Multi-modal AI methods in the context of heterogeneous oceanic observations and multisensor maritime surveillance

Matteo Zambra

COLLEGES BRETAGNE LOIRE
SCIENCES POUR L'INGENIEUR ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE
MINES-TÉLÉCOM ATLANTIQUE BRETAGNE
PAYS DE LA LOIRE – IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 648
Sciences pour l'Ingénieur et le Numérique
Spécialité : *Signal, Image et Vision*

Par
Matteo ZAMBRA

Méthodes IA multimodales dans des contextes d'observation océanographique et de surveillance maritime multi-capteurs hétérogènes

Thèse présentée et soutenue à l'IMT Atlantique, Brest, le 19 Janvier 2024
Unité de recherche : LabSTICC, UMR CNRS 6285
Thèse N° : 2024IMTA0391

Rapporteurs avant soutenance :

Ricard MARXER	Professeur, Université de Toulon, France
Yannick BERTHOUMIEU	Professeur, Université de Bourdeaux, France

Composition du Jury :

Président :	**François ROUSSEAU**	Professeur, IMT Atlantique, France
Rapporteurs :	**Ricard MARXER**	Professeur, Université de Toulon, France
	Yannick BERTHOUMIEU	Professeur, Université de Bourdeaux, France
Examinateurs :	**Marie CHABERT**	Professeur, INPT - ENSEEIHT, France
	Julien BONNEL	Chercheur, WHOI, États-Unis
Dir. de thèse :	**Ronan FABLET**	Professeur, IMT Atlantique, France
Co-encadrement de thèse :	**Nicolas FARRUGIA**	Maître de conférences, IMT Atlantique, France
	Dorian CAZAU	Maître de conférences, ENSTA Bretagne, France

Invité(s) :

Alexandre GENSSE	Ingenieur, Naval Group, France

Summary (English)

This thesis aims to investigate the potential benefit of the simultaneous exploitation of heterogeneous and multi-sensor observations for the sea-surface state characterization. The significance of this work can be viewed in the perspectives of the availability of large oceanic data sets and the recent surge of interest in data-based AI methods for geoscientific applications.

Our contributions articulate in two complete case studies. The first investigation targets quantitative and qualitative improvement of sea-surface wind speed estimation using underwater passive acoustic data. A first axis of improvement is tied to model design choices. We show that an end-to-end neural network-based architecture that parameterizes a variational data assimilation scheme of the type weak-constrained 4DVar outperforms classic regressive predictors and purely neural network-based inversion models. The second axis addresses the task performance imputable to a multi-modal dataset, in which underwater acoustics is jointly used with sea-surface wind speed reanalyses (data assimilation hind-casting products). The inclusion of this second modality represents a supplementary information source for the wind speed reconstruction.

The second case study deepens the multi-modal approach including spatio-temporally heterogeneous data. The target variable is the sea-surface wind speed considered on a large spatial extent. The model chosen for this analysis is the same as for the first case study, i.e. a neural network-based parametrization of the weak-constrained 4DVar. Our analyses aim to assess the impact of each input source on the overall high-resolution wind fields reconstruction performance. We use simulated data in order to avoid any availability and compatibility issues in the input data. The heterogeneous input sources are obtained by the original data in order to have spatial high-resolution and low-resolution fields and local point-wise time series. These pseudo-observations aim to emulate satellite imagery, numerical weather prediction products and in-situ observations. The results are promising on both the scientific and operational aspects. We prove the model capability to flexibly incorporate the complementary information conveyed by the heterogeneous data and we show how this framework can be used to optimize the oceanic observation networks.

Résumé (Français)

Cette thèse vise à étudier les avantages potentiels de l'exploitation simultanée d'observations hétérogènes et multicapteurs pour la caractérisation de l'état de la surface de la mer. L'importance de ce travail peut être considérée dans la perspective de la disponibilité de grands ensembles de données océaniques et de l'intérêt récent pour les méthodes d'IA basées sur les données pour les applications géoscientifiques.

Nos contributions s'articulent autour de deux études de cas complètes. La première étude vise à améliorer quantitativement et qualitativement l'estimation de la vitesse du vent à la surface de la mer à l'aide de données acoustiques passives sous-marines. Un premier axe d'amélioration est lié aux choix de conception du modèle. Nous montrons qu'une architecture de-bout-en-bout basée sur un réseau neuronal qui paramètre un schéma d'assimilation variationnelle de données du type *weak-constrained* 4DVar est plus performante que les prédicteurs régressifs classiques et les modèles d'inversion purement basés sur un réseau neuronal. Le deuxième axe concerne les performances imputables à un ensemble de données multimodales, dans lequel l'acoustique sous-marine est utilisée conjointement avec des réanalyses de la vitesse du vent à la surface de la mer (produits d'assimilation de données à posteriori). L'inclusion de cette deuxième modalité représente une source d'information supplémentaire pour la reconstruction de la vitesse du vent.

La deuxième étude de cas approfondit l'approche multimodale en incluant des données spatio-temporelles hétérogènes. La variable cible est la vitesse du vent à la surface de la mer, considérée sur une large étendue spatiale. Le modèle choisi pour cette analyse est le même que pour la première étude de cas, c'est-à-dire une paramétrisation basée sur un réseau neuronal du *weak-constrained* 4DVar. Nos analyses visent à évaluer l'impact de chaque source d'entrée sur la performance globale de la reconstruction des champs de vent à haute résolution. Nous utilisons des données simulées pour éviter tout problème de disponibilité et de compatibilité des données d'entrée. Les sources d'entrée hétérogènes sont obtenues à partir des données originales afin d'avoir des champs spatiaux à haute et basse résolution et des séries temporelles locales ponctuelles. Ces pseudo-observations visent à imiter l'imagerie satellitaire, les produits de prévision météorologique numérique et les observations in situ. Les résultats sont prometteurs tant sur le plan scientifique

qu'opérationnel. Nous prouvons la capacité du modèle à incorporer de manière flexible les informations complémentaires véhiculées par les données hétérogènes et nous montrons comment ce cadre peut être utilisé pour optimiser les réseaux d'observation océaniques.

Remerciements

Ce travail est le résultat de trois années d'engagement personnel et professionnel au sein de l'école IMT Atlantique Bretagne-Pays de la Loire. Je tiens à remercier tout particulièrement mon directeur de thèse, le Professeur Ronan Fablet, pour son encadrement constant et surtout patient. Son expértise scientifique et professionnelle a été un exemple precieux dont j'ai l'honneur de pouvoir garder un souvenir direct. Avec la même gratitude, je remercie les deux autres membres de la direction de la thèse, M Nicolas Farrugia et M Dorian Cazau (Maîtres de conférences). Le pluralisme des compétences et des formations personnels n'a fait que rendre les échanges constants mieux nourris par des points de vue hétérogènes, avec une âme aux multiples facettes—multimodale, on dirait ! D'une manière générale, je dois un grand merci aux trois membres de la direction de ma thèse pour la confiance qu'ils m'ont accordée dès le moment où ils m'ont proposé de m'engager pour ce travail, et qu'ils ont continué à m'accorder tout au long de ces trois années. Ce que je peux dire d'avoir appris au cours de ma thèse, c'est aussi grâce à leur soutien et leur exemple.

Je addresse également mes remerciements les plus sincères à Naval Group pour avoir soutenu le projet de thèse à travers le financement. En particulaire, je souhaite exprimer ma gratitude à M Alexandre Gensse pour l'excellent accueil que m'a réservé pendant ma visite (bref mais très constructive) au sein du site Naval Group à Toulon. Par ailleurs, je tiens à remercier l'ANR pour le financement (Chaire AI Oceanix, ANR-19-CHIA-0016) et la région Bretagne pour la mise à disposition des ressources informatiques (à travers le projet CPER AIDA 2021 - 2027).

Je voudrais adresser mes remerciements à M Roberto Bozzano et Mme Sara Pensieri (CNR - Consiglio Nazionale delle Ricerche) pour avoir partagé les données sur lesquelles l'analyse présentée au Chapitre 3 est basée et pour avoir pris le temps d'échanger des informations sur cet ensemble de données et les applications précédentes.

Je souhaite dire un grand merci au personnel de l'école pour l'aide impeccable que j'ai reçu pour les démarches administratives. En particulaire j'addresse un remerciement à Mme Cynthia Nougé, Mme Aurelie Marcet, Mme Flora Christien, Mme Magali Gouez et Mme Marie-Dominique Pazat pour leur soutien et disponibilité. Autant, j'addresse ma

gratitude et reconnaissance au personnel des services du campus de l'IMT Atlantique de Brest pour leur gentillesse, leur disponibilité cordiale et leur travail. Tout ça m'a donné un sens d'accueil familiale depuis mes premières jours à l'école.

D'un côté plus personnel, je tiens à mentionner les collegues. C'est principalement grâce à vous que mes compétences linguistiques ont pu passer d'un niveau débutant à un niveau avancé, d'où ces pages de remerciements sont intentionellement écris en français (sans oublier de remercier les outils basés sur l'IA pour le traitement de language naturel). Je garderai un bon souvenir des moments de convivialité qui m'ont permis de m'immerger dans un monde différent et parallèle. Du tourisme, on revient le même. En voyageant, on se perd dans les autres et on revient changé. Ou on revient pas du tout.

Un grand merci à mes anciens amis. Nos chemins ont commencé à prendre des directions différentes, parallèles et parfois divergentes. L'important, ce qui me remplit de joie, c'est que—contrairement à la géométrie—ces différents chemins finissent par se rejoindre et, à ce moment-là, c'est comme s'ils ne s'étaient jamais séparés. Rares sont les amitiés qui résistent à l'épreuve du temps et de la distance, et je suis heureux de pouvoir dire cela de vous.

Je tiens à remercier tout particulièrement ma famille. À mes parents, merci de m'avoir toujours apporté votre soutien et votre confiance inconditionnels. Merci d'avoir toujours été à mes côtés et de m'avoir élevé dans la culture de la bonté et de l'intégrité. Merci également à ma sœurette. Hier, tu étais ma petite sœur, aujourd'hui tu es une grande femme et une personne dont je suis fier d'être le grand frère.

Enfin, un immense merci à ma chérie. Pour avoir cru fermement en ce projet, même et surtout dans les moments où je n'y arrivais pas. Ton soutien et ton infinie patience ont été tout simplement essentiels et irremplaçables.

List of contributions

Conferences

- **Zambra, M.**, Cazau, D., Farrugia, N., Gensse, A., Pensieri, S., Bozzano, R., & Fablet, R. (2023, June). Trainable dynamical estimation of above-surface wind speed using underwater passive acoustics. In OCEANS 2023-Limerick (pp. 1-6). IEEE.

Journal Papers

- **Zambra, M.**, Cazau, D., Farrugia, N., Gensse, A., Pensieri, S., Bozzano, R., & Fablet, R. (2023). "Learning-Based Temporal Estimation of In-Situ Wind Speed From Underwater Passive Acoustics", in *IEEE Journal of Oceanic Engineering*, vol. 48, no. 4, pp. 1215-1225, Oct. 2023, doi: `https://doi.org/10.1109/JOE.2023.3288970`.

Preprints

- **Zambra, M.**, Cazau, D., Farrugia, N., R., & Fablet, R. (2023). "Multi-Modal Learning-based Reconstruction of High-Resolution Spatial Wind Speed Fields", arXiv preprint arXiv:2312.08933.

TABLE OF CONTENTS

General introduction

Context and motivation

The ocean is a vast complex dynamical system that occupies the majority of the Earth's surface. As it is a main infrastructure for human activities such as commerce, fishing and energy production, its exploitation prompted a surge in interest and concern due to increasing pressure exerted on the marine ecosystems [1]. Much of the work devoted to ocean and sea research has involved scientific and governmental interventions [2] gathering the efforts of diverse communities to better study, characterize and describe the oceanic medium. On the strictly scientific side, the disciplinary fields contributing in sea and ocean studies include mathematical physics, biology, climatology, signal processing and, more recently, artificial intelligence (AI) and big data science. The disciplinary field of this thesis work is AI, in particular deep learning, for **oceanic data processing**. The objective is the characterization of the sea-surface state. This interest is encouraged by the large availability of multi-source and multi-sensor data. The availability of these observations is tied to the increasing number of maritime data collection infrastructures deployed worldwide [3]–[5]. In the wake of recent developments and advances in machine learning applied to geosciences [6]–[8], the motivation is to leverage such novel and promising techniques to fully exploit potential synergies in the information carried out by different data sources.

Aim and objectives

Given this context, the fundamental scientific questions that this thesis aims to approach are the following. We ask in which measure an hybrid approach aiming to join data-driven and physics-informed modelling is competitive with respect to classical methods based only on mathematical physical models or purely data-driven statistical methods. In a complementary way, we may further ask in which measure spatio-temporally heterogeneous and multi-sensor input data allow to improve the model performance. These general objectives in the framework of the practical thesis work can be restated as fol-

lows. The first application aims to evaluate in which measure a learning-based approach can improve the reconstruction performance of sea-surface wind speed using underwater acoustics. A further analysis concerns the performance improvement as a function of a multi-source dataset which is composed on underwater acoustics and synthetic model-derived wind speed. A second application concerns the exploitation of multi-scale and multi-dimensional information about sea-surface wind speed to reconstruct high-resolution time series of spatial wind speed fields. This second application aims to demonstrate the potential of an explicit treatment of spatio-temporally diverse data sources.

Structure of the manuscript

This manuscript is organized as follows. Chapter 1 gives the physical context and provides a general background on the most common observation and measurement techniques for wind speed at sea surface. Alongside with observations, numerical weather prediction and AI-based methods are presented and discussed. Chapter 2 introduces the methodological aspects that are approached in the development of the thesis work. In particular, the chapter is focused on data assimilation and deep learning concepts. The conclusive section is devoted to the introduction of the 4DVarNet framework, which is based on the previously mentioned concepts. Chapters 3 and 4 present the contributions of this thesis.

Part I

Background and methodology

Chapter 1

Background

1.1 Introduction

The oceans, the atmosphere and their interactions play a central role on life on Earth. They are responsible for global temperature regulation and heat distribution [9], carbon balance equilibrium between water and the atmosphere [10] and are the primary driving forces in the hydrological cycle [11], affecting climate and meteorological phenomena at all the spatio-temporal scales. Sea-air interactions influence ocean circulation and climate processes. This work concentrates on the strict interface area. The processes taking place at this compartment are manifold, schematically represented in Figure 1.1. Among the others, the wind speed dynamics is targeted and taken as main process of interest. Wind speed is a relevant phenomenon for a wide range of scientific and operational applications. To cite a few: natural resources exploitation, weather extreme events forecast and climate projections, marine environment and ecosystems monitoring.

Despite its paramount importance, sea-surface wind characterization is made difficult by the turbulent nature of geophysical fluid flows and the large spatial extent of the sea-air interface. Atmospheric phenomena behave differently at different spatio-temporal scales. Climate scale wind speed patterns are characterized by spatial dimensions of more that 5000 km with several-years duration. Weather phenomena (taking place at the *mesoscale*) develop on spatial dimensions of few to 1000 km and in rapid (sub-daily) time windows. The turbulence makes fluid flow fields three-dimensional, chaotic and multi-scale. These features make the prediction and description of flow fields extremely difficult. Moreover, the inter-scales spatio-temporal dependence make any simplification a potential source of inaccuracy. The interested reader may refer to the textbooks by Wallace and Hobbs [12] and Vallis [13] for a broader and more detailed overview on atmospheric dynamics. Turbulence is extensively treated by Batchelor [14] and Holmes *et al.* [15].

Due to the complexity of this physical system and its primary importance for the scientific and operational purposes mentioned above, a large set of observation sources

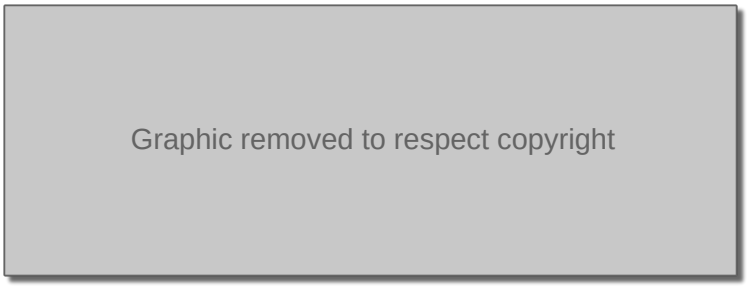


Figure 1.1 – A schematic illustration of the physical processes taking place at the sea-air interface. Image source: Woods Hole Oceanographic Institution `https://www2.whoi.edu/site/casimas/`.

and modelling techniques have been employed over the years. The most relevant sources and techniques in the context of this thesis are (i) numerical weather prediction (NWP) **models**, (ii) direct (in-situ) and indirect (remote) **observations**, (iii) sea-surface variables **reanalyses** (obtained with data assimilation schemes) and (iv) **AI**-based methods. These sources of information will be detailed in the following sections. Despite the abundance of sources of information, these can describe the spatio-temporal variability of the wind speed realization on one given instant. These sources of information are inter-dependent and target different compartments of the Earth system at different spatio-temporal realization scales. Mathematical modelling lays upon scientific understanding of natural phenomena and is expressed in mathematical formalism [16]. See [17] for a complete textbook dedicated to the subject. In-situ and remote observations [18]–[20] provide "physical" measurements of some environmental variable. Such data are valuable *per se* in weather and climate research but are also essential to validate the outputs of mathematical models numerical integration [21]. Conversely, well established models may provide reference (synthetic) data to evaluate the profitability of new data sources by means of observations system simulation experiments [22]. Data and models interact at many levels [23]. An higher level of this fusion is represented by data assimilation schemes [24]. In data assimilation, the numerical integration of the model is combined with the information coming from massive amounts of real observations. This data-model interaction allows to obtain the best estimate of the system state. In addition, recent years have seen a surging interest in AI data-driven modelling to approach the challenges posed by the oceans characterization [25], [26]. Unlike data assimilation, the most widespread AI-based methods do not need, by their design, a prior physical knowledge to operate but they are primarily and foremostly dependent on large volumes of data.

Method	Sensor / Model	Spartial coverage	Spatial resolution	Temporal resolution	Typical RMSE
In-situ	WindSonic2D anemometer [27]	None	Local	Sub-hourly	0.2 $\mathrm{m\,s^{-1}}$
Reanalyses	ECMWF ERA5 [28]	Global	$\sim$ 31 km	Hourly	1.5 – 2 $\mathrm{m\,s^{-1}}$
SAR	Sentinel-1A [29]	$\sim$ 600 km	0.02 km	Daily	1.6 $\mathrm{m\,s^{-1}}$

Table 1.1 – Summary table of the spatio-temporal scales targeted by three selected observation techniques.

The rest of this chapter is structured as follows. Section 1.2 discusses the observations techniques available for monitoring wind speed at sea surface. The sensors and platforms most relevant for the scope of this thesis are presented. Section 1.3 gives an overview of the methods based on NWP modelling and data assimilation schemes. Section 1.4 closes the chapter giving an overview of AI methods that stand as complementary to the traditional methods outlined in the other sections referred to above. Section 1.5 defines formally the terms used in this thesis to reference the observations scales and natures diversity.

Take-home message This introductory section aimed to present the complexity of the ocean/atmosphere description tasks. Due to the heterogeneous spatio-temporal scales involved and the physical conceptual difficulty, no such thing as a universally suited, simple and all-comprehensive model is available and is not going to be so in the foreseeable future. To remediate this, the necessity of exploring and understanding the oceanic environments is addressed by a set of methods and techniques that involve both observations, modelling and interaction between them. Table 1.1 gives an introductory overview of the wind speed measurement or reconstruction performance expected for three selected methods. These methods and techniques will be discussed in further detail in the following sections. As a methodological note, the root mean squared error is widely used in the operational practice as well as in scientific research. This score provides an average value of the quadratic error between an estimation and the true value of a given variable. The root mean squared error is expressed in physical dimensions. In the case of wind speed, it is expressed in meter per second ($\mathrm{m\,s^{-1}}$). Figure 1.2 provides a graphical visualization of Table 1.1. This visualization helps to see the different spatio-temporal scales targeted by the methods chosen.

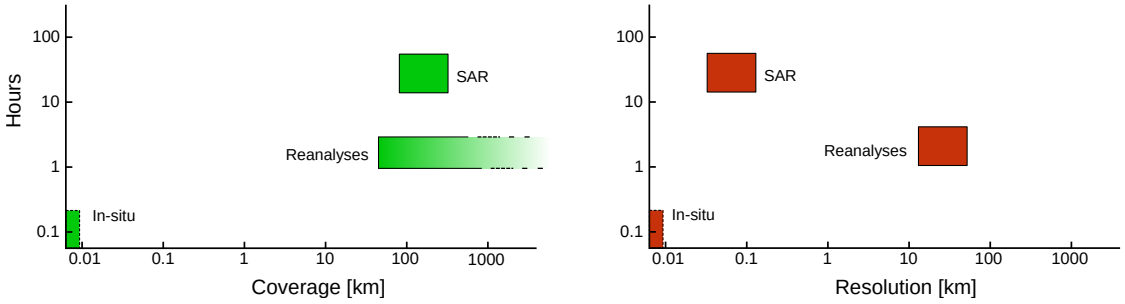


Figure 1.2 – Graphical representation of the spatial coverage and resolution as a function of temporal resolution for the selected observation and measurement techniques stated in Table 1.1. Note that in-situ techniques are represented graphically by rectangles for visual convenience but their coverage in space is point-wise w.r.t. the position of the sensor. The same remark applies for the spatial resolution. A local sensor can not have a spatial resolution since it can not provide spatial observations.

1.2 Observation techniques

As mentioned above, observations are divided into direct (local, in-situ) and remote. In the following, an overview of both is provided, with a particular focus on the techniques and methods commonly used to estimate the wind speed at sea surface.

1.2.1 In-situ observations

All the observation techniques that involve a direct contact between the instrument and the phenomenon to measure are called *in-situ* methods. The main advantages of in-situ data are the extensive temporal coverage and high-rate temporal sampling frequency, allowing the analysts to have a sound description of the phenomenon temporal evolution. However, such measures should be understood and interpreted in the sense of point-wise description since one sensor measurement does not have any spatial coverage. In this sense, in-situ data provide a fine-scale measure of the observed process. This feature represents a main limitation of in-situ techniques. We may further emphasize that in-situ observations do not provide any meaningful information about wind speed spatial variability, unless a dense and regular buoys network is deployed. This operation, however, would imply prohibitive efforts in terms of both economical costs and personnel safety. In this section, a self-contained presentation of the main sensors measurement platforms for wind speed is provided. Figure 1.3 gives a graphical overview of the most common in-situ observation

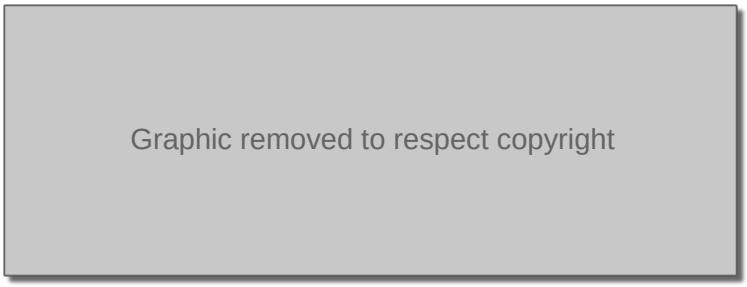


Figure 1.3 – Main in-situ observation methods and techniques. Source: Ocean Explainers by Copernicus Marine Service / Mercator Ocean International `https://marine.copernicus.eu/explainers/operational-oceanography/monitoring-forecasting/in-situ`

techniques, not only for wind speed.

Direct measures of wind are mainly made with anemometers. In general, an anemometer is an instrument that gives a measure of wind speed (some models can measure direction as well), based on different physical processes. **Cup anemometers** [30] are the simplest version and they put in relationship wind speed with the mechanical action of wind with the movement of three and four cups installed on a fixed pivot. Wind speed moves these cups and the angular velocity of the cups is used to compute the wind intensity. **Hot-wire anemometers** [31] allow to evaluate the wind speed intensity by the cooling rate of an hot wire. This hot wire is maintained at constant temperature. The energy required to do so can give information about the wind speed. **Ultra-sonic anemometers** use two transducers to measure the back-and-forth travel time of an ultrasonic pulse. This can be put in relationship with wind speed. The advantage of this sensor, other than the accuracy, is the capability to measure wind speed direction, unlike cup and hot-wire anemometers.

These sensors can find place on different observation infrastructures and platforms. One common such installation is the **moored buoy** [32]–[34]. Moored buoys are fixed stations that may be equipped with multiple sensors to measure temperature, pressure and wind speed, among others. They provide reliable, high-quality measurements and represent a cost-effective method [35]. Although, the measurements of moored buoys relate to the phenomenon realization on the strict geographical proximity of the platform. Moreover, due to the effort to install them, they are typically deployed in the strict proximity of coastlines [36]. Another solution is the installation and deployment of wind speed-related sensors on **research vessels** [37], [38]. This alternative allows to extend the portion of

surface explored at limited deployment cost. Still, the limited coverage problem persists. A similar solution is represented by **drifters** [39], [40], that is, floating weather buoys. These platforms are free to drift with the ocean currents.

To conclude this brief introduction to in-situ sensors for wind speed measurement, another point of interest in the scope of this work is represented by **underwater acoustics** measurements [41]. The discipline of acoustic meteorology aims to infer the above-surface atmospheric state from the underwater soundscape. The work of Nystuen [42] is a first example of this research area. Underwater noise is measured with fixed hydrophones [41], [43], floating profilers [44], [45] or bio-logged mammals [46]. These instruments collect underwater noise that can be related to different geophysical phenomena such as wind speed and rainfall.

1.2.2 Remote sensing

Remote sensing is the process of relating one given physical variable on a given area and the Earth surface-reflected radiation. This process is done with air or space-borne carried sensors. Unlike in-situ techniques, remote sensing methods capture spatially wide observations of the Earth surface. Applications involving sea-surface wind speed benefit from the simultaneous spatial variability captured by remotely sensed fields [47]. Remote sensing is distinguished in passive [48] and active [49] methods. Passive sensors receive and interpret electro-magnetic radiation naturally backscattered by the Earth surface. Active sensors emit the radiation themselves, so they can capture and interpret the backscattered information. The geophysical parameters observed by remote sensors are manifold. The measurement of wind speed at sea surface can be accomplished with **scatterometers** [50] and **Synthetic Aperture Radar** (**SAR**) [51]. SAR imagery can deliver surface wind observations with a spatial resolution of 0.5 to 1 km. SAR sensors estimate sea-surface wind speed by treating the normalized radar cross section (NRCS). NRCS, called σ^0, is a non-linear function of the emitted pulse frequency, the polarization and the wind speed and direction [29]. Several inversion algorithms exist for this inversion [52]. These inversions are based on a chosen geophysical model function (GMF, [53]) that relates the raw signal σ^0 and the wind speed and directions. SAR data need NWP or in-situ data to be properly calibrated. The interested reader is referred to [54] for a complete review on SAR-based sea-surface wind speed measurement and to [29] for a complete explanation of SAR imaging.

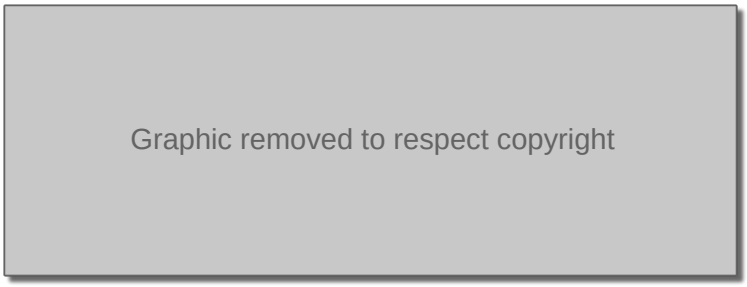


Figure 1.4 – An example of Sentinel-1 SAR product. Data source: French access to Sentinel product service `https://peps.cnes.fr/rocket/#/home`.

To give a practical example, C-band [1] SAR sensors are mounted on Sentinel-1 satellites launched by the European Space Agency. These satellites provide 1×1 km resolution ocean wind products of extent 170×80 km. The Sentinel-1 A and B satellites have been launched in 2014 and 2016 respectively and have an expected lifetime of seven years. The reader is referred to [55] and [56] for a detailed overview of the Sentinel missions.

For the purpose of wind speed monitoring, remotely acquired information represents a valuable source of information. However, the main limitation of satellite observations resides in the poor revisit period of the satellite sensor on a given region. Due to the rapidly evolving surface wind speed patterns, a revisit period of 12 hours or more (as usual for SAR imagery) prevents a time-continuous description of the phenomenon.

1.3 Mathematical modelling and data reanalysis

In the following subsections, the products provided by mathematical atmospheric models and data assimilation systems are discussed. This presentation is kept essential for the purpose of introducing some examples of both products, which are directly touched by these thesis' contributions.

1.3.1 NWP models

A NWP model conceptualizes a given physical law. In geoscientific applications, these laws involve momentum, mass, energy conservation balances and thermodynamic state relations. Due to the mathematical complexity of these models, a direct analytical solution

1. C-band refers to the sensor-emitted pulse frequency. The C-band is associated to a 5.3 GHz frequency.

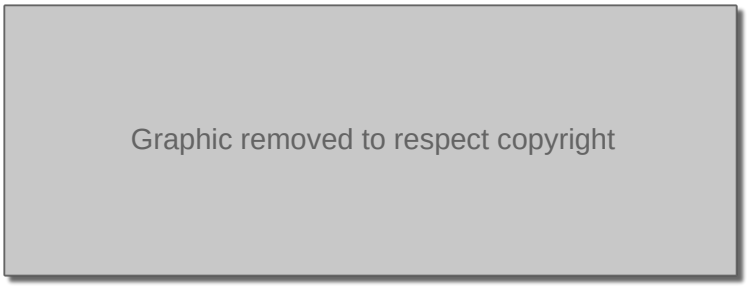


Figure 1.5 – Surface wind speed forecast by the WRF model. Image Source: `https://www.mmm.ucar.edu/models/wrf`

is practically not possible. The alternative is represented by numerical simulations [57]. The simulation scheme depends on the spatio-temporal scales involved by the processes addressed, e.g. Earth rotation and inertial forces become relevant at large and small spatial scales respectively. We list here some relevant examples of NWP models.

The **WRF** (Weather Research and Forecasting model [58]) is an example of mesoscale model, developed by the National Ocean and Atmospheric Administration (NOAA) and National Center for Atmospheric Research (NCAR). WRF is an atmospheric model but can be coupled with ocean models. It has regional coverage and has grid spacing ranging from meters to kilometers, depending on the configuration. Its applications involve short-range weather forecast and climate research. Figure 1.5 shows an example of the WRF wind speed forecast. The **IFS** (Integrated Forecasting System [59]) and **GFS** (Global Forecast System [60]) models, on the other hand, are global weather prediction models and, as such, they have larger horizontal grid spacings, from several to tens of kilometers. They are developed, respectively, by the European Center of Medium-Range Weather Forecast (ECMWF) and by NOAA. They are used for weather and climate prediction at short and medium term. The coverage of such models is larger than the mesoscale dimension but they are mainly used for continental forecast services on Europe and North America.

1.3.2 Data assimilation

Data assimilation refers to a set of methods that aim to estimate the atmospheric or ocean state [61]–[63]. The main use of data assimilation schemes is to combine information from a NWP model integration and real observations to find the optimal model state and/or parameters. Among the most important parameters, the initial and boundary con-

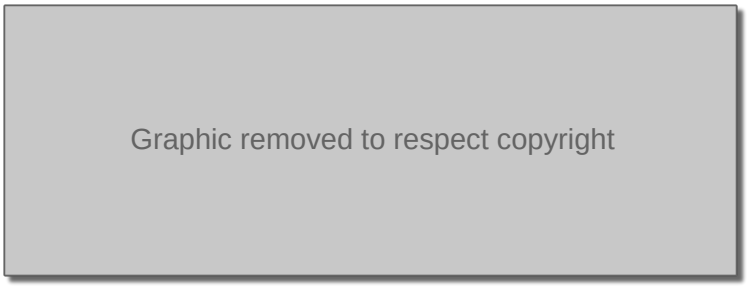


Figure 1.6 – ERA-5 and CERRA products. Image source: Climate Reanalyses by Copernicus Climate Change Service, accessible at `https://climate.copernicus.eu/climate-reanalysis`.

ditions play a crucial role to initialize the NWP model forecast. At the time that new observations are available, the data assimilation scheme produces an *analysis*, that is the optimal estimation of the system state given the model forecast and the observations. In this way, the forecast and analysis steps are repeated sequentially to estimate and update the system state and parameters. The same process can be performed on assimilation time windows before current time. This *hind-casting* procedure provides *re-analyzed* historical data. Reanalyses benefit from quality improvement, such as resolution and grid homogeneity. For its capability of forecasting and hind-casting, data assimilation systems are implemented by several operational weather services centers. Notable examples are the aforementioned ECMWF for Europe and the National Centers of Environmental Prediction (NCEP) in the United States of America. Notable example of reanalyses are the ERA-5 dataset by ECMWF [64] and NCEP/NCAR dataset by the agencies referred to in the name itself [65].

