can, in principle, be mitigated by careful choice of the data generation mechanism (of the training and validation datasets) and the loss function in the training.

- The biggest difference, between pointwise and image based implicit learning procedures is that image based implicit learning requires an outside (implicit) interpolation between the learned implied volatilities in order to compute the implied volatility of an option with an arbitrary strike or maturity, not aligned with the grid. At face value, this is of course an advantage of the pointwise (explicit) approach, where the interpolation is rather performed by the deep neural network. On the other hand, we note that the function $(T,k) \mapsto \sigma^{\mathcal{M}}(\theta;T,k)$ (for fixed model parameters θ) is usually a very well understood smooth function.

- Indeed, this very same structure induces a reduction of variance in the training data for the grid-based approach as compared to the pointwise approach. Overall, in the grid-wise approach, the network can be trained on a (remarkably) small number of Mote Carlo paths, as the same generated paths can be re-used and evaluated at different maturities (corresponding to the points on the grid). This efficient use of MC paths would a priori would not be possible without a fixed grid, i.e. in the pointwise method. Formally speaking, in the image based approach only the model parameters are sampled, while the strike and maturities of the underlying instruments are deterministic. As a side note, keep in mind that we should always compare the two approaches based on a fixed number N_{Train} of total training data.

- It is easier to take into account the *structure* of real financial data into the data generation for the pointwise approach by adjusting the (random) sampling distribution on the surface accordingly. Clearly, not all options are equally *important* for the purpose of calibration, but we would like to concentrate on liquid options. The pointwise approach is ideally suited to take into account historical numbers of liquidity: it is easy to adjust the sampling distribution for strikes and maturities,which is the method of choice if such considerations should be taken into account. In the grid-based approach, specific sampling distributions as above could be mimicked (to some extent) by the choice of a weight vector $\eta_i \in \mathbb{R}_{>0}$ in (9). Alternatively it could be taken into account by some hibrid method using non-uniform, non-tensorized (or even bespoke quasirandom sampling grids), with higher density of points in regions with higher liquidity, which however, is beyond the scope of the current discussions.

- The image-based approach may be seen as an efficient dimension-reduction technique as compared to the pointwise one. Of course, the price we pay is that we only learn the values of the implied volatilities on a fix grid Δ of option parameters. In this example, this price is, however, worth paying since the regularity of the volatility surface is well understood. This implies that we know very well the number and location of grid points required to get good fits globally in terms of the chosen interpolation.

In the particular calibration example presented in Section 4 below, the image-based approach performed somewhat better than the pointwise approach, which indicates that the variance and dimension reduction features may be more important than the other aspects in the above comparison.

Remark 5. *In principle, the two-step approach is also amenable to other numerical interpolation methods. For instance, we could also use Chebyshev interpolation to approximate model implied volatilities such as [GKS20, GHMP19].*

Remark 6. *In line with Remark 5, we note that the image-based approach (in conjunction with the outside interpolation) is a hybrid between a pure DNN approximation such as the pointwise approach and a standard polynomial interpolation method, such as Chebyshev approximation, see [GKS20] for example. Of course, other, more specialized interpolation methods on the implied volatility surface are also possible, for instance using the SVI volatility parameterization [Itk15] or using no-arbitrage constraints [CRW20].*

Remark 7. *We remark that* inversion *of a trained DNN in the grid-base two-step approach might help training the neural network underlying the one-step approach, for instance by providing a good initial guess for the parameters of the neural network. Continued training with market data will, however, certainly be necessary to take into account the model error as well as the special structure of market data.*

3 Practical implementation

We start by describing the approximation network (Step **(i)** of Section 2 with objective functions (8) and (9)) and leave the discussion of calibration (Step **(ii)**) for Section 3.2 below. While several related works [HLP94, CD17, McG18] have demonstrated that learning the pricing map (Step **(i)**) in the Black-Scholes model and in certain classical stochastic volatility models (such as the lognormal SABR model in [McG18]) can be done to a satisfactory accuracy with a single hidden layer, the situation is–as often–more delicate in the case of rough volatility models. Since these models are highly nonlinear nature, they also require deeper networks for an accurate approximation of their pricing functional.

3.1 Network architecture and training

We present the architecture used for the grid-based approach in some detail, as this approach was used for most of the numerical examples below.

1. A fully connected feed forward neural network with 3 hidden layers and 30 nodes on each layers;
2. Input dimension = n, number of model parameters
3. Output dimension = 11 strikes$\times$ 8 maturities for this experiment, but this choice of grid can be enriched or modified.
4. The three inner layers have 30 nodes each, which adding the corresponding biases results on a number

$$(n+1)\times 30 + 2\times(1+30)\times 30 + (30+1)\times 88 = 30n + 4618$$

of network parameters to calibrate.

5. We choose the Elu $\sigma_{\text{Elu}} = \alpha(e^x - 1)$ activation function for the network and a linear activation $\sigma_{linear}(x) = x$ in the output layer

The use of Elu and linear activations is motivated by the fact that we want the neural network to be continuously differentiable in order to use gradient methods in the calibration. We train the neural network using gradient descent, the so-called 'Adam' minibatch training scheme due to Kingman and Ba [KB15], which is a version of the Stochastic Gradient Descent algorithm. In the following, w denotes the set of parameters – weights and biases – of a neural network $F = F(w, x)$. Given parameters $0 \leq \beta_1, \beta_2 < 1, \epsilon,\ \alpha$, initial iterates $u_0 := 0, v_0 := 0, w_0 \in \Omega$, the Adam scheme has the following iterates:

$$\begin{aligned} g_n &:= \nabla^w \sum_{i=1}^{m} \mathscr{L}\left(F(w_{n-1}, X_{n,m}^{\text{batch}}), F^*(X_{n,m}^{\text{batch}})\right) \\ u_{n+1} &:= \beta_1 u_n + (1-\beta_1) g_n \\ v_{n+1} &:= \beta_2 v_n + (1-\beta_2) g_n^2 \\ w_{n+1} &:= w_n - \alpha \frac{u_{n+1}}{1-\beta_1^{n+1}} \frac{1}{\sqrt{v_n/(1-\beta_2^{n+1})} + \epsilon}. \end{aligned}$$

We used the default parameters recommended in [KB15] and we chose a batch size of 32 and iterated through 200 epochs with random shuffling in place to perform the off-line training procedure (we refer the reader to GitHub: NN-StochVol-Calibrations for further details).

3.2 The calibration step

Once the pricing map approximator $\widetilde{F}$ for the implied volatility is found, only the calibration step is left to solve. We use the Levenberg-Marquart algorithm as presented in Section 1.

3.2.1 Bayesian Analysis of the Calibration

Intuitively, we are interested in *quantifying the uncertainty* about model parameter estimates obtained by calibrating with the approximative implied volatility map map $\widetilde{F}$. To this end, we switch to a Bayesian viewpoint and treat model parameters θ as random variables. The fundamental idea behind Bayesian parameter inference is to update prior beliefs $p(\theta)$ with the likelihood $p(y \mid \theta)$ of observing a given point cloud $y \in \mathbb{R}^N$ of implied volatility data to deduce a posterior (joint) distribution $p(\theta \mid y)$ over model parameters θ.
Formally, for pairs (T_i, k_i) of time to maturity and log-moneyness, let an implied volatility point cloud to calibrate against be given by

$$y = \left[y_1(T_1, k_1), \ldots, y_N(T_N, k_N)\right]^T \in \mathbb{R}^N$$

and analogously, collect model implied volatilities for model parameters θ

$$\widetilde{F}(\theta) = \left[\widetilde{F}(\theta, T_1, k_1), \ldots, \widetilde{F}(\theta, T_N, k_N)\right]^T \in \mathbb{R}^N.$$

We perform a liquidity-weighted nonlinear Bayes regression. Mathematically, for heteroskedastic sample errors $\sigma_i > 0, i = 1, \dots, N$, we postulate

$$y = \widetilde{F}(\theta) + \varepsilon, \quad \varepsilon \sim \mathcal{N}\left(0, \mathrm{diag}(\sigma_1^2, \dots, \sigma_N^2)\right),$$

so that for some diagonal weight matrix $W = \mathrm{diag}(w_1, \dots, w_N) \in \mathbb{R}^{N \times N}$, the liquidity-weighted residuals are distributed as follows

$$W^{\frac{1}{2}}\left[y - \widetilde{F}(\theta)\right] \sim \mathcal{N}\left(0, \mathrm{diag}(w_1\sigma_1^2, \dots, w_N\sigma_N^2)\right).$$

In other words, we assume that the joint likelihood $p\left(y \mid \theta\right)$ of observing data y is given by a multivariate normal. In absence of an analytical expression for the posterior (joint) probability $p(\theta|y) \propto p(y|\theta)p(\theta)$, we approximate it numerically using MCMC techniques [FMHLG13] and plot the one- and two-dimensional projections of the four-dimensional posterior by means of an MCMC plotting library [FM16].

Remark 8. *Of course, from a statistical point of view, loss functions of sum of squares form corresponds to a normality assumption on the error distribution when interpreted as an MLE, for instance. The normality assumption above, hence, merely mirrors the common choice of sum-of-squares as loss function for calibration in finance.*

4 Numerical experiments

4.1 Speed and accuracy of the price approximation networks

As mentioned in Section 2.2 one crucial improvement in comparison with direct neural network approaches, as pioneered by Hernandez [Her16], is the separation of (i) the implied volatility approximation function, mapping from parameters of the stochastic volatility model to the implied volatility surface–thereby bypassing the need for expensive Monte-Carlo simulations in the on-line phase—and (ii) the calibration procedure, which (after this separation) becomes a simple deterministic optimisation problem.

Table VIB.1 shows the CPU computation time for functional evaluation of a full surface under the rough Bergomi model of Example VIB.1. Here, we take the forward variance ξ_0 as constant. In a future work we take a similar approach to constract a network that can consistently approximate a variaty of models including the rough Bergomi model with a forward variance curve that is approximated (more generally) by piecewise constant function.