For the sake of introducing the contributions of our work, a brief overview of ERA-5, the fifth generation of ECMWF reanalyses databases, is provided. ERA-5 reanalyses are available from 1940 onwards. They have spatial grid spacing of $0.25° \times 0.25°$ and temporal resolution of 1 hour. This database comprehends a large list of atmospheric variables, such as the horizontal components of wind speed at 10 m level, temperature at 2 m, convective precipitation, geopotential and many others. The ERA-5 reanalyses are obtained by integrating the IFS model referred to above, in particular its Cy41r2 version [66]. These forecast are integrated in a 4DVar assimilation scheme. Chapter 2 will provide an explanation about 4DVar. The reader may refer to [64] for a complete review on the ERA-5 reanalyses database. Despite being state-of-the-art methods for wind speed forecasting and reconstruction, reanalyzed wind speed fields suffer some limitations. The

ECMWF reanalyses users feedback often points out that the spatial resolution of the products may not suffice to resolve the smaller scales. Indeed, the 30 km resolution of ERA-5 may neglect smaller scales phenomena. One other limitation concerns *model errors*. These errors imply notably timing and/or intensity errors [67].

To conclude this section, it is worth recalling that ERA-5 products have global coverage. As such, their capability of performing accurate description of smaller-scale phenomena may be inadequate. Regional reanalyses (RRA), for example the Copernicus RRA for Europe (CERRA, [68]), address this issue. ERA-5 is used as forcing and boundary conditions for higher resolution grids with horizontal spacing of 5.5 km. These reanalyses are obtained by integrating regional-scale numerical models, as the IFS is more suited for larger-scales forecasts. Figure 1.6 provides a visualization of ERA-5 and CERRA products.

1.4 AI and statistical modelling

Recently, the application of AI (in particular deep learning) has been largely considered as a methodological tool complementary to the state-of-the-art methods in geosciences [69], [70]. This kind of synergy has been prompted by the increasing affordability of computational resources, advances in the machine and deep learning field and the availability of bigger volumes of geophysical data bases [7]. Machine Learning models provide flexible representation of high-dimensional dynamical systems and computational efficiency. This makes it well suited for simulation and inverse problems [7], long-term forecast [6], remote sensing applications [71] and geophysical fields super-resolution [72]. The particular application to remote sensing is itself an active research field [73] due to the promising cross-fertilization of computer vision and computational imaging techniques, deep learning modelling and the availability of remotely sensed data. One other interesting intersection point regards the deep learning-assisted satellite products compression. Recent work proposed the use of neural networks [74] to perform multi-rate compression of remotely sensed products. The authors of [75] use a learnable probabilistic framework to reduce the time and memory complexity of the same task. Generally, deep learning represents an attracting methodology to approach the solution of inverse problems [76]. Geoscientific and geophysical applications often have to face such problems. The relevant deep learning concepts will be introduced in depth in Section 2.3.

The data assimilation techniques are considered the state-of-the-art in geoscientific modelling. For the reasons mentioned above, AI-based methods present themselves as

complementary techniques. In some cases, deep learning-based models are used as auxiliary modules of data assimilation schemes, for example to improve the resolution of the system state [77] or to characterize errors dynamics [78], [79]. In other cases, data assimilation schemes are embedded in an end-to-end trainable architecture [80]. This means that the inversion is performed by an end-to-end trainable model [7]. The complementarity between deep learning and data assimilation stems from the effectiveness of flexibility and efficiency of deep learning and the physical and theoretical back-end of data assimilation. To complete this discussion, data assimilation can be viewed as an algorithm to solve a geophysical inverse problem. Deep learning can be seen as a programming framework. A data assimilation scheme could be designed and implemented numerically with the programming frameworks used for deep learning. In this way, the data assimilation scheme would benefit from the automatic differentiation capabilities of deep learning tools [81]. Both data assimilation and deep learning as strategies to tackle inverse problems in geophysical applications will be introduced in detail in the following chapter.

Deep learning on its own has also rapidly outperformed state-of-the-art results is some specific applications. Some notable examples are provided in the following. Inpainting and interpolation are techniques that allow to fill spatially occluded data. **Image inpainting** [82] refers to the technique of restoring deteriorated images. Remote sensing products may suffer of field occlusions due to meteorologic conditions or other causes. Deep learning-based approaches have been proposed to address this problem [83]–[85]. **Interpolation** refers to the technique aiming to find intermediate values of a given field from discrete and sparse observations [86]. The flexibility of deep learning-based modelling allows to perform trainable interpolation for geophysical fields [87]–[89]. For the scope of this thesis, the **multi-modal** approach is particularly relevant. Multi-modal machine learning [90] aims to jointly learn from multi-modal data sources. Different sources of data convey complementary information about the domain. Multi-modal deep learning approaches have been proposed to treat the heterogeneity of geophysical observations [91]–[93]. Other interesting recent developments are **physics-informed machine learning** [94], [95] and **theory-guided data science** [96]. These approaches endow machine learning and data science models with physical constraints and informs such models with theoretical physical knowledge, otherwise not encoded in data-driven modelling. The model is trained to simulate the system behavior accounting for such constraints. The applications of these methods include learning partial differential equations [97] and weather systems forecasting [98].

1.5 Glossary of data-related terms

Through this work, we refer extensively to observations and sources of information that have diverse nature and spatio-temporal sampling patterns. The introductory sections presented in this chapter made explicit reference to the spatio-temporal variety and the deployment of different sensors to capture many features of one given natural phenomena, with particular reference to wind speed at the sea surface. This section closes the chapter providing the definitions of the data variety that will be referenced in the chapters devoted to the contributions.

- **Heterogeneity**. With heterogeneity we mean the different spatio-temporal resolutions and coverage that characterize two or more sources of information. For example, a dataset that involves both satellite images and reanalyses (for example of sea-surface wind speed) is inherently heterogeneous inasmuch the spatial resolution of a satellite (for example SAR) product may be one order of magnitude finer than that of the reanalysis product (cfr. Sections 1.2.2 and 1.3.2).
- **Multi-modality**. The concept of multi-modality has a more relevant impact at the level of model design and implementation. A multi-modal dataset contains diverse information that one given model should be capable to ingest and process simultaneously. For example, a multi-modal dataset may be composed of satellite images and in-situ time series. These two sources of information benefit of a large body of independent processing techniques. We may refer to [99] and to [100] for complete accounts on time series analysis and computational image processing, respectively. Multi-modality may also refer to the different nature of the variables involved, for example in the case where time series of rainfall and pressure are analysed.
- **Multi-sensor**. A multi-sensor dataset is composed of different observation channels that are captured necessarily with different sensors. For example recall the measurement instruments introduced in Section 1.2.1 in relation with the wind speed measurement. In this sense, one given variable may be measured with different kinds of instruments for the different measurement properties of each instrument.

We may emphasise that these definitions are not mutually exclusive, in fact they often overlap. A multi-sensor dataset is very likely a multi-modal and heterogeneous dataset since different sensors may capture different features of the phenomenon at dif-

ferent spatio-temporal scales. By contrast, a multi-modal dataset may not necessarily be heterogeneous, since this dataset may be composed of wind speed, pressure and temperature reanalyses products defined on the same spatial grid with the same temporal resolution and coverage.

To anticipate briefly and to apply these definitions to the case studies presented in the contributions chapters, we may characterize the dataset used as follows. In the case study presented in Chapter 3, we use a dataset composed of underwater acoustic data and wind speed reanalyses. The objective is to retrieve sea-surface wind speed. In this case, the wind speed reanalyses are referred to $1° \times 1°$ grid cells. For the purpose of our analyses, we consider grid value as representative of the local neighborhood of the observation infrastructure hosting hydrophone that collects acoustic data. In this case, the dataset is multi-modal and multi-sensor but not heterogeneous. The data have different natures (underwater acoustics and reanalysed wind speed) but both are local in space and have been chosen in such a way to have the same temporal coverage. In the second case study, presented in Chapter 4, we use model data (cfr. Section 1.3.1) to simulate satellite images, reanalyses and local in-situ time series. This dataset is heterogeneous (due to the different spatio-temporal resolutions of each simulated modality) and multi-modal (due to the different nature of these pseudo-observations). The use of model data does not allow to define such dataset multi-sensor. This would be the case if real data are used, like remotely sensed spatial fields and point-wise time series.

Chapter 2

Methodology

2.1 Inverse problems in geophysics

In geophysics, one often encountered problem is the estimation of a set of parameters or one system's state given a set of observable quantities [101], [102]. Focusing on a given physical system, let us denote **model** the formal representation of the system behavior, subjected to physical laws. Let the **parameters** be a set of quantities that characterize the model. The **system state** is the complete description of the system configuration at any instant of a given realization that involves the system. Given a complete model characterization (parameters comprehended), the system states trajectory can be fully described, retrieved and predicted. Mathematical modelling is concerned with the formulation of the governing equations of one given physical system. These equations describe the past, present and future of this system. For example, a simple system could be a point particle P, the system's state is the position $\mathbf{x}$ of the point at any time t, that is $\mathbf{x}(t)$. The model is the equation of motion that describes the dynamical behavior of P, $\dot{\mathbf{x}}(t) = f(\mathbf{x}, t; m)$. The mass of the point, m is the parameter that fully characterizes the model.

Given a model, any information that can be obtained about the system state is provided by an **observation operator**, $\mathcal{H}$, that maps the state space X (where system states reside) to the observations space Y. The observation operation can be stated as (omitting the temporal dependence not to lose generality)

$$\mathbf{y} = \mathcal{H}(\mathbf{x}) \tag{2.1}$$

In the case of the motion of the point P, one could directly observe the object up to some observation noise, such that $\mathbf{y}(t) = \mathbf{x}(t) + \boldsymbol{\epsilon}(t)$, where ϵ is a noise process. This observation process relates with the "sampling" operation: a continuous signal is converted to a discrete signal. Given the knowledge of the model and the observation operator, this is a simple

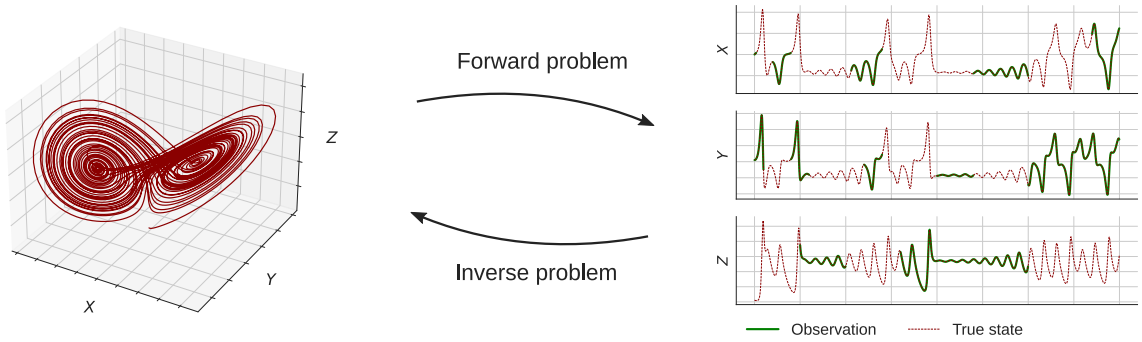


Figure 2.1 – Schematization of the forward and inverse problem realizations for the dynamical system proposed by Lorenz [104].

process that can deliver information about the system states at any instant. This is a **forward** problem, since the action of the observation model $\mathcal{H}$ samples the system state and returns its observable version.

The **inverse problem** is the inversion of the forward problem. It accounts for finding the system state given its observations. The model reconstruction from the observations aims to retrieve the system state $\mathbf{x}$ or the model parameters. In the simple example of the point particle, there is a straightforward relationship between states and observations. The information about the system state $\mathbf{x}$, the particle point position, can be directly converted to a time series of coordinates of the point, with the addition of measurement noise. In real world applications this is never the case. A direct inversion would require an exact theory and infinite and noise-free observations [103], but the cases in which it is possible are rare. Major limitations stem from the (necessarily) simplistic modellization of the problem domain (e.g. solid Earth, atmospheric or ocean media physical models unavoidably contain conceptual errors). Furthermore, the model to be reconstructed is a function of infinite-dimensional space variables and the observations are necessarily finite-dimensional. This makes the inversion an ill-posed problem.

This work is particularly concerned with geosciences, and in particular the wind speed dynamics at the sea surface. Notable examples of inversions for this case, treated in the Chapters 3 and 4 of this thesis, are the following. Our first study case exploits underwater passive acoustic data collected with hydrophones, which relate with the above-surface state. The inverse problem is to quantify the wind speed intensity shearing at the sea-air interface from the underwater ambient noise. Our second study case consists in reconstructing high-resolution time series of spatial wind speed fields at the sea surface from

(i) the interpolation of sparse high-resolution large-extent observations, (ii) the super-resolution of NWP products. In these cases, the inversion aims to retrieve the original high-resolution state from its partial observations.

The remainder of this chapter introduces the inversion methods that have been chosen for the works presented in this thesis. Section 2.2 presents data assimilation. Section 2.3 gives an overview on deep learning and Section 2.4 introduces the 4DVarNet framework.

2.2 Variational data assimilation

2.2.1 Preliminary statements

Data assimilation is the instance in the field of geosciences of dynamical systems state estimation and system parameters estimation. The objective is to estimate the state of the geophysical system using the knowledge of both real observations and the physical model describing the phenomena. The problem of estimating the state of a large and complex dynamical system, such as the atmosphere or the ocean, starting from a finite base of observations falls under the set of inverse problems described above. The process involves the extraction and fusion of information coming from different sources at a given time. The natural formalism for data assimilation is often that of discrete evolutionary equations [1] of the system state and the state-observable relations. The standard starting point is the *state-space formulation* at a discrete time t, as follows [2]

$$\begin{cases} \mathbf{x}_t &= \mathcal{M}_t(\mathbf{x}_{t-1}) + \boldsymbol{\eta}_t \\ \mathbf{y}_t &= \mathcal{H}_t(\mathbf{x}_t) + \boldsymbol{\epsilon}_t \end{cases} \tag{2.2}$$

The symbols are the following. $\mathbf{x} \in X$ and $\mathbf{y} \in Y$ are the system's state and the observations respectively. The state space and observations spaces $X \subset \mathbb{R}^m$ and $Y \subset \mathbb{R}^d$ typically have different dimensions as $d < m$. This means that the observations have a much smaller dimensionality than the true state. The operator $\mathcal{M}_t$ is a non-linear dynamical operator considered at time t, that maps one system state into its next realization. The operator $\mathcal{H}_t$ is the possibly non-linear observation model that maps the state space to the observations space, delivering observable information about the state $\mathbf{x}$. The processes $\boldsymbol{\eta}_t$ and

1. The more general continuous formulation may equally be used as formal starting point, although it must necessarily be discretized for the proper application.

2. A complete statement would include the model parameters explicitly in this formulation. Since parameter estimation is beyond the scope of this introductory presentation, its dependence is omitted.

$\boldsymbol{\epsilon}_t$ are respectively the model errors and the observation errors. The former represent the cumulative numerical errors, in the model parameterization as well as effects of unresolved scales, while the latter is an observation error and is imputable to measurement noise and representativeness error due to the measurement instrument itself [61].

Given the fundamental random nature of the states and observation processes, a statistical framework is often chosen to frame the data assimilation problem into. The focus shifts on the probability distributions of the states and observations, especially the posterior distribution on the system's state *conditioned* by the observations, $p(\mathbf{x}|\mathbf{y})$. The posterior above can be stated by the Bayes theorem as follows

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x})}{\int p(\mathbf{x},\mathbf{y})\, p(\mathbf{x})\, d\mathbf{x}} \tag{2.3}$$

Call $p(\mathbf{y}|\mathbf{x})$ the *likelihood*, that is the probability of having observed the data $\mathbf{y}$ due to the value attained by the state $\mathbf{x}$. Call $p(\mathbf{x})$ the prior, which carries information about the realization of the state. The density $p(\mathbf{x},\mathbf{y})$ is the *joint* distribution and its explicit evaluation, alongside with the normalization constant of the last side of Equation (2.3), is seldom analytically and computationally tractable.

In data assimilation, it is important to lay solid assumptions about the errors. The common assumption is to consider i.i.d. processes. The equations of the state-space formulation (2.2) can be restated in terms of probability densities, in light of these remarks, as

$$\begin{cases} p(\mathbf{x}_t|\mathbf{x}_{t-1}) &= p_{\boldsymbol{\eta}}[\mathbf{x}_t - \mathcal{M}_t\left(\mathbf{x}_{t-1}\right)] \\ p(\mathbf{y}_t|\mathbf{x}_t) &= p_{\boldsymbol{\epsilon}}[\mathbf{y}_t - \mathcal{H}_t(\mathbf{x}_t)] \end{cases} \tag{2.4}$$

Enforcing temporally independent errors, then the probability distributions for the errors processes can be factorized in products of probabilities. One additional common assumption is that the states process is a first order process, i.e. $p(\mathbf{x}_i|\mathbf{x}_{0:i-1}) = p(\mathbf{x}_i|\mathbf{x}_{i-1})$. These assumptions allow to state Equation (2.3) as

$$p(\mathbf{x}_{0:T}|\mathbf{y}_{0:T}) = C^{-1}\, p(\mathbf{x}_0) \prod_{t=1}^{T} p_{\boldsymbol{\epsilon}}[\mathbf{y}_t - \mathcal{H}_t(\mathbf{x}_t)]\, p_{\boldsymbol{\eta}}[\mathbf{x}_t - \mathcal{M}_t\left(\mathbf{x}_{t-1}\right)] \tag{2.5}$$

where C is the normalization constant and $p(\mathbf{x}_0)$ is the prior on the initial state.

This framework may be adapted in both sequential (and ensemble) and variational data assimilation schemes. Thanks to the probabilistic formulation, it is possible to detail and make more specific assumptions about the errors distributions. With no loss of

generality, the bayesian framework detailed above allows to treat three cases, depending on the temporal horizon of the state estimation. The first case is **prediction**, where the objective is to predict a future state, related to a time step where no observation is available yet. This accounts to the estimate of the function $p(\mathbf{x}_{T+s}|\mathbf{y}_{0:T})$; with $s > T$. This is a prediction ahead-in-time and must necessarily leverage the prior information of the observation up to the time step T.

The other case, referred to as **filtering**, assumes that both the observation $\mathbf{y}_t$ and a prior estimate of the state $\mathbf{x}_t$ are available at time t. The prior estimate $\mathbf{x}_t$ may be produced by the model forecast given the knowledge of the observation $\mathbf{y}_{t-1}$. The one-step-ahead prediction (for example with a prediction step, as mentioned above) from $\mathbf{x}_{t-1}$ gives a version of $\mathbf{x}_t$ (forecast step), which, thanks to filtering, can be further corrected (analysis step). This is equivalent to finding $p(\mathbf{x}_t|\mathbf{y}_{0:t})$. The *sequential* actualization of alternate forecast-analysis steps is common in geophysics [61], since the information from new observations are steadily implemented for a new forecast as they are available.

The third use case of the general bayesian formulation is **smoothing**, that is re-analysing a sequence of states $\mathbf{x}_{0:T}$, given the observations $\mathbf{y}_{0:T}$ have been observed yet. Stated otherwise, smoothing aim to estimate the distribution $p(\mathbf{x}_{0:T}|\mathbf{y}_{0:T})$ that allows to reanalyse the model states obtained up to the time step [3] T. The ERA-5 products mentioned in Section 1.3.2 are an example of this process.

These three cases, as they are stated, remain general. To actualize them in practical applications, some further assumptions are needed, since the probability density function (2.5) is not explicitly tractable. High-dimensionality, non-gaussianity and non-linearity prevent a more explicit characterization. An assumption, that simplifies to problem to the point of having closed-form equations an optimal exact solutions for the filtering and smoothing cases, is that the model dynamics and observation operators are *linear*. The state-space formulation (2.2) begets

$$\begin{cases} \mathbf{x}_t &= \mathbf{M}_t\,\mathbf{x}_{t-1} + \boldsymbol{\eta}_t & \boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \\ \mathbf{y}_t &= \mathbf{H}_t\,\mathbf{x}_t + \boldsymbol{\epsilon}_t & \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \end{cases} \tag{2.6}$$

with $\mathcal{N}$ denoting the normal distribution. From this point, the estimator for system's state $\mathbf{x}$ as required by the filtering and smoothing cases, can be found as least squares estimation [105]. This allows to formulate closed-form equations for the forecast and analysis steps for

3. The time T can, in this case, be anterior to the present time. Re-analysis is typically done on historical data.

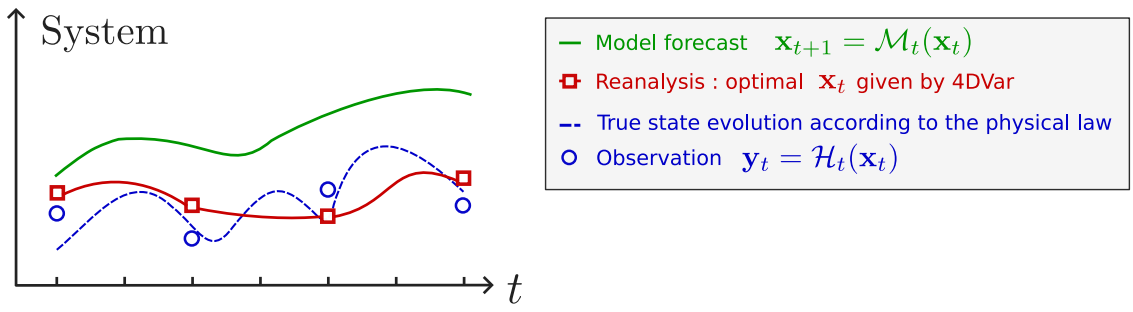


Figure 2.2 – Illustration of the 4DVar scheme.

the filtering problem and closed-form recursive equations for the smoothing case. These two instantiations are called **Kalman filter** and **Kalman smoother** respectively [62], [106].

2.2.2 Variational schemes

The starting point for the variational formulation is the probabilistic statement (2.5) [107]. The Gaussian errors assumption is retained. The objective could be phrased as follows: find the system's states in such a way to maximise the posterior on the states in the trajectory $\mathbf{x}_{0:T}$ given the time series of observations $\mathbf{y}_{0:T}$. This is called maximum-a-posteriori estimation (MAP) [61]. Typically, the probabilities are transformed in the log-space to avoid numerical underflow. By taking the logarithm of Equation (2.5), and calling this objective function $J(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) = -\log p(\mathbf{x}_{0:T}|\mathbf{y}_{0:T})$, the expression becomes

$$J(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) = \frac{1}{2}\sum_{t=0}^{T} \|\mathbf{y}_t - \mathcal{H}_t(\mathbf{x}_t)\|^2_{\mathbf{R}_t^{-1}} + \frac{1}{2}\sum_{t=0}^{T} \|\mathbf{x}_t - \mathcal{M}_t(\mathbf{x}_{t-1})\|^2_{\mathbf{Q}_t^{-1}} + \frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}_0^b\|^2_{\mathbf{B}^{-1}} \quad (2.7)$$

The quantity $\mathbf{x}_0^b$ is referred to as the "background" state. It represents some prior information about the system state at the initial time. The quantities $\mathbf{R}_t$, $\mathbf{Q}_t$ and $\mathbf{B}$ are the covariance matrices associated with the observation, model and background errors respectively. The subscript t refers to the time step where such quantities (but the background term) are evaluated. The maximization of the posterior probability on the state is equivalent to the minimization of the objective function $J(\mathbf{x}, \mathbf{y})$, called **variational cost**. This problem is called **weak-constrained 4DVar**. 4DVar refers to the four dimensional character of the data assimilation scheme, because it considers both the three spatial coordinates and the fourth temporal dimension. The term "weak-constraint" refers

to the fact that in the variational cost the data-model misfit is accounted for at any time step, thus not making the assumption that the model error is zero at any time step. On the opposite, the **strong-constrained 4DVar** only imposes the proximity condition on observations. Its variational cost has the simpler form

$$J(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) = \frac{1}{2}\sum_{t=0}^{T} \|\mathbf{y}_t - \mathcal{H}_t(\mathcal{M}_{0:t}(\mathbf{x}_0))\|^2_{\mathbf{R}_t^{-1}} + \frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}_0^b\|^2_{\mathbf{B}^{-1}} \tag{2.8}$$

Note that in the instances of the 4DVar scheme, the totality of the time steps of the assimilation window, $t = 0, \ldots, T$ is explicitly referenced. This means that the assimilation of observation is intended to be performed at each of such time steps. One other version of this variation approach, the 3DVar, neglects the temporal order and considers the assimilation to be performed at one time step, at the end of the assimilation window. The expression of the **3DVar** [108] variational cost is expressed by the form

$$J(\mathbf{x}_0) = \frac{1}{2}\,\|\mathbf{x}_0 - \mathbf{x}_0^b\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\,\|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^2_{\mathbf{R}^{-1}} \tag{2.9}$$

The 3DVar is obviously computationally cheaper than the 4DVar counterparts. The application of variational data assimilation is mainly directed to the estimation of the best initial condition for the initialization of a model run, assimilating all the observations over a given time window and optimizing for the initial model state. We invite the reader to connect the application of these schemes to the general inversion problem. In this case, the forward process is stated by the general state-space model (2.2). These variational data assimilation schemes aim to *invert* the problem and return the desired sequence of system states.

The optimization of the objective function of variational data assimilation schemes is necessarily performed with iterative methods. The gradient descent algorithm and its variations are commonly used optimization methods. The evaluation of the variational cost gradient, on its right, is not straightforward since it involves a large number of computations, at least as many as the computational grid points. In this sense, the finite-difference gradient evaluation is impracticable since it would require N evaluations, N being the size of the control variable $\mathbf{x}$. This dimension may involve the number of assimilation steps, resulting in a larger dimension than that of the state vector itself. The solution (currently implemented by meteorological services centers, such as the aforementioned ECMWF) is to lean on the **adjoint method**. In its simplest form, the adjoint method aims to put in

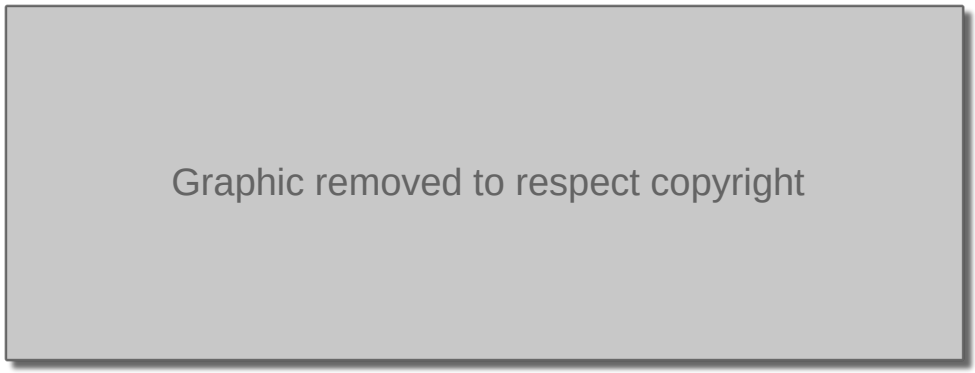


Figure 2.3 – Visualization of the backward-in-time adjoint dynamics. (Top) The state sequence is evaluated in a feed-forward way. The state on each time step is used for the cost function computation. (Bottom) Given the final state computed as above, the adjoint sensitivity method allows to propagate backward the information. Image source: [109].

direct relationship the first-order perturbation of the variational cost with the perturbation of the initial system state $\mathbf{x}_0$. This process involves two steps. The first step computes the sequence of the system states on the chosen assimilation window $\{\mathbf{x}_t;\, 0 \leq t \leq T\}$. The second step uses the information of the last state $\mathbf{x}_T$ to evaluate a sequence of auxiliary (adjoint) vectors in a backward-in-time recursion. This step aims to write the gradient of the variational cost as

$$\boldsymbol{\delta} J(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) \simeq \boldsymbol{\lambda}_0^\top \, \boldsymbol{\delta}\mathbf{x}_0 \tag{2.10}$$

The vector $\boldsymbol{\lambda}_0$ is the adjoint vector evaluated at time step $t = 0$ and is dependent on the adjoint vector of the time step $t = 1$ by the relation

$$\boldsymbol{\lambda}_t = \mathbf{M}_t^\top \, \boldsymbol{\lambda}_{t+1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} (\mathcal{H}_t(\mathbf{x}_t) - \mathbf{y}_t)\,; \qquad 0 \leq t < T \tag{2.11}$$

The symbols $\mathbf{H}_t$ and $\mathbf{M}_t$ denote the Jacobians of the dynamical and observation operators, respectively. This approach allows to compute the variational cost gradient by storing the sequences of T states and adjoint vectors. This method is less expensive in terms of computational resources and makes it possible to implement variational data assimilation schemes for operational purposes. A broader and more detailed treatment of the subject is provided by Talagrand [110] and Bannister [111].

Intriguingly, this forward-backward mechanism for the gradient computation resembles the gradient *back-propagation algorithm* that is the base for deep neural network training. To anticipate the contents of the next section, a neural network can be seen as a cascade of

computations that provide the activation state of the output layer. In the continuous limit, this cascade reduces to a dynamical operator, as exposed by Chen *et al* [109]. In this case, the adjoint method coincides with the back-propagation algorithm. The adjoint method, as applied in Variational Data Assimilation, evaluates analytically the derivative of the variational cost and in a second moment discretizes the dynamical operator to implement the gradient computation. In back-propagation, the network output is evaluated as a discrete cascade of computations and in a second step the gradients are evaluated thanks to automatic differentiation. Figure 2.3 depicts schematically forward-backward mechanism of the adjoint sensitivity method discussed in [109]. The next section details deep learning and the back-propagation mechanism.

2.3 Deep learning

2.3.1 Preliminary definitions

Deep learning is a sub-field of machine learning, which is itself a sub-field of Artificial Intelligence. The goal is to automatically (machine) extract knowledge from a dataset to find the best model (learning) to describe the data. The fundamental components of most machine learning models are: (i) a parametric function f, (ii) a dataset composed of inputs $\mathbf{y}$ and outputs $\mathbf{x}$ for the function f, (iii) an algorithm $\mathcal{A}$ to optimize the function parameters and (iv) a performance metric $\mathcal{L}$ to evaluate the model performance. The performance is the capacity of the function f to relate the inputs and outputs of the dataset. The interested reader is referred to the textbooks by Mitchell [112], Hastie *et al.* [113] and Hart *et al.* [114] for comprehensive and hype-free treatment of the subject.

The simplest example of machine learning is linear regression. The dataset is composed of input-output pairs $\{(\mathbf{y}_i, x_i)\}_{i=0}^{N}$, with $\mathbf{y} \in \mathbb{R}^d$, $d \geq 1$ and $x \in \mathbb{R}$. The class of function considered is that of linear functions

$$\mathcal{F} = \{f_{\boldsymbol{\theta}} : \mathbf{y} \mapsto \boldsymbol{\theta}^T [\mathbf{y}, 1];\ \boldsymbol{\theta} \in \mathbb{R}^{d+1}\} \tag{2.12}$$

The objective is to find the optimal slope and intercept values $\boldsymbol{\theta}$. A good choice for the cost function to minimize is the mean squared error between ground-truth labels and model predictions, i.e. $\mathcal{L}(x, f(\mathbf{y}; \boldsymbol{\theta})) = \sum_i \|x_i - f(\mathbf{y}_i; \boldsymbol{\theta})\|^2$. The algorithm $\mathcal{A}$ is the gradient descent, which solves the problem of minimizing the function $\mathcal{L}$, giving the optimal parameters $\hat{\boldsymbol{\theta}}$. The parameters optimization is performed by the gradient descent updates

expressed as follows

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \alpha \nabla_{\boldsymbol{\theta}^{(k)}} \mathcal{L}(x, f(\mathbf{y}; \boldsymbol{\theta}^{(k)})) \tag{2.13}$$

The gradient of the objective function is evaluated with respect to the model parameters. The quantity α, called *learning rate*, is a small positive real value. Linear *regression* is the linear case of regression problems, that is the task to determine the numerical value of one system's response to a given input.

In recent years machine learning gained attention thanks to advances in **deep learning** [115]. Deep learning modelling achieved impressive results in computer vision [116], [117] and natural language processing [118], [119]. Deep learning is based on deep artificial neural networks [120]. The interested reader may refer to the comprehensive review by Schmidhuber [121] for an historical overview on the subject. The reader is also invited to consult the textbook by Rumelhart *et al.* [122] and Goodfellow *et al.* [123] for a complete treatment of deep learning concepts and methodologies. Feed-forward neural networks are universal function approximators [124], [125]. As such, they can be used to parameterize more complex functions that the simple linear case discussed above. A neural network is a cascade of non-linear functions. With no loss of generality, let the following expression represent the functional form of one such building blocks, called **layer**

$$\mathbf{h} = \sigma\left(W\mathbf{y} + \boldsymbol{b}\right) \tag{2.14}$$

In this expression W and $\boldsymbol{b}$ are the weights and biases parameters respectively. The transformation σ is a non-linear activation function. The vectors $\mathbf{y}$ and $\mathbf{h}$ represent an input and its projection onto the hidden layer (the next after input layer). The non-linearity gives neural network a larger representational capability than a linear function. This makes the data input-output relationship arbitrarily complex. A stack of multiple neural layers allows to increase this complexity level by building a progressive hierarchy of features [126]–[128]. Recent work [129] showed that the representative power of deep neural networks benefits from the application of second-order features maps processing. This means that the deeper features maps operations do not only involve the computation of piece-wise averages but also covariances.

One fundamental aspect of neural networks design is the computation of the loss function gradients with respect to the model parameters (easily millions in standard network models [130]). The output of a neural network is the result of a cascade of tensor computations. For this reason, the gradients of the cost function with respect to the model

parameters can be evaluated using the chain-rule. This principle applied to neural networks is called **back-propagation** [131]. Let the computational cascade be expressed as follows (the output may be a vector)

$$\hat{\mathbf{x}} = f_L(\cdot\,;\boldsymbol{\theta}_L) \circ f_{L-1}(\cdot\,;\boldsymbol{\theta}_{L-1}) \circ \cdots \circ f_1(\cdot\,;\boldsymbol{\theta}_1)(\mathbf{y}) \tag{2.15}$$

where $\hat{\mathbf{x}}$ is the output of the network and f represents the operation stated in Equation (2.14). The parameters of this function are expressed as $\boldsymbol{\theta} = \{W, \boldsymbol{b}\}$. Each parameter is accountable for its contribution to the overall loss $\mathcal{L}(\mathbf{y}, \hat{\mathbf{x}}; \{\boldsymbol{\theta}_l\,;\ 0 \leq l \leq L\})$. The derivative of this loss with respect to the parameter $\boldsymbol{\theta}_l$ is expressed as

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}_l} = \frac{\partial \mathcal{L}}{\partial \hat{\mathbf{x}}} \cdot (f_L)' \cdot (f_{L-1})' \cdot \ \ldots \ \cdot \frac{\partial f_l}{\partial \boldsymbol{\theta}_l} \tag{2.16}$$

The full chain of computations is stored during the **forward pass**, forming a computational graph. The derivatives of the scalar cost function with respect to each parameter are evaluated with the **backward pass**, that implements the chain-rule as expressed in Equation (2.16). The reader is invited to recall the forward and backward evaluations in the adjoint method, evoked in Section 2.2.2, and the consideration exposed therein about the equality of the adjoint method and back-propagation for a network replaced by a continuous dynamical operator [109]. The implications of this method are profound and go beyond deep learning itself. This capability can be exploited to perform *automatic differentiation* [132], [133]. As mentioned at the end of Section 2.2.2, this gives the possibility to evaluate derivatives in a fast an efficient way, which can be used in many computational and numeric applications [134], [135].

In this introduction, the underlying assumption is that of labelled dataset. Each datum $\mathbf{y}_i$ is associated to a numerical or categorical descriptor $\mathbf{x}_i$. In real world applications, however, data collected may be not labelled. We can distinguish **supervised** and **unsupervised** learning algorithms [136]. In the case of supervised learning algorithms (such as regression [137] and classification [138]) the objective is to find input-output correspondences patterns in the dataset and make decisions accordingly. Unsupervised learning algorithms [139] treat data with no labels associated. Examples of these tasks are data reconstruction [140], de-noising [141] and clustering [142]. In this class of methods, patterns and knowledge are extracted from the data domain with no need for supervision.

A notable example of unsupervised learning is **generative deep learning** [143], [144], in which the objective for the model is to learn the underlying probability distribution

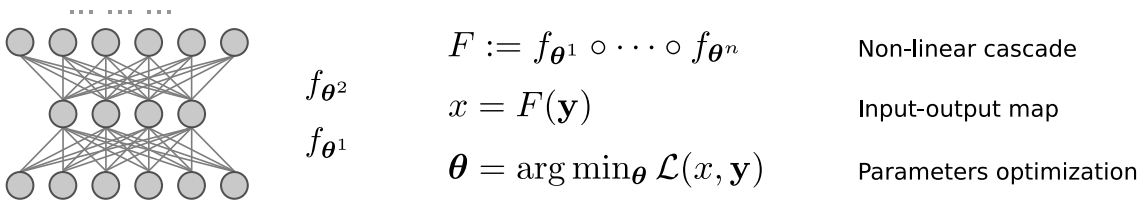


Figure 2.4 – Schematic illustration of the computational flow in a feed-forward neural network.

of the dataset. This gives to the model the capability generating new data samples resembling the input distribution. The early deep generative modelling state-of-the-art was represented by Variational Auto-encoders [145] (VAE) and Generative Adversarial Networks [146] (GAN). More recently, generative learning has been drastically improved by deep diffusion models [147]. These latter gave impressing results in image processing and generation [148].

2.3.2 Deep network layers

Equation (2.14) expressed in a general way one single building block on the computations cascade in a deep neural network. There exist several instances of such layers implementations. In the following, the main neural network layers are listed and discussed.

Fully connected layers

The simplest layer is the **fully-connected** (or **dense**) layer. This operation is a simple matrix-vector product. The weights W are collected in a tabular matrix having dimensions $n \times m$, where m and n are the input and output dimensions respectively. The bias vector has dimension n. The output of a dense layer is a weighted sum of the input compared with the bias (or activation threshold). The basic computation is expressed by the following equation

$$\mathbf{h}_l = \sigma\left(W_l\, \mathbf{h}_{l-1} + \boldsymbol{b}_l\right) \;; \quad 1 \leq l \leq L \quad \text{and} \quad \mathbf{h}^0 = \mathbf{y} \tag{2.17}$$

The signal $\mathbf{y}$ is propagated in the L network layers, $\mathbf{h}_l$ is the activity of the hidden layer l. The non-linearity σ makes the network layers a composition of non-linear functions.

Fully-connected networks suffer of some limitations. Such models are not able to capture translation and rotations in the input data. This represents a non-negligible deficiency in image processing. In addition, fully-connected layers constraint the network to absorb

input elements of fixed size. For real data (images or time series) this may not always apply.

Convolutional layers

Convolutional layers [149] are well suited to respond to fully-connected layers limitations. In convolutional neural networks (CNN), the matrix-vector multiplication is restricted to a local subset of the whole input item. This operation accounts for the spatial or sequential coherence in the input data. The weights matrices are replaced by 1D or 2D **filters** (or **kernels**), respectively for time series or images. A filter is a group of weights that is *locally* convolved with the input. This operation is performed on the complete input elements by sliding the filter along the input dimensions. This operation is crucial for learning local features in data. In addition, this weight sharing technique allows to significantly reduce the number of the network parameters.

In common types of data the information is conveyed on multiple **channels**. In the RGB color model, the number of channels is $C = 3$, for red, green and blue colors. Likewise, temporal signals may have different channels, giving multi-variate time series. The number of channels is a supplementary dimension added to the data shape. Convolutional layers can shrink or expand the data number of channels to target the different information sources. A common choice in 2D convolutional networks is to increase the number of channels with the depth of the network to capture the whole set of features extracted in deep layers [150].

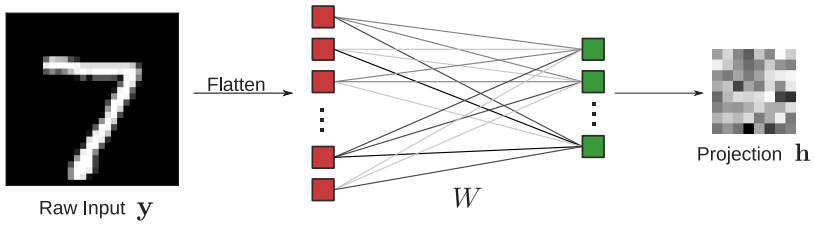
For sequential data, such as multivariate time series, 1D convolutional (Conv1d) layers are often used [151]. The computation of 1D convolutions may be stated as follows. The symbols to denote input and output signals is the same as in the previous equations, except that in this case the dimensions for an input $\mathbf{y}$ are $T \times C_{\text{in}}$ and for the output $\mathbf{h}$ are $T \times C_{\text{out}}$, T is the sequence length [4]

$$\mathbf{h}_c = \sigma\left(\sum_{k=0}^{C_{\text{in}}} W_{ck} * \mathbf{y}_k + \mathbf{b}_c\right) \; ; \quad 0 \leq c \leq C_{\text{out}} \tag{2.18}$$

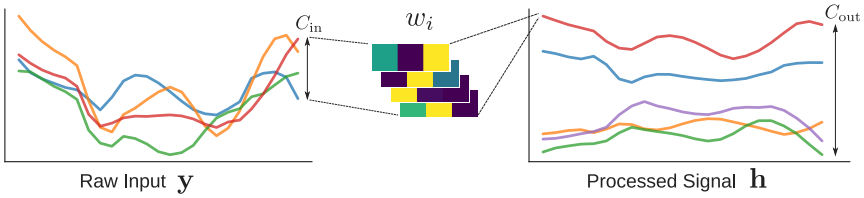
Here W is the convolutional filter of dimensions $C_{\text{in}} \times C_{\text{out}} \times s_k$, with s_k the dimension of the filter. The operation denoted with the symbol $*$ is the cross-correlation [5] between the

4. In real cases, the input and output sequences lengths may differ, depending on the filters sliding behaviors.

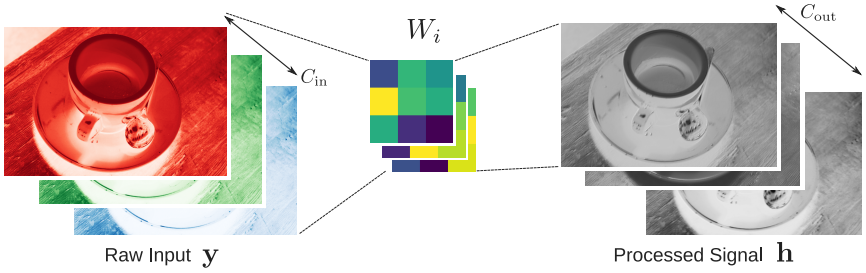
5. Unlike in signal processing, in deep learning the terms cross-correlation and convolution operations



(a) Fully Connected Layers



(b) 1D Convolutional Layers



(c) 2D Convolutional Layers

Figure 2.5 – Graphical description of most notable neural network layers. Panel (a): Fully connected layer. Panel (b): 1D convolutional layer. Panel (c): 2D convolutional layer.

k-th filter W_k and the k-th component (in the channels dimension) of the input tensor, in formulae

$$(f * g)(t) = \sum_{\tau=-s_k}^{s_k} f(t)\, g(t+\tau) \tag{2.19}$$

In the case of spatial inputs (as images), the basic computation has the same form as for Equation (2.18) but the dimensions of input, output and filter are different. Let H and W be the image heigth and width, respectively. The input $\mathbf{y}$ has dimensions $C_{\text{in}} \times H \times W$, the output $\mathbf{h}$ has dimensions[6] $C_{\text{out}} \times H \times W$. The filter has dimensions $C_{\text{in}} \times C_{\text{out}} \times \boldsymbol{s}_k$;

are synonyms, so the terms can be used interchangeably.

6. As before, also in this case the dimensions H, W of the input may differ from those of the input. For the sake of a light notation, this is omitted.

and $\boldsymbol{s}_k = [s_{k,1}, s_{k,2}]$. The cross-correlation operation in this case is

$$(f * g)(i,j) = \sum_{k_1=-s_{k,1}}^{s_{k,1}} \sum_{k_2=-s_{k,2}}^{s_{k,2}} f(i,j)\, g(i+k_1, j+k_2) \tag{2.20}$$

This defines the computation in a 2D convolutional (Conv2d) layer. The layers just discussed form the basis for the most widespread modern deep learning models. Figure 2.5 depicts schematically the functioning principles of fully-connected and convolutional 1- and 2D layers.

Convolution operations in neural networks are characterized by some layer-specific parameters. One is the aforementioned **kernel size**, that controls the size of the filter to convolve the signal. The bigger the filter, the larger the features that such filter can capture. **Padding** controls the filter behavior as it approaches the signal borders. *Unpadded* convolutions have the filter not to exceed the border pixels and this implies that the output has smaller shape than the input. A common strategy is to add a buffer of zeros at the border of the signal in such a way for the filter to be able to finish its horizontal and vertical sliding. In this way, the output can retain its original shape. **Stride** controls the filter displacement value. This can be used to drastically reduce the output size, if necessary. The interested reader may refer to [152] for a technical and graphical explanation of the padding and striding mechanisms and to Wu [153], Stanković [154] and Dumoulin [155] for comprehensive and complete introductions to convolutional neural networks and convolutional arithmetic.

2.3.3 Common neural architectures

This section provides an essential list of the most widely used models for research and industrial applications.

Auto-encoders

In a vast number of applications, the objective is to reconstruct the data in their original aspect rather than making a decision or associating a numerical value to them. Auto-encoders [156], [157] are well suited for this task. An auto-encoder propagates the input signal through a cascade of dimensionality-reducing layers so to obtain a compact representation of the original data (the encoder). This can be seen as a non-linear version of the Principal Components Analysis algorithm [158]. The activations of the latent space

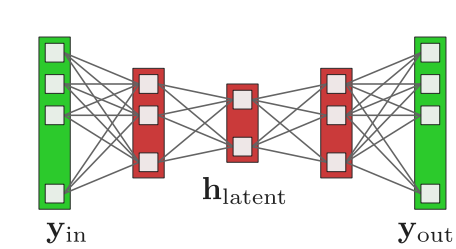
are then expanded with a cascade of neural layers ending with a layer which has the same dimension of the input (the decoder). This is an example of unsupervised learning, as input data do not need a label. Data are labels themselves because the original input element is used to compare the reconstruction with and then to evaluate the cost function. Auto-encoders can implement any of the computational layers discussed above, depending on the input data. One common choice for the loss function is the mean squared error, since the objective is to reconstruct data as best as possible. A schematic illustration of an auto-encoder is reported in Panel (a) of Figure 2.6.

U-Nets

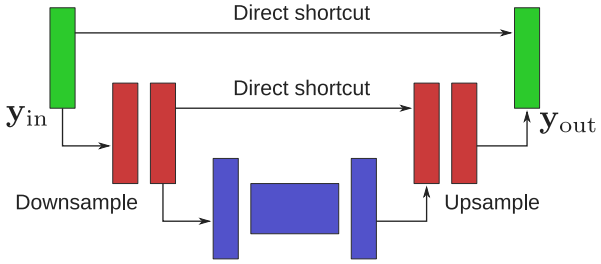
U-Nets were initially conceived for semantic medical image segmentation, i.e. assigning a categorical value to each pixel and not to the global image. The objective is to associate contextual and spatial information. The authors of the original U-Net research [159] designed a network architecture in which each *down-sampling* operation in the encoder is directly connected with the respective up-sampling operation in the decoder. This allows to effectively join context and spatial information so to localize an object in its actual position in the image. This workflow can be visualized in Panel (b) of Figure 2.6. The encoding branch processed the original input down-sampling the signal and producing a lower-dimensional representation of the data. The decoding branch up-scales such features map by improving the spatial resolution of the encoded data and, thanks to direct pathways between down- and up-scaling operations, the classes contextual information is combined with the spatial specification, i.e. *where* each object is placed in the image scene. U-Nets layers are typically 2D convolutional for standard convolutions, and the down-sampling steps are performed by max-pooling operations.

LSTMs

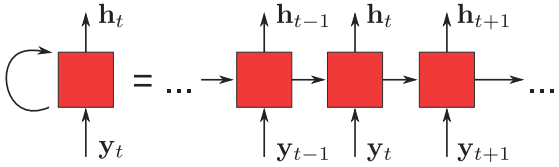
Deep learning applications involve time series processing. Problems as Natural Language Processing require more sophisticated models than simple 1D convolutional layers to successfully handle time series, especially to address long-term recurrence. Recurrent neural networks have been conceived for this problems [162]. A Recurrent Neural Network can be visualized as the recursion of a trainable module on the elements of a sequence, as schematically depicted in Panel (c) Figure 2.6. Each element of the time series is passed through the network which treats and retains useful information. The objective is to use a time series to make prediction of future unseen elements or classify temporal sequences.



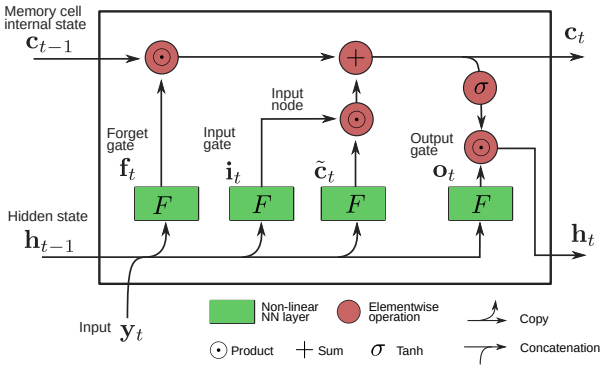
(a) Auto-encoder



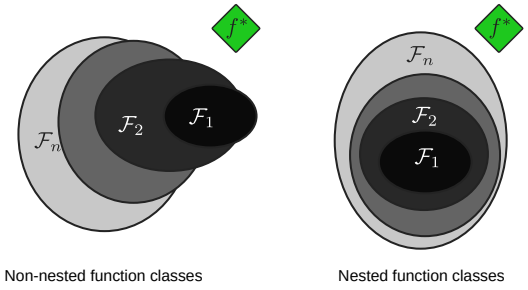
(b) U-Net



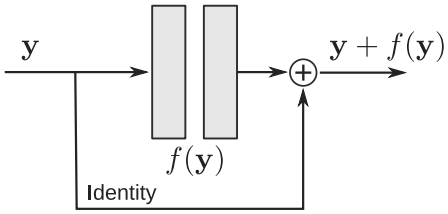
(c) Recurrent Network



(d) LSTM Cell



(e) Functional spaces nesting



(f) Residual Block

Figure 2.6 – Models discussed in the text. Panel (a): Auto-encoder. In this picture the auto-encoder has fully connected layers but the same scheme applies for convolutional layers. Panel(b): U-Net. Panel (c): Recurrent neural network. Panel (d): LSTM cell. Graphic by the author, inspired by [160]. Panel (e): Functional spaces nesting strategies. Graphic by the author, inspired by [161]. Panel (f): Residual block. The layers are typically 2D convolutional.

However, the simplicity of these recurrent models raises the problem of long-term interdependence of the sequence elements (non-recent information). Long-Short Term Memory (LSMT) networks [163], [164] have been designed with the aim of making the network capable of retaining contextual information through long and complex textual sequences. A single LSTM cell is composed of four interacting network layers: input, forget, cell and output gates. These operations are characterized by different sets of parameters. The difference between plain RNNs and LSTMs is that in these latter, the interaction between the mentioned multiplicative gates filters the relevant information to update the LSTM internal state (input gate), what is the information that the internal state can delete (forget gate) and what is the relevant information that the cell can propagate to the next sequence elements (output gate). Panel (d) of Figure 2.6 gives a visual intuition of this mechanics. For completeness, we may mention that newer models, namely Transformer-like architectures [165], replaced recurrent architectures as the state-of-the-art for sequential data processing. The appeal of Transformers stems from the *attention* mechanism that they implement [166]. This mechanism allows the model to process sequential data in a parallel way, while recurrent models processing is essentially sequential.

Residual networks

In Residual Networks [167] the computational cascade expressed in Equation (2.15) implements identity shortcuts between the layers transformations. Assume that the hidden activity $\mathbf{h}_{l-1}$ is fed to the layer f_l. The computation performed is the following

$$\mathbf{h}_l = \mathbf{h}_{l-1} + f_l(\mathbf{h}_{l-1}) \tag{2.21}$$

Thanks to the manually added identity shortcut, f_l is trained to learn the *residual* $\mathbf{h}_l - \mathbf{h}_{l-1}$. The authors of the original paper [167] show experimentally that deeper networks experience a performance loss. They argue that this is due to the difficulty for the network to learn the identity mapping. This phenomenon can be intuitively stated as follows. The copy of $\mathbf{h}_{l-1}$ onto its transformation $f_l(\mathbf{h}_{l-1})$ prevents the layer f_l to completely overwrite the information of $\mathbf{h}_{l-1}$, by explicitly enforcing the identity mapping. Stated otherwise, the residual connection allows the network to build a nested sequence of function spaces where the propagated signal resides [161]. This prevents the network layers to "forget" the previous layer information. Panel (e) of Figure 2.6 gives a visual intuition of this phenomenon. Panel (f) describes the residual connection behavior. Intriguingly, the residual

skip connection expressed in Equation (2.21) has an interesting resemblance with ordinary differential equations numerical integration schemes. Previous work put in relationship residual networks with dynamical systems identification and forecasting [168], [169] continuous flow diffeomorphisms [170] and neural ordinary differential equations [109], [171].

2.3.4 Deep learning for inverse problems

The statement evoked in Equation (2.1) expresses a possibly non-linear relationship between an observable space and a model space which can be partially observed. The previous section introduced the basics of deep learning modelling and its appealing points. Among these points of interest we mentioned the capability of representing complicated input-output maps. Recent work was devoted to the direct application of deep learning modelling to the solution of some inverse problems in medical imaging [172] and computer tomography [173], computational photography [174] and geophysics (cfr. Section 1.4). In geophysical applications one often encounters inverse problems that share the conceptual structure of many computer vision applications. For example, the recent work by Gastineau *et al.* [175] proposed to use generative deep learning models to perform *pansharpening*, that is the augmentation of the spatial resolution of low-resolution multi-spectral images using the spatial resolution of panchromatic images.

Generally speaking, the inversion can be sought in a general way as a maximum-a-posteriori estimator, namely

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \left[-\ln p(\mathbf{y}|\mathbf{x}) - \ln p(\mathbf{x}) \right] \tag{2.22}$$

which, under the assumption of Gaussian distributions, simplifies to

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left[\frac{1}{2} \|\mathbf{y} - \mathcal{H}(\mathbf{x})\|^2 + \mathcal{R}(\mathbf{x}) \right] \tag{2.23}$$

where the term $\mathcal{R}(\mathbf{x})$ is a regularization term to enforce prior domain knowledge. The authors of [76] argue that some strategies exists to leverage deep learning for such problems. On the one hand, one may parameterize the regularization term and perform the rest of the inversion in an analytical way. A more general strategy to solve the problem is to parameterize the global optimization task with a trainable neural network-based model

f. The general expression reads

$$\hat{\mathbf{x}} = f(\mathbf{y}; \boldsymbol{\theta}) \tag{2.24}$$

This method provides a direct learning-based inversion to relate the data $\mathbf{y}$ to the desired output $\mathbf{x}$ in an end-to-end way. In any case, the domain engineering tied to the representation of $\mathcal{R}$ can be delegated to the neural model used. Note that the inversion stated in Equation (2.24) is possible to implement if the dataset is composed of consistent input-output pairs, which can not always be the case [76]. Direct inversion learning must take into account the data qualitative characteristics that may influence the training dynamics.

The authors of [176] argue that the pixel-wise reconstruction loss (often implemented by a Mean Squared Error) could be coupled with a perceptual loss to enforce visual resemblance. A pixel-wise errors enforces the data spatial structure. Perceptual losses, on the other hand, evaluate the distance between high-level data descriptors. This allows to learn computer vision tasks, such as super-resolution and style transfer, that require deep understanding of the data semantic and stylistic information [177]. The pixel-perceptual losses coupling strategy can be pursued by training a second network ψ on a discriminatory task for original data and the neural-inverted ones. This accounts for training the inversion network with the following loss

$$\mathcal{L}(\hat{\mathbf{x}}, \mathbf{u}) = \frac{1}{N}\sum_{i=1}^{N} \|\hat{\mathbf{x}}_i - \mathbf{u}_i\|^2 + \frac{1}{N}\sum_{i=1}^{N} \|\psi(\hat{\mathbf{x}}_i) - \psi(\mathbf{u}_i)\|^2 \tag{2.25}$$

where $\mathbf{u}$ represents the ground truth available and ψ projects its inputs into features maps that encode meaningful representations, for example the image style [178].

This paragraph aimed to give a brief overview on the use of deep learning modelling for inverse problems. While this represents a viable strategy, the inversion benefits from deep learning modelling if a sufficient amount of input-output example pairs are available and if the forward model tends to be complicated. We refer the reader to the work of Ongie *et al.* [76], Lucas *et al.* [176] and Scarlett *et al.* [179] for complete and exhaustive accounts on deep learning in inverse problems.

2.4 The 4DVarNet scheme

This section describes in depth the 4DVarNet model, which has been selected to perform the analyses presented in the subsequent chapters. Section 2.4.1 outlines the model

and its features. Section 2.4.2 provides the key definitions and states formally the model backbone. Section 2.4.3 instantiates the model in the class of problems treated in this work, namely geophysical applications. Section 2.4.4 provides some remarks about the links between the 4DVarNet and the 4DVar scheme and the deep learning modelling practice. Section 2.4.5 discusses the practical implementation details. This last section aims to provide the interested reader with some practical ideas to implement the scheme on real case studies.

2.4.1 Motivation

The 4DVarNet scheme relies on variational data assimilation and deep learning concepts [180], [181]. In particular, it aims to leverage the computational efficiency and automatic differentiation tools of modern deep learning frameworks, e.g. Pytorch [182], to parameterize classic variational data assimilation schemes. The 4DVarNet retains the weak-constraint 4DVar methodological backbone (cfr. Section 2.2.2), but the neural network parameterization allows to jointly learn the underlying dynamical process and the solver for the 4DVar assimilation problem [180] (cfr. Section 2.2.2).

The 4DVarNet application proved promising in a variety of practical case studies involving satellite observations [183], Automatic Identification Systems [184] and local underwater acoustics dataset [185]. It wraps the core 4DVar scheme introduced in Section 2.2.2 in an end-to-end [7] trainable framework. Since the 4DVarNet has been extensively used in the case studies of this thesis, the last section of this introductory chapter is devoted to its formal and detailed introduction. One of the geoscientific problems that inspired the 4DVarNet framework is the sampling patterns of real observations which can deliver missing and partially observed data. Indeed, while a ever growing number of satellite missions collecting Earth remote observations are launched, these observations still do not cover entirely the Earth surface and are unavoidably gappy and incomplete. The 4DVarNet has been conceived to propose a trainable interpolation technique to process such incomplete, albeit fundamental and profitable, data sets coming especially from (but not limited to) satellite observations. In the following, the framework is introduced and detailed in depth. The reader is referred to the 4DVarNet original publications for an extensive account [180], [187], [188].

7. End-to-end refers to the architecture mapping the input to the desired output with no intermediate steps [186].

2.4.2 Formulation

The starting point of the 4DVarNet formulation considers an interpolation problem. Let $\{\mathbf{y}_t;\ 0 \leq t \leq T\}$ be a partially occluded observation process. The objective is to retrieve the sequence $\{\mathbf{x}_t;\ 0 \leq t \leq T\}$ that, through the forward process (cfr. Section 2.1), generates the observations. In the specific case of occluded and gappy data, the observation operator simply masks the state realizations and makes the state variables available on a subset of the spatio-temporal domain considered, called Ω. This observation model can be stated as

$$\mathbf{y} = \mathbb{I}_{\Omega}(\mathbf{x}) \qquad \Omega \subset \mathbb{R}^2 \times \mathbb{R}^+ \tag{2.26}$$

The objective is to invert this operation. The problem is ill-posed as the indicator operator $\mathbb{I}_{\Omega}$ occludes a significant part of the original state. Following the work by Fablet *et al.* [188], this problem can be stated as the minimization of an energy functional $U : X \times Y \to \mathbb{R}$ as follows

$$\begin{cases} \hat{\mathbf{x}}_t = \arg\min_{\mathbf{x}} U(\mathbf{x}_{0:T}, \mathbf{y}_{0:T}) \\ \text{subject to} \quad \mathbf{x}_t|_{\Omega_t} = \mathbf{y}_t|_{\Omega_t} \end{cases} \tag{2.27}$$

In its basic form, this statement only delivers the optimal state $\mathbf{x}$ subjected to one constraint. In addition, the problem constraint suggests a suitable form of the energy functional U. As the requirement is for the state and observations to match on a given spatio-temporal interval of the domain, one could opt for a quadratic expression of the data-state misfit. Including these two remarks in the statement (2.27), it may be restated as follows

$$\hat{\mathbf{x}}_t = \Psi(\mathbf{y}_t\,; \Omega_t, \mathcal{R}) = \arg\min_{\mathbf{x}} \left\{ \|\mathbf{x}_t - \mathbf{y}_t\|^2_{\Omega_t} + \mathcal{R}(\mathbf{x}_t) \right\} \tag{2.28}$$

where Ψ represents the optimal interpolation operator and $\mathcal{R}$ is a regularization term that informs the problem with any a-priori knowledge about the system states. Practical implementations of the regularization term range from physical priors to spatio-temporal statistics [189].

One other alternative is to parameterize $\mathcal{R}$ with a set of parametric variables $\boldsymbol{\theta}$. In this case, the problem closure requires a further optimization step for the parameters, leading to a bi-level parameterization that aims to adjust in parallel both the state $\mathbf{x}$ and the

variables $\boldsymbol{\theta}$. The previous statement is then expanded as follows

$$\begin{cases} \text{State optimization step} & \Psi(\mathbf{y}\,;\Omega_t,\mathcal{R}_{\boldsymbol{\theta}}) = \arg\min_{\mathbf{x}} \left\{ \|\mathbf{x}_t - \mathbf{y}_t\|^2_{\Omega_t} + \mathcal{R}_{\boldsymbol{\theta}}(\mathbf{x}_t) \right\} \\ \text{Parameters optimization step} & \hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{t=0}^{T} \|\mathbf{y}_t - \Psi(\mathbf{y}_t\,\Omega_t,\mathcal{R}_{\boldsymbol{\theta}})\|^2 \end{cases} \tag{2.29}$$

As now, the formulation remains general since neither the regularization term nor the interpolation operator parameterization strategies have been chosen explicitly. A good choice for the parameterization of the regularization term is an auto-encoder, U-Net (cfr. Section 2.3.3) or a Gibbs model [188] (a model based on Gaussian Markov random fields [190]). This choice of model $\phi_{\boldsymbol{\theta}_r}$ allows to restate the regularization term as

$$\mathcal{R}_{\boldsymbol{\theta}_r}(\mathbf{x}) = \lambda \|\mathbf{x} - \phi_{\boldsymbol{\theta}_r}(\mathbf{x})\|^2 = \sum_{t=0}^{T} \|\mathbf{x}_t - \phi_{\boldsymbol{\theta}_r}(\mathbf{x}_t)\|^2 \tag{2.30}$$

The second step concerns the inversion of the problem stated in Equation (2.26). This parameterization involves the interpolation operator Ψ. Two options are available for this operator. A first one would be a direct (fixed-point iteration) inversion that uses a projection $\psi(\mathbf{x})$ to directly retrieve the state variable from the sequence of observations. This strategy has been discussed in Section 2.3.4. As mentioned there, the direct inversion allows to retrieve the target variable $\mathbf{x}$ directly from the observations $\mathbf{y}$ and this allows to overlook the energy functional optimization-based inversion stated in the first row of the statement (2.29). Given the initialization of the state variable $\mathbf{x}^{(0)}|_{\Omega} = \mathbf{y}|_{\Omega}$ and $\mathbf{x}^{(0)}|_{\bar{\Omega}} = \mathbf{0}$, where $\bar{\Omega}$ represents the missing data spatio-temporal segments, the direct inversion is expressed by the recursive projection

$$\begin{cases} \hat{\mathbf{x}}^{(k+1)} &= \psi(\mathbf{x}^{(k)}) & \\ \mathbf{x}^{(k+1)} &= \mathbf{y} & \text{on } \Omega \\ \mathbf{x}^{(k+1)} &= \hat{\mathbf{x}}^{(k+1)} & \text{on } \bar{\Omega} \end{cases} \tag{2.31}$$

Practically, the number of iterations may be set to 1. Note that in this case there is only one optimization to be achieved, that is the one of the parameters of model ψ and the first row of the statement (2.29) reduces to the operation expressed by the relations (2.31) and the explicit parameterization of the regularization term can be superseded.