MC Pricing Full Surface	NN Pricing Full Surface	NN Gradient Full Surface	Speed up NN vs. MC
500.000 μs	14,3 μs	47 μs	21.000 – 35.000

Table VIB.1: Computational time of pricing map (entire implied volatility surface) and gradients via Neural Network approximation and Monte Carlo (MC) for the image-based approach

Table VIB.1 provides the speed of evaluating the trained neural network for the image-based approach, the numbers for the pointwise approach are very similar. We used

- Total number of parameteres: 5.668
- Training set of size 34.000 and testing set of size 6.000
- Rough Bergomi sample: $(\xi_0, \nu, \rho, H) \in \mathscr{U}[0.01, 0.16] \times \mathscr{U}[0.5, 4.0] \times \mathscr{U}[-0.95, -0.1] \times \mathscr{U}[0.025, 0.5]$
- Strikes: $\{0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5\}$
- Maturities: $\{0.1, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.0\}$
- Training data samples of Input-Output pairs are computed using Algorithm 3.5 in Horvath, Jacquier and Muguruza [HJM17] with 60.000 sample paths and the spot martingale condition i.e. $\mathbb{E}[S_t] = S_0,\ t \geq 0$ as control variate.

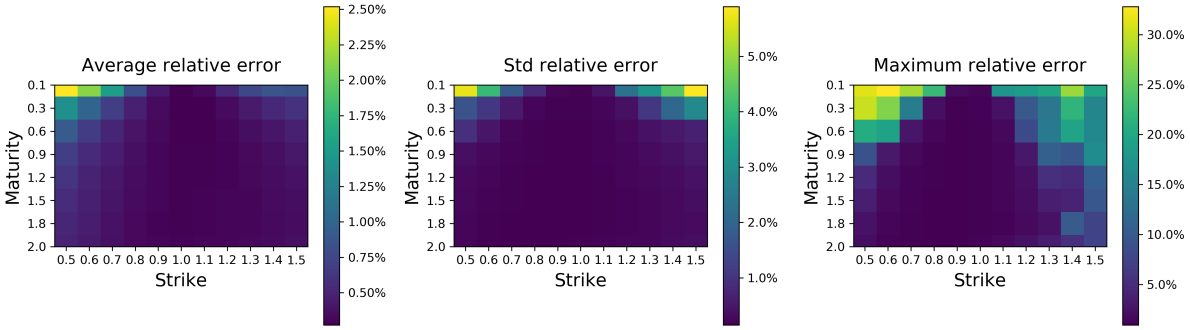


Figure VIB.2: **Error metrics for prices in the rough Bergomi model computed using Monte Carlo evaluations**.
As benchmark we recall relative errors of Monte Carlo prices computed across 34,000 random parameter combinations of the Rough Bergomi model. Relative errors are reported in terms of Average-Standard Deviation-Maximum (Left-Middle-Right) on implied volatility surfaces in the Rough Bergomi model, computed using 95% confidence intervals.

Figure VIB.2 reports the Monte Carlo errors of the training set as a reference. Figure VIB.3 show that the average (across all parameter combinations) relative error[3] between neural network and Monte Carlo approximations is far less than 0.5% consistently (left image in Figure VIB.3) with a standard deviation of less than 1% (middle image in Figure VIB.3). Nevertheless, the maximum relative error goes as far as 25%. As previously stated, the beauty of this approach is the ability to asses whether the approximation is suitable and if not, where exactly fails or is more delicate. In this case, we observe that the approximation is less precise for short maturities and deep out-of-the-money/in-the-money options. Theses errors are consistent with the errors of the Monte Carlo training set.

[3]Relative here is computed here as $|\sigma^{NN}(T,k) - \sigma^{MC}(T,k)|/|\sigma^{MC}(T,k)|$.

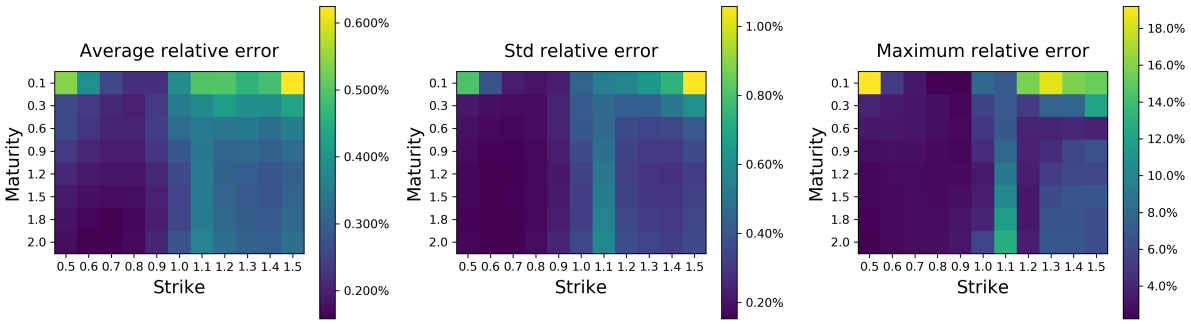


Figure VIB.3: **Error metrics for prices in the rough Bergomi model computed using the neural network approximation**.
We compare surface relative errors of the neural network approximator against the Monte Carlo benchmark across all training data (34.000 random parameter combinations) in the rough Bergomi model. Relative errors are given in terms of Average-Standard Deviation-Maximum (Left-Middle-Right).

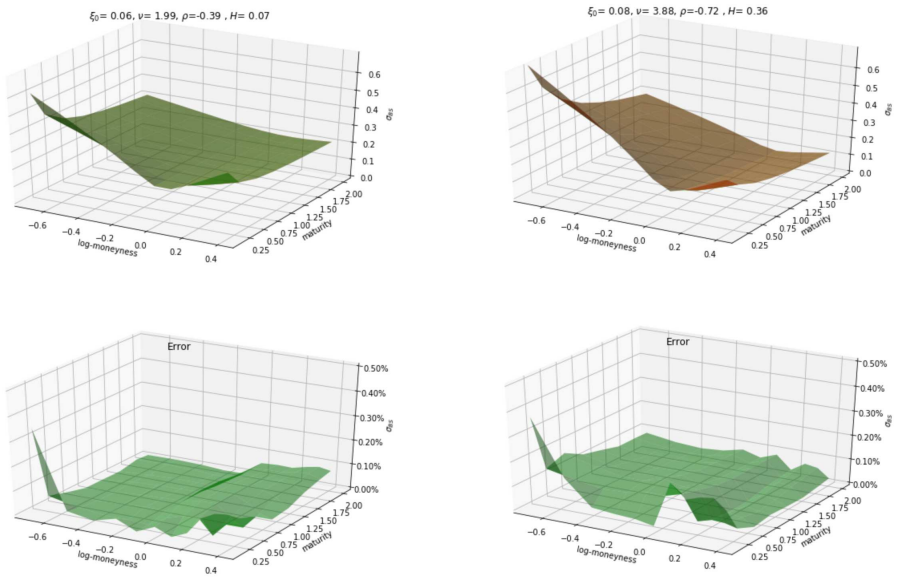


Figure VIB.4: **Quality of fit of the neural network approximation**.
The Figure illustrates the distribution of the approximation error in space after the interpolation to a full implied volatility surface in two examples of model parameter choice.

4.2 Calibration speed and accuracy

To demonstrate the advantage of our two-step approach we obtain calibration times less than 40 milliseconds for the full implied volatility surface in the rough Bergomi model, which was notoriously slow to calibrate (several seconds) by Monte Carlo methods due to its non-Markovian nature. Note that these calibration times become much lower (usually under 10 milliseconds) for Markovian stochastic volaility models. This considerable speedup is due to the 21000-35000 factor speedup (reported in Tabe 1) of the approximation network.

In order to asses calibration the accuracy compared to synthetic data in a controlled experiment, the accuarcy of calibrated model parameters $\widehat{\theta}$ compared to the synthetically generated data with the set of parameters $\overline{\theta}$ that was chosen for the generation of our synthetic data. We measure the accuracy of the calibration via parameter relative error i.e.

$$E_R(\widehat{\theta}) = \frac{|\widehat{\theta} - \overline{\theta}|}{|\overline{\theta}|}$$

as well as the root mean square error (RMSE) with respect to the original surface i.e.

$$\text{RMSE}(\widehat{\theta}) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(\widetilde{F}(\widehat{\theta})_{ij} - \sigma_{BS}^{MKT}(T_i, k_j))^2}.$$

Therefore, on one hand a measure of good calibration is a small RMSE. On the other hand, a measure of parameter sensitivity on a given model is the combined result of RMSE and parameter relative error. For this set of tests, we again restrict ourselves to the image-based approach for learning the price (implied volatility) function in the model.

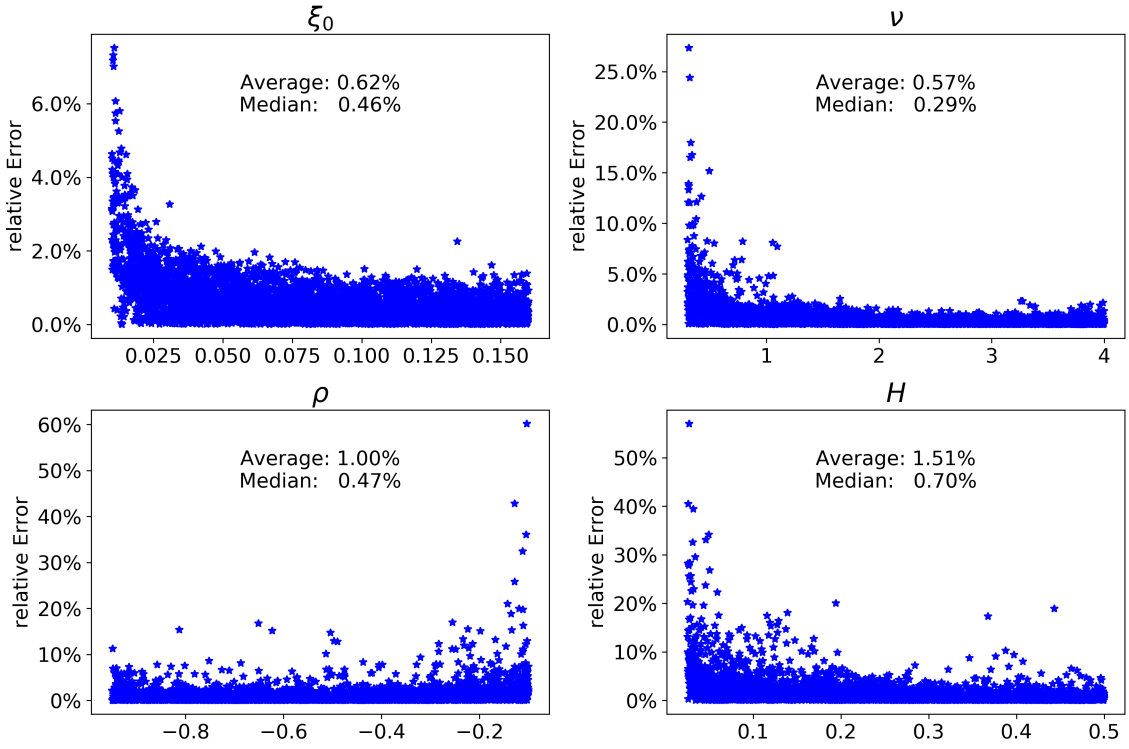


Figure VIB.5: **Calibration relative error per parameter in the test set in the rough Bergomi model.**

Figures VIB.5 shows relative errors after calibration via Levengerg-Marquardt in the rough Bergomi model. We observe that largest errors are concentrated for small H or small vol of vol ν situations. Naturally, the relative error is more sensitive around 0 as well. Once again, we emphasise that by understanding the error zones of the pricing function P (see Figure VIB.3) along with parameter relative errors in Figure VIB.5, we are able to asses its quality and detect parameter configurations that might yield a lower performance of the calibration process.

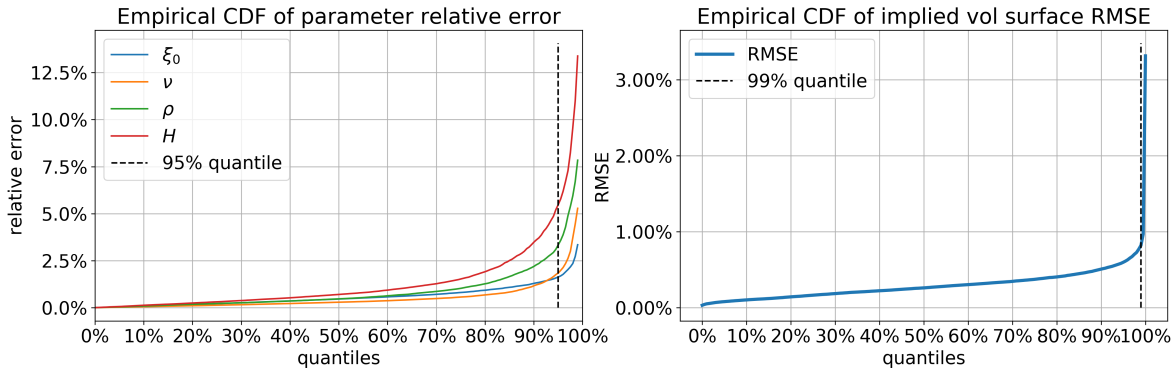


Figure VIB.6: **Parameter calibration error for the rough Bergomi model using the two-step approach**.
Cumulative Distribution Function (CDF) of rough Bergomi parameter relative errors (left) and RMSE (right) after Levengerg-Marquardt calibration across test set random parameter combinations.

To finalise our analysis, Figure VIB.6 shows that the 99% quantile of the RMSE is below 1%, even though parameter relative errors might be higher (see VIB.5 as well), particularly when the parameters are close to 0. Notably, the maximum RMSE across the full surface (i.e. the 88 grid points) is below 4%, which suggests a surprisingly good accuracy.

4.3 A Bayes calibration experiment

We next test the deep calibration procedure using the Bayesian point of view sketched in Section 3.2.1. Here, we use the pointwise approach for learning the model implied volatility map. We perform two experiments. First, fixing $\theta = \theta^\dagger$, we generate a synthetic implied volatility point cloud

$$y_{\text{synth}} = \left[P\left(\theta^\dagger, T_1, k_1\right), \ldots, P\left(\theta^\dagger, T_N, k_N\right)\right] \in \mathbb{R}^N$$

using Monte Carlo simulation as in Section 4.2 above. Next, we perform a non-weighted Bayesian calibration against the synthetic surface and collect the numerical results in Figure VIB.7.

More precisely, the figure shows histograms from the posterior distribution of the one-dimensional marginal distribution of the (four-dimensional) parameter θ in the rough Bergomi model, together with contour plots of all pairs of two-dimensional marginal distributions based on kernel density estimates of the joint densities. The titles of the histogram-windows report the empirical medians together with the differences to the 2.5% and 97.5% quantiles, respectively. The dashed lines in the histogram plots show those quantiles.

If the map $\widetilde{F}$ is sufficiently accurate for calibration, the computed posterior should attribute a large probability mass around $\theta^\dagger$. The results in Figure VIB.7 are quite striking in several

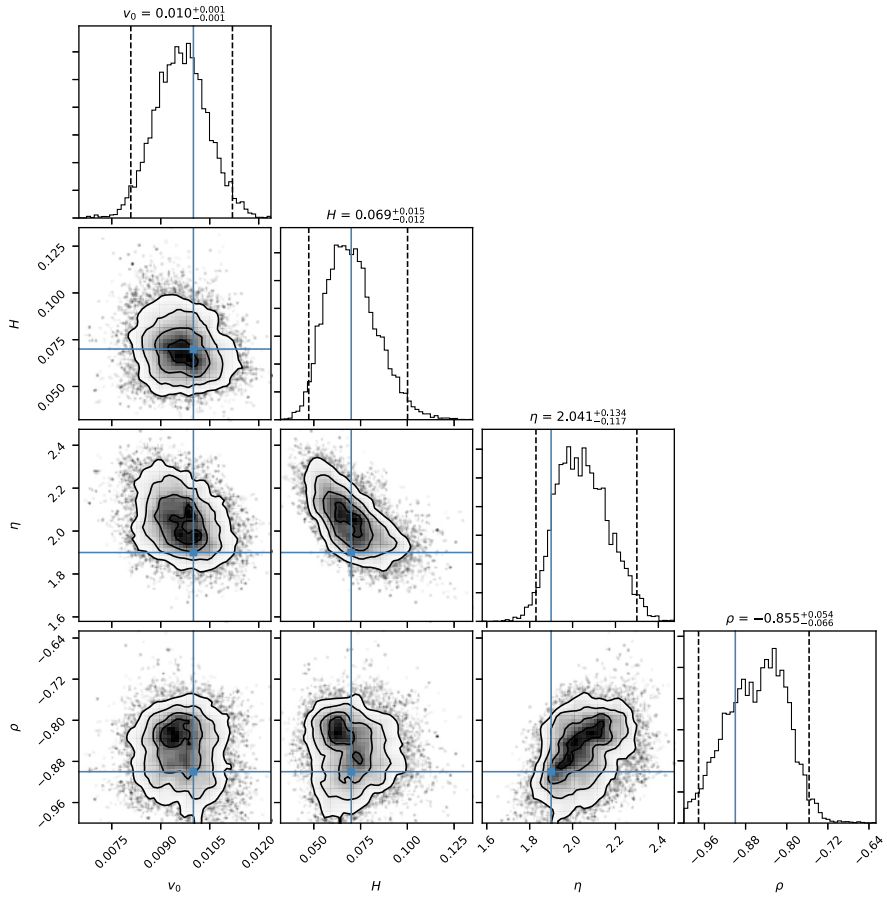


Figure VIB.7: **Bayes calibration against synthetic implied volatility surface.** Solid vertical blue lines indicate true parameter values.

ways: (1) From the univariate histograms on the diagonal it is clear that the calibration routine has identified sensible model parameter regions covering the true values. (2) Histograms are unimodal and its peaks close or identical to the true parameters. (3) The isocontours of the 2d Gaussian KDE in the off-diagonal pair plots for (η, H) and (η, ρ) show exactly the behaviour expected from the reasoning in the last section: Since increases or decreases in one of η, H or ρ can be offset by adequate changes in the others with no impact on the calculated IV, the Bayes posterior cannot discriminate between such parameter configurations and places equal probability on both combinations. This can be seen by the diagonal elliptic probability level sets.

In a second experiment, we want to check whether the inaccuracy of $\widetilde{F}$ allows for a successful calibration against market data. To this end, we perform a liquidity-weighted Bayesian regression against SPX implied volatilities from 19th May 2017. For bid and ask IVs $a_i > 0$

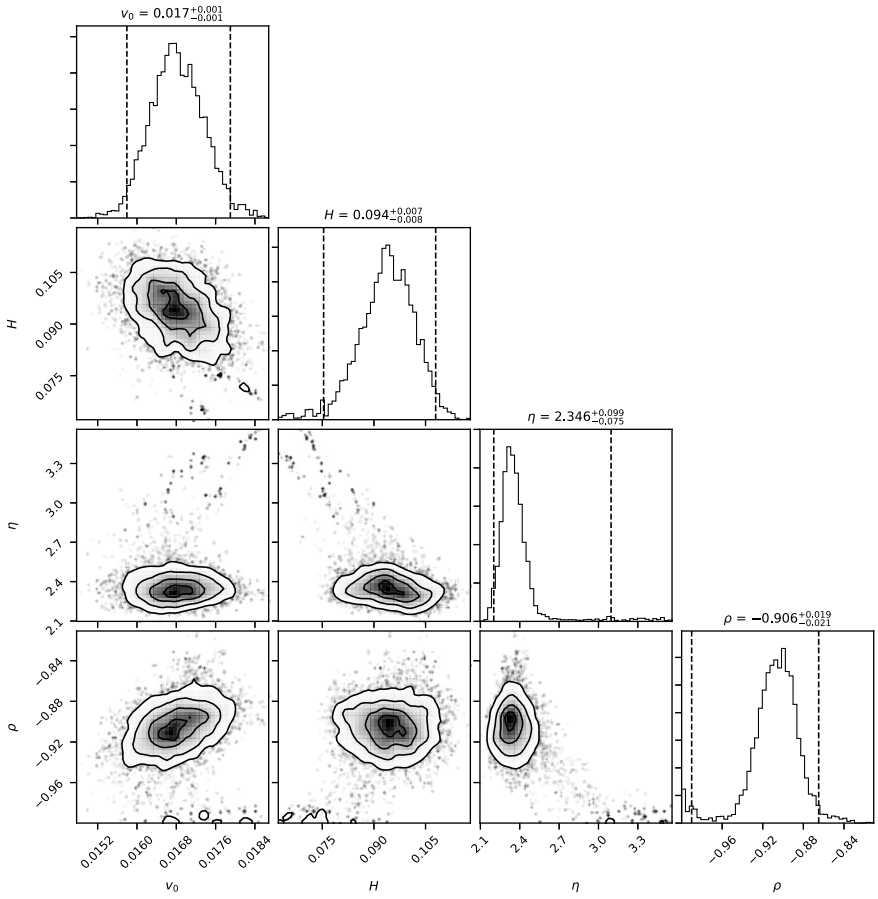


Figure VIB.8: **Liquidity-weighted Bayes calibration against SPX market implied volatility surface from 19th May 2017.**
Liquidity proxies given by inverse bid-ask-spreads.

and $b_i > 0$ respectively, we proxy the IV of the mid price by $m_i := \frac{a_i + b_i}{2}$. With spread defined by $s_i = a_i - b_i \geq 0$, all options with $s_i / m_i \geq 5\%$ are removed because of too little liquidity. Weights are chosen to be $w_i = \frac{m_i}{a_i - m_i} \geq 0$, effectively taking inverse bid-ask spreads as a proxy for liquidity. Finally, σ_i are proxied by a fractional of the spread s_i. The numerical results in Figure VIB.8 further confirm the accuracy of $\widetilde{F}$: (1) As can be seen on the univariate histograms on the diagonal, the Bayes calibration has again identified sensible model parameter regions in line with what is to expected. (2) Said histograms are again unimodal with peaks at or close to values previously reported in the literature. (3) Quite strikingly, at a first glance, the effect of the diagonal probability level sets in the off-diagonal plots as documented in Figure VIB.7 cannot be confirmed here. However, the scatter plots in the diagrams do reveal some remnants of that phenomenon.