A second and more interesting option for the neural network-based interpolation operator would be to parameterize it as a trainable gradient solver to perform the oper-

ation stated in the first equation of the expression (2.29). This choice is inspired by meta-learning and optimizer learning schemes [191]. In this way the update recursion for state variable would be informed by the gradient descent direction, improving the convergence. In this case the first row of the statement (2.29) can not be transcended and the solution of the problem is not carried out directly in an end-to-end way. It is then necessary to explicitly parameterize the regularization term of the energy functional. Calling $U_{\boldsymbol{\theta}_r}(\mathbf{x}, \mathbf{y}) = \sum_{t=0}^{T} \left\{ \|\mathbf{x}_t - \mathbf{y}_t\|^2_{\Omega_t} + \mathcal{R}_{\boldsymbol{\theta}_r}(\mathbf{x_t}) \right\}$ the energy functional and defining $\Gamma_{\boldsymbol{\theta}_s}$ the trainable gradient solver, the interpolation outcome can be expressed by the recursion

$$\begin{cases} \boldsymbol{\delta}^{(k+1)} &= \Gamma_{\boldsymbol{\theta}_s}(\nabla_{\mathbf{x}} U_{\boldsymbol{\theta}_r}(\mathbf{x}^{(k)}, \mathbf{y})) \\ \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} - L\, \boldsymbol{\delta}^{(k+1)} \end{cases} \tag{2.32}$$

In the previous expression L denotes a linear operator to reshape the increment $\boldsymbol{\delta}^{(k+1)}$ to the dimension of $\mathbf{x}$, if necessary. The superscript k denote the solver iteration. Once the optimal state $\mathbf{x}$ is found, the model performs the optimization step expressed by the second row of the statement (2.29) for the model parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_r, \boldsymbol{\theta}_s\}$. A common choice to parameterize the trainable neural solver expressed above is an LSTM model (cfr. Section 2.3.3). This is not the only possible choice for the trainable solver. The authors of the reference paper of 4DVarNet [180] argue that the solver may be parameterized by a simpler cascade of convolutional layers. Interestingly, the solver iterations expressed by the relations (2.32) are used as residual blocks (cfr. Section 2.3.3) that define a Residual Network with a fixed number of iterations. A common choice is to perform 5 to 20 solver iterations [180].

The framework presented in this subsection is the methodological backbone of the 4DVarNet but no explicit reference of geophysical dynamics has been made yet. In the following we detail the case of geoscientific applications, hence its name referring to the 4DVar scheme.

2.4.3 The 4DVarNet for geophysical inversions

This section details the application of the 4DVarNet framework in geophysical problems and draws the analogy with the 4DVar framework discussed in Section 2.2.2. The starting point for the practical application of the 4DVarNet framework in the scope of

this thesis is a continuous state-space formulation, expressed as

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathcal{M}(t, \mathbf{x}(t)) + \boldsymbol{\eta}(t) \\ \mathbf{y}(t) &= \mathcal{H}(t, \mathbf{x}(t)) + \boldsymbol{\epsilon}(t) \end{cases} \tag{2.33}$$

The meaning of the symbols is the same as in Section 2.2. To make the physical context sounder, the observation operator $\mathcal{H}$ is assumed to operate from the state space X to a subset of the spatio-temporal space Ω, which defines a precise spatial and temporal sampling regime, delivering possibly incomplete and occluded observations of the state variable. As the data assimilation practice lays on the discrete formulation, a **flow operator** that integrates $\mathcal{M}$ from one time step t to the next one, assumed of amplitude Δt, is introduced and stated as

$$\Phi(\mathbf{x}(t + \Delta t)) = \mathbf{x}(t) + \int_t^{t+\Delta t} \mathcal{M}(t, \mathbf{x}(t))\, dt \tag{2.34}$$

The numerical implementation of flow operator Φ involves a numerical integration scheme such as Euler and Runge-Kutta explicit integration schemes [192], [193] using time discretization. Once this discretization step has been done, the problem can be cast in the weak-constrained 4DVar framework and its associated variational cost is expressed as

$$U_\Phi(\mathbf{x}, \mathbf{y}) = \lambda_d \sum_{t=0}^{T} \|\mathbf{y}_t - \mathcal{H}(\mathbf{x}_t)\|^2 + \lambda_r \sum_{t=0}^{T} \|\mathbf{x}_t - \Phi(\mathbf{x}_t)\|^2 \tag{2.35}$$

We may recall that the variational cost comes from the maximum-a-posteriori estimation of the state likelihood. This leads to a variational cost function composed by the *residuals* of the state-space formulation equations. The parameters $\lambda_{\{d,r\}}$ weight the importance of each term. The form of this equation is the same as in the energy functional on the left hand side of the first row of the statement (2.29). The analogies with the framework defined in Section 2.4.2 are clear. The operator $\mathcal{H}$ implements the forward process of observation and the objective is to retrieve the full sequence of states $\{\mathbf{x}_t\,;\ 0 \leq t \leq T\}$. The variational cost used to perform the inversion, choosing the maximum a-posteriori estimation linked with the bayesian framework that allows to formulate such functional as in Equation (2.35), has one term of data-model misfit and a regularization term that enforces the compliance to the dynamical law expressed by the first equation in the state-space formulation (2.33). The flow operator Φ should be parameterized and a suitable parameterization for the inversion operation should be chosen. This latter choice allows

to treat multi-scale problems [180]. Recent studies have explored how to bridge deep learning schemes and data assimilation methods [194]–[199] as a way to combine physically-sound formulations with the computational efficiency and the versatility of deep learning frameworks. As reported in [180], we have a generic end-to-end deep learning scheme for time-related inverse problems which explicitly relies on a variational data assimilation formulation.

The other fundamental component of the 4DVarNet framework is the neural gradient solver that treats the gradients of the cost function (2.35) to speed up the convergence of the state variable update. In the previous section, the LSTM parameterization has been illustrated. The neural network parameterizations for the physical prior Φ and the gradient solver Γ define an end-to-end architecture Ψ that, given the input observations, returns the state variable $\hat{\mathbf{x}}$ as solution of the inversion problem (2.29). The last step for the complete specification of the 4DVarNet architecture is the learning loss that allows to optimize the neural network models parameters $\boldsymbol{\theta}$.

We recall the distinction made in Section 2.3.1 between supervised and unsupervised learning, respectively whether or not a ground truth value is associated to input data. In experimental settings, a common choice is to use synthetic model data. In this way the observations can be manufactured from the original data in order to impose some sampling pattern. The original data are used as ground truths. In this case the loss function used to optimize the model and solver parameters can be expressed as follows

$$\mathcal{L}(\mathbf{x}, \mathbf{y}) = \sum_{t=0}^{T} \|\mathbf{x}_t^{\mathrm{gt}} - \Psi(\mathbf{y}_t\,;\, \Omega_t, \Phi, \Gamma)\|^2 \tag{2.36}$$

where $\mathbf{x}^{\mathrm{gt}}$ symbolizes the ground truths. In more realistic cases, the input data come from real observation campaigns, for example satellite missions. As mentioned above, satellite observations do not homogeneously cover the Earth surface and this prevents the data to be associated with a ground truth. In this case, the 4DVarNet report [180] recommends to use the variational cost itself as loss function. However, this learning strategy requires to have a calibrated ODE or PDE representation of the dynamical model $\mathcal{M}$. In alternative, a neural network parameterization of this model can be learned in a supervised way by choosing a representative subset of the dataset. In this case, let $n = 0, \ldots, N$ be the

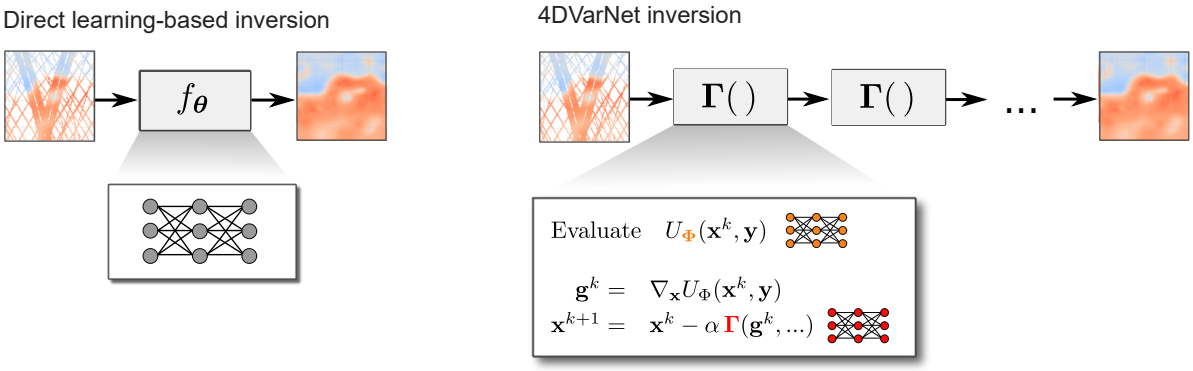


Figure 2.7 – Comparison between the direct and 4DVarNet-based inversions. The observation and reconstructed fields for these visualizations are sea-surface height fields, adapted from [187].

indexes of this subset. The learning objective function can be stated as follows

$$\mathcal{L}(\mathbf{x}, \mathbf{y}) = \sum_{n=0}^{N} U_\Phi(\Psi(\mathbf{y}_n\,;\,\Omega_n, \Phi, \Gamma),\ \mathbf{y}_n;\,\Omega_n) \tag{2.37}$$

Recall the expression of the variational cost (2.35). This second strategy aims to minimize a reconstruction term that measures the mismatch between the data $\mathbf{y}$ and the output of the model $\hat{\mathbf{x}}$. The additional term $\|\mathbf{x} - \Phi(\mathbf{x})\|^2$ acts as a regularization term.

2.4.4 Connections with deep learning and 4DVar

The reader is invited to draw the parallel between the 4DVarNet and the 4DVar schemes. The 4DVar is the state-of-the-art method to treat geophysical inversion problems. In the 4DVar method the essential part of the practical implementation is the adjoint method (cfr. Section 2.2.2). The 4DVarNet framework circumvents this operation thanks to the automatic differentiation capabilities of deep learning frameworks (cfr. Section 2.3.1), which completely automatizes the gradients computation.

Deep learning-based direct inversion schemes, on the other hand, provide a method to perform the inversion in an end-to-end manner. A neural network-based model can be used as inversion black-box. The 4DVarNet scheme is practically implemented by deep learning methodologies but the inversion $\mathbf{y} \mapsto \mathbf{x}$ remains based on the variational cost optimization and on the (parametric) state-space model. Figure 2.7 illustrates graphically the two approaches. This feature constraints the inversion to a sound physical formulation

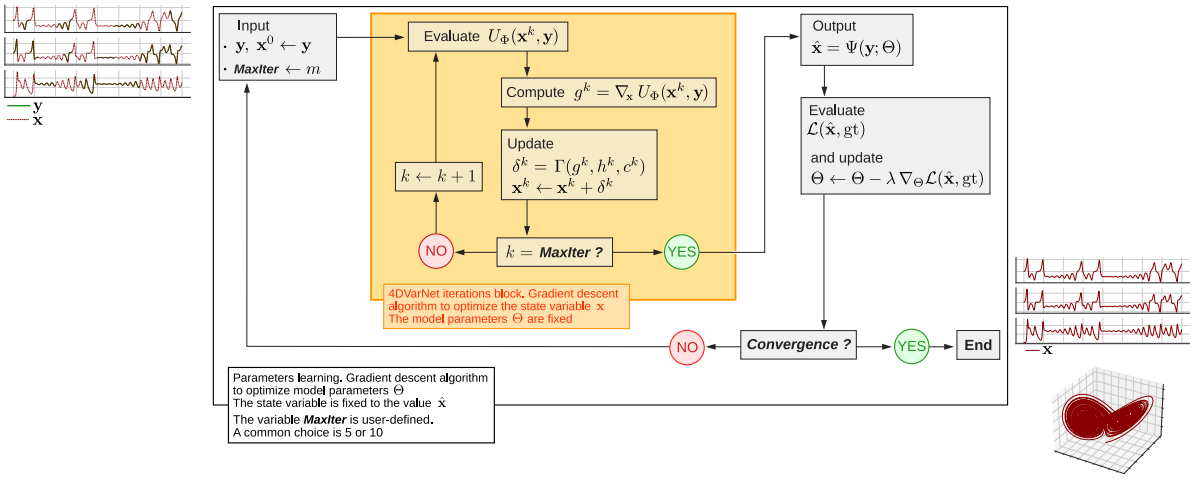


Figure 2.8 – Workflow diagram of the 4DVarNet framework. The yellow block represents the recursive state updates of the 4DVar inversion. The neural network parameterization of the 4DVar allows to learn the underlying model and solver.

of the problem but still letting the model learn automatically the underlying dynamical process. Experimental evidence [180] shows that this jointly learned model and solver outperforms ODE and PDE-based integration schemes.

2.4.5 The 4DVarNet workflow

In this conclusive section, we provide the reader with some practical considerations about the actual implementation of the 4DVarNet scheme. We refer to the stable version of the 4DVarNet implementation. The starter version is available at [200] and the core version (used for the numerical simulations presented in Chapters 3 and 4) is available at [201]. The starter version has been tested on Gulf Stream SSH data [187] and on the Lorenz-63 [104] and Lorenz-96 [202] synthetic models [180].

This description is supported by the graphical explanation provided by Figure 2.8. In this section, we refer explicitly to the elements of the core version of the 4DVarNet implementation. In particular, the interested reader is referred to the source code in `solver.py` of the core version. The source code is publicly available, see the link associated to the reference [201]. The yellow block in the image represent the 4DVarNet iterations. That is, the gradient-based solver iterations that allow to obtain the optimal value of the state variable $\mathbf{x}$. The information entering the iterations block is that available from the observations, cfr. Equations (2.30). Globally the iterative gradient descent procedure and

state variable update (together with the operators mentioned above) is embedded in the routines of the class `Solver_Grad_4DVarNN`. The variational cost evaluation is made by the class `Model_Var_Cost`. The variational cost gradient is evaluated using the automatic differentiation capabilities of the Pytorch library. This gradient is processed by the convolutional LSTM Γ, coded by the class `model_GradUpdateLSTM`. This class allows to choose whether to use Conv2d or Conv1d layers, depending on the nature of the signals to be processed. Once the solver performs the predefined number of iterations, the state variable exits the solver and is used to evaluate the training loss in order to optimize the models parameters. These models are the ones mentioned above plus the parameterization of the operator Φ. Globally, the 4DVarNet mechanism described, is embedded in a neural network training procedure. This means that the 4DVarNet iterations are performed for each training epoch and for each data minibatch. Overall, the operations stated in the yellow block implement the optimization stated in the first equation of the expression (2.29). The models parameters update achieves the optimization stated in the second equation of the same expression.

Part II

Contributions

Chapter 3

Learning-based temporal estimation of in-situ wind speed from underwater passive acoustics

3.1 Context and motivation

In the context of oceanic and marine applications, acoustic data are extensively used. The analysis of underwater acoustic signals is not only limited to geophysical applications, but provides a valuable mean to monitor the marine environment and wildlife. Best *et al.* [203] use underwater passive acoustic data to automatically recognise fin whales pulses with a neural network classifier. Other work [204] approach the problem of deep learning knowledge transfer to cope with the shortage of large and clean bioacoustic datasets. Other recent efforts [205] applied multi-task learning to perform baleen whales calls detection and range estimation using a single underwater hydrophone.

This chapter [1] presents the application of the 4DVarNet framework (cfr. Section 2.4) to the reconstruction of sea surface wind speed using underwater acoustic data. As mentioned in Section 1.2.1, the underwater soundscape relates to the above-surface atmospheric state and can provide useful information about wind speed and rainfall. The complexity of physical processes governing the underwater sound propagation prevents an accurate model-based inversion scheme. Previous work [206] proposed a linear model relating wind speed and acoustic sound pressure. However, this approach is limited by the sensibility of the linear model to the wind speed regime. Indeed, the acoustics-wind speed relationship is piece-wise linear w.r.t. the wind speed intensity. In addition, the

1. The contents of this chapter are adapted from the following publication. Zambra, M., Cazau, D., Farrugia, N., Gensse, A., Pensieri, S., Bozzano, R., & Fablet, R. (2023). Learning-Based Temporal Estimation of In-Situ Wind Speed From Underwater Passive Acoustics. *IEEE Journal of Oceanic Engineering*, vol. 48, no. 4, pp. 1215-1225, Oct. 2023

parameters of the linear model are site-specific. This prevents the model to be general enough and the typical wind reconstruction error expected range from 1.4 to 2 $\mathrm{m\,s^{-1}}$. In this work, the 4DVarNet application represents . Taylor *et al.* [207], in a 2020 work, proposed a data-driven approach for the acoustics-wind speed inversion through regression models. A regression model learns the relationship between one acoustic spectrum and the associated scalar value of the wind speed. This data-driven approach allows us to supersede the features engineering phase and to use the raw sound spectra to predict the wind speed value. The typical wind reconstruction error expected for this trainable multi-variate regression approach is about 1 $\mathrm{m\,s^{-1}}$.

In our work, we apply the 4DVarNet scheme to have a physics-informed end-to-end trainable solution that circumvents the conceptual difficulties related to the acoustics-to-wind relationship. In addition, thanks to the underlying state-space formulation of the 4DVarNet framework, we can explicitly account for the temporal dimension of the phenomenon. Indeed, the 4DVarNet scheme has been used in recent studies to model dynamical processes and time-related patterns [180], [187]. The numerical experiments performed in this analysis aim to benchmark the 4DVarNet performance. In addition, a multi-modal version of the dataset is used to assess how the reconstruction performance benefits of heterogeneous observations. The other modality used is the reanalysed wind speed, available through the ECMWF ERA-interim data base. The multi-modal version of the model is tested against missing data. This chapter is concluded by a categorical classification performance test.

The rest of this chapter is structured as follows. Section 3.2 details the dataset used. Section 3.3.4 describes the methodological aspects, models and training scheme. Section 3.4 presents the results associated to each test configuration and Section 3.5 critically discusses the results.

3.2 Data

Data used in this case study are underwater ambient noise spectra, synthetic reanalyses of wind speed provided by the European Center of Medium-Range Weather Forecast (ECMWF) and in-situ measurements of wind speed at 10 meters above the sea surface. ECMWF reanalyses are publicly available in the ERA-interim database [208] and, underwater passive acoustic (UPA) data and in-situ wind speed are sampled on the W1M3A marine observatory. A brief explanation on the physical functioning principles of the

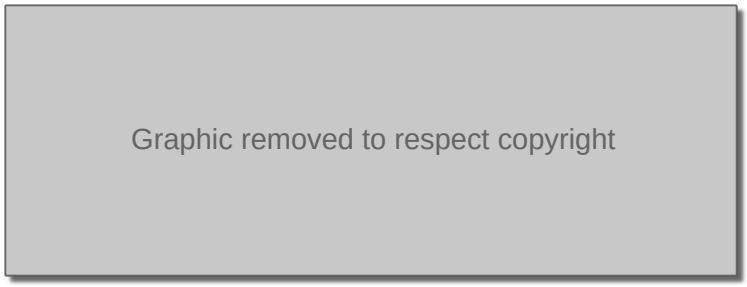


Figure 3.1 – Panel (a): Photography and geographical position of the ODAS Italia-1 buoy. Image source: [210] Panel (b): A schematic illustration of the buoy submerged segment. Image source: [32]

Underwater Passive Acoustic listener is provided in the subsection devoted to W1M3A. Figure 3.2 gives a visual intuition of the data involved. Note that in-situ and ECMWF wind speed values are mildly correlated, $R^2 = 0.71$ and the root mean squared error between them is 1.71 $\mathrm{m\,s^{-1}}$. The in-situ wind speed distributions are displayed in the bottom right panel and show an high density on mild wind values. Higher grades of the Beaufort scale, up to grade 8^{th} are occasionally attained with a wind speed value of 20.71 $\mathrm{m\,s^{-1}}$.

3.2.1 ECMWF wind speed values

ECMWF wind speeds come from model reanalyses, cfr. Section 1.3.2. Since ECMWF wind is an estimation obtained with a numerical model, it is smoothed but implicitly carries information about the physical evolution of the meteorological variables involved. The wind speed reanalyses come from the ERA-Interim dataset, based on the global atmosphere model reanalysis developed at the ECMWF. ERA-interim [209] is the predecessor of the aforementioned ERA-5. All these global reanalyses are obtained with the assimilation of a large body of different in situ and satellite data. The atmospheric model is coupled to an ocean-wave model with a $1.0° \times 1.0°$ latitude and longitude grid. A detailed description of the ERA-Interim product archive can be found in [208], [209].

3.2.2 The W1M3A observation system

The observation site is the Western 1-Mediterranean Moored Multisensor Array (hereafter abbraviated as W1M3A) which is part of EMSO and ICOS networks of European research infrastructures. It is located in the Gulf of Genoa (Italy) at a distance of 80 km from the coast. The W1M3A system is composed of the ODAS Italia 1 spar buoy and a subsurface moored component. Figure 3.1 shows a picture the buoy and its geographical location. Pensieri *et al.* [211] provide a detailed explanation of the W1M3A observation site. Underwater noise data are collected with an Underwater Passive Acoustic Listener (UPAL) [2]. A detailed description of the functioning of UPAL are provided in Zuba *et al.* [212], Nystuen *et al.* [213] and Pensieri *et al.* [211]. For the sake of context, we provide here some insights of UPA data collection and preprocessing. UPAL is equipped with an hydrophone but, due to hardware constraints (battery life and memory) it is not possible to sample noise continuously during the whole instrument duty cycle. Noise is sampled for 4.5 seconds at 100 kHz and then processed to obtain a spectral representation of the signal having 64 frequency bands. The data used in this case study are indeed time series of such spectra. UPAL electronics is endowed with a recognition algorithm that classifies ambient noise sources. The duty cycle of the instrument is slightly adapted based on such rough classification, and on average, a recording is acquired every 5 minute. The interested reader may refer to [213] for a comprehensive explanation on the UPAL functioning. The frequency bands associated with the wind speed signature are those related to 8 and 20 kHz. Nevertheless, we choose to retain all the frequency bands, since one advantage of deep learning modelling stems from the minimal handmade feature engineering required on input data. Thanks to the capability of the deep models to extract high level representation of the data features in a hierarchical way, there is no necessity of manually selecting the features (in our case, the sound pressure level associated to each frequency band). Each 64-dimensional acoustic spectrum will then be used to predict one in-situ wind speed values.

In-situ wind speed is measured by means of the WindSonic-2D anemometer (cfr. the Ultra-sonic anemometer in Section 1.2.1). This instrument measures the two horizontal components of the wind speed vector and thus the scalar speed modulus.

2. Not to be confused with UPA. The acronym UPAL refers to the instrument. UPA refers to the data that this instrument provides.

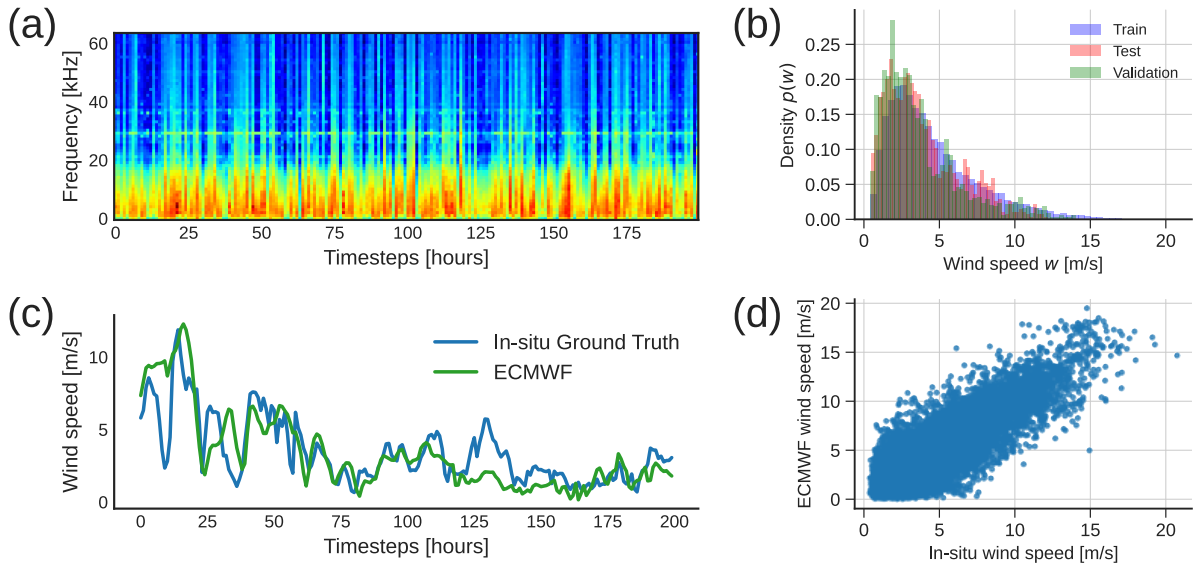


Figure 3.2 – Data set graphical description. Panel (a) A time series of UPA spectra, visualized as spectrogram. Panel (b) The histograms of in-situ wind speed associated to the train-validation-test set partitions. Panel (c) Time series of ECMWF ERA-interim and in-situ wind speed. Panel (d) A scatterplot comparing ECMWF ERA-interim and in-situ wind speed values.

3.2.3 Temporal resolutions

The underwater acoustics data are sampled from 2011-06-17 at 00:50 to 2013-09-06 at 18:50 almost continuously with an hourly resolution, except a period of time between 2013-04-26 and 2013-06-06 in which no observations are available. In-situ wind speed data are available from 2011-06-17 at 00:50 to 2013-09-06 at 18:50, except for a time window from 2012-11-06 to 2013-06-06 where no data are available. In-situ wind speed data are provided as the hourly average of the monitored wind speed values. In an open sea context, such as the W1M3A system, there is almost always non-negligible wind speed, except for some short periods of time [211]. The hourly resolution is motivated by the fact that UPA and in-situ wind speed values are available as hourly averages. In the previous subsection we mentioned the functioning principles of the UPA measurement instrument. Recall that the instrument records underwater noise for a time window of 4.5 seconds, as long as no acoustic sources other than wind speed are present. This time period should not be confused with the temporal resolution of our dataset. For a given hour, the noise samples recorded are averaged and this defines the temporal resolution. Time averaging does imply a smoothing of the information contained in the raw data. However, we assume that such

a temporal resolution does suffice to retain the dynamical information of wind speed evolution. ECMWF wind speed values are available from 2011-06-01 00:00 to 2019-06-30 23:00. Note that this latter data modality does not suffer from the presence of missing data, unlike UPAL and in-situ wind, since it is the output of an operational model. UPAL and in-situ wind speed time series, on the other hand, may have one or more missing time steps of observation, due to many reasons, including among others instrument failure or maintenance operations.

3.2.4 Pre-processing scheme

Our first pre-processing step consists in co-locating UPA, ECMWF and in-situ wind time series, according to their respective time resolutions. If one time step has only UPA and/or ECMWF but has not an in-situ wind speed value associated, this time step is removed from the overall dataset. If one time step has no UPA and/or ECMWF but has the in-situ value, it is kept since it may be a proof of robustness in case of missing data in our time series. In other words, in the 100 % of the time steps of the dataset temporal extent, UPA and ECMWF data are validated by their respective in-situ wind speed value, but the opposite is not true. To complete the data clean-up, we remove all the observation days that do not have full time series of 24 in-situ wind speed values, because we want the dataset to be divisible in time series of length 24 (one day from 00:00 to 23:00). This results in keeping about 98% of the original dataset. By thus doing, we obtain a collection in temporal order of 14088 triples constituted by a 64-dimensional UPA vector (each UPA spectrum has 64 frequency bands), a 1-dimensional ECMWF wind speed value and a 1-dimensional in-situ wind speed value.

For our analyses we need two versions of the same dataset, one time-independent version and one time-dependent version. The time-independent version is simply the collection fo 14088 UPA, ECMWF and in-situ wind triples. For the time-dependent version of the dataset two choices are possible. The first would group the time-independent version in series of 24 hours according to the hours. Each series would start at 00:00 and end at 23:00. By thus doing, the overall number of time series available would be $14008/24 = 587$. A second choice could be to randomly extract 2000 time series of length 24 from the time-independent version. In this case, the start/end hour of the series is irrelevant.

We choose the second alternative since 14088 samples are sufficient to fit, validate and test a regression model, but 587 samples (considered to be time series) may not suffice to fit, validate and test the 4DVarNet model, since it comprehends two deep network

parameterized models (the dynamical prior and the gradient solver). Note that this version of the dataset comprehends 2000×24 single UPA, ECMWF, in-situ wind triples, unlike the 14088 instances of the time-independent version. But in the time-dependent case, the *elemental* unit of the dataset is the 24-hours time series and not the single hour. In other words, a time-independent model takes one data triple, referred to one given hour, and returns the in-situ wind for that hour. The time-dependent model takes as input the time series obtained as described and returns a time series of 24 in-situ wind speed values.

3.3 Proposed method

This section details the proposed end-to-end deep learning scheme based on a variational data assimilation formulation for the retrieval of in-situ wind speed from multi-source data, namely underwater acoustics and ECMWF data.

3.3.1 Problem statement

In the introduction the main difference between the state-of-the-art data-driven techniques for the underwater acoustics-to-wind speed inversion has been evoked. Formally, regression models learn the relationship

$$\mathbf{y}_t \mapsto \mathbf{x}_t \qquad \text{for all} \quad t = 0, \ldots, T \tag{3.1}$$

Following the symbols meanings adopted in the previous chapters, $\mathbf{y}$ represents the acoustic observations and $\mathbf{x}$ the target wind speed. The symbol T represents the dataset length. This relationship holds for each time step but the time evolution of the phenomenon is not explicitly accounted for. By contrast, the resolution of inversion problems in geoscience for time-related processes generally relies on a data assimilation formulation [214], [215]. It explicitly accounts for the underlying temporal dynamics of the process of interest through a state-space model, cfr. 2.2.1. In this application case, the state variable $\mathbf{x}$ is assumed to host information related to UPA, ECMWF and in-situ wind speed. For this augmented state dynamical problem, the observation operator $\mathcal{H}$ reduces to a binary mask. This masking operation hides the in-situ wind speed information and may emulate an irregularly sampled dataset.

The problem can be stated as follows. Given the augmented state dynamics described above, the objective is to retrieve time series of in-situ wind speed from time series of

input UPA. In the multi-modal case, the input data involve the ECMWF wind speed values as well. Formally the problem can be written as follows

$$\{\mathbf{y}_t\,;\; 0 < t \leq T\} \mapsto \{\mathbf{x}_t\,;\; 0 < t \leq T\} \tag{3.2}$$

The difference between this expression and the relationship stated in Equation (3.1) is clear. Here the reconstruction does not only target the acoustics-to-wind inversion but also the explicit time evolutionary feature. In this expression, T symbolizes the length of one time series of observations. This convention will be used throughout the rest of the chapter. The reader is invited to recall the standard data assimilation solution scheme, that involves the optimization of a variational cost (2.35). For convenience, this functional is restated for this approach as

$$U_\Phi(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_d \left\|\mathbf{x} - \mathbf{y}\right\|_\Omega^2 + \lambda_r \left\|\mathbf{x} - \Phi(\mathbf{x})\right\|^2 \tag{3.3}$$

The symbols $\lambda_{\{d,r\}}$ represent adjustable weights. The standard data assimilation framework is bridged with a learning-based scheme thanks to the trainable neural operator Φ, that parameterizes the dynamical operator $\mathcal{M}$, cfr. Section 2.4.3.