VIB.A A numerical experiment with the inverse map

To motivate the main drawbacks of the inverse map approach of Section 2.1, we calibrate rough Bergomi model with it, i.e., we consider the simple map

$$\Pi^{-1}(\Sigma_{\text{BS}}^{\text{rBergomi}}) \to (\hat{\xi}_0, \hat{\nu}, \hat{\rho}, \hat{H})$$

where $\Sigma_{\text{BS}}^{\text{rBergomi}} \in \mathbb{R}^{n\times m}$ is a rBergomi implied volatility surface and $(\hat{\xi}_0, \hat{\nu}, \hat{\rho}, \hat{H})$ the optimal solution to the corresponding calibration problem.

Remark 9. *For simplicity we consider the strikes and maturities to be fixed for all implied volatility surfaces.*

Inverse Map Architecture

- 1 convolutional layer with 16 filters and 3×3 sliding window
- MaxPooling layer with 2×2 sliding window
- 50 Neuron Feedforward Layer with *Elu* activation function
- Output layer with *linear* activation function
- Total number of parameters: 10.014
- Train Set: 34.000 and Test Set: 6.000
- $(\xi_0, \nu, \rho, H) \in \mathcal{U}[0.01, 0.16] \times \mathcal{U}[0.3, 4.0] \times \mathcal{U}[-0.95, -0.1] \times \mathcal{U}[0.025, 0.5]$
- strikes=$\{0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5\}$
- maturities=$\{0.1, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.0\}$
- Implied volatilities computed using Algorithm 3.5 in Horvath, Jacquier and Muguruza [HJM17] with 60.000 sample paths and the spot martingale condition i.e. $\mathbb{E}[S_t] = S_0,\ t \geq 0$ as control variate.

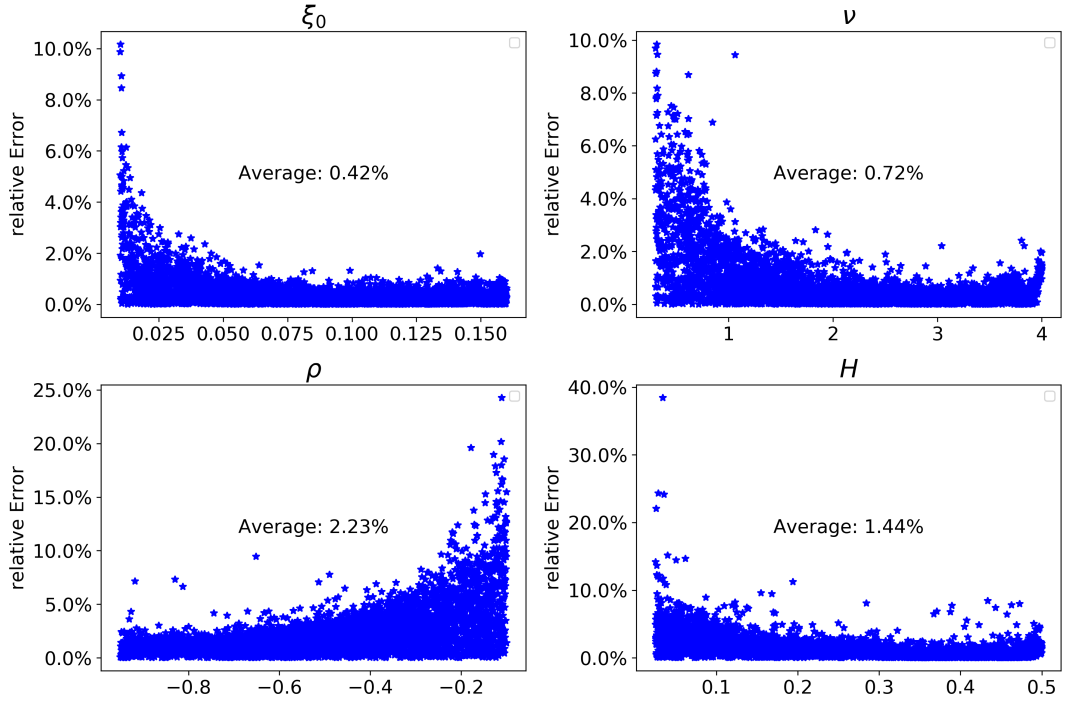


Figure VIB.9: **Out of sample relative errors per parameter calibration**.

Figure VIB.9 shows that, indeed it is possible to approximate the inverse map and very sharply calibrate model parameters with a relatively small network. Convolutional networks make sense in this context, since a implied volatility surface has many features both in the strike and maturity direction that can be extracted, similar to image recognition problems. Notice also that the biggest error come from parameter configurations where the Monte Carlo input is more delicate i.e. very small H or very small volatility. Hence, the shape of the errors is intuitively natural and expected beforehand.

Out of sample performance of the approach

Let us now consider a more substantial variation of our out-of-sample data. We now do not only present implied volatilities to the neural network which have been generated from parameter configurations that are "unseen" by the network, but instead, the presented implied volatitlites are generated by a different model than the one that has been used in the training phase. By this variation we would like to illustrate the scenario of a more substantial regime change in the market. To this end we do not use a rough Bergomi model for the out of sample surfaces but instead generate implied volatility surfaces using the 2-factor Bergomi model, which is described by the stochastic dynamics

$$\begin{aligned} dX_t &= -\frac{1}{2}V_t\,dt + \sqrt{V_t}\,dW_t \\ V_t &= \xi_0(t)\mathscr{E}\left(\nu\left((1-\theta)\int_0^t \exp(-\kappa_X(t-s))\,dZ_s + \theta\int_0^t \exp(-\kappa_Y(t-s))\,dY_s\right)\right), \end{aligned} \tag{10}$$

where W, Y and Z are correlated standard Brownian motions. We run two experiments: one with the one-step inverse map approach in Figure VIB.10 and one with the two-step approach in Figure VIB.11. In the first experiment we present smiles from the model (10) to our (one-step) neural network as out-of-sample (input) data and seek to obtain the corresponding optimal (rough Bergomi) parameters to the presented implied volatilities as an output. The benchmark used in these experiments is a standard calibration routine where (rough Bergomi) prices are obtained by Monte Carlo (MC) simulation, which is then combined with Levenberg-Marquardt (LM) [Lev44, Mar63] algorithm to obtain the calibrated rough Bergomi paremeters. Clearly, since we are calibrating a rough Bergomi model to a market that has already shifted and is now described by a 2-factor Bergomi model, we expect some instances of larger calibration errors, even for the benchmark MC-LM calibration case. Figure VIB.10 however shows that the (ones-step) neural network performs overall visibly worse than the benchmark: While for some samples (marked in blue) the root mean square error of the neural network calibration is smaller than the benchmark, for the majority of the samples (orange) the neural network error is significantly larger. However, we must emphasize that when the neural network is exposed to familiar situations i.e. surfaces close to the ones generated by the rBergomi model it may work just as well (or better) than the benchmark standart MC approach, see the points (in blue) below the dashed black line in Figure VIB.10. The results by Hernandez [Her16] also support this conclusion. There, some out of sample scenarios (based on different historical period) reasonably worse than the in-sample ones. Similar observations can be made here in Figure VIB.10 This is likely due to delicate parameter configurations i.e. very low variance, where MC suffers to obtain accurate estimates whereas the network benefits from the smoothing effect.

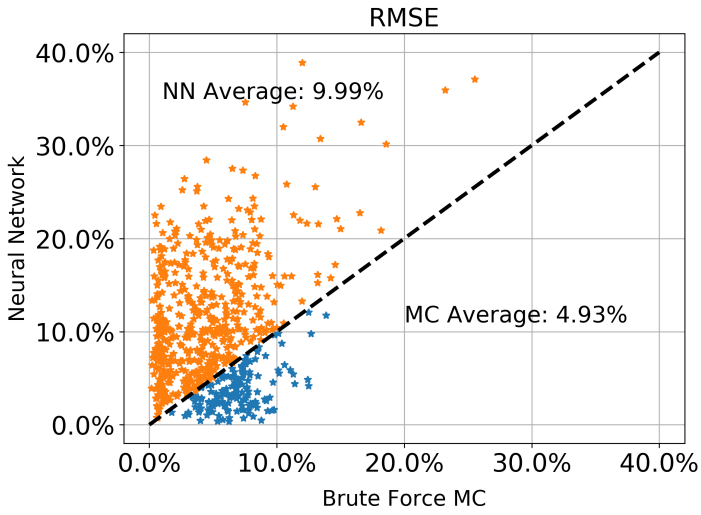


Figure VIB.10: **Calibration accuracy for the inverse map compared to Monte Carlo calibration**.
Stars represent the out of sample RMSE via neural network (NN) in the one-step approach on the vertical axis and the standard benchmark calibration via Monte Carlo method (MC) with Levenberg-Marquardt (LM) optimisation on the horizontal axis. The dashed black line represents the identity function. Points below the identity are samples where the neural network one-step approach has outperformed the Monte-Carlo benchmark. These are highlighted in blue. The majority of the sample are above the identity line marked orange represent samples where the neural network one-step approach generated a significantly larger error than the benchmark.

The *one-step* approach does not generalise the problem to all possible settings, only the ones already observed in the (non-stationary) data. By design it is not possible to train Π^{-1} on all possible (arbitrage-free) market scenarios. Moreover, there is a lack of understanding in the highly non-trivial function Π^{-1}, hence from a risk-managing perspective is more difficult to justify the use of this inverse approach than of the direct approach. On the contrary, Figure VIB.11 reproduces the same experiment with the two-step approach. As expected, by construction the two-step approach solves the calibration problem for any market condition.

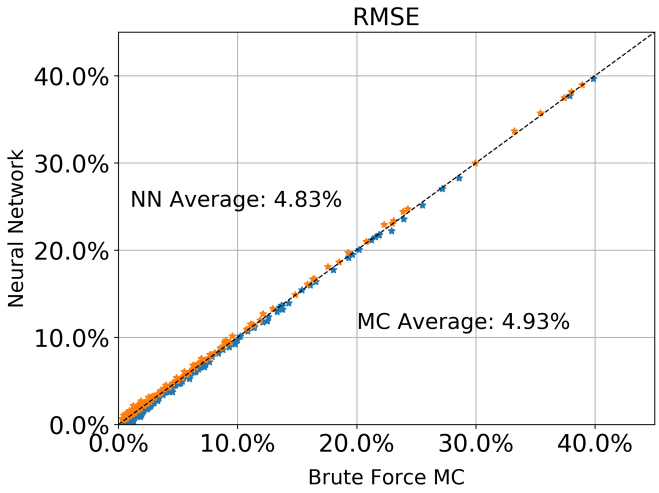


Figure VIB.11: **Calibration accuracy for the two-step approach compared to Monte Carlo calibration**.
Stars represent the out of sample RMSE via neural network (NN) using the two-step approach and brute force Monte Carlo (MC) on different samples. The dashed black line represents the identity function. Points below the identity are samples where the neural network one-step approach has outperformed the Monte-Carlo benchmark and represented in blue. Points above the identity represent samples where the neural network one-step approach underperformed and represented in orange.

We believe that these differences highlighted in the graphs above are not due to some inherent flaw of the one-step model architecture, but rather due to the structures of the data: due to non-stationarity effects in markets, an unprecedented behaviour may be difficult to reproduce directly in the learned network, depending on the training data previously provided to the network. Therefore, we would like to emphasize here that the above experiments should by no means suggest that the one-step approach should be abandoned all together, but rather that the merit of that approach lies in situations, where data should take a more central role rather than–as we suggest here–using the DNN as a speedup for pricing.

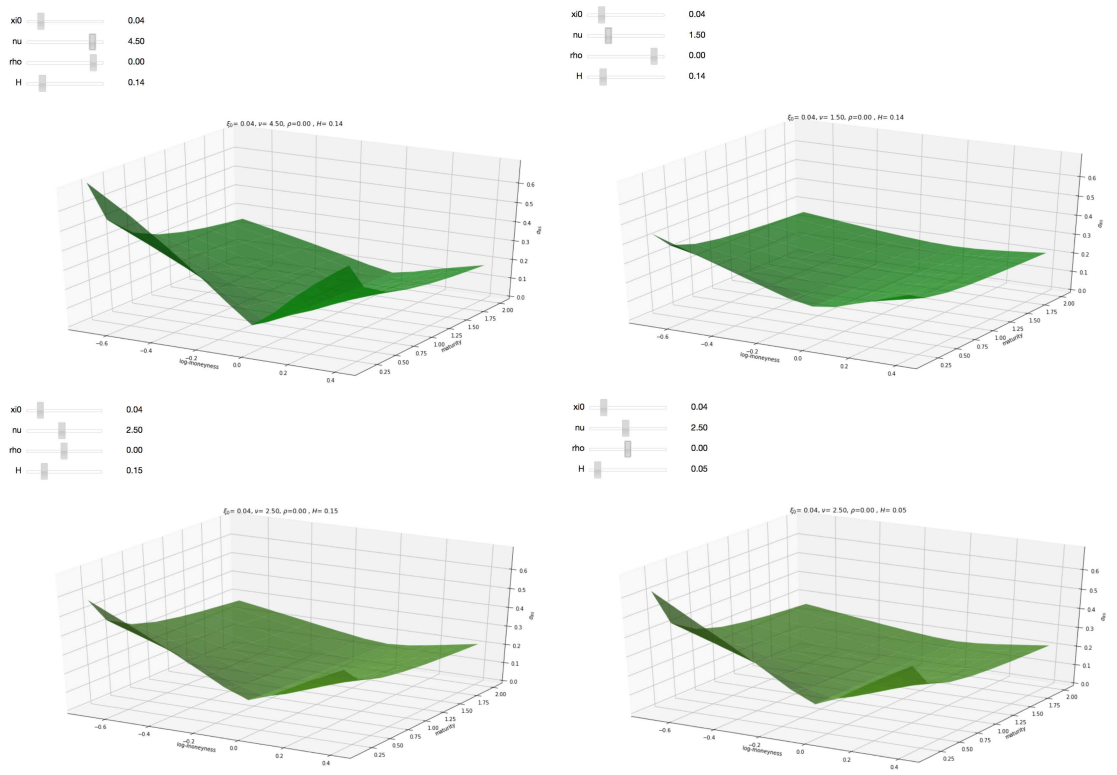
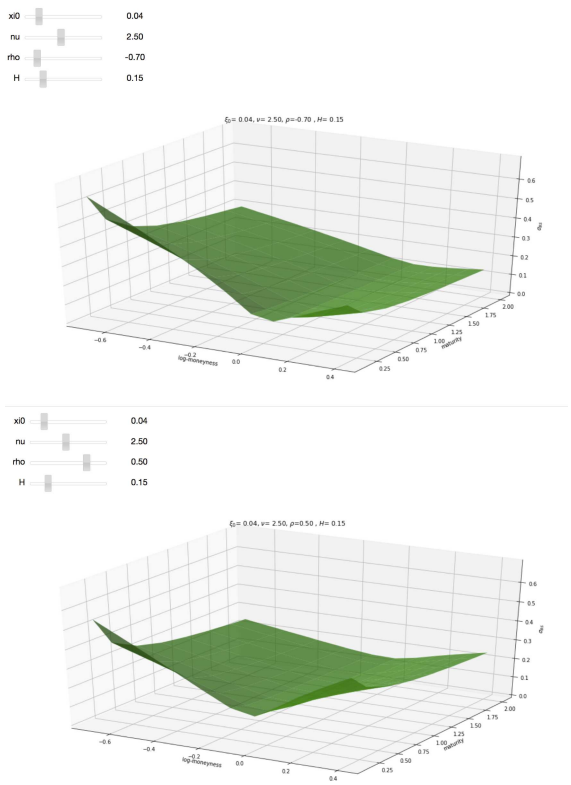


Figure VIB.12: **Influence of model parameters on the implied volatility surface in the rough Bergomi model.**

VIB.B Illusration of model parameters & the pricing engine in the rBergomi model

We showcase here the influence of the model parameters in the rough Bergomi model on the shape of the implied volatility surface using the hands-on pricing engine we generated via the DNN of **step (i)** for the rough Bergomi model. Our findings are in line with asymptotic results presented in [BFG+19] and with [MP18] for the role of the model parameters.

The model parameters ν, ρ and H correspond to the *smile* (ν), *skew* (ρ) and the *explosion (H) parameters* of the surface, while ξ_0 is the one-point approximation of the forward variance. The images illustrate that the parameters ν and H influence the slope of the smile, and an explosive behaviour for short maturities can be achieved (without calibrating slice by slice) with a single surface if the Hurst parameter is $H << \frac{1}{2}$. And finally, as usual in stochastic volatility models, the parameter ρ introduces skewness in the surface as illustrated below.


xi0 0.04
nu 2.50
rho -0.70
H 0.15
ξ0= 0.04, ν= 2.50, ρ=-0.70 , H= 0.15
σBS
log-moneyness
maturity
xi0 0.04
nu 2.50
rho 0.50
H 0.15
ξ0= 0.04, ν= 2.50, ρ=0.50 , H= 0.15
σBS
log-moneyness
maturity

Bibliography

[AGLM18] Elisa Alos, David García-Lorite, and Aitor Muguruza. On smile properties of volatility derivatives and exotic products: understanding the vix skew. *arXiv preprint arXiv:1811.05524*, 2018.

[AJLP19] Eduardo Abi Jaber, Martin Larsson, and Sergio Pulido. Affine volterra processes. *The Annals of Applied Probability*, 29(5):3155–3200, 2019.

[AKS13] Alexander Antonov, Michael Konikov, and Michael Spector. SABR spreads its wings. *Risk (August issue), pp*, pages 58–63, 2013.

[AKS16] Aurélien Alfonsi, Florian Klöck, and Alexander Schied. Multivariate transient price impact and matrix-valued positive definite functions. *Mathematics of operations research*, 41(3):914–934, 2016.

[AKS19] Alexander Antonov, Michael Konikov, and Michael Spector. Modern SABR analytics: Formulas and insights for quants, former physicists and mathematicians. *SpringerBriefs in Quantitative Finance*, 2019.

[ALV07] Elisa Alos, Jorge A León, and Josep Vives. On the short-time behavior of the implied volatility for jump-diffusion models with stochastic volatility. *Finance and Stochastics*, 11(4):571–589, 2007.

[ATHL05] Robert Almgren, Chee Thum, Emmanuel Hauptmann, and Hong Li. Direct estimation of equity market impact. *Risk*, 18(7):58–62, 2005.

[Bar94] Andrew R Barron. Approximation and estimation bounds for artificial neural networks. *Machine Learning*, 14(1):115–133, 1994.

[BBDG18] Jean-Philippe Bouchaud, Julius Bonart, Jonathan Donier, and Martin Gould. *Trades, Quotes and Prices*. Cambridge University Press, 3 2018.

[BBLB18] Frédéric Bucci, Michael Benzaquen, Fabrizio Lillo, and Jean-Philippe Bouchaud. Slow decay of impact in equity markets: insights from the ancerno database. *Market Microstructure and Liquidity*, 4(03n04):1950006, 2018.

[BBLB19] Frédéric Bucci, Michael Benzaquen, Fabrizio Lillo, and Jean-Philippe Bouchaud. Crossover from linear to square-root market impact. *Physical review letters*, 122(10):108302, 2019.

[BCT+15] Giacomo Bormetti, Lucio Maria Calcagnile, Michele Treccani, Fulvio Corsi, Stefano Marmi, and Fabrizio Lillo. Modelling systemic price cojumps with hawkes factor models. *Quantitative Finance*, 15(7):1137–1156, 2015.

[BDHM13] Emmanuel Bacry, Sylvain Delattre, Marc Hoffmann, and Jean-François Muzy. Modelling microstructure noise with mutually exciting point processes. *Quantitative finance*, 13(1):65–77, 2013.