3.3.2 Proposed variational data assimilation model

In section 3.3.1 we introduced the formal statement of the problem. Recall that in a variational data assimilation problem, one has access to some observations $\mathbf{y}$ and wants to reconstruct their state variable $\mathbf{x}$. Section 2.4.2 introduced the 4DVarNet scheme. In the classical applications, the state variable is partially observed, so the observations can be represented as a masked version of the state variable and the observation operator is simply a binary mask. In the introduction of this chapter, we mentioned the conceptual difficulty to relate explicitly sea-surface wind speed and underwater acoustics through empirical models. Our choice is to set up the 4DVarNet scheme as an *augmented-state dynamical system* [216]. The state variable is assumed to be the concatenation of the different available data modalities: underwater acoustics, in-situ wind speed and, in the multi-modal case, ECMWF wind speed. In this way, the observation operator can be stated again as a binary mask. This operator masks the in-situ wind speed information and returns only the underwater (and ECMWF in the multi-modal case) information. Figure 3.3 represents graphically these configurations.

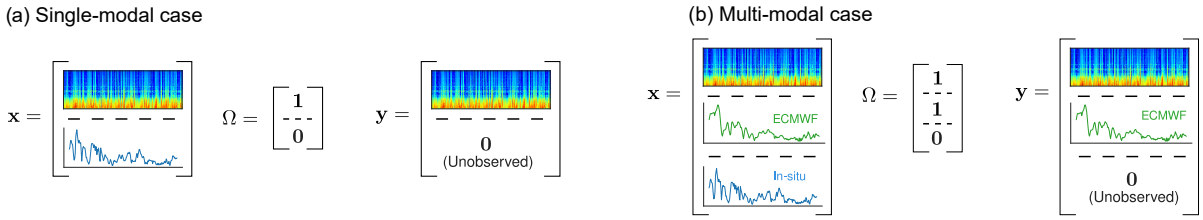


Figure 3.3 – Visualization of the augmented states used in the simulations.

We consider in the following the outputs of single-modal and multi-modal versions of the proposed 4DVarNet framework. Assume that α indicates the UPA or UPA with ECMWF modality and β the in-situ wind speed modality. In the single-modal version, the state variable is the concatenation of UPA and in-situ wind speed, $\mathbf{x} \in \mathbb{R}^{N_\alpha + N_\beta}$, and the observable part is represented by UPA data, $\mathbf{y} \in \mathbb{R}^{N_\alpha}$ with $N_\alpha = 64$ and $N_\beta = 1$. In the multi-modal version, the observable part is composed by the UPA and ECMWF data, and the state variable is still considered to be the concatenation of UPA, ECMWF and in-situ wind speed, i.e. with $N_\alpha = 65$ and $N_\beta = 1$. Since we prepare these data as time series, the observation effectively used in the model will be a batch of vectors of shape $(24, N_\alpha)$. Given each time series of 24 UPA and ECMWF instances, the model will return a time series of 24 wind speed values.

Given these definitions of the state variable and of the observations, we can consider the following observation operator $\mathcal{H}$

$$\mathbf{y} = \left[\mathbf{y}^\alpha, 0 \times \mathbf{x}^\beta\right] = \mathcal{H}\left(\left[\mathbf{x}^\alpha, \mathbf{x}^\beta\right]\right) \tag{3.4}$$

The square brackets symbolize a simple concatenation operation. The quantity $\mathbf{x}^\alpha$ represents the part of the latent variable hosting the acoustics and ECMWF information and $\mathbf{x}^\beta$ represents the part of the state variable hosting in-situ wind information. The observation operator has as argument the concatenation of the information related to observable quantities (marked with α) and in-situ wind speed (marked with β), and returns the only measurable information $\mathbf{y}^\alpha$, i.e. related to UPA and ECMWF. This observation operator simply states that no direct observation of the in-situ wind speed is available. The dependence between the in situ wind speed and the observed variables derives from the parameterization of prior operator Φ. Rather than exploring an explicit ODE-based parameterization as in model-driven data assimilation schemes [63], we rely on neural auto-encoder architectures as in [180], [187]. In such architectures, the inputs and out-

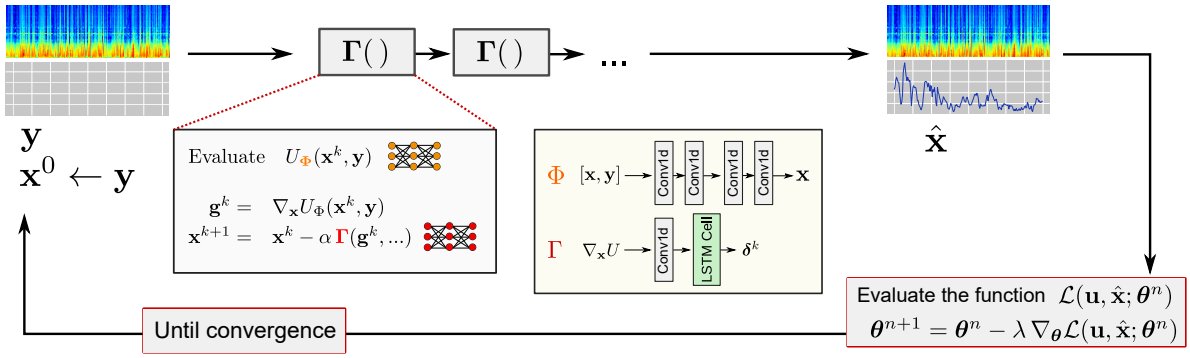


Figure 3.4 – Diagram of the 4DVarNet scheme applied to our case study. Recall that the block denoted by Γ contains one iteration of the variational cost optimization explicitly stated in Figure 2.8.

puts share the same shape. Auto-encoder architectures [217] have been widely used for denoising, reconstruction and simulation tasks [156]. They generally exploit a latent lower-dimensional representation of the input data [156]. This property seems appealing here to enforce the assumption of some underlying latent space jointly encoding sea surface wind speed and available observation data.

Here we chose to use a 1D convolutional auto-encoder architecture (Conv-AE) [218]. This is motivated by the fact that convolutional networks can leverage trainable convolution operators to model translation-invariant features throughout data examples. Our encoder is composed of two 1D convolutional layers, having input-output shapes of $(N_\alpha + N_\beta, 128)$ and $(128, 20)$. After the first layer a Leaky Rectified Linear Unit non-linear activation function is placed, its negative slope is set to 10^{-1}. The decoder has the same structure, but in reverse order and the same non-linear activation functions are used after the two layers. All convolution layers involve a zero-padding and a kernel dimension of 3. The number of channels, 128 and 20 respectively, was set empirically from cross-validation experiments. Figure 3.4 adapts the general scheme presented in Figure 2.7. In the multi-modal case, the concatenation involves also the ECMWF wind speeds time series as depicted in Figure 3.3.

3.3.3 Associated trainable solver

Within the proposed approach, the reconstruction of the state variable relies on the minimization of variational cost. We solve this optimization problem using another neural network model included in the end-to-end architecture. This trainable gradient solver,

referred to as Γ, exploits a convolutional Long-Short Term Memory (LSTM) network [163], the latter being particularly suited for time series modelling [219] and optimizer learning [191]. Overall, the end-to-end architecture implements a predefined number of iterations of the iterative rules states in the relations (2.32). In our implementation, the LSTM cell dimension was set to 100, again based on numerical experiments. A typical choice for the total number of iterations ranges from 5 to 10. This choice derives both the computational complexity of the underlying automatic differentiation [220] as well as the ability of this gradient-based iterative update to converge with only very few gradient-based steps.

3.3.4 Learning scheme

Overall, the training procedure involves two nested gradient descents: an inner minimization of variational cost (3.3) with respect to the state variable $\mathbf{x}$ and an outer minimization of the training loss with respect to model parameters, especially the parameters Θ of models Φ and Γ. The objective function for the former has been already discussed. Let Ψ_Θ be the 4DVarNet end-to-end system and $\hat{\mathbf{x}} = \Psi_\Theta(\mathbf{y})$ the reconstructed state variable and $\mathbf{u}$ the ground truth values. We choose to train the models parameters by the minimization of a MSE training loss. This loss function involves two reconstruction terms. The first one relates to the reconstruction of the visible information associated to the acoustics and, in the multi-modal case, the ECMWF wind speed. These elements are denoted by the superscript α. The second term relates to the reconstruction of in-situ wind speed, denoted by β. The expression of our loss function is the following

$$\mathcal{L}(\hat{\mathbf{x}}, \mathbf{u}) = \frac{1}{M} \sum_{i=0}^{M} \sum_{t=0}^{T} \left\{ \lambda_d \left\| \mathbf{u}_{it}^{\alpha} - \hat{\mathbf{x}}_{it}^{\alpha} \right\|_{\Omega^\alpha}^2 + \lambda_p \left\| \mathbf{u}_{it}^{\beta} - \hat{\mathbf{x}}_{it}^{\beta} \right\|_{\Omega^\beta}^2 \right\} \tag{3.5}$$

In this equation, Ω_α and Ω_β are domain masks that account for missing data. These masks may be simple binary matrices that represent the data sparsity pattern. The parameters $\lambda_{\{d,p\}}$ weight the importance of the acoustic data and wind speed reconstructions. The weight associated to the in-situ wind speed reconstruction is dominant.

3.3.5 Numerical implementation

Parameters optimization is achieved with two different instances of the Adam algorithm [221]. For both the Conv-AE and the LSTM solver the learning rate is set to 10^{-3}

and the weight decay to 10^{-5}. The weights λ_d and λ_p are set to 0.5 and 1.5 for the Conv-AE and 4DVarNet experiments. These weights were tuned empirically. The weight on wind prediction is greater because it is the term associated with the task of major interest. The full end-to-end architecture Ψ_Θ is trained for 200 epochs, with no early stopping criteria. Our learning protocol consists of two consecutive steps. First we perform a full training procedure with 5 gradient iterations for the 4DVarNet scheme and save the best model based on validation loss. Note the best model is not necessarily the one at the last epoch of the training procedure. Then, this best model is further trained through another full training procedure, this time using 10 gradient iterations. The simulations are performed on a machine equipped with 3 Nvidia Quadro RTX 8000 units. Each of these units has a TU102 graphical processor operating at a frequency of 1395 MHz and has 48 GB memory size.

3.4 Results

This section reports the numerical experiments carried out to assess performance of the proposed approach with respect to state-of-the-art data-driven methods [3]. We first detail our evaluation setting and the benchmarked models. We then report and discuss the reconstruction performance for three case studies for the reconstruction of in situ wind speed (i) using only underwater acoustics data, (ii) using underwater acoustics and ECMWF data, (iii) when dealing with random gaps in the underwater acoustics data, again in a multi-modal UPA and ECMWF dataset case.

3.4.1 Evaluation framework

Our evaluation procedure follows the one reported in [207]. We chose as evaluation metrics the root mean squared error (RMSE) between reconstructed and in situ wind speed values. We evaluate this RMSE on data period from 2011-06-18 00:00 to 2011-08-07 23:00 (50 days) whereas data period 2011-08-08 00:00 to 2013-09-05 23:00 is used as an independent training dataset, except for the period between 2011-08-25 00:00 to 2011-10-14 23:00 (50 days) which is used as validation set. From the 50-day time series with an hourly time resolution, we extract all associated one-day time windows, which result in a dataset of $(50 - 1) \times 24 = 1176$ 24-hour samples.

3. We refer the reader to the following in the repository for our associated implementation `https://github.com/CIA-Oceanix/4DVarNet-wsp`

For benchmarking purposes, our numerical experiments involve machine learning models proposed in [207] as well as state-of-the-art neural network architectures. The latter have been chosen in accordance with the parametrization of 4DVarNet scheme. Overall, the first category of methods involves machine learning schemes, referred to as time-independent, which predict in-situ wind speed at one time step from the underwater acoustics spectrum at the same time step, as stated in (3.1). The multi-modal approach is not implemented for this class of models. This category includes:

- **CatBoost** [222] A gradient-boosting algorithm [223], [224] which can manage effectively categorical features. CatBoost uses as loss function the root mean square error as loss function.
- **Random Forest** [225] An ensemble of decision trees, either trained for classification or regression. Random Forest has set up with the maximum depth of the tree to 10, the number of features to consider when splitting equal to the actual number of features, the minimum number of samples required for leaf nodes to 1 and the minimum number of samples required to split an internal node to 2. The number of estimators is set to 100.
- **FC-AE**. This Fully-Connected Auto-Encoder (FC-AE) architecture comprises a fully-connected encoder composed of 2 linear layers with input and output shapes of $(N_\alpha + N_\beta, 128)$ and $(128, 20)$, respectively. After the first layer a Leaky Rectifier Linear Unit with negative slope of 0.1 is applied. The decoder has the same architecture but reversed, with the same non-linearity applied after all layers. The learning rate is set to 10^{-3} and the weight decay is set to 10^{-6}. The weights of the loss terms are chosen to be 0.5 and 1.5. The intermediate and latent dimensions, learning rate and weight decay and finally the loss terms weights were set empirically using cross-validation.

The second category of machine learning methods, referred to as time-dependent methods, predicts a time series of in-situ wind speed from a time series of underwater acoustics data and may benefit from time-related features. The link between the two temporal configurations is the FC-AE model. FC-AE is used in both time-independent and time-dependent configurations and shows, compared to CatBoost, which is the quota of improvement imputable to time-dependence alone. So the absolute performance level of the time-dependent models listed below is to be understood as the aggregated performance of temporal dependence inclusion and model choice. The category of time-dependent models comprises the following schemes:

- **Fully-connected auto-encoder**. The FC-AE parametrization as described above has been reused in the time-dependent configuration. There are no differences in the architecture as the input size does not change, except the temporal dimension which is neglected in the time-independent setting.
- **Conv-AE-UPA**. This convolutional auto-encoder architecture refers to the operator Φ in (3.3) for the proposed 4DVarNet scheme. Here, we use this architecture to train a direct inversion scheme to map input data $\mathbf{y}$ to a reconstructed state $\mathbf{x}$. We may point out that this direct inversion scheme can be regarded as a single iteration of a fixed-point iterative solver for the minimization of variational cost (3.3).
- **Conv-AE-UPA+ECMWF**. this architecture refers to an extension of the previous one when the input data $\mathbf{y}$ includes both UPA and ECMWF wind speed. Besides, the reconstruction x also comprises UPA, ECMWF and in-situ wind speed states.
- **4DVarNet-UPA**. Using the Conv-AE as previously described and the trainable solver as detailed in section 3.3.3, the first 4DVarNet configuration accounts for the observations $\mathbf{y}$ of UPA data only. The output $\mathbf{x}$ is a concatenation of UPA reconstructions and in-situ wind speed predictions.
- **4DVarNet-UPA+ECMWF**. In this second case, the architecture of the 4DVarNet architecture is the same, except for the observations and the state variable. The input data comprehend both UPA and ECMWF wind speed.

The loss function used to train the deep models is a simple mean squared error, formulated in (3.5). For evaluation on the test set, the root mean squared error is instead used. Since the task studied is the reconstruction of wind speed time series, we consider performance metrics based on the in-situ wind speed data considered as ground-truth. In the following tables two columns are present. A first column, named "Mean $\pm$ std" reports the average RMSE and the quartiles over the 10 runs. In order to compare our framework with ensemble models previously discussed, another evaluation strategy is used. Each of the 10 models trained can produce a reconstruction of wind speed sequence given test data. Similarly to what is done in *bagging* [226], we chose to compute the median of the wind speed values reconstruction as aggregated output. We express this aggregated output formally as

$$\hat{\mathbf{x}}_{\text{median}} = \text{Median}(\ \{\hat{\mathbf{x}}_n^{\beta}\ ;\ 0 \leq n < N_{\text{runs}}\}\ ) \tag{3.6}$$

The n-Median metric is defined as the RMSE between this quantity and the ground

Model		Metrics in $\mathrm{m\,s^{-1}}$			Relative
		RMSE	Mean ± std	n-Median	Gain [%]
Time-independent cases					
ECMWF 4DVarDA		1.71	–	–	–
CatBoost		0.95	–	–	–
Random Forest		0.97	–	–	–
FC-AE		–	0.98 ± 0.03	0.95	–
Time-dependent cases					
Single-modal	FC-AE	–	0.97 ± 0.04	0.92	3.2
	Conv-AE	–	0.94 ± 0.04	0.88	7.4
	4DVarNet	–	0.89 ± 0.04	0.84	11.6
Multi-modal	Conv-AE	–	0.88 ± 0.02	0.83	12.6
	4DVarNet	–	**0.84 ± 0.02**	**0.80**	**15.8**

Table 3.1 – Results for the time-independent and time-dependent analyses. The dataset for the two cases is described in Section 3.2.4. The models are described in Section 3.4.1. For completeness, the time-independent case has been completed with a row reporting the average error between ECMWF ERA-interim and in-situ wind speed values.

truths. In order to quantify the improvement of the proposed class of models with respect to the baselines, we may define a relative gain metric. Call p_B and p_M the baseline and improved performance metrics, chosen to be the ensemble n-Median scores. Then define the relative gain as

$$\eta = \left(1 - \frac{p_M}{p_B}\right) \times 100 \tag{3.7}$$

In the following tables, the relative gain scores are reported for each of the associated models. In the results related to the time-independent models, such a comparison is not extremely informative. Rather in this first comparison we assess what is the most indicative baseline model to perform the subsequent comparisons.

3.4.2 UPA-only time-independent models

The performance of time-independent models are reported in the first part of Table 3.1. FC-AE performs as well as classical regression models. In a time-independent scenario, where the interest is only the prediction of a wind speed label given a single acoustic

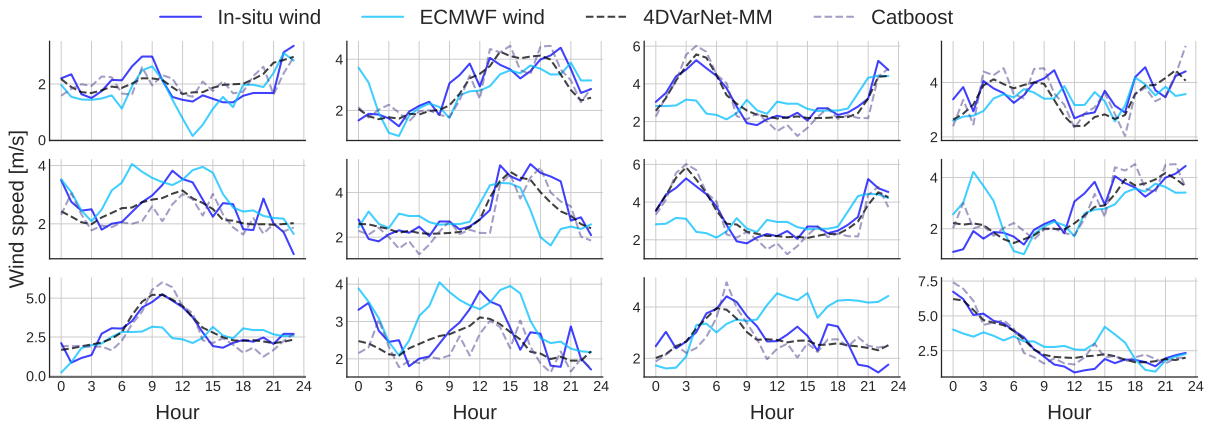


Figure 3.5 – Wind speed reconstructions: examples of wind speed reconstruction over different 24-hour windows in the test dataset. We depict the in situ wind speed and compare the reconstructions issued from the proposed approach to Catboost reconstruction [207] and ECMWF wind speed. Each panel represents a 24 hours time window of the test set. There is no temporal contiguity between each subplot.

spectrum, a framework like the one described in (3.1) could then be preferable. Since in the time-independent configuration the fully connected auto-encoder and the CatBoost have a similar performance, see Table 3.1, the performance metric of these models is now taken as baseline in order to evaluate the improvement hereafter. In this table, no data are provided for the mean and quartiles for CatBoost and Random Forest, since these are ensemble methods and hence they give yet an aggregated output.

3.4.3 UPA-only time-dependent case

The second part of Table 3.1 displays the results given by the model detailed in section 3.3.2. The case of single-modal dataset presents a relative improvement of the performance with respect to the time-independent configurations. FC-AE applied in a time-dependent scenario has a similar performance as in the time-independent case. Indeed the Conv-AE model yet suffices to improve the time-independent FC-AE baseline by 7.4 %. The 4DVarNet-UPA leads to a relative gain of 8.4 % with respect to the FC-AE and regression models baselines.

3.4.4 Multi-modal time-dependent case

Results in Table 3.1 confirm the potential of the multi-modal approach. The Conv-AE trained on a heterogeneous dataset gives a gain of 12.3 % with respect to the FC-AE benchmark. The 4DVarNet-UPA+ECMWF model, the multi-modal counterpart of the single modal 4DVarNet-UPA, yields a performance gain of 14.7 % and 15.8 % with respect to the FC-AE benchmark and the state-of-the-art presented by previous work. Recall that such a result by the 4DVarNet was obtained through the training protocol explained in section 3.3.3, that is 5 gradient iterations on the first training step and 10 iterations on the second step after selection by best score on the validation set. Figure 3.5 presents a visual comparison between ground truth wind speed time series and the reconstructions obtained with selected models. ECMWF wind values are also super-imposed. These time series are compared with the reconstruction performed by a time-independent and a time-dependent model, the CatBoost and the multi-modal 4DVarNet, respectively.

Figure 3.6 shows a scatterplot between the reconstructed and ground truth values. The top left panel clearly highlights a bias in the reconstruction of high wind speed values. This could be due to saturation of underwater acoustic data for large wind speeds. The bottom panel presents the average hourly error. This plot shows the effect of 1D convolutional filters, since the boundaries of the time observation intervals are not entirely involved by the striding of the filters hence there is no coverage backward and forward. One other interesting feature of this plot is the worse predictive performance on the central part of the day. This might be due to the diurnal cycle of winds trend, with the mildest wind speed during the day and strongest winds during the nights. Additionally, winds modeled by ECMWF are also known to suffer from a bias usually ascribable to a misrepresentation of mesoscale convective variability and wind shear [227].

3.4.5 Multi-modal time-dependent case with missing data

One of the points of interest of 4DVarNet is that it can handle time series with missing and/or corrupted data. Since underwater acoustic data in our dataset are almost complete, experiments on missing data were designed by arbitrarily artificially masking data batches during training and testing. The missing data percentage is a parameter chosen to range between 10 % and 90 %. Figure 3.7 describes graphically the experiments outcomes. Note that when 10 % of the data is missing, 4DVarNet performs as good as the best multi-modal model trained with complete data. One may argue that removing 10 % of data is

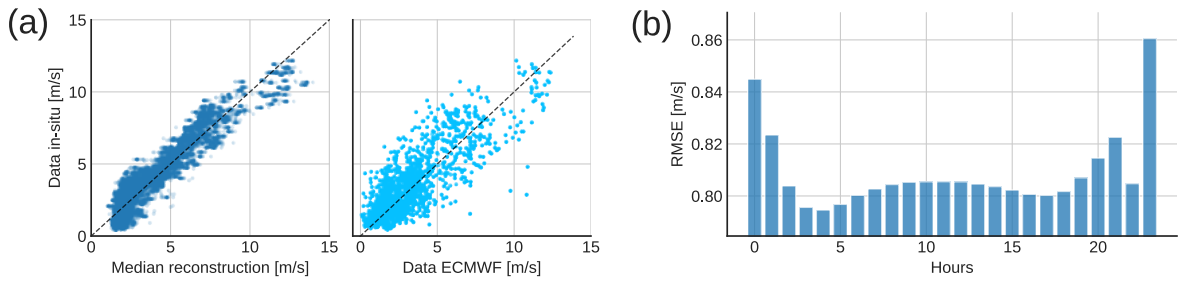


Figure 3.6 – Overall reconstructions and error patterns. Top panel: Scatterplots of in-situ ground-truths against the reconstructions of wind speed obtained with 4DVarNet and ECMWF wind speeds. Bottom panel: Hourly averages of reconstruction error.

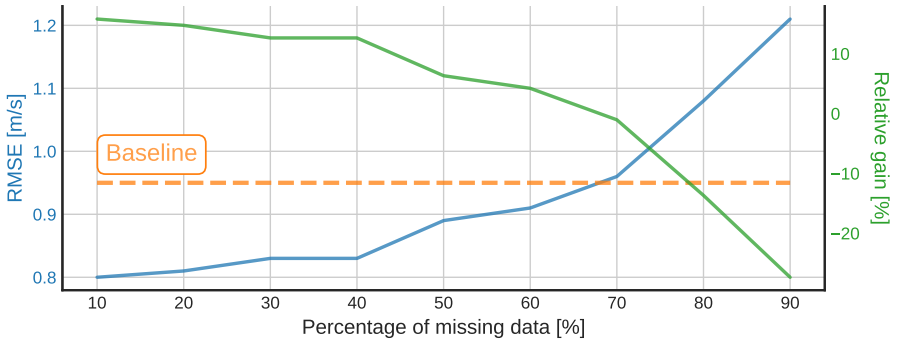


Figure 3.7 – Missing data, 4DVarNet-UPA+ECMWF 10 iteration. The orange dashed line represents the baseline performance level. The value 0 for the relative gain (right y-axis) is attained at about 70 % of missing input data.

analogous to dropout mechanics, artificially removing features and/or noising the data samples [228], [229]. We may also note that the proposed approach reaches almost the same performance as the CatBoost model presented in [207] up to 70 % of missing data.

3.4.6 A-posteriori classification performance

We propose a complementary analysis to assess the performance of our model in a classification perspective. For operational applications it is more practical to express the wind speed as wind classes rather than numerical values. An empirical way to classify wind speed categories is the Beaufort scale, a subdivision based on observable effects of wind on land and sea scenarios. The Beaufort scale is composed on 12 classes. The class 0, called *calm* is associated to wind speed values lesser than 0.5 $\mathrm{m\,s^{-1}}$. The class 12, called *Hurricane-force* is associated with wind speed greater than 32.7 $\mathrm{m\,s^{-1}}$. Due to the wind

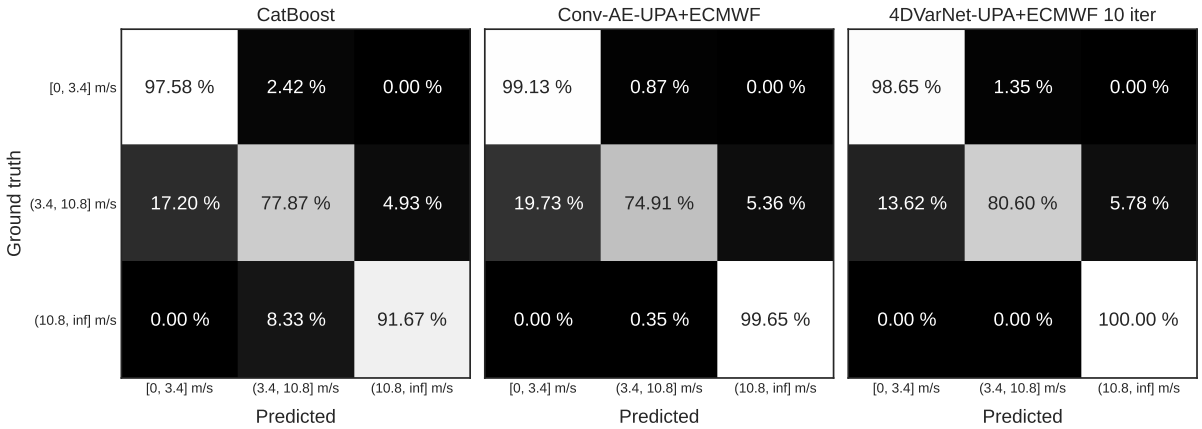


Figure 3.8 – Confusion matrices related to the a-posteriori classification task of wind speed.

speed distribution of the test set, visualized in the bottom-right panel of Figure 3.2, we choose to define three macro-classes as follows. The first class, "low wind" comprehends the wind speed values ranging from 0 to 3.4 $\mathrm{m\,s^{-1}}$. In terms of the Beaufort scale these are winds up to the *gentle breeze*. The second class, "median wind" comprehends wind speeds from 3.4 to 10.8 $\mathrm{m\,s^{-1}}$, this latter is the *strong breeze* category according to the Beaufort scale. The last class, "high wind" comprehends all the wind speed values stronger than the *strong breeze*, larger than 10.8 $\mathrm{m\,s^{-1}}$.

The a-posteriori analysis on wind speed classification shows that the 4DVarNet model trained on regression performs better in terms of classification accuracy with respect to CatBoost and the multi-modal direct inversion model. Figure 3.8 reports the confusion matrices of the classification task for each of these three models. The diagonal structure of the matrix is evident for each case. The multi-modal 4DVarNet has a larger precision for medium and high wind values. The average accuracy scores for each class are 89 %, 91 % and 93 % for the CatBoost, multi-modal direct inversion and multi-modal 4DVarNet, respectively. These results show that our framework could be valuable as an instrument to provide reliable qualitative estimates of wind speed, offering operationally useful information about the atmospheric conditions. To conclude, we may remark that a proper classification task on wind speed categories could be approached designing the model, the dataset and the evaluation criterion in such a way to account explicitly for the wind speed values in form of classes. That would be a classfication problem rather than a regression problem, as done in this case study.

3.5 Conclusion

This chapter presented a novel robust and efficient framework for managing time dependence in data sets comprehending underwater acoustics and wind speed values. While previous work successfully demonstrated that machine learning approaches are promising tools to perform wind speed estimation given underwater acoustic data, this work highlights and proves the importance of explicitly accounting for time dependence. This concept could be further expanded to short then long-term forecasting problems, that is predicting a time series of wind speed given a small amount of acoustic data, in such a way to forecast the wind trend in a near future window. We believe that the effective representation of time dependence is particularly relevant, as in an operational scenario, no data are available after a given time. In that case, the model used should be capable of *predict* wind speed in the near future. For this task, a complete characterization on wind speed dynamical behavior is necessary.

Further work may also consider the joint use of acoustic data and satellite imagery, such as Synthetic Aperture Radar (SAR) images. While acoustic data have a limited spatial coverage but have a rich resolution in time, SAR images display opposite characteristics, as they are scarce in time but offer a wider spatial resolution. A multi-modal approach that bridges these two temporally-rich and spatially-rich features could lead interesting research directions aiming to fit trainable models in which learning one modality helps in learning the other. Cross-modal learning and generation is a salient and important feature of multi-modal machine learning [90].

A further improvement could address the target variable to be modelled. In this case study, we mainly focused on the prediction of an environmental variable, but other important applications could benefit from the use of underwater ambient noise for anthropic activities such as submarine recognition or sea wildlife observation.

Chapter 4

Multi-modal learning-based reconstruction of high-resolution spatial wind speed fields

4.1 Context and motivation

This chapter presents the application of the 4DVarNet framework to the reconstruction of sea-surface wind speed fields using spatio-temporally heterogeneous data. The information typically available in an operational setting comes from satellite images, reanalyses and in-situ observations. Section 1.3.2 introduced the data assimilation products and stated that these are a crucial knowledge for climate research and weather forecasting. However, these products do not resolve the finest horizontal scales observed by satellite sensors. For instance, as introduced in Section 1.2.2, satellite Synthetic Aperture Radar (SAR) sensors deliver an observation of the sea surface winds with a 1 km spatial resolution, when operational reanalysis products typically resolve horizontal scales between a few tens and one hundred kilometers [64], [209]. However, remotely sensed products can not continuously monitor one given area. This limitation prevents the analysis to be informed by the rapidly evolving wind speed patterns. In-situ sensors (cfr. Section 1.2.1) on the other hand, can provide local sub-hourly measurements that capture the temporal variability of surface winds. These remarks on the characteristics of available information suggest that this kind of analysis is inherently multi-modal.