[BDM12] Emmanuel Bacry, Khalil Dayri, and Jean-François Muzy. Non-parametric kernel estimation for symmetric Hawkes processes. Application to high frequency financial data. *The European Physical Journal B*, 85(5):1–12, 2012.

[BEGV18] Philippe Bergault, David Evangelista, Olivier Guéant, and Douglas Vieira. Closed-form approximations in multi-asset market making. *arXiv preprint arXiv:1810.04383*, 2018.

[Ber15] Lorenzo Bergomi. *Stochastic volatility modeling*. CRC press, 2015.

[BFG16] Christian Bayer, Peter Friz, and Jim Gatheral. Pricing under rough volatility. *Quantitative Finance*, 16(6):887–904, 2016.

[BFG+19] Christian Bayer, Peter K Friz, Archil Gulisashvili, Blanka Horvath, and Benjamin Stemper. Short-time near-the-money skew in rough fractional volatility models. *Quantitative Finance*, 19(5):779–798, 2019.

[BFG+20] Christian Bayer, Peter K Friz, Paul Gassiat, Jorg Martin, and Benjamin Stemper. A regularity structure for rough volatility. *Mathematical Finance*, 30(3):782–832, 2020.

[BGTW19] Hans Buehler, Lukas Gonon, Josef Teichmann, and Ben Wood. Deep hedging. *Quantitative Finance*, 19(8):1271–1291, 2019.

[BHL+20] Hans Buehler, Blanka Horvath, Terry Lyons, Imanol Perez Arribas, and Ben Wood. A data-driven market simulator for small data environments. *Available at SSRN 3632431*, 2020.

[BHM+19] Christian Bayer, Blanka Horvath, Aitor Muguruza, Benjamin Stemper, and Mehdi Tomas. On deep calibration of (rough) stochastic volatility models. *arXiv preprint arXiv:1908.08806*, 2019.

[BILL15] Emmanuel Bacry, Adrian Iuga, Matthieu Lasnier, and Charles-Albert Lehalle. Market impacts and the life cycle of investors orders. *Market Microstructure and Liquidity*, 1(02):1550009, 2015.

[BLP17] Mikkel Bennedsen, Asger Lunde, and Mikko S Pakkanen. Hybrid scheme for brownian semistationary processes. *Finance and Stochastics*, 21(4):931–965, 2017.

[BLZ17] Bruno Bouchard, Grégoire Loeper, and Yiyi Zou. Hedging of covered options with linear market impact and gamma constraint. *SIAM Journal on Control and Optimization*, 55(5):3319–3348, 2017.

[BM96] Pierre Brémaud and Laurent Massoulié. Stability of nonlinear Hawkes processes. *The Annals of Probability*, pages 1563–1588, 1996.

[BMEB17] Michael Benzaquen, Iacopo Mastromatteo, Zoltan Eisler, and Jean-Philippe Bouchaud. Dissecting cross-impact on stock markets: An empirical analysis. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(2):23406, 2017.

[BMM15] Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015.

[BR13] Nataliya Bershova and Dmitry Rakhlin. The non-linear market impact of large trades: Evidence from buy-side order flow. *Quantitative finance*, 13(11):1759–1778, 2013.

[CBT+18] Lucio Maria Calcagnile, Giacomo Bormetti, Michele Treccani, Stefano Marmi, and Fabrizio Lillo. Collective synchronization and high frequency systemic instabilities in financial markets. *Quantitative Finance*, 18(2):237–247, 2018.

[CD17] Robert Culkin and Sanjiv R Das. Machine learning in finance: the case of deep learning for option pricing. *Journal of Investment Management*, 15(4):92–100, 2017.

[CFD02] Rama Cont, Jose da Fonseca, and Valdo Durrleman. Stochastic models of implied volatility surfaces. *Economic Notes*, 31(2):361–377, 2002.

[CK94] Jordi Caballe and Murugappa Krishnan. Imperfect competition in a multi-security market with risk neutrality. *Econometrica (1986-1998)*, 62(3):695, 1994.

[Con10] Rama Cont. Model calibration. *Encyclopedia of quantitative finance*, 2010.

[COVDW11] Bin Chen, Cornelis W Oosterlee, and Hans Van Der Weide. Efficient unbiased simulation scheme for the SABR stochastic volatility model. *Preprint*, 2011.

[CRW20] Samuel N. Cohen, Christoph Reisinger, and Sheng Wang. Detecting and repairing arbitrage in traded option prices. *Applied Mathematical Finance*, 27(5):345–373, 2020.

[CT19] Christa Cuchiero and Josef Teichmann. Markovian lifts of positive semidefinite affine Volterra type processes. *arXiv preprint arXiv:1907.01917*, 2019.

[DB15] Jonathan Donier and Julius Bonart. A million metaorder analysis of market impact on the bitcoin. *Market Microstructure and Liquidity*, 1(02):1550008, 2015.

[DBMB15] Jonathan Donier, Julius Bonart, Iacopo Mastromatteo, and Jean-Philippe Bouchaud. A fully consistent, minimal model for non-linear market impact. *Quantitative Finance*, 15(7):1109–1121, 2015.

[DD70] Catherine Doléans-Dade. Quelques applications de la formule de changement de variables pour les semimartingales. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 16(3):181–194, 1970.

[DFZ19] José Da Fonseca and Wenjun Zhang. Volatility of volatility is (also) rough. *Journal of Futures Markets*, 39(5):600–611, 2019.

[DJR19] Aditi Dandapani, Paul Jusselin, and Mathieu Rosenbaum. From quadratic Hawkes processes to super-Heston rough volatility models with Zumbach effect. *arXiv preprint arXiv:1907.06151*, 2019.

[DRF18] Georgi Dimitroff, Dirk Roeder, and Christian P. Fries. Volatility model calibration with convolutional neural networks. *Available at SSRN 3252432*, 2018.

[DSMRS18] Jan De Spiegeleer, Dilip B. Madan, Sofie Reyners, and Wim Schoutens. Machine learning for quantitative finance: fast derivative pricing, hedging and fitting. *Quantitative Finance*, 18(10):1635–1643, 2018.

[DVJ08] Daryl J. Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure.* Springer New York, 2008.

[EEFR18] Omar El Euch, Masaaki Fukasawa, and Mathieu Rosenbaum. The microstructural foundations of leverage effect and rough volatility. *Finance and Stochastics*, 22(2):241–280, 2018.

[EEGR19] Omar El Euch, Jim Gatheral, and Mathieu Rosenbaum. Roughening heston. *Risk*, pages 84–89, 2019.

[EEGRR20] Omar El Euch, Jim Gatheral, Radoš Radoičić, and Mathieu Rosenbaum. The zumbach effect under rough heston. *Quantitative Finance*, 20(2):235–241, 2020.

[EER18] Omar El Euch and Mathieu Rosenbaum. Perfect hedging in rough heston models. *The Annals of Applied Probability*, 28(6):3813–3856, 2018.

[EER19] Omar El Euch and Mathieu Rosenbaum. The characteristic function of rough heston models. *Mathematical Finance*, 29(1):3–38, 2019.

[ELMK18] Ibrahim Ekren, Ren Liu, and Johannes Muhle-Karbe. Optimal rebalancing frequencies for multidimensional portfolios. *Mathematics and Financial Economics*, 12(2):165–191, 2018.

[EMK19] Ibrahim Ekren and Johannes Muhle-Karbe. Portfolio choice with small temporary and transient price impact. *Mathematical Finance*, 29(4):1066–1115, 2019.

[ES16] Ronen Eldan and Ohad Shamir. The power of depth for feedforward neural networks. In *Conference on learning theory*, pages 907–940. PMLR, 2016.

[EV18] David Evangelista and Douglas Vieira. New closed-form approximations in multi-asset market making. *arXiv preprint arXiv:1810.04383*, 2018.

[FG18] Ryan Ferguson and Andrew Green. Deeply learning derivatives. *arXiv preprint arXiv:1809.02233*, 2018.

[FM16] Daniel Foreman-Mackey. corner. py: Scatterplot matrices in python. *The Journal of Open Source Software*, 1:24, 2016.

[FMHLG13] Daniel Foreman-Mackey, David W Hogg, Dustin Lang, and Jonathan Goodman. emcee: the mcmc hammer. *Publications of the Astronomical Society of the Pacific*, 125(925):306, 2013.

[FS12] Vladimir Filimonov and Didier Sornette. Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Physical Review E*, 85(5):056108, 2012.

[Fuk11] Masaaki Fukasawa. Asymptotic analysis for stochastic volatility: martingale expansion. *Finance and Stochastics*, 15(4):635–654, 2011.

[FZ17] Martin Forde and Hongzhong Zhang. Asymptotics for rough stochastic volatility models. *SIAM Journal on Financial Mathematics*, 8(1):114–145, 2017.

[Gat04] Jim Gatheral. A parsimonious arbitrage-free implied volatility parameterization with application to the valuation of volatility derivatives. *Presentation at Global Derivatives & Risk Management, Madrid*, page 0, 2004.

[Gat10] Jim Gatheral. No-dynamic-arbitrage and market impact. *Quantitative Finance*, 10(7):749–759, 2010.

[Gat11] Jim Gatheral. *The volatility surface: a practitioner's guide*, volume 357. John Wiley & Sons, 2011.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[GdMMBB20] Luis Carlos Garcia del Molino, Iacopo Mastromatteo, Michael Benzaquen, and Jean-Philippe Bouchaud. The multivariate kyle model: More is different. *SIAM Journal on Financial Mathematics*, 11(2):327–357, 2020.

[GHL12] Julien Guyon and Pierre Henry-Labordère. Being particular about calibration. *Risk*, 25(1):88, 2012.

[GHMP19] Kathrin Glau, Paul Herold, Dilip B Madan, and Christian Pötz. The chebyshev method for the implied volatility. *Journal of Computational Finance*, 23(3), 2019.

[GJ14] Jim Gatheral and Antoine Jacquier. Arbitrage-free svi volatility surfaces. *Quantitative Finance*, 14:59–71, 2014.

[GJR18] Jim Gatheral, Thibault Jaisson, and Mathieu Rosenbaum. Volatility is rough. *Quantitative finance*, 18(6):933–949, 2018.

[GKS20] Kathrin Glau, Daniel Kressner, and Francesco Statti. Low-rank tensor approximation for chebyshev interpolation in parametric option pricing. *SIAM Journal on Financial Mathematics*, 11(3):897–927, 2020.