In this work, we explore the design of learning-based schemes to make the most of available multi-modal datasets for the reconstruction of sea surface winds. In particular, we design an Observation System Simulation Experiment [22] to assess quantitatively the impact of each input source on the overall reconstruction performance. The analyses articulate as follows. We base our analyses on synthetic data, in order to avoid the problem of missing data and we manufacture high-resolution, low-resolution spatial wind fields and

point-wise time series to emulate satellite images, reanalyses and in-situ observations, respectively. We define four experimental data configurations to simulate different data availability configurations. We benchmark the 4DVarNet scheme against two baseline models to assess the improvement related to the model itself. In addition, we assess the impact of artificially biased low-resolution data to test the robustness of the model against model errors inherently present in reanalyses products. We also assess the impact of the temporal sampling frequency for the satellite high-resolution spatial fields and we perform a sensitivity test with respect to the in-situ pseudo-observations.

The rest of this chapter is structured as follows. Section 4.2 presents the dataset used and how the original data are prepared for our simulations. Section 4.3 details the model experimental setup. Section 4.4 presents the results obtained for each test case and Section 4.5 critically discusses these results.

4.2 Data

We focus here on the exploitation of simulation datasets. We may emphasize that the direct exploitation of real observation datasets is not straightforward. It would imply a huge amount of work for mining and co-locating data on the same spatio-temporal grid, with no guarantee to build a sufficiently large and consistent dataset on a regional scale to run the targeted learning-based experiments. We then prioritize a preliminary study with simulations datasets. Recent studies [230] also suggest the potential of such simulation-based datasets to train deep learning schemes which apply to real observation datasets.

We use the output of the RUWRF model (Rutgers University Weather Research and Forecast [231]), based on the version 4.1.2 of the Weather Research and Forecast model (WRF, cfr. Section 1.3). The RUWRF is developed by the Rutgers University Center for Ocean Observing Leadership. This model runs a parent nest with resolution 9 km for a time interval of 120 hours and then a child nest with resolution 3 km out of 48 hours. The model is run daily. This implies a discontinuity in the data time series between 23:00 and 00:00 of two consecutive days. This motivates us to use 24 hours as reference time series length. The wind speed data are available in terms of the horizontal components. Our analyses target the modulus of wind speed. We process the components into the vector norm in order to treat the wind speed modulus. These wind speed fields are chosen to have a spatial extent of roughly 644 $\times$ 645 km, with a spatial resolution of $0.03^\circ \times 0.03^\circ$ (on

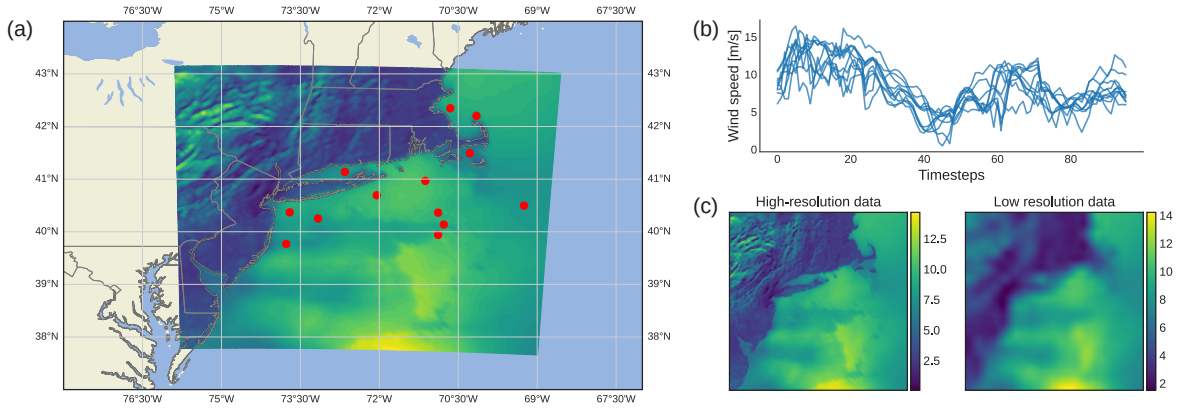


Figure 4.1 – Dataset qualitative characteristics. Panel (a): Geographical region considered. The red markers represent the buoys positions. Panel (b): Sample in-situ time series pseudo-observations obtained from wind speed values at buoys positions. Panel (c): The downsampling-reinterpolation step to obtain low-resolution pseudo-observations from ground-truths. The spatial resolution for this case is 30 kilometers.

average 3 km). The time window selected spans from 01/01/2019 at 00:00 to 01/01/2021 at 23:00. We prepare the wind speed fields and in-situ data as time series of length 24. Figure 4.1 displays examples of wind speed fields on the spatial region considered with associated pseudo-observations.

4.2.1 Low-resolution data

We simulate low-resolution (LR) data similar to reanalysis data bases obtained using state-of-the-art data assimilation schemes (cfr. 1.3.2). Such products typically resolve a wide range of spatial and temporal scales, depending on whether the numerical weather prediction (NWP) model is configured to address mesoscale, synoptic or climate scales phenomena. The spatial scales involves range between few tens to thousands kilometers. Likewise, temporal scales range from sub-daily to yearly. We adopt the following strategy. LR wind speed fields are obtained by down-sampling the original data. These fields are then re-interpolated on the reference grid to match the spatial resolution of the original data, as in Panel (c) Figure 4.1. We choose to manufacture the LR data to have spatial resolution of 30 to 100 km and a time-step of 1 to 6 hours. This allows us to emulate products that resolve different spatio-temporal scales. The 30 km and 1 h configuration matches with the ECMWF ERA-5 data base [64]. The configuration 100 km and 6 h matches with the previous ECMWF reanalyses data base ERA-interim [209].

Reanalyses datasets may also involve local errors and biases imputable to Gaussian assumptions (cfr. Section 2.2). These errors may relate to random delays in the weather forecast or random phase alteration. This kind of anomalies are observable in real-world NWP outputs, that may predict a given weather phenomenon at a given time, but this phenomenon is observed earlier or later, and with an intensity that may be different than the one predicted. In our experiments, we account for these possible biases by artificially injecting a phase delay or an amplitude re-modulation. Throughout the rest of this chapter, we refer to LR fields and NWP products (or pseudo-observations) interchangeably.

4.2.2 High-resolution data

We aim to mimick to some extent the sampling pattern of SAR satellite sensors [29], [51]. We assume pseudo-SAR observations as noise-free HR snapshots of the sea surface wind speed. We focus here on the time sampling pattern. We assume that we are respectively provided by one and two HR observations over each 24-hour window. With a view to assessing the added value of a second SAR observation within a 24-hour window, we assume for the sake of simplicity that the two HR observations are sampled with a 12 hours delay.

Overall, we simulate here two HR observation datasets: for the first one, we provide a HR observation at the center of the 24-hour time window, namely at 12:00. For the second one, two HR observations are placed at 06:00 and 18:00, so to be 12-hours away. In the two cases the temporal sampling frequency of HR data is 24 and 12 hours respectively. In the experiments, we tested both these frequencies. Figure 4.2 gives a visual explanation of observations sampling patterns on a 24-hours time window. HR observations are masked in order to keep the wind speed information associated with the sea surface. This choice is motivated by the fact that SAR imagery can provide wind speed information by the sea surface roughness. This is not possible to achieve on the land surface. Satellite-based remote sensing products have been introduced in Section 1.2.2.

4.2.3 In-situ time series

In-situ observation infrastructures (cfr. Section 1.2.1) provide real-time in-situ measurements. In-situ data have a fine temporal resolution and directly measure the quantity of interest, thus not being prone to any model error. The limitation of in-situ sensors is the deployment cost and limited spatial coverage. We simulate pseudo in-situ time se-

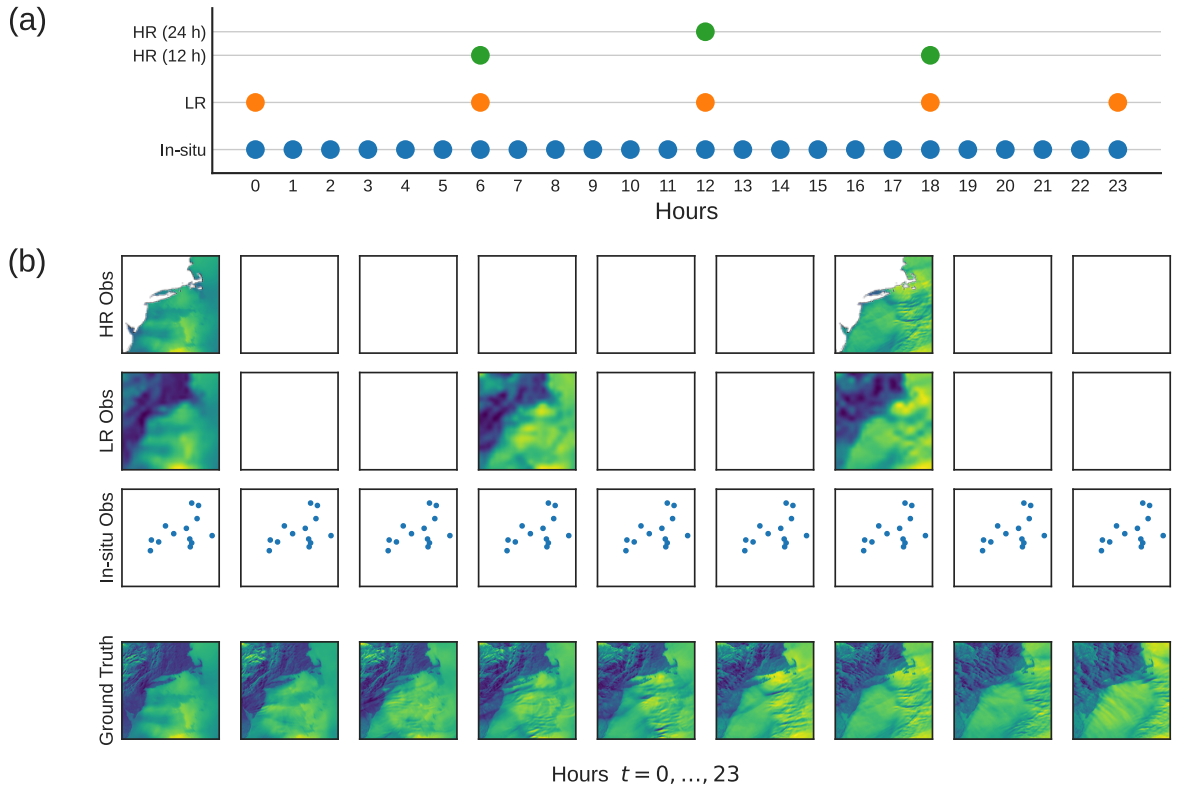


Figure 4.2 – Panel (a): Temporal sampling patterns of the data used. The items "HR (12 h)" and "HR (24 h)" refer to the datasets in which the HR observations are simulated to have temporal frequency of 12 or 24 hours. These items refer to different experimental configurations and are depicted on the same plot for graphical convenience. Panel (b): An example of the dataset items. The temporal sampling frequencies of HR and LR fields are fictitious and aim to illustrate the dataset.

ries accounting for the positions of the weather buoys of the NOAA National Data Buoy Center network (website: `https://www.ndbc.noaa.gov/`), depicted in panel (a) of Figure 4.1. Refer to Green [232] for an overview of NOAA buoys installations. We select the buoys included in chosen region that were active in the chosen dataset temporal window. Using the positions of these buoys on the data grid, we keep the pixel values of the HR spatial fields and we extract these set of positions so to have multi-variate time series of point-wise wind speed values. As a result, we have thirteen buoys in the dataset. A portion of these time series is shown in panel (b) of Figure 4.1.

4.2.4 Preprocessing scheme

The dataset is composed of 732 time series of 24 wind fields. We allocate the first 432 series for the training set, the subsequent 200 series for the test set and the last 100 series for the validation set. In order to implement the simulations on LR data biased by random phase delays and amplitude remodulations we extract 36-hours time series in an early data processing stage. This choice is motivated by the fact that the hour 00:00 of each series may be modified by the random delay. If negative, this delay assigns to the field at 00:00 a value of the previous series. In this way, the early-stage series cover a time window from 18:00 of day $T-1$ to 06:00 of day $T+1$. This preprocessing step implies the loss of the first and last days of each data set. The training, validation and test set will then have lengths of 430, 198 and 98, respectively. Once the LR bias has been injected, if the test case requires it, the 36-hours time series are cropped in order to have 24-hours long sequences starting at 00:00 and ending at 23:00 of day T.

To sum up, the final dataset is composed of the three modalities mentioned above: the HR fields, the LR fields (with the injected bias if the training configuration requires) and the in-situ time series. From a practical point of view the dataset object provides a time series for each modality. Each time series is composed by 24 matrices having the shape of the spatial domain. The series are masked according to the temporal sampling frequency and the spatial features of each modality. For example, the in-situ observations are obtained by setting to zero the whole field except for the positions of the buoys. Panel (b) of Figure 4.2 gives a visual intuition of the data appearance. The three time series related to the pseudo-observations are obtained from the ground truths, the bottom series.

Data of the training, validation and test sets are normalized by a field-wise division by the standard deviation of the the respective set. For example, each field of the training set is divided by the standard deviation of the entire training set. Once the model is trained, data are de-normalized in order to evaluate the model performance in terms of physical dimensions.

4.3 Proposed method

Here we state formally the problem and we provide the methodological and numerical details and the learning scheme used throughout our experiments.

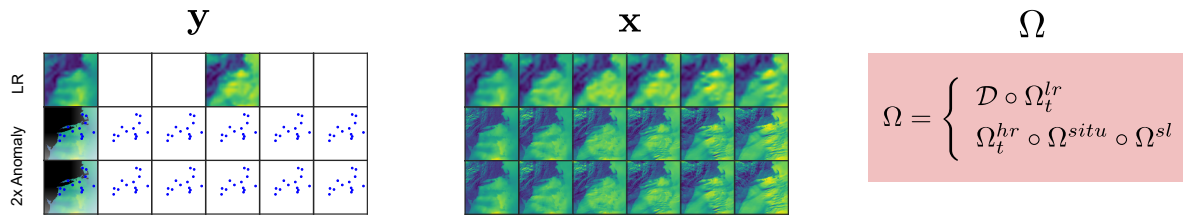


Figure 4.3 – Representation of the augmented states for the observations and the state variable. The leftmost bottom panels of the observations object are the anomaly fields masked by the land-sea mask. The blue dots represent the pixel values associated with the in-situ pseudo-observations.

4.3.1 Problem statement

The data assimilation approach

The formal problem statement follows closely the state-space model introduced in Section 2.2.1. The objective is to invert the (forward) observation process in order to retrieve the state variable starting from its partial observations. We choose to use the 4DVarNet framework to perform this inversion. We may recall that the appeal of the 4DVarNet model stems from its underlying state-space formulation. This allows to parameterize in an end-to-end manner the data assimilation problem exploiting both the physical relevance and the computational and representational flexibility of deep learning modelling. The reader is referred to Section 2.4 for the extensive introduction of the 4DVarNet scheme. In our case, the state variable $\mathbf{x}$ is the temporal sequence of HR wind speed field. The forward process $\mathcal{H}$ provides the observations $\mathbf{y}$ described in the previous section. More specifically, let $\mathbf{y}^{lr}$, $\mathbf{y}^{hr}$ and $\mathbf{y}^{situ}$ be the LR, HR and in-situ pseudo-observations respectively.

Given the spatio-temporal heterogeneity of the partial observations, we choose to set up the problem in terms of augmented-state dynamics [216]. The state variable is prepared as a concatenation of the LR and anomaly fields. We define the anomaly as the difference between the HR and the LR fields. In formulae,

$$\mathbf{y}^{an} = \mathbf{y}^{hr} - \mathbf{y}^{lr} \tag{4.1}$$

We choose to use the anomaly field in place of the HR because the LR part conveys information about the field average and the anomaly is the deviation of the HR field from the LR average. Using the full HR field alongside with the LR field would imply an information redundancy. In the 4DVarNet computational scheme, we will use the *hr*

index to refer to the anomaly parts. Figure 4.3 illustrates the preparation choices for the observations and the state variable and the associated observation operator binary mask.

Let Ω^{lr}, Ω^{hr} and Ω^{situ} be the binary masks that identify the temporal sampling patterns associated to HR, LR and in-situ observations. Let Ω^{sl} be a mask that covers the pixel locations of the land surface. This mask enforces the SAR-like observations to be available only on the sea surface. Formally we can express the observation operator as follows

$$\mathcal{H}(\mathbf{x}) = \begin{cases} \mathbb{I}^{hr}(\mathbf{x}) & = \mathbf{y}^{hr} \\ \mathcal{D} \circ \mathbb{I}^{lr}(\mathbf{x}) & = \mathbf{y}^{lr} \\ \mathbb{I}^{situ}(\mathbf{x}) & = \mathbf{y}^{situ} \end{cases} \tag{4.2}$$

The symbols $\mathbb{I}^{hr}$, $\mathbb{I}^{lr}$ and $\mathbb{I}^{situ}$ represent the indicator functions associated with the spatio-temporal domains $\Omega^{hr} \cup \Omega^{sl}$, Ω^{lr} and Ω^{situ} respectively. The symbol $\mathcal{D}$ represents the downsampling-reinterpolation operation performed to obtain the LR fields. The objective is to retrieve the complete time series of HR surface wind speed using these partial observations. This problem is solved by optimizing the variational cost function that relates the observations and the dynamical model integration as in the weak-constrained 4DVar scheme. For convenience, we restate the expression of the variational cost function used to solve the 4DVar problem.

$$U_{\Phi}(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_1 \left\| \mathcal{H}(\mathbf{x}) - \mathbf{y} \right\|^2 + \lambda_2 \left\| \mathbf{x} - \Phi(\mathbf{x}) \right\|^2 \tag{4.3}$$

The operator Φ is the neural network parameterization of the dynamical operator $\mathcal{M}$ and the λ parameters are weights for the data-model proximity and regularization terms. The complete treatment of the 4DVarNet scheme was provided in Section 2.4.

The deep learning end-to-end approach

Section 1.4 introduced the application of deep learning modelling to geophysical inversion. An alternative way to state the inversion problem involves the direct application of deep learning models. These techniques can be viewed in the perspective of computer vision algorithms, such as inpainting (cfr. Section 1.4) and classification applied to geophysical fields. The flexibility of deep neural networks in the approximation of input-output relationships and their representative power has been mentioned in Section 2.3 and Section 2.3.4 specified the application of deep learning modelling for the direct inversion of

a general forward process. This strategy allows to directly retrieve the state variable $\mathbf{x}$ as the output target of a multi-layered architecture as in Equation (2.24). For convenience, we restate this expression as

$$\mathbf{x} = f(\mathbf{y}; \boldsymbol{\theta}) \tag{4.4}$$

where $\boldsymbol{\theta}$ are the model f parameters. The capability of deep networks to build effective feature maps and learn the input-output relationships from data allows to retrieve the system state directly, with no physical constraints. The difference between the inversion approach based on variational data assimilation methods is clear. A learning-based direct inversion as stated by a relationship as that stated in Equation (4.4) does not require the knowledge of the dynamical model $\mathcal{M}$.

Deep learning methods can also be useful for super-resolution problems. This computer vision task aims to improve an image resolution. Previous work applied learning-based modelling to this end [233]. Impressive results are achieved by Generative Adversarial Networks (GANs [146]). Recent work applied GANs to the super-resolution of wind speed [234], temperature [235] and precipitation [236]. However, GANs may be difficult to train due to the sensible choice of hyper-parameters and the model size. Recent advances in the deep learning field presented diffusion models [148] as appealing alternatives for their efficiency and performance. Recent work applied diffusion models to high-resolution solar radiance forecast [237], seismic waves [238] and medium-range weather forecast super-resolution [239]. While generative deep learning-based super-resolution allows to generate graphically-realistic fields, our purpose is to apply deep learning modelling to reconstruct a sequence of wind speed fields as close as possible to reality. A wind speed field generated by a GAN or diffusion model may look more realistic than a direct inversion-produced one but there is no guarantee for the former to be physically plausible.

4.3.2 Trainable data assimilation scheme

The classical way to approach the problem stated in Section 4.3.1 is to approximate the one-step-ahead predictor as defined by Equation (2.34) with Euler or Runge-Kutta schemes [192], [193]. In the 4DVarNet scheme, we parameterize this operator with a trainable neural operator. This defines a bi-level optimization [188] that aims to both minimize the variational cost to find the state variable and another cost function that measures the reconstruction performance of the model in order to adjust the neural network-based model parameters.

The parameterization of the observation term depends on the experimental configuration. In the simplest case, the observation term is the data-state misfit restricted to the observation spatio-temporal domain. For example, if the sampling frequency of HR fields is 12 hours, then the observation operator delivers the HR observations of the spatial wind speed at hours 06 and 18 for each daily time series. This reduces the problem to the task of learning an interpolation operator that fills the gaps imputable to the observation process and to improve the spatial resolution of the interpolated fields. Recalling the form of the variational cost (4.3), restated in matrix form, this simple case of non-trainable data fidelity term (DFT) reduces to the optimization of the following cost function

$$U_\Phi(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_1 \left\|\mathbf{x} - \mathbf{y}\right\|_\Omega^2 + \lambda_2 \left\|\mathbf{x} - \Phi(\mathbf{x})\right\|^2 \tag{4.5}$$

We call this case "single-modal DFT". In a second case, we still have to enforce the sampling pattern with a binary mask but we add a second observation term in the variational cost. This term is the distance between the higher-level descriptors extracted from the observations and the state variable. This "multi-modal DFT" learns feature maps from spatial and/or sequential observations and allows us to relate the state variable and the observations at a higher abstraction level that transcends the spatio-temporal characteristics of the raw observations. We think that this is particularly relevant as spatial scales which characterize spatial fields and point-wise time series are drastically different. We may draw an analogy between this choice for the second observation term of the 4DVarNet variational cost and the composite loss function design mentioned in Section 2.3.4. The trainable term acts as an *perceptual loss* [177] that relates the observations and the state variable at an higher level of representation. In this case the variational cost (4.3) can be stated as follows

$$U_\Phi(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_1 \left\|\mathbf{x} - \mathbf{y}\right\|_\Omega^2 + \lambda_1 \|\psi_{\mathbf{x}}(\mathbf{x}) - \psi_{\mathbf{y}}(\mathbf{y})\|_\Omega^2 + \lambda_2 \|\mathbf{x} - \Phi(\mathbf{x})\|^2 \tag{4.6}$$

In this equation, $\psi_{\mathbf{x}}$ and $\psi_{\mathbf{y}}$ are the neural networks. Depending on whether the observations $\mathbf{y}$ are spatial (HR wind fields) or sequential (point-wise time series), the networks classes $\psi_{\mathbf{x}}$ and $\psi_{\mathbf{y}}$ are 2D or 1D convolutional, respectively. The state variable $\mathbf{x}$ is intended to be a spatial field in any case. The observations $\mathbf{y}$ are multi-variate time series or spatial fields, or both, depending of the experimental configuration.

The trainable gradient solver Γ (cfr. Section 2.4.2) is parameterized with a 2D convolutional LSTM network. The optimization of the state variable is achieved by an iterative

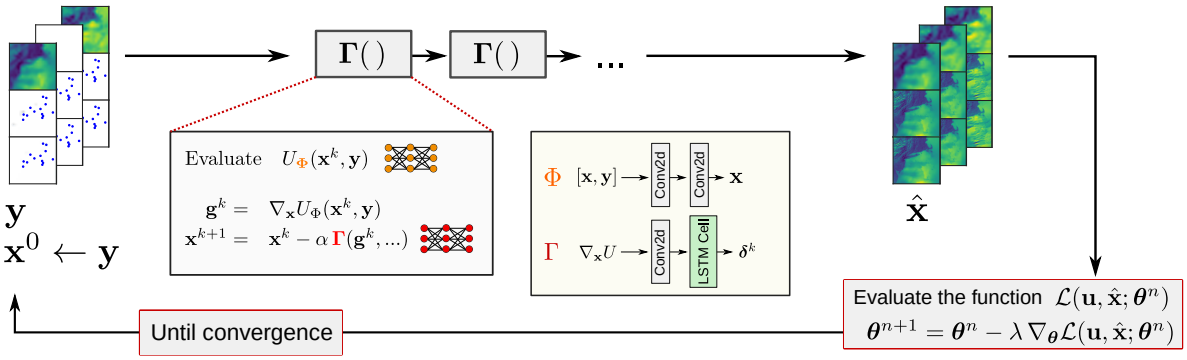


Figure 4.4 – Diagram and configuration of the 4DVarNet scheme for this case study.

application of the solver to the gradients of the variational cost. The expression of the iterative state variable update is stated in Equation (2.32). Figure 4.4 visualizes the workflow of the 4DVarNet scheme applied to this case study.

4.3.3 Learning scheme

The training loss chosen to optimize the model parameters is a mean squared error (MSE) composed of terms related to LR and HR reconstructions. The input data to the model are the LR and the anomalies of the spatial fields. Figure 4.3 visualizes the compositon of observations and state variable. As reported in the figure, the augmented states involve one LR and two HR (anomaly) fields. The first instance of the anomaly is used in the 4DVarNet computations, the state variable optimization and the second is used as reconstructed output. This choice is motivated by the fact that the part processed by the 4DVarNet operators may have undesired artifacts in the reconstruction [240]. The training loss comprehends also one term which enforces the spatial gradients of the HR reconstructions and ground-truth data to be similar. Let $\mathbf{u}$ represent the ground-truths used in the problem design and $\hat{\mathbf{x}}$ the model output. The training loss can be formally stated as

$$\mathcal{L}(\hat{\mathbf{x}}, \mathbf{u}) = \frac{1}{M} \sum_{i=0}^{M} \sum_{t=0}^{T} \left\{ \|\mathbf{u}_{it}^{lr} - \hat{\mathbf{x}}_{it}^{lr}\|^2 + \|\mathbf{u}_{it}^{hr} - \hat{\mathbf{x}}_{it}^{hr}\|^2 + \|\nabla \mathbf{u}_{it}^{hr} - \nabla \hat{\mathbf{x}}_{it}^{hr}\|^2 \right\} \tag{4.7}$$

The symbol ∇ identifies the spatial gradients of a field w.r.t. the spatial coordinates. The terms of the loss function involving gradients are needed to enforce the spatial variation patterns of the reconstructions to match those of the ground-truths. The training criterion (4.7) is complemented by a L_2 regularization term to prevent overfitting.

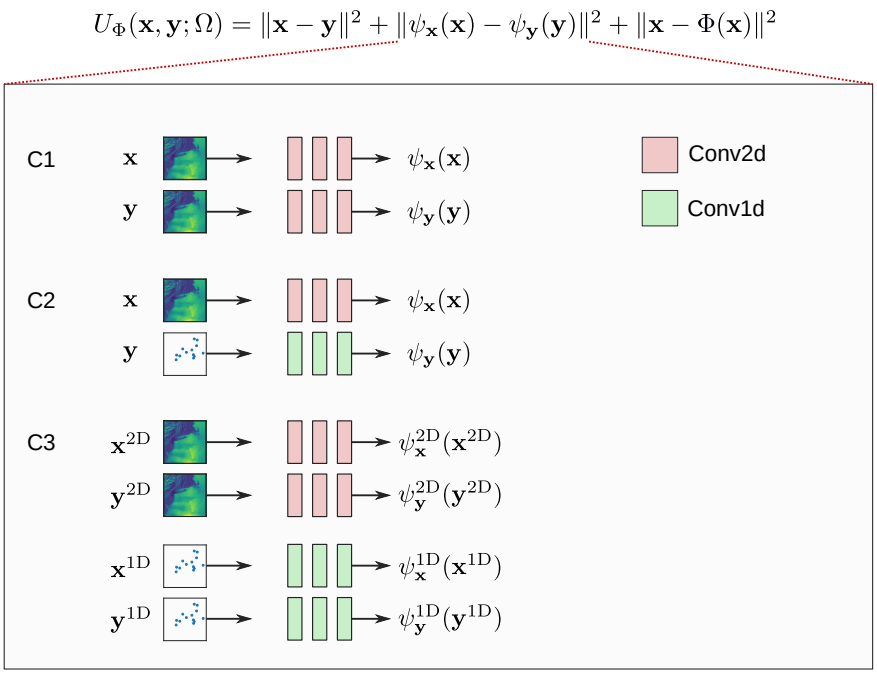


Figure 4.5 – Schematic representation of the features extraction networks.

4.3.4 Numerical implementation

The optimization of the cost function (4.7) is done with the Adam algorithm [241]. The model is trained for 50 epochs. The weights configuration used for the test stage is selected according to the validation loss during training. Table 4.1 resumes the learning rates and L_2 regularization weight decay coefficients for the trainable models deployed. Our choice is to set the weights of the variational cost terms $\lambda_{1,2}$ to trainable parameters. In the following we report briefly the neural architectures used for this work. The interested reader may refer to the code available at `https://github.com/CIA-Oceanix/4DVN-MM-W2D/tree/main` for the complete and detailed models design.

- The operator Φ is parameterized by a 2D convolutional network with two layers. The kernel size is chosen to be 5 with padding 2 (both are isotropic). The input tensor is a concatenation of three times the spatial fields (LR and anomalies), so the temporal dimension becomes 72. For this reason, the input channels are set to 72. The first layer squeezes the channels to 32 and the second layer expands the channels to 72. The model does not have non-linearities.
- The operator Γ is parameterized by a 2D convolutional LSTM. The dimension of the hidden cell is set to 100. The gates layer is parameterized by a 2D convolutional

	Operator Φ	Solver Γ	Models $\psi_{\mathbf{x}}$, $\psi_{\mathbf{y}}$	Weights $\lambda_{1,2}$
Learning rate	$5 \cdot 10^{-5}$	$9 \cdot 10^{-5}$	10^{-4}	10^{-4}
L_2 weight decay	10^{-7}	10^{-8}	10^{-7}	10^{-5}

Table 4.1 – Hyper-parameters used for the simulations. Depending on the case, some columns may not apply. For example, in the case of non-trainable DFT, the column $\psi_{\mathbf{x}}$, $\psi_{\mathbf{y}}$ is not considered.

layer with kernel size 3 and padding 1. The input channels are 96 and the gates layer expands this number to 384. A linear Conv2d layer (kernel size 1 and no padding) reshapes the LSTM output to the original input size.

- The models $\psi_{\mathbf{x}}$ and $\psi_{\mathbf{y}}$ are involved in the multi-modal DFT case. They process the system state and observations respectively. The model for the system state $\psi_{\mathbf{x}}$ is a 2D convolutional model. According to whether the observations are spatial wind fields or in-situ time series, the model $\psi_{\mathbf{y}}$ is a cascade of 2D convolutional layers and Average pooling layers or a Conv1d layer. We refer to Section 2.3.2 for the convolutional layers explanation. In the case where both spatial and time series are involved, two different models $\psi_{\mathbf{y}}$ are deployed. The extracted features maps are flattened to be compared by the variational cost observations term. Figure 4.5 represents graphically the networks used as features extraction models.

The simulation are run on a machine mounting 8 Nvidia A100-SWM4 graphical processor units. These units have 80 GB memory and GA100 graphics processors with 1593 MHz memory clock speed. We provide the motivation for the design choice of the model Φ and supporting sensitivity tests in Appendix A.

4.4 Results

Results articulate as follows. We detail the baseline models that we choose to compare the 4DVarNet framework with. We provide the technical details about the evaluation metrics and we detail each experimental configuration. The baseline models and the 4DVarNet are compared. The analyses on robustness against biased LR fields, buoys sensitivity and scale analyses are made using the best model as of the benchmark configuration. The results of the analyses on high and low-resolutions sensitivity are obtained by models trained on the specified data sampling configurations.

4.4.1 Evaluation framework

The reconstruction performance is evaluated quantitatively in terms of root mean squared error (RMSE) between ground-truth data and the output of each model. The reconstruction error is evaluated on the complete region considered and on both land and sea areas separately. We perform 10 runs for each configuration and we compare the median reconstruction against the ground-truth. This procedure can be viewed as a *voting* average to aggregate an ensemble of models [242]. Ensemble Machine Learning methods construct a group of independently trained models. The overall ensemble output has an overall reduced variance [243]. The aggregated median reconstruction of the models ensemble is obtained as

$$\hat{\mathbf{x}}_{\text{median}} = \text{Median}\left(\{\hat{\mathbf{x}}_n \,;\; 0 \leq n < N_{\text{Runs}}\} \right) \tag{4.8}$$

where $\hat{\mathbf{x}}_n$ is the run-wise model output. The model performance is evaluated against a chosen baseline. The next Section provides a detailed overview of models and baselines. We may define a relative gain to compare the percentage improvement of each model w.r.t. the selected baseline. This gain in defined as

$$\eta = \left(1 - \frac{p_M}{p_B}\right) \times 100 \tag{4.9}$$

where p_M and p_B are the reconstruction RMSEs of a given model M and the reference baseline B respectively. The performance p_M is referred to the ensemble reconstruction. This gain may attain virtually any real value. If it is negative, we refer to it as "degradation" as the performance of a model M may be worse than the baseline B performance.

4.4.2 Benchmark analysis results

In order to benchmark the 4DVarNet model we propose the following configuration of available data: (i) LR wind speed fields simulating a NWP product with a sampling frequency of 6 hours; (ii) HR pseudo-observations that simulate the satellite images, available with a temporal sampling frequency of 12 and 24 hours, respectively at hours 06:00 and 18:00 for the first case and 12:00 for the second; (iii) in-situ time series, with hourly resolution. We identify four test cases given by the combination of these data modalities. Table 4.2 resumes these configurations. The 4DVarNet scheme is compared with two base-

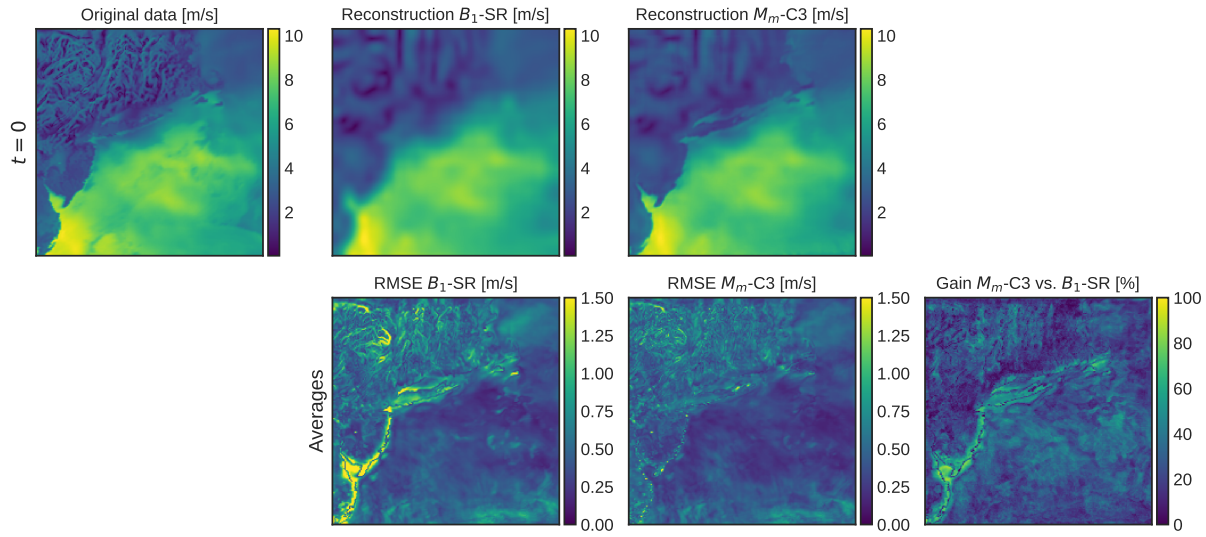


Figure 4.6 – First row: (left) Original data, (middle) reconstruction of the B_1-SR baseline, (right) reconstruction of the M_m-C3 4DVarNet. Second row: (left) Map of average MSE related to the B_1-SR baseline, (middle) M_m-C3 4DVarNet and (right) map of the average relative gain of the 4DVarNet w.r.t. the baseline. The temporal sampling frequency for high-resolution fields is 12 hours. The two rows are displaced in order for the baseline and model reconstructions and error maps to match vertically.

line models. The first baseline used is a temporal interpolation of LR fields, with no HR observations. In this case, the interpolation operator is not trainable. The second baseline model is a learning-based direct inversion (cfr. Section 4.3.1) applied to the data configurations of Table 4.2. The target state variable $\mathbf{x}$ is obtained by the direct application of a trainable operator to the observed data as stated by Equation (4.4). The direct inversion for the data configuration SR solves a *super-resolution* task. This means that the model is trained to retrieve the finer-scale information using only the LR fields. Let B_0 and B_1 represent the interpolation and direct inversion baselines respectively. Let M_s and M_m denote the 4DVarNet framework in the single-modal and multi-modal DFT settings respectively (cfr. Section 4.3.2). In the following, the notation $\{B_0, B_1, M_s, M_m\}$-$\{\mathrm{SR}, \mathrm{C1}, \mathrm{C2}, \mathrm{C3}\}$ is used to identify the combination of model and experimental configuration. This convention is used hereafter in the rest of the paper. The 4DVarNet dynamical operator Φ shares the same architecture of the trainable direct inversion model.

Table 4.3 reports systematically the simulation results considering the data configurations mentioned above. The model M_m-C3 4DVarNet is the configuration that gives the best reconstruction performance. In Figure 4.6 the first row shows original data and

Configuration	LR data	HR data	In-situ data
SR	6 h	✗	✗
C1	6 h	{12, 24} h	✗
C2	6 h	✗	1 h
C3	6 h	{12, 24} h	1 h

Table 4.2 – Experimental data configurations. The curly brackets for the cases C1 and C3 represents the set to HR sampling frequencies inspected.

reconstructions examples of both the baseline B_1-SR and the model M_m-C3. The second row shows the average RMSE map of B_1-SR and M_m-C3 (first and second bottom panels) and the average relative gain of M_m-C3 w.r.t. B_1-SR. This result may not raise any surprise, since the more the data used the more information is available for the reconstruction. Nevertheless, a very interesting result concerns the performances of the two instances of the 4DVarNet scheme, M_s and M_m. Despite the model M_s outperforms the direct inversion baseline, the addition of in-situ observations in the configuration C3 does not seem to be beneficial w.r.t. the configuration C1. Contrarily, the M_m model benefits more from in-situ time series in configuration C3. The reconstruction performance is roughly 2 % superior w.r.t. the case C1 for both the temporal frequencies. This result proves the capability of an explicit multi-modality processing to make the most out of both the sources of HR information.

In Figure 4.7 we report the visual representation of the improvement due to the multi-modality of the DFT by comparing the reconstruction gains of cases C3 and C1. For this visualization, the temporal sampling frequency chosen is 12 hours. The left panel represents the average gain of the model M_s-C3 w.r.t. M_s-C1, and the right panel is the average gain of model M_m-C3 w.r.t. M_m-C1. Intriguingly, the gains are not only restricted to areas surrounding the buoys but extend to regions of about one order of magnitude larger than the spatial scale of a local point-wise measurement. This improved reconstruction can even reconstruct the profile of the coastline. This result shows that a trainable multi-modal approach may incorporate the heterogeneous information of in-situ observations and spatial HR observations. The local small-scale information is learned and reused to a larger scale, which is defined by the spatial HR data.

Model		ω_t^{hr}	Full		Sea		Land	
			RMSE	Gain	RMSE	Gain	RMSE	Gain
B_0			1.1234		1.1384		1.0983	
B_1	SR	–	0.9960		0.9817		1.0192	
	C1	12 h	0.9605	3.56	0.9389	4.36	0.9951	2.36
	C1	24 h	0.9741	2.20	0.9555	2.67	1.0040	1.49
	C2	–	0.9957	0.03	0.9814	0.03	1.0187	0.05
	C3	12 h	0.9571	3.91	0.9341	4.85	0.9938	2.49
	C3	24 h	0.9711	2.50	0.9515	3.08	1.0024	1.65
M_s	C1	12 h	0.9000	9.64	0.8930	9.04	0.9114	10.58
	C1	24 h	0.9012	9.52	0.8953	8.80	0.9108	10.64
	C2	–	0.9619	3.42	0.9508	3.15	0.9798	3.870
	C3	12 h	0.8999	9.65	0.8939	8.94	0.9096	10.75
	C3	24 h	0.9015	9.49	0.8958	8.75	0.9107	10.65
M_m	C1	12 h	0.8802	11.63	0.8695	11.43	0.8974	11.95
	C1	24 h	0.8907	10.57	0.8836	9.99	0.9022	11.48
	C2	–	0.9197	7.66	0.9207	6.21	0.9180	9.93
	C3	12 h	0.8617	**13.48**	0.8481	**13.61**	0.8836	13.30
	C3	24 h	0.8692	12.73	0.8606	12.34	0.8832	**13.34**

Table 4.3 – Benchmark test results. B_0: interpolation. B_1 learning-based direct inversion. M_s and M_m: 4DVarNet model with non-trainable and trainable observation operator, respectively. The gains are referred to the B_1-SR configuration and are computed as in Equation (4.9). RMSE is expressed in $\mathrm{m\,s^{-1}}$ and relative gain in percentage. The six three columns report the RMSE and relative gain for the full region, the sea and land portions respectively. The symbol ω_t^{hr} refers to the temporal sampling frequency of high-resolution fields.

4.4.3 Biased low-resolution data

The prediction made by a NWP system and the true realization of the phenomenon may differ in both timing and intensity. To simulate this scenario, the LR fields are biased by either a random phase delay Δt in the interval $[-4, +4]$ hours or a random amplitude re-modulation α in the interval $[0.5, 1.5]$. This LR data modification is performed randomly at train time. More specifically, each field $\mathbf{y}^{lr}(t)$ [1] is modified by a randomly chosen phase

1. The time step t complies with the temporal sampling frequency prescribed by the observation process.

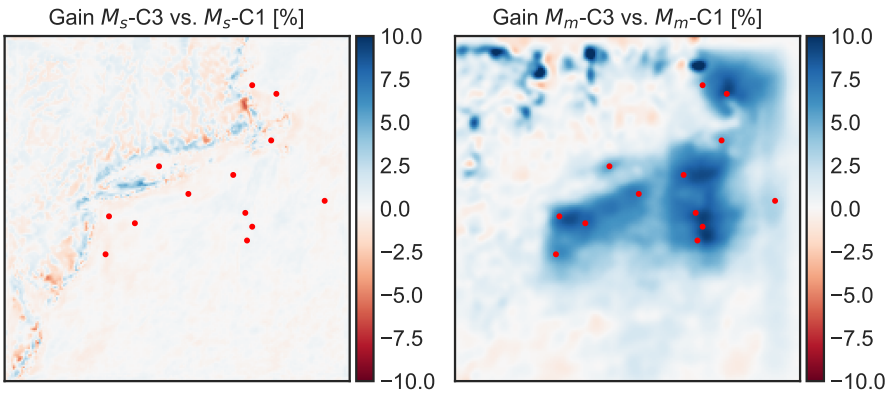


Figure 4.7 – Maps of average relative gains. Left panel: Non-trainable observation operator. Average gain of model M_s-C3 w.r.t. M_s-C1. Right panel: Trainable observation operator. Average gain of model M_m-C3 w.r.t. M_m-C1. The temporal sampling frequency for high-resolution fields is 12 hours.

delay or remodulation. Formally

$$\mathbf{y}^{lr}(t) = \begin{cases} \mathbf{y}^{lr}(t + \Delta t) & \text{with } \Delta t \sim \mathcal{U}(-4, +4) \\ \alpha\, \mathbf{y}^{lr}(t) & \text{with } \alpha \sim \mathcal{U}(0.5, 1.5) \end{cases} \tag{4.10}$$

The symbol $\mathcal{U}$ represents the uniform probability distribution. Interestingly, this procedure has a clear resemblance with dynamic data augmentation [244]. The modification stated in the equations 4.10 involves all the time steps of the time series of the training set. The model M_m-C3 with HR sampling frequency of 12 hours is trained with these two LR data modification separately. HR spatial and in-situ observations are not modified. We expect a degradation in the reconstruction performance, but the foremost interest of this analysis is to prove the robustness of the trainable variational scheme against the presence of model errors in the LR pseudo-observations. The test procedure is performed as follows. The trained model is evaluated on an ensemble of modified test sets, where each of these sets is obtained modifying all the LR elements by one constant biasing value. For example, in the case of random delay the first test set has its LR elements modified by the delay -4 hours. The second test set's LR elements are modified by -3 hours and so on. The same method is applied for the case of the re-modulation bias. This procedure gives an ensemble of 9 and 11 performance metrics for the delay and re-modulation cases.

Figure 4.8 reports the curves of reconstruction error as functions of the bias magnitude for both the delays (first row) and re-modulations (second row). Intriguingly, the model

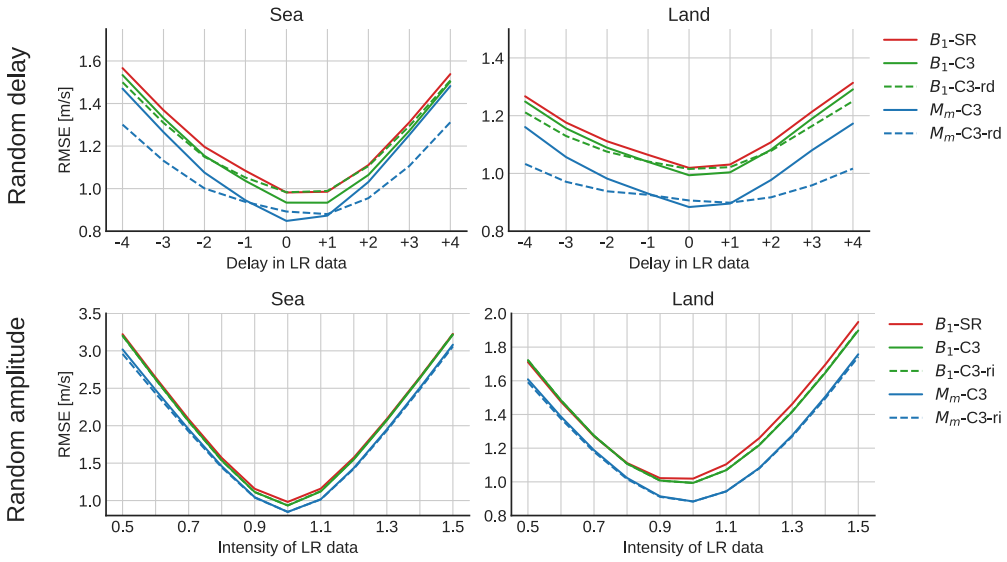


Figure 4.8 – Top row: test case of LR simulated delay. Bottom row: test case of LR simulated re-modulation. The suffixes "-rd" and "-ri" identify the models trained in case of random delay and re-modulation, respectively. The experimental configuration used for this experiment is the case C3 for the M_m model and the SR, C3 cases for the B_1 baseline. The temporal sampling frequency of high-resolution fields is 12 hours.

M_m-C3 trained on biased LR data outperforms the model trained on unbiased LR data for large biases values. This result can be interpreted as the model capability to learn the LR data correction, thanks to the trainable multi-modal observation operator that extracts meaningful features from the HR observations. This is particularly clear in the case of random delay, while for random re-modulation the effect is milder. This may be explained noting that model intensity errors in the range $[0.5, 1.5]$ are not strong enough to spoil the results. Moreover, such biases may, contrarily, have a positive regularizing effect on the model training. The performance of model M_s w.r.t. model M_m is better discussed in Appendix A.

4.4.4 Buoys sensitivity analysis

In this work we focus on the added-value of having in-situ sensors that measure directly the variable of interest. In real-world applications it is not possible to deploy a very dense network of observation infrastructures as such installation operations are expensive and demanding. In the following experiment, we aim to evaluate the impact of missing buoys on test stage using a model trained with both HR spatial wind fields and in-situ

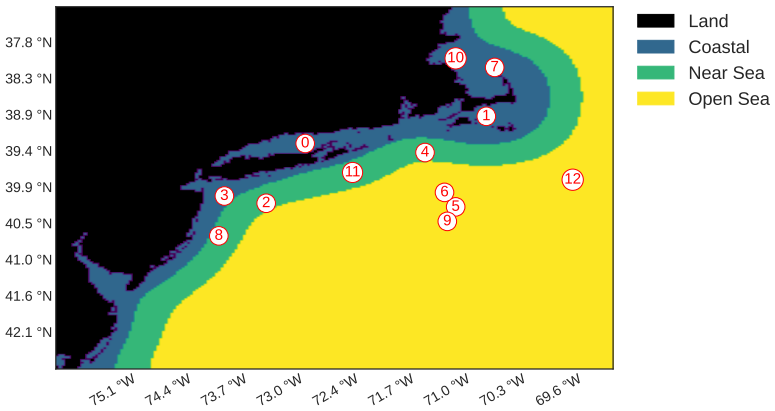


Figure 4.9 – Division of the region of interest in the three zones that host the three buoys groups. The circled numbers represent the buoys identifiers on the respective buoy position. The surfaces of the coastal, intermediate and open sea regions are approximately 34938, 34245 and 154548 km^2. The average distances from the coastline to the limit of the coastal and intermediate region are respectively 38 and 76 km.

Zone	Average degradation $\omega_t^{hr} = 12$ h	$\omega_t^{hr} = 24$ h
Coastal	−0.69 %	−1.61 %
Near-sea	−0.66 %	−1.48 %
Open Sea	−0.52 %	−1.38 %

Table 4.4 – Average degradation values of the case of buoys group removal.

measurements. We use the best model of the case described in Section 4.4.2, considering both the sampling frequencies of 12 and 24 hours for the HR spatial fields. The choice of performing the test with both the sampling frequencies is motivated by the necessity to assess the impact of in-situ observations related to two levels of HR spatial information availability. We set up two test cases. In the first case, we test the model removing one single buoy to see whether some buoys are more relevant for the reconstruction task. This test is repeated for each of the 13 buoys of the observation network. In the second case, we identify three zones on the region on interest, as illustrated in Figure 4.9. A first zone is the strict proximity of the coastline, the second is an intermediate buffer zone between the coast and the open sea and the third zone is the open sea, from approximately 76 kilometers from the coastline. The objective of this second test case is to check whether one group of buoys provides more meaningful measurements for the task of wind speed reconstruction on a mixed land-sea region.

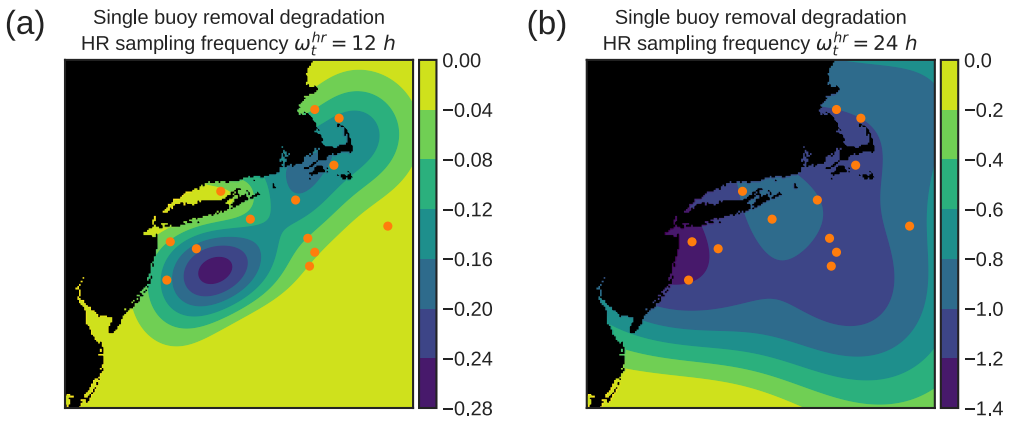


Figure 4.10 – Single buoys removal tests. Degradation maps obtained interpolating spatially the buoy-wise degradation values. The interpolation is made with a Gaussian process regression. Panel (a): HR temporal sampling frequency of 12 hours. Panel (b): HR temporal sampling frequency 24 hours.

We start the analysis with the results related to the case of single buoy removal, cfr. Section 4.4.4. In order to visualize the degradation trend as a function of the single buoys, we produce a fictitious map as follows. The starting point is a blank matrix having the same dimensions as the input spatial data. We can identify on this map the buoys locations, as shown in Figure 4.9. On the point associated with the buoy n, with $n = 1, \ldots, 13$, we set the value of degradation due to the removal of the buoy n. In this way we obtain a group of discrete values of degradation. These values are interpolated spatially with a Gaussian process [245]. The resulting map can be interpreted as the individual buoy responsibility to degradation. Panel (a) of Figure 4.10 displays the degradation map for the case of HR spatial fields temporal frequency of 12 hours, i.e. a HR field at hours 06 and 18. The degradation is mainly concentrated on a spatial belt encompassing the intermediate and coastal regions. This means that the most relevant buoys for the reconstruction task are those located in the strict proximity of the coast. The degradation map changes in the case of HR fields temporal frequency of 24 hours, see Panel (b) of Figure 4.10. In-situ observations are more that 1 % more relevant as the HR observation regime is halved. In other words, the fine-scaled information provided by in-situ data is exploited more extensively by the model. In addition, the importance of the in-situ data is more distributed among all the buoys and there are not particularly preferred regions. We discuss now the results of the buoys groups removal case. Again, the average degradation values are larger in the case of HR sampling frequency of 24 hours. The reason for this

	Spatial resolution	Temporal resolution
Group A	30 km	6 h
Group B	30 km	1 h
Group C	100 km	6 h
Group D	100 km	1 h

Table 4.5 – Combinations of the HR and LR fields spatio-temporal resolutions.

is again the relevance of HR information of in-situ data. The difference w.r.t. the single buoy removal tests is the constant tendency of degradation, which attains larger values for the coastal region buoys and progressively decreases in the open sea direction. Table 4.4 reports values of error metrics, which may be explained as follows. The coastal buoys are placed on the area that experiences the most *shelter effects*, involving strong wind speed gradients associated with the presence of the morphology of the coast [246], [247]. These results mean that the coastline proximity is the most sensitive region for the wind speed reconstruction task. The largest impact of in-situ observations is associated to this particular zone. Appendix A presents the same analysis performed on a synthetic buoys network. The purpose of this analysis is to study whether an arbitrary constellation may be more suited for the task of large extent wind speed reconstruction.

4.4.5 Spatial fields resolution sensitivity analysis

To conclude this work, we propose an aggregated analysis on the impact of each data source on the reconstruction performance of the M_m model. The tests involve the comparison between the C1 and C3 configurations (in-situ time series), the LR data divided in four groups as in Table 4.5. The group A corresponds to the LR configuration used in the test cases detailed above. Using the best benchmark model, for each group we test both the C1 and C3 configurations as described above and both the 12 hours and 24 hours temporal resolutions for HR spatial fields. These additional tests allow to better appreciate the impact of the temporal and spatial resolution of spatial fields, both HR and LR.

Figure 4.11 visualizes the results of the performance of M_m in all the configurations mentioned above. There are two evident remarks. The first one is that the difference between C3 and C1 is larger for the configurations where LR fields have temporal resolution of 6 hours. This can be explained as follows: if LR data have temporal resolution of 1

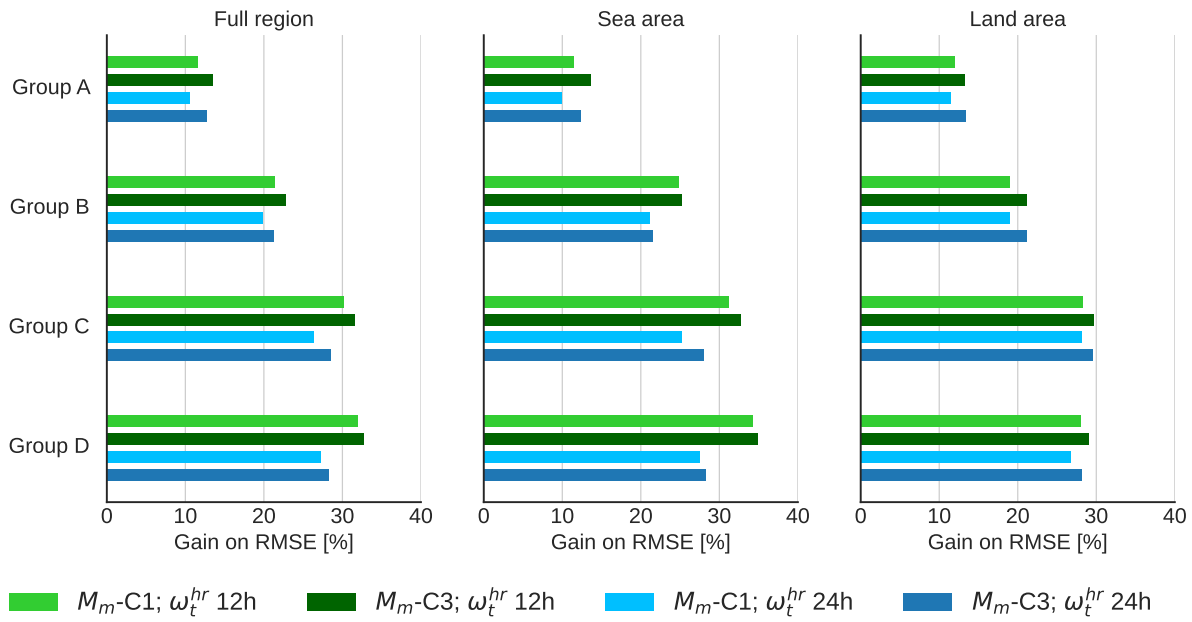


Figure 4.11 – Sensitivity analysis results. The labels on the left identify the groups of LR spatio-temporal resolutions. The performance metrics illustrated are relative gains on RMSE referred to the B_1-SR baseline applied to each LR combination. The symbol ω_t^{hr} refers to the high-resolution spatial fields sampling frequency.

hour, the dynamical information of in-situ time series is partially hidden by the larger temporal availability of LR fields. Again, this result supports the importance of in-situ observations as source of time-dependent information. The second remark concerns the spatial resolution of LR data. For the higher resolution of 100 km, the fine-scale information delivered by both the spatial fields and the in-situ data are more largely exploited by the multi-modal trainable observation operator of the 4DVarNet.

Figure 4.12 summarizes the visualizations of Figure 4.11. We compare: (i) the difference between the relative gain brought by the in-situ time series (C1 against C3, the solid and dashed dark gray curves) and (ii) the difference between the relative gains brought by the spatial HR fields sampled at 12 hours against 24 hours (solid and dashed light gray curves). Referring to Figure 4.11, the dark gray solid line points represent the differences between dark and light green marked results, the dark gray dashed line points represent the differences between dark and light blue marked results. In the same way, the light gray solid point refer to the difference between the performances marked by dark green and blue bars and the points of the dashed light gray line refer to the differences between performances marked by light green and blue bars. For example, consider the point on

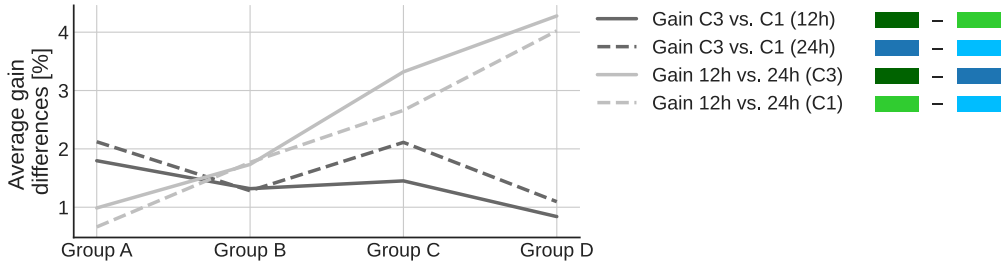


Figure 4.12 – Analysis on the gains referred to the contributions of either the in-situ time series and a finer temporal resolution of high-resolution wind speed fields. The gains are referred to the $B1$-SR baseline associated to the low-resolution spatio-temporal resolutions of each group. The groups name convention follows the definitions given in Section 4.4.5.

the light gray curve in correspondence of the low-resolution combination group C. There, one may consider both the performance gain (w.r.t. the baseline of group C) of M_m-C3 and M_m-C1. The point of the light gray curve for the group C represents the difference between these two gains. Stated otherwise, the performance surplus imputable to the configuration C3 against the performance of configuration C1, which is the value added by in-situ time series.

The reader is referred to Appendix A for the results on a sensitivity test in which we choose a different temporal configuration for the HR fields. More precisely, we study the effect of the HR fields placed on different time steps.

4.5 Conclusions

In this work we presented an observation system simulation experiment based on an hybrid data assimilation and deep learning framework. The objective of our analyses was to evaluate the impact and value of different input sources of information on sea-surface wind speed. Our investigation has both scientific and operational relevance. On the scientific side, our analyses prove the potential of a trainable approach for the simultaneous exploitation of diverse oceanic observations. This approach allows the model to ingest spatio-temporally heterogeneous data and to use this complementary information to reconstruct time series of spatial wind speed fields. The reconstruction performance associated with this explicit multi-modal strategy for the 4DVarNet (M_m) proves superior to the performance of the same 4DVarNet model not implementing the trainable multi-modal approach (M_s). We showed that this framework brings the model to auto-

matically learn the correction of possibly biased input data, thanks to the high-level data representation encouraged by the multi-modal induced features extraction. On the operational side, beside proving the added value of in-situ observations, our results on the buoys sensitivity analyses may open the road to future work devoted to the predictive learning-based optimization of in-situ sensors installation. The results presented in this work can not provide a direct answer to this problem but are an encouraging starting point to further investigate this aspect. To conclude, we may recall that the state-of-the-art of super-resolution modelling is based on generative AI methods. Further work may be devoted to the inclusion of these models in our data assimilation-based framework. This would allow to improve the reconstructed fields appearance. The output time series would benefit from a photo-realistic aspect and a stronger spatial gradients consistence.

Part III

General conclusion

Conclusion and future perspectives

Conclusions

This thesis aimed to investigate the use of multi-modal deep learning-based methods to treat oceanic and atmospheric data. The compartment that we considered for our analyses was the sea-air interface, focusing in particular on wind speed. The introduction of the thesis started with an overview on the conceptual and practical limitations that prevent a full and complete characterization and description of sea-surface wind speed. The introductory matter also provided a (non-exhaustive) list of techniques for the measurement and reconstruction of wind speed. The methodological aspects, regarding data assimilation and deep learning, are introduced in depth and precede the formal statement of the 4DVarNet framework. This model has been extensively used in our analyses for its appealing relation with classical variational data assimilation schemes. These schemes have the desirable property of modelling explicitly the state-space dynamics of a given phenomenon, in order to state formally both the physical and the observation processes.

We investigated two case studies. The relevant points of each contribution are here stated in relationship with the questions that were posed at the beginning of this thesis.

- In the first contribution, we performed a spatially local analysis. The main objective was to test systematically regressive models to estimate the wind speed at the sea surface using underwater passive acoustics data. An ancillary objective was to assess the reconstruction performance improvement imputable to a multi-modal approach (cfr. Section 1.5). Our analyses show that the 4DVarNet improves the best result of a purely data-driven method of 4 %. The same analyses performed with a multi-modal dataset bring a further 5 % improvement.
- The second contribution treats the problem of wind speed high-resolution reconstruction from its partial (pseudo-)observations. Due to the spatio-temporal scale information of sea-surface wind speed data, the approach used for this case study has to account necessarily for the dataset multi-modality and heterogeneity. Our

analyses prove that this multi-modality is most effectively exploited with a learning-based approach. Stated otherwise, heterogeneous data "interact" more effectively if related at a higher abstraction level. This is accomplished by the extraction of learned data feature maps, that transcend the spatio-temporal domain of raw data. The use of 4DVarNet in this case brings an average improvement of 6 % with respect to the baseline and the learning-based multi-modal approach is responsible of a further 3 % of reconstruction performance improvement.

These two case studies clearly show that there are two axes of improvement, in some measure independent. The first is imputable to modelling choices (i.e. 4DVarNet framework) and the second to the multi-modal nature of the dataset. The most encouraging result is that the improvements of these two aspects are cumulative. A clear example is provided by the first contribution, where the multi-modal approach is optional and we can distinguish the two independent improvements.