[GP16] Nicolae Gârleanu and Lasse Heje Pedersen. Dynamic portfolio choice with frictions. *Journal of Economic Theory*, 165:487–516, 2016.

[Gre15] Andrew Green. *XVA: credit, funding and capital valuation adjustments*. John Wiley & Sons, 2015.

[Gué17] Olivier Guéant. Optimal market making. *Applied Mathematical Finance*, 24(2):112–154, 2017.

[Haw71a] Alan G. Hawkes. Point spectra of some mutually exciting point processes. *Journal of the Royal Statistical Society: Series B (Methodological)*, 33(3):438–443, 1971.

[Haw71b] Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

[HBB13] Stephen J Hardiman, Nicolas Bercot, and Jean-Philippe Bouchaud. Critical reflexivity in financial markets: a hawkes process analysis. *The European Physical Journal B*, 86(10):1–9, 2013.

[Her16] Andres Hernandez. Model calibration with neural networks. *Available at SSRN 2812140*, 2016.

[Hes93] Steven L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies*, 6(2):327–343, 1993.

[Hig08] Nicholas J Higham. *Functions of matrices: theory and computation*, volume 104. Siam, 2008.

[HJL19] Blanka Horvath, Antoine Jacquier, and Chloé Lacombe. Asymptotic behaviour of randomised fractional volatility models. *Journal of Applied Probability*, 56(2):496–523, 2019.

[HJM17] Blanka Horvath, Antoine Jacquier, and Aitor Muguruza. Functional central limit theorems for rough volatility. *Available at SSRN 3078743*, 2017.

[HJW18] Jiequn Han, Arnulf Jentzen, and E Weinan. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.

[HKLW02] Patrick S. Hagan, Deep Kumar, Andrew S. Lesniewski, and Diana E. Woodward. Managing smile risk. *The Best of Wilmott*, 1:249–296, 2002.

[HL17] Pierre Henry-Labordere. Deep primal-dual algorithm for bsdes: Applications of machine learning to cva and im. *Available at SSRN 3071506*, 2017.

[HLP94] James M. Hutchinson, Andrew W. Lo, and Tomaso Poggio. A nonparametric approach to pricing and hedging derivative securities via learning networks. *The journal of Finance*, 49(3):851–889, 1994.

[HLW15] Patrick Hagan, Andrew Lesniewski, and Diana Woodward. Probability distribution in the SABR model of stochastic volatility. In *Large deviations and asymptotic methods in finance*, pages 1–35. Springer, 2015.

[HM17] Sebas Hendriks and Claude Martini. The extended ssvi volatility surface. *Available at SSRN 2971502*, 2017.

[HMT21] Blanka Horvath, Aitor Muguruza, and Mehdi Tomas. Deep learning volatility: a deep neural network perspective on pricing and calibration in (rough) volatility models. *Quantitative Finance*, 21(1):11–27, 2021.

[HO74] Alan G. Hawkes and David Oakes. A cluster process representation of a self-exciting process. *Journal of Applied Probability*, 11(3):493–503, 1974.

[Hor91] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[HR18] Blanka Horvath and Oleg Reichmann. Dirichlet forms and finite element methods for the SABR model. *SIAM Journal on Financial Mathematics*, 9(2):716–754, 2018.

[HS01] Joel Hasbrouck and Duane J Seppi. Common factors in prices, order flows, and liquidity. *Journal of financial Economics*, 59(3):383–411, 2001.

[HSW89] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[HSW90] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural networks*, 3(5):551–560, 1990.

[HW90] John Hull and Alan White. Pricing interest-rate-derivative securities. *The review of financial studies*, 3(4):573–592, 1990.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[Itk15] Andrey Itkin. To sigmoid-based functional description of the volatility smile. *The North American Journal of Economics and Finance*, 31:264–291, 2015.

[Jai15] Thibault Jaisson. Market impact as anticipation of the order flow imbalance. *Quantitative Finance*, 15(7):1123–1135, 2015.

[JCLP19] Eduardo Abi Jaber, Christa Cuchiero, Martin Larsson, and Sergio Pulido. A weak solution theory for stochastic Volterra equations of convolution type. *arXiv preprint arXiv:1909.01166*, 2019.

[JL19] H. Jang and J. Lee. Machine learning versus econometric jump models in predictability and domain adaptability of index options. *Physica A: Statistical Mechanics and its Applications*, 513:74–86, 2019.

[JMM18] Antoine Jacquier, Claude Martini, and Aitor Muguruza. On vix futures in the rough bergomi model. *Quantitative Finance*, 18(1):45–61, 2018.

[JPS18] Antoine Jacquier, Mikko S Pakkanen, and Henry Stone. Pathwise large deviations for the rough bergomi model. *Journal of Applied Probability*, 55(4):1078–1092, 2018.

[JR15] Thibault Jaisson and Mathieu Rosenbaum. Limit theorems for nearly unstable Hawkes processes. *The Annals of Applied Probability*, 25(2):600–631, 2015.

[JR16] Thibault Jaisson and Mathieu Rosenbaum. Rough fractional diffusions as scaling limits of nearly unstable heavy tailed Hawkes processes. *The Annals of Applied Probability*, 26(5):2860–2882, 2016.

[JR18] Paul Jusselin and Mathieu Rosenbaum. No-arbitrage implies power-law market impact and rough volatility. *Mathematical Finance*, 2018.

[KB15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *3rd International Conference for Learning Representations*, 2015.

[Kon18] Alexei Kondratyev. Learning curve dynamics with artificial neural networks. *Available at SSRN 3041232*, 2018.

[Kra88] D. Kraft. A software package for sequential quadratic programming. *DFVLR-FB*, pages 88–28, 1988.

[Kyl85] Albert S. Kyle. Continuous auctions and insider trading. *Econometrica: Journal of the Econometric Society*, pages 1315–1335, 1985.

[LBGO19] Shuaiqiang Liu, Anastasia Borovykh, Lech A Grzelak, and Cornelis W Oosterlee. A neural network-based framework for financial model calibration. *Journal of Mathematics in Industry*, 9(1):1–28, 2019.

[LC14] Mehdi Lallouache and Damien Challet. Statistically significant fits of hawkes processes to financial data. *Available at SSRN*, 2014.

[LCBP99] Laurent Laloux, Pierre Cizeau, Jean-Philippe Bouchaud, and Marc Potters. Noise dressing of financial correlation matrices. *Physical review letters*, 83(7):1467, 1999.

[Lev44] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

[LGO17a] Álvaro Leitao, Lech A Grzelak, and Cornelis W Oosterlee. On a one time-step monte carlo simulation approach of the SABR model: Application to european options. *Applied Mathematics and Computation*, 293:461–479, 2017.

[LGO17b] Álvaro Leitao, Lech A Grzelak, and Cornelis W Oosterlee. On an efficient multiple time step monte carlo simulation of the SABR model. *Quantitative Finance*, 17(10):1549–1565, 2017.

[LM19] Charles-Albert Lehalle and Charafeddine Mouzouni. A mean field game of portfolio trading and its consequences on perceived correlations. *arXiv preprint arXiv:1902.09606*, 2019.

[LMPR18] Giulia Livieri, Saad Mouti, Andrea Pallavicini, and Mathieu Rosenbaum. Rough volatility: evidence from option prices. *IISE Transactions*, 50(9):767–776, 2018.

[Loe18] Gregoire Loeper. Option pricing with linear market impact and nonlinear Black–Scholes equations. *The Annals of Applied Probability*, 28(5):2664–2726, 2018.

[Luk70] Eugene Lukacs. *Characteristic functions*. Griffin, 1970.

[MA99] F. Stella M. Avellaneda, A. Carelli. Following the bayes path to option pricing. *Journal of Computational Intelligence in Finance*, 8(4), 1999.

[Mar63] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[MBC+07] John G McWhirter, Paul D Baxter, Tom Cooper, Soydan Redif, and Joanne Foster. An EVD algorithm for para-hermitian polynomial matrices. *IEEE Transactions on Signal Processing*, 55(5):2158–2169, 2007.

[MBEB17] Iacopo Mastromatteo, Michael Benzaquen, Zoltan Eisler, and Jean-Philippe Bouchaud. Trading lightly: Cross-impact and optimal portfolio execution. *arXiv preprint arXiv:1702.03838*, 2017.

[McG18] William A McGhee. An artificial neural network representation of the SABR stochastic volatility model. *Available at SSRN 3288882*, 2018.

[Mha93] H. N. Mhaskar. Approximation properties of a multilayered feedforward artificial neural network. *Advances in Computational Mathematics*, 1:61–80, 1993.

[MP18] Ryan McCrickerd and Mikko S Pakkanen. Turbocharging monte carlo pricing for the rough bergomi model. *Quantitative Finance*, 18(11):1877–1886, 2018.

[MTB14] Iacopo Mastromatteo, Bence Toth, and Jean-Philippe Bouchaud. Agent-based models for latent liquidity and concave price impact. *Physical Review E*, 89(4):042805, 2014.

[Mug19] Aitor Muguruza. Not so particular about calibration: Smile problem resolved. *arXiv preprint arXiv:1909.13366*, 2019.

[NM65] John A Nelder and Roger Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[NW06] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer-Verlag, New York, 2006.

[PB18] Felix Patzelt and Jean-Philippe Bouchaud. Universal scaling and nonlinearity of aggregate price impact in financial markets. *Physical Review E*, 97(1):012304, 2018.

[Pow94] Michael JD Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Kluwer Academic (Dordrecht)*, pages 51–67. S. Gomez and J-P. Hennart, 1994.

[PV15] Paolo Pasquariello and Clara Vega. Strategic cross-trading in the us stock market. *Review of Finance*, 19(1):229–282, 2015.

[RAB11] Pierre-Alain Reigneron, Romain Allez, and Jean-Philippe Bouchaud. Principal regression analysis and the index leverage effect. *Physica A: Statistical Mechanics and its Applications*, 390(17):3026–3035, 2011.

[RS13] Luis Miguel Rios and Nikolaos V Sahinidis. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3):1247–1293, 2013.

[RT21a] Mathieu Rosenbaum and Mehdi Tomas. A characterisation of cross-impact kernels. *arXiv preprint arXiv:2107.08684*, 2021.

[RT21b] Mathieu Rosenbaum and Mehdi Tomas. From microscopic price dynamics to multidimensional rough volatility models. *Advances in Applied Probability*, 53(2):425–462, 2021.

[RY13] Daniel Revuz and Marc Yor. *Continuous martingales and Brownian motion*, volume 293. Springer Science & Business Media, 2013.

[Sai19] Emilio Said. How option hedging shapes market impact. *Available at SSRN 3470915*, 2019.

[SCC18] Uri Shaham, Alexander Cloninger, and Ronald R Coifman. Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis*, 44(3):537–557, 2018.

[SL19] Michael Schneider and Fabrizio Lillo. Cross-impact and no-dynamic-arbitrage. *Quantitative Finance*, 19(1):137–154, 2019.

[SP97] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[Sto20] Henry Stone. Calibrating rough volatility models: a convolutional neural network approach. *Quantitative Finance*, 20(3):379–392, 2020.

[TEB16] Bence Tóth, Zoltán Eisler, and Jean-Philippe Bouchaud. The square-root impact law also holds for option markets. *Wilmott*, 2016(85):70–73, 2016.

[Tit48] Edward C Titchmarsh. *Introduction to the theory of Fourier integrals*. Clarendon Press, 1948.

[TLD+11] B. Tóth, Y. Lempérière, C. Deremble, J. de Lataillade, J. Kockelkoren, and J.-P. Bouchaud. Anomalous price impact and the critical nature of liquidity in financial markets. *Phys. Rev. X*, 1:021006, Oct 2011.

[TMB20] Mehdi Tomas, Iacopo Mastromatteo, and Michael Benzaquen. How to build a cross-impact model from first principles: Theoretical requirements and empirical results. *arXiv preprint arXiv:2004.01624*, 2020.

[TMB21] Mehdi Tomas, Iacopo Mastromatteo, and Michael Benzaquen. Cross impact in derivative markets. *arXiv preprint arXiv:2102.02834*, 2021.

[Tor97] Nicolo Torre. BARRA market Impact model handbook. *BARRA Inc., Berkeley*, 1997.

[TWG19] Gerry Tsoukalas, Jiang Wang, and Kay Gieseckec. Dynamic portfolio execution. *Management Science*, 65(5):2015–2040, 2019.

[Ver12] Mark Veraar. The stochastic Fubini theorem revisited. *Stochastics An International Journal of Probability and Stochastic Processes*, 84(4):543–551, 2012.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[VRD09] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[VSS18] Marc Sabate Vidales, David Siska, and Lukasz Szpruch. Unbiased deep solvers for parametric pdes. *arXiv preprint arXiv:1810.05094*, 2018.

[WA59] Norbert Wiener and E.J. Akutowicz. A factorization of positive hermitian matrices. *Journal of Mathematics and Mechanics*, pages 111–120, 1959.

[WM57] Norbert Wiener and Pesi Masani. The prediction theory of multivariate stochastic processes. *Acta Mathematica*, 98(1-4):111–150, 1957.

[WMW15] Zeliang Wang, John G McWhirter, and Stephan Weiss. Multichannel spectral factorization algorithm using polynomial matrix eigenvalue decomposition. In *2015 49th Asilomar conference on signals, systems and computers*, pages 1714–1718. IEEE, 2015.

[WNG17] Shanshan Wang, Sebastian Neusüß, and Thomas Guhr. Grasping asymmetric information in market impacts. *arXiv preprint arXiv:1710.07959*, 2017.

[WSG15] Shanshan Wang, Rudi Schäfer, and Thomas Guhr. Price response in correlated financial markets: empirical results. *arXiv preprint arXiv:1510.03205*, 2015.

[WSG16] Shanshan Wang, Rudi Schäfer, and Thomas Guhr. Cross-response in correlated financial markets: individual stocks. *The European Physical Journal B*, 89(4):105, 2016.

[ZBLN97] Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23(4):550–560, 1997.

[ZTFL15] Elia Zarinelli, Michele Treccani, Doyne J. Farmer, and Fabrizio Lillo. Beyond the square root: Evidence for logarithmic dependence of market impact on size and participation rate. *Market Microstructure and Liquidity*, 1(02):1550004, 2015.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Le processus de formation de prix multivarié et l'impact croisé

Mots clés : microstructure des marchés, cross-impact, couts de transaction

Résumé : Cette thèse comprend six parties. La première lie les flux d'ordres anonymes et les variations de prix à l'aide de modèles d'impact croisé statiques et linéaires. Nous énumérons les propriétés souhaitables de ces modèles, caractérisons ceux qui les satisfont et les testons sur différents marchés. La deuxième partie étend cette approche aux produits dérivés afin d'obtenir une méthode d'estimation pour l'impact croisé que nous appliquons aux options SP500 et aux contrats à terme VIX. Dans la troisième partie, nous généralisons les modèles précédents pour prendre en compte l'influence des ordres passés sur les prix. La quatrième partie utilise des données de méta-ordres sur les actions et les contrats à terme pour proposer une formule d'impact croisé qui généralise la loi de la racine carrée de l'impact. Dans la cinquième partie, nous présentons un modèle tick-par-tick pour la dynamique des prix multivariés en utilisant les processus de Hawkes. La dernière partie résout le problème de la calibration des modèles de volatilité en utilisant des réseaux de neurones. La première partie étudie les modèles statiques linéaires pour l'impact croisé. Ces modèles dépendent des covariances des transactions et des prix. Nous introduisons des propriétés pour qu'un modèle se comporte correctement dans différentes conditions de marché. Nous montrons qu'il existe un seul modèle qui satisfait toutes ces propriétés. Nous appliquons différents modèles sur des actions et des contrats à terme. Le modèle précédent est l'un des deux modèles robustes sur les marchés étudiés. Il s'agit donc d'un bon candidat pour une vision unifiée du processus de formation des prix. La deuxième partie généralise l'approche précédente aux produits dérivés. Nous dérivons une méthode d'estimation pour l'impact croisé à partir de covariances de faible dimension. Sur des données de produits dérivés sur le SP500 et des contrats à terme VIX, le modèle explique en partie les fluctuations du sous-jacent et de la surface de volatilité implicite. Dans la deuxième partie, nous étudions des modèles linéaires à noyaux pour l'impact croisé. Nous examinons deux classes de modèles : ceux qui anticipent le flux d'ordres pour fixer des prix martingale et ceux qui empêchent l'arbitrage statistique. Nous montrons qu'il existe au plus un noyau appartenant aux deux classes mais qu'il n'empêche pas nécessairement l'arbitrage. Pour résoudre ce problème, nous introduisons un second noyau qui empêche l'arbitrage statistique et qui est le plus proche possible à donner des prix martingale. Enfin, nous calibrons ces noyaux sur des données de contrats à terme. La troisième partie mesure l'impact croisé avec deux bases de données d'ordres envoyés par des gestionnaires d'actifs sur des actions et des contrats à terme. Nous proposons une formule pour l'impact croisé qui généralise la loi de la racine carrée et donne des prédictions plus précises sur nos données. Dans la quatrième partie, nous modélisons le processus de prix tick-par-tick avec des processus de Hawkes. Pour capturer l'endogénéité des marchés financiers, nous étudions la limite où la norme L^1 du rayon spectral du noyau de Hawkes devient égale à un. Certains modèles de volatilité rugueuse multivariée émergent alors comme la limite macroscopique de la dynamique microscopique des prix. Le processus de volatilité de ces modèles est une combinaison de facteurs de variance entraînés par un mouvement brownien fractionnaire d'indice de Hurst commun. Enfin, la dernière partie de cette thèse examine la calibration des modèles de volatilité à l'aide de réseaux de neurones. Nous approchons la fonction donnant les prix des contrats à partir des paramètres du modèle en utilisant des réseaux de neurones. Cette approximation est ensuite utilisée pour obtenir les paramètres du modèle à partir de prix de marché des contrats. Nous mettons en évidence l'applicabilité de la méthode en utilisant des données de marché synthétiques et réelles.

Title : The multivariate price formation process and cross-impact

Keywords : market microstructure, cross-impact, transaction costs

Abstract : This thesis comprises six parts. The first relates anonymous order flow and price changes using static, linear cross-impact models. We list desirable properties of such models, characterise those which satisfy them and test their predictions on different markets. The second part extends this approach to derivatives to obtain a tractable estimation method for cross-impact which is applied to SP500 options and VIX futures. In the third part, we generalise the previous setup to derive and estimate cross-impact models which account for the influence of past trades on current prices. The fourth part uses meta-order databases on stocks and futures to propose a formula for cross-impact which generalises the square-root law of market impact. In the fifth part, we propose a tick-by-tick model for price dynamics using Hawkes processes. We investigate scaling limits of prices in the high endogeneity regime to derive multivariate macroscopic price dynamics of rough Heston type. Finally, the last part solves the calibration problem of volatility models using neural networks. In the first part, we study linear cross-impact models which relate asset prices to anonymous order flow. These models are functions of the covariances of these variables. We introduce properties models should satisfy to behave well across market conditions and show that there exists a unique model which satisfies all such properties. We apply models on stocks and futures and find that the latter model is one of two robust across markets. Thus, it is a good candidate model for a unifying view of the price formation process on stocks and futures. The second part leverages the candidate model identified in the first part to extend the previous setup to derivatives. We derive an estimation method for the large cross-impact matrix which depends on low-dimensional covariances. Using SP500 options and VIX futures data, we show cross-impact captures salient features of the price formation process on derivatives. The second part examines cross-impact kernels, which account for the lasting influence of past trades on current prices. We focus on two kernel classes : kernels that anticipate future order flow to set martingale prices and those that prevent statistical arbitrage. We show that there is at most one kernel belonging to both classes. This kernel sets martingale prices but may not prevent arbitrage. To fix this, we introduce a methodology to obtain a second kernel which prevents statistical arbitrage and is the closest to setting martingale prices. Finally, we derive a calibration methodology for both kernels and apply it to futures data. The third part measures cross-impact from using two databases of proprietary orders sent by asset managers on U.S stocks and futures. These databases allow us to study the cross-impact of individual investor orders. We propose a formula for cross-impact which generalises the square-root law to account for price and order correlations. On both stocks and futures, we find that this generalisation gives more precise predictions than the square-root law. In the fourth part, we model the tick-by-tick price process using Hawkes processes. To capture the high endogeneity of financial markets, we investigate the limit where the L^1 norm of the spectral radius of the Hawkes kernel goes to one. We show that some multivariate rough volatility models emerge as the macroscopic limit of the microscopic price dynamics. In these models, volatility is a combination of underlying variance factors, each driven by a fractional Brownian motion of common Hurst index. Finally, the last part examines the calibration of volatility models by using neural networks. We first approximate the map from model parameters to contract prices using neural networks. This approximation can then be used to recover model parameters given market prices of contracts. We highlight the applicability of the method using synthetic and real market data.

Institut Polytechnique de Paris
91120 Palaiseau, France