One additional remark about the multi-modal approach relates to the gain experienced by the deep learning-based direct inversion and the 4DVarNet inversion schemes detailed in the two contributions. Figure 4.13 resumes systematically the results presented in the contributions chapters. The right part of Panel (a) shows that the direct inversion model benefits more of the multi-modal dataset than the 4DVarNet model. This fact may be interpreted as follows. The direct inversion is an end-to-end trainable model that is optimized to relate the observations and the true system state. Given the observations, the model is trained to directly return the state variable. On the other hand, the plain 4DVarNet scheme is based on two optimization procedures: a first for the state variable and a second for the neural network-based operators parameters. The state variable is found by optimizing the variational cost. The reader is referred to Sections 2.3.4 and 2.4.2 for the formal statements of the direct inversion and 4DVarNet-based inversion schemes. The variational cost function is composed of two quadratic distance terms. The first between the observations and the state variables and the second is a regularization term involving the prior knowledge on the dynamical operator. In this variational cost there are no explicit constraints about the observations modalities. The 4DVarNet, as presented in Section 2.4.2 and as implemented in the first case study, is not informed by the data diversity. For this reason, it does not make the most from the multi-modal observations.

The second case study, as depicted by Panel (b) of Figure 4.13, presents better results. As a reminder, the models B_1, M_s and M_m represent the direct inversion, the plain 4DVarNet and the multi-modal 4DVarNet versions. The model M_m had an additional

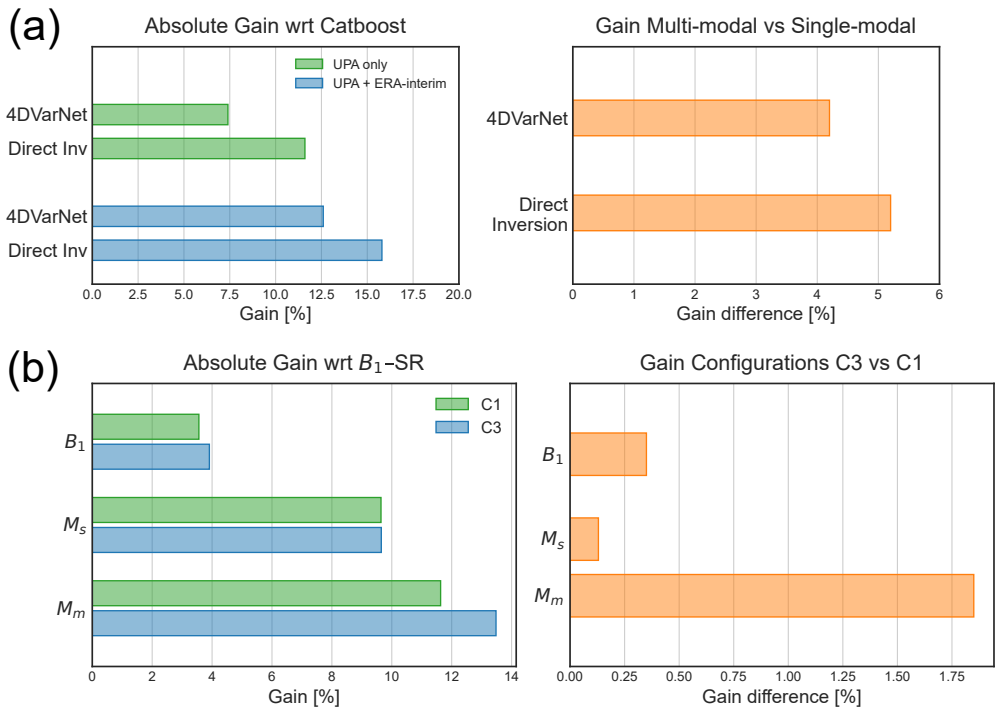


Figure 4.13 – Graphical summary of the results of the two case studies presented as contributions. Panel (a): First case study, Chapter 3. The left part reports the absolute gains of each model used w.r.t. the CatBoost baseline. The right part reports the gains imputable to the multi-modal dataset, computed as differences between the "UPA+ECMWF" and "UPA only" instances. Panel (b): Second case study, Chapter 4. The left part reports the absolute performances of the deep learning-based inversion (B_1), single and multi-modal versions of the 4DVarNet schemes (M_s and M_m respectively) w.r.t. the B_1-SR super-resolution baseline.

term in the variational cost. Section 4.3 discussed in depth the rational behind this modified variational cost. Briefly, the additional term enforced the similarity between the *features maps* of observations and state variable in order to relate these objects with no constraints posed by the heterogeneous spatio-temporal characteristics. The right part of Panel (b) is consistent with the results of the first case study. The plain 4DVarNet, whose variational cost does not contain any information about the data heterogeneity, can not fully exploit the complementary information in the input dataset and the direct inversion baseline still benefits more from the multi-modality. However, the results are different in the case of the multi-modal 4DVarNet M_m. The right part Panel (b) of Figure 4.13 reports the gain differences between the performance levels of the C3 and C1 data configurations. We remind briefly that the configuration C1 has low-resolution fields and

high-resolution pseudo-observations and the configuration C3 has low and high-resolution pseudo-observations and in-situ time series. In this sense, the difference between C3 and C1 can be seen as the added value of in-situ time series. The M_m inversion is based on the optimization of a variational cost that accounts explicitly for the data heterogeneity thanks to the additional features maps term. The better performance of the case M_m w.r.t. the case M_s is imputable to the model skill to use the time-series information and this skill is due to the better model design choice of the variational cost.

To take up explicitly the questions posed at the beginning of this manuscript, we are in the position of giving a positive answer to both. We can state that an hybrid data-driven and physics-informed scheme constitutes a promising solution, especially when the target problem is a time process where the temporal dynamics plays a central role. We may recall that the 4DVarNet is based on classical variational data assimilation schemes, which explicitly account for temporal dependence. This feature is crucial in geophysical applications, where the characterization of the phenomenon dynamics is as important as the data-to-model inversion itself. These conceptual foundations allow us to make the most of temporal data and to propose a methodology that benefits as much from the physical knowledge as from fully data-driven approaches. Moreover, our results shed light on the potential of data multi-modality. The spatio-temporal heterogeneity can be interlaced and be an added value again due to proper modelling choices. In this sense, we think that deep learning-based methodologies may be a relevant mean to this end, thanks to their autonomous feature extraction capabilities.

Future work

This thesis work represents a contribution for the specific case of wind speed at the sea-surface estimation and reconstruction. The results presented and discussed raise some further related scientific questions. We close this thesis listing some of the possible future research directions.

Using multiple physical variables

Recent work focused on multi-modal approaches for geophysical problems [69]. Some examples include multi-modal data fusion for hurricane forecasting using reanalyses and numerical tabular data [248], Automatic Identification Systems and Sea Surface Height ob-

servations to reconstruct sea-surface currents [249], satellite and seasonal and geographical information for weather event classification [250]. Our analyses focused and were mainly limited to wind speed at the sea-air interface. The global ocean and atmospheric circulations, however, involve a larger set of environmental variables that play a crucial role. For example the sea-surface temperature and height, salinity, atmospheric pressure, and rainfall to cite a few. The complete and accurate description of oceanic and atmospheric phenomena involves the description of all these processes and their interaction. A more developed and mature multi-modal approach would aim to couple these interactions. Based on the evidence seen in Chapter 4, we think that the extraction of high-level feature maps from different variables could help to better relate different physical processes. This would improve the representation of the common underlying driving phenomena.

Towards the extension to forecasting problems

The case studies treated in this thesis focused on time series reconstruction. A relevant application in geosciences is forecasting, that is predicting the phenomenon with time advance. Deep learning modelling has been successfully applied to time series analysis [251]. This application has inspired much work devoted to weather forecast, thanks to the ever growing availability of observation data sets [252]–[254]. Previous work compared systematically the conventional numerical weather prediction methods and learning-based end-to-end strategies in weather prediction [70]. The authors of this research argue that deep learning models for weather forecasting applications perform well in a 24 hours time range since the last observations are available. We discussed the appealing capabilities of deep learning modelling in Chapters 1 and 2. Our analyses and previous work (cfr. Section 2.4 and references therein) highlighted that the model-based and data-driven approaches coupling is promising. We think that the application of models of the type of 4DVarNet could play a role in bridging the numerical weather prediction state-of-the-art prediction capabilities and the deep learning strength points.

Investing the 4DVarNet to optimize in-situ observation networks

In-situ observations, together with remote sensing products, are essential information to perform data assimilation and deep learning-based analyses. The installation of in-situ infrastructures is a costly process. This makes the realization of large-scale and dense buoys networks infeasible. Recent studies focused on the study of optimal buoys layouts

and placement [255]–[257]. The second case study mentioned an operational outcome related to the position of the in-situ sensors. The analyses were limited to the assessment of the most relevant region to observe with in-situ infrastructures. Inspired by the cited previous work, these results may be further extended using the proposed OSSE-based analysis to determine the best possible buoys layout to maximize the income of useful information. The most valuable outcome of such analysis would be the optimization of the infrastructures installation process, in terms of both effort and cost.

Towards deployment on real data

The first case study treated real data. The W1M3A dataset is composed of time series of acoustic, wind speed and rainfall observations. The case in which real data associated with ground truths for long time windows are available is not ordinary. Indeed, the second case study fully relied on simulated data. Using synthetic data to train deep learning model is a commonly used practice in research applications [258]–[260]. However, for an operational deployment and implementation, it is preferable for the model to be tested on real data. Transfer learning [261], [262] can be used to fine-tune the model trained on a synthetic dataset [69], if only a small real dataset is available. However, if the synthetic data set can capture the most important features of real data, the model trained on such dataset could be deployed on real data with no transfer [71]. As demonstrated by the authors of [230], machine learning models trained on simulation dataset may lead to a good performance on real datasets, for example sea-surface altimetry data. As encouraging as our results for the second case study are, a solid framework should be tested on real data sets in order to prove its robustness and applicability in real-world scenarios.

Managing quantitatively the uncertainties

The case studies proposed in this thesis focused on the application of the 4DVarNet scheme to reconstruct the *most likely* system state. This kind of analysis stems from a maximum-a-posteriori estimation, where one is concerned with the reconstruction of a representative state. Recent works focused on the uncertainty quantification associated with the mean state. For example, the authors of [263] adapted the 4DVarNet scheme to perform variational Bayes inference [264] in order to estimate the posterior on the system state given the observations. This method allows to interpret the variables involved as random quantities, with which the uncertainty can be quantatively associated. Likewise,

the authors of [265] proposed an hybrid framework that leverages both the 4DVarNet scheme and stochastic partial differential equations [266] to address jointly the interpolation and uncertainty quantification issues. An interesting further research direction could be the application of these concepts to the multi-modal sea-surface state reconstruction, with particular attention to the reconstruction of the state probability distribution.

BIBLIOGRAPHY

[1] M. Visbeck, « Ocean science research is key for a sustainable future », *Nature Communications*, vol. 9, *1*, p. 690, Feb. 2018. DOI: 10.1038/s41467-018-03158-3. [Online]. Available: https://doi.org/10.1038/s41467-018-03158-3.

[2] E. Commission, D.-G. for Maritime Affairs, and Fisheries, *Summary of the results of the targeted consultation on international ocean governance*. Publications Office, 2021. DOI: doi/10.2771/673816.

[3] R. Cole, J. Kinder, W. Yu, C. L. Ning, F. Wang, and Y. Chao, « Ocean climate monitoring », *Frontiers in Marine Science*, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00503. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00503.

[4] B. deYoung, M. Visbeck, M. C. de Araujo Filho, *et al.*, « An integrated all-atlantic ocean observing system in 2030 », *Frontiers in Marine Science*, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00428. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00428.

[5] Y. Liu, M. Qiu, C. Liu, and Z. Guo, « Big data challenges in ocean observation: a survey », *Personal and Ubiquitous Computing*, vol. 21, *1*, pp. 55–65, Feb. 2017, ISSN: 1617-4917. DOI: 10.1007/s00779-016-0980-2. [Online]. Available: https://doi.org/10.1007/s00779-016-0980-2.

[6] A. Karpatne, I. Ebert-Uphoff, S. Ravela, H. A. Babaie, and V. Kumar, « Machine learning for the geosciences: challenges and opportunities », *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, *8*, pp. 1544–1554, 2018.

[7] K. J. Bergen, P. A. Johnson, M. V. de Hoop, and G. C. Beroza, « Machine learning for data-driven discovery in solid earth geoscience », *Science*, vol. 363, *6433*, 2019.

[8] M. Sonnewald, R. Lguensat, D. C. Jones, P. D. Dueben, J. Brajard, and V. Balaji, « Bridging observations, theory and numerical simulation of the ocean using machine learning », *Environmental Research Letters*, vol. 16, *7*, p. 073 008, Jul. 2021. DOI: 10.1088/1748-9326/ac0eb0. [Online]. Available: https://dx.doi.org/10.1088/1748-9326/ac0eb0.

[9] R. Ferrari and D. Ferreira, « What processes drive the ocean heat transport? », *Ocean Modelling*, vol. 38, *3-4*, pp. 171–186, 2011.

[10] E. Y. Kwon, F. Primeau, and J. L. Sarmiento, « The impact of remineralization depth on the air–sea carbon balance », *Nature Geoscience*, vol. 2, *9*, pp. 630–635, 2009.

[11] A. L. Gordon, « The marine hydrological cycle: the ocean's floods and droughts », *Geophysical Research Letters*, vol. 43, *14*, pp. 7649–7652, 2016.

[12] J. M. Wallace and P. V. Hobbs, *Atmospheric science: an introductory survey*. Elsevier, 2006, vol. 92.

[13] G. K. Vallis, *Atmospheric and oceanic fluid dynamics*. Cambridge University Press, 2017.

[14] G. K. Batchelor, *The theory of homogeneous turbulence*. Cambridge university press, 1953.

[15] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry* (Cambridge Monographs on Mechanics). Cambridge University Press, 1996. DOI: 10.1017/CBO9780511622700.

[16] G. Galilei, *Il saggiatore*, 1623.

[17] A. Fowler *et al.*, *Mathematical geoscience*. Springer, 2011, vol. 36.

[18] G. Balsamo, A. Agusti-Parareda, C. Albergel, *et al.*, « Satellite and in situ observations for advancing global earth surface modelling: a review », *Remote Sensing*, vol. 10, *12*, p. 2038, 2018.

[19] J. Gould, B. Sloyan, and M. Visbeck, « In situ ocean observations: a brief history, present status, and future directions », *International Geophysics*, vol. 103, pp. 59–81, 2013.

[20] T. Dickey, M. Lewis, and G. Chang, « Optical oceanography: recent advances and future directions using global remote sensing and in situ observations », *Reviews of geophysics*, vol. 44, *1*, 2006.

[21] C. E. Synolakis, E. N. Bernard, V. V. Titov, U. Kânoğlu, and F. I. Gonzalez, « Validation and verification of tsunami numerical models », *Tsunami Science Four Years after the 2004 Indian Ocean Tsunami: Part I: Modelling and Hazard Assessment*, pp. 2197–2228, 2009.

[22] R. N. Hoffman and R. Atlas, « Future observing system simulation experiments », *Bulletin of the American Meteorological Society*, vol. 97, *9*, pp. 1601–1616, 2016.

[23] A. Gettelman, A. J. Geer, R. M. Forbes, *et al.*, « The future of earth system prediction: advances in model-data fusion », *Science Advances*, vol. 8, *14*, eabn3488, 2022.

[24] M. Asch, M. Bocquet, and M. Nodet, *Data assimilation: methods, algorithms, and applications*. Dec. 2016, ISBN: 978-1-611974-53-9.

[25] Z. Sun, L. Sandoval, R. Crystal-Ornelas, *et al.*, « A review of earth artificial intelligence », *Computers & Geosciences*, vol. 159, p. 105 034, 2022.

[26] J. S. Dramsch, « 70 years of machine learning in geoscience in review », *Advances in geophysics*, vol. 61, pp. 1–55, 2020.

[27] Campbell-Inc, *Windsonic1 and windsonic4 two-dimensional sonic anemometers*, 2016.

[28] P. Potisomporn, T. A. Adcock, and C. R. Vogel, « Evaluating era5 reanalysis predictions of low wind speed events around the uk », *Energy Reports*, vol. 10, pp. 4781–4790, 2023.

[29] F. M. Monaldo, C. R. Jackson, and W. G. Pichel, « Seasat to radarsat-2: research to operations », *Oceanography*, vol. 26, *2*, pp. 34–45, 2013.

[30] S. Pindado, J. Cubas, and F. Sorribes-Palmer, « The cup anemometer, a fundamental meteorological instrument for the wind energy industry. research at the idr/upm institute », *Sensors*, vol. 14, *11*, pp. 21 418–21 452, 2014.

[31] F. Motallebi, « A review of the hot-wire technique in 2-d compressible flows », *Progress in aerospace sciences*, vol. 30, *3*, pp. 267–294, 1994.

[32] E. Canepa, S. Pensieri, R. Bozzano, M. Faimali, P. Traverso, and L. Cavaleri, « The odas italia 1 buoy: more than forty years of activity in the ligurian sea », *Progress in Oceanography*, vol. 135, Apr. 2015. DOI: 10.1016/j.pocean.2015.04.005.

[33] K. Nittis, T. C, B. R, *et al.*, « The m3a multi-sensor buoy network of the mediterranean sea », *Ocean Science*, vol. 3, p. 243, May 2007. DOI: 10.5194/osd-3-1399-2006.

[34] R. Venkatesan, K. Ramesh, A. Kishor, N. Vedachalam, and M. A. Atmanand, « Best practices for the ocean moored observatories », *Frontiers in Marine Science*, vol. 5, 2018, ISSN: 2296-7745. DOI: 10.3389/fmars.2018.00469. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2018.00469.

[35] A. Meindl, « Guide to moored buoys and other ocean data acquisition systems. », 1996.

[36] L. R. Centurioni, J. Turton, R. Lumpkin, *et al.*, « Global in situ observations of essential climate and ocean variables at the air–sea interface », *Frontiers in Marine Science*, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00419. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00419.

[37] B. I. Moat and M. J. Yelland, « Going with the flow: state of the art marine meteorological measurements on the new nerc research vessel », *Weather*, vol. 63, *6*, pp. 158–159, 2008.

[38] V. Kumar, S. Khalap, and P. Mehra, « Instrumentation for high-frequency meteorological observations from research vessel », *in OCEANS'11 MTS/IEEE KONA*, IEEE, 2011, pp. 1–10.

[39] P. P. Niiler and J. D. Paduan, « Wind-driven motions in the northeast pacific as measured by lagrangian drifters », *Journal of Physical Oceanography*, vol. 25, *11*, pp. 2819–2830, 1995.

[40] R. Lumpkin, T. Özgökmen, and L. Centurioni, « Advances in the application of surface drifters », *Annual review of marine science*, vol. 9, pp. 59–81, 2017.

[41] J. A. Nystuen, S. E. Moore, and P. J. Stabeno, « A sound budget for the southeastern bering sea: measuring wind, rainfall, shipping, and other sources of underwater sound », *The Journal of the Acoustical Society of America*, vol. 128, *1*, pp. 58–65, 2010. DOI: 10.1121/1.3436547.

[42] J. A. Nystuen, « Rainfall measurements using underwater ambient noise », *The Journal of the Acoustical Society of America*, vol. 79, *4*, pp. 972–982, 1986. DOI: 10.1121/1.393695.

[43] B. B. Ma and J. A. Nystuen, « Detection of rainfall events using underwater passive aquatic sensors and air–sea temperature changes in the tropical pacific ocean », *Monthly Weather Review*, vol. 135, *10*, pp. 3599–3612, 2007. DOI: 10.1175/MWR3487.1.

[44] J. Yang, S. C. Riser, J. A. Nystuen, W. E. Asher, and A. T. Jessup, « Regional rainfall measurements using the passive aquatic listener during the spurs field campaign », *Oceanography*, Mar. 2015.

[45] Y. Jie, W. E. Asher, and S. C. Riser, « Rainfall measurements in the north atlantic ocean using underwater ambient sound », *in 2016 IEEE/OES China Ocean Acoustics (COA)*, 2016, pp. 1–4. DOI: 10.1109/COA.2016.7535834.

[46] D. Cazau, J. Bonnel, J. Jouma'a, Y. le Bras, and C. Guinet, « Measuring the marine soundscape of the indian ocean with southern elephant seals used as acoustic gliders of opportunity », *Journal of Atmospheric and Oceanic Technology*, vol. 34, *1*, 2017. DOI: 10.1175/JTECH-D-16-0124.1.

[47] A. Fischer, « Remote sensing vs. in situ measurements: Two poles of the same planet? », *in EGU General Assembly Conference Abstracts*, ser. EGU General Assembly Conference Abstracts, Apr. 2013, EGU2013-3220, EGU2013–3220.

[48] M. Amani, A. Ghorbanian, M. Asgarimehr, *et al.*, « Remote sensing systems for ocean: a review (part 1: passive systems) », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 210–234, 2021.

[49] M. Amani, F. Mohseni, N. F. Layegh, *et al.*, « Remote sensing systems for ocean: a review (part 2: active systems) », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1421–1453, 2022.

[50] F. Naderi, M. Freilich, and D. Long, « Spaceborne radar measurement of wind velocity over the ocean-an overview of the nscat scatterometer system », *Proceedings of the IEEE*, vol. 79, *6*, pp. 850–866, 1991. DOI: 10.1109/5.90163.

[51] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, « A tutorial on synthetic aperture radar », *IEEE Geoscience and remote sensing magazine*, vol. 1, *1*, pp. 6–43, 2013.

[52] M. Portabella, A. Stoffelen, and J. A. Johannessen, « Toward an optimal inversion method for synthetic aperture radar wind retrieval », *Journal of Geophysical Research: Oceans*, vol. 107, *C8*, pp. 1–1, 2002.

[53] Y. Lu, B. Zhang, W. Perrie, A. A. Mouche, X. Li, and H. Wang, « A c-band geophysical model function for determining coastal wind speed using synthetic aperture radar », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, *7*, pp. 2417–2428, 2018.

[54] M. A. Bourassa, T. Meissner, I. Cerovecki, *et al.*, « Remotely sensed winds and wind stresses for marine forecasting and ocean modeling », *Frontiers in Marine Science*, vol. 6, p. 443, 2019.

[55] R. Torres, P. Snoeij, D. Geudtner, *et al.*, « Gmes sentinel-1 mission », *Remote sensing of environment*, vol. 120, pp. 9–24, 2012.

[56] P. Snoeij, E. Attema, M. Davidson, *et al.*, « The sentinel-1 radar mission: status and performance », *in 2009 International Radar Conference" Surveillance for a Safer World"(RADAR 2009)*, IEEE, 2009, pp. 1–6.

[57] A. Quarteroni and A. Valli, *Numerical approximation of partial differential equations*. Springer Science & Business Media, 2008, vol. 23.

[58] J. G. Powers, J. B. Klemp, W. C. Skamarock, *et al.*, « The weather research and forecasting model: overview, system efforts, and future directions », *Bulletin of the American Meteorological Society*, vol. 98, *8*, pp. 1717–1737, 2017.

[59] E. C. of Medium-Range Weather Forecast (ECMWF), *IFS Documentation CY48r1* (IFS Documentation). ECMWF, 2023. DOI: 10.21957/0f360ba4ca.

[60] N. C. for Environmental Prediction (NCEP), « Global forecast system », 2023. [Online]. Available: https://www.ncei.noaa.gov/products/weather-climate-models/global-forecast.

[61] A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen, « Data assimilation in the geosciences: An overview of methods, issues, and perspectives », *Wiley Interdisciplinary Reviews: Climate Change*, vol. 9, *5*, e535, Sep. 2018. DOI: 10.1002/wcc.535. [Online]. Available: https://hal.science/hal-02905891.

[62] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*. Berlin, Heidelberg: Springer-Verlag, 2006, ISBN: 354038300X.

[63] R. N. Bannister, « A review of operational methods of variational and ensemble-variational data assimilation », *Quarterly Journal of the Royal Meteorological Society*, vol. 143, *703*, pp. 607–633, 2017. DOI: https://doi.org/10.1002/qj.2982. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2982. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2982.

[64] H. Hersbach, B. Bell, P. Berrisford, *et al.*, « The era5 global reanalysis », *Quarterly Journal of the Royal Meteorological Society*, vol. 146, *730*, pp. 1999–2049, 2020. DOI: https://doi.org/10.1002/qj.3803. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803.

[65] E. Kalnay, M. Kanamitsu, R. Kistler, *et al.*, « The ncep/ncar 40-year reanalysis project », *in Renewable Energy*, Routledge, 2018, Vol1_146–Vol1_194.

[66] ECMWF, « Press release 10 march 2016 », EMCWF, Tech. Rep., 2016, Visited on 25 September 2023. [Online]. Available: https://www.ecmwf.int/sites/default/files/ECMWF_41r2_PressRelease.pdf.

[67] A. Storto, A. Alvera-Azcárate, M. A. Balmaseda, *et al.*, « Ocean reanalyses: recent advances and unsolved challenges », *Frontiers in Marine Science*, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00418. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00418.

[68] S. S., R. M., L. M. P., *et al.*, « Cerra sub-daily regional reanalysis data for europe on single levels from 1984 to present », Visited on 09 September 2023. DOI: 10.24381/cds.622a565a. [Online]. Available: https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-cerra-single-levels?tab=overview.

[69] S. Yu and J. Ma, « Deep learning for geophysics: current and future trends », *Reviews of Geophysics*, vol. 59, *3*, e2021RG000742, 2021.

[70] M. G. Schultz, C. Betancourt, B. Gong, *et al.*, « Can deep learning beat numerical weather prediction? », *Philosophical Transactions of the Royal Society A*, vol. 379, *2194*, p. 20 200 097, 2021.

[71] Q. Yuan, H. Shen, T. Li, *et al.*, « Deep learning in environmental remote sensing: achievements and challenges », *Remote Sensing of Environment*, vol. 241, p. 111 716, 2020.

[72] K. Stengel, A. Glaws, D. Hettinger, and R. N. King, « Adversarial super-resolution of climatological wind and solar data », *Proceedings of the National Academy of Sciences*, vol. 117, *29*, pp. 16 805–16 815, 2020.

[73] M. Yasir, W. Jianhua, L. Shanwei, H. Sheng, X. Mingming, and M. Hossain, « Coupling of deep learning and remote sensing: a comprehensive systematic literature review », *International Journal of Remote Sensing*, vol. 44, *1*, pp. 157–193, 2023.

[74] S. M. i Verdú, M. Chabert, T. Oberlin, and J. Serra-Sagristà, « Reduced-complexity multi-rate remote sensing data compression with neural networks », *IEEE Geoscience and Remote Sensing Letters*, 2023.

[75] V. Alves de Oliveira, M. Chabert, T. Oberlin, *et al.*, « Reduced-complexity end-to-end variational autoencoder for on board satellite image compression », *Remote Sensing*, vol. 13, *3*, p. 447, 2021.

[76] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, « Deep learning techniques for inverse problems in imaging », *IEEE Journal on Selected Areas in Information Theory*, vol. 1, *1*, pp. 39–56, 2020.

[77] S. Barthélémy, J. Brajard, L. Bertino, and F. Counillon, « Super-resolution data assimilation », *Ocean Dynamics*, vol. 72, *8*, pp. 661–678, 2022.

[78] C. Buizza, C. Q. Casas, P. Nadler, *et al.*, « Data learning: integrating data assimilation and machine learning », *Journal of Computational Science*, vol. 58, p. 101 525, 2022.

[79] A. Farchi, M. Bocquet, P. Laloyaux, M. Bonavita, M. Chrust, and Q. Malartic, « Model error correction with data assimilation and machine learning », *in EGU General Assembly Conference Abstracts*, 2022, EGU22–5692.

[80] R. Arcucci, J. Zhu, S. Hu, and Y.-K. Guo, « Deep data assimilation: integrating deep learning with data assimilation », *Applied Sciences*, vol. 11, *3*, p. 1114, 2021.

[81] W. Castaings, D. Dartus, M. Honnorat, F.-X. L. Dimet, Y. Loukili, and J. Monnier, « Automatic differentiation: a tool for variational data assimilation and adjoint sensitivity analysis for flood modeling », *in Automatic Differentiation: Applications, Theory, and Implementations*, Springer, 2006, pp. 249–262.

[82] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, « Image inpainting: a review », *Neural Processing Letters*, vol. 51, pp. 2007–2028, 2020.

[83] R. Wong, Z. Zhang, Y. Wang, F. Chen, and D. Zeng, « Hsi-ipnet: hyperspectral imagery inpainting by deep learning with adaptive spectral extraction », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4369–4380, 2020.

[84] M. Czerkawski, P. Upadhyay, C. Davison, *et al.*, « Deep internal learning for inpainting of cloud-affected regions in satellite imagery », *Remote Sensing*, vol. 14, *6*, p. 1342, 2022.

[85] Q. Zheng, L. Zeng, and G. E. Karniadakis, « Physics-informed semantic inpainting: application to geostatistical modeling », *Journal of Computational Physics*, vol. 419, p. 109 676, 2020.

[86] M. P. Foster and A. N. Evans, « Performance evaluation of multivariate interpolation methods for scattered data in geoscience applications », *in IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium*, IEEE, vol. 4, 2008, pp. IV–565.

[87] C. Kirkwood, T. Economou, N. Pugeault, and H. Odbert, « Bayesian deep learning for spatial interpolation in the presence of auxiliary information », *Mathematical Geosciences*, vol. 54, *3*, pp. 507–531, 2022.

[88] T. Bai and P. Tahmasebi, « Accelerating geostatistical modeling using geostatistics-informed machine learning », *Computers & Geosciences*, vol. 146, p. 104 663, 2021.

[89] L. Liu and J. Ma, « Dl2: dictionary learning regularized with deep learning prior for simultaneous denoising and interpolation », *Geophysics*, vol. 88, *1*, WA13–WA25, 2023.

[90] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, « Multimodal machine learning: a survey and taxonomy », *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, *2*, pp. 423–443, 2018.

[91] D. Hong, L. Gao, N. Yokoya, *et al.*, « More diverse means better: multimodal deep learning meets remote-sensing imagery classification », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, *5*, pp. 4340–4354, 2020.

[92] S. Alyaev and A. H. Elsheikh, « Direct multi-modal inversion of geophysical logs using deep learning », *Earth and Space Science*, vol. 9, *9*, e2021EA002186, 2022.

[93] Z. Xue, X. Yu, A. Yu, B. Liu, P. Zhang, and S. Wu, « Self-supervised feature learning for multimodal remote sensing image land cover classification », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[94] K. Kashinath, M. Mustafa, A. Albert, *et al.*, « Physics-informed machine learning: case studies for weather and climate modelling », *Philosophical Transactions of the Royal Society A*, vol. 379, *2194*, p. 20 200 093, 2021.

[95] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, « Physics-informed machine learning », *Nature Reviews Physics*, vol. 3, *6*, pp. 422–440, 2021.

[96] A. Karpatne, G. Atluri, J. H. Faghmous, *et al.*, « Theory-guided data science: a new paradigm for scientific discovery from data », *IEEE Transactions on knowledge and data engineering*, vol. 29, *10*, pp. 2318–2331, 2017.

[97] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, « Scientific machine learning through physics–informed neural networks: where we are and what's next », *Journal of Scientific Computing*, vol. 92, *3*, p. 88, 2022.

[98] M. Chantry, H. Christensen, P. Dueben, and T. Palmer, « Opportunities and challenges for machine learning in weather and climate modelling: hard, medium and soft ai », *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 379, *2194*, p. 20 200 083, 2021. DOI: 10.1098/rsta.2020.0083. eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2020.0083. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2020.0083.

[99] J. D. Hamilton, *Time series analysis*. Princeton university press, 2020.

[100] A. Bhandari, A. Kadambi, and R. Raskar, *Computational Imaging*. MIT Press, 2022.

[101] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics, 2005. DOI: 10.1137/1.9780898717921. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898717921. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9780898717921.

[102] A. Tarantola, « Popper, bayes and the inverse problem », *Nature physics*, vol. 2, *8*, pp. 492–494, 2006.

[103] R. Snieder and J. Trampert, « Inverse problems in geophysics », *in Wavefield Inversion*, A. Wirgin, Ed., Vienna: Springer Vienna, 1999, pp. 119–190, ISBN: 978-3-7091-2486-4.

[104] E. N. Lorenz, « Deterministic nonperiodic flow », *Journal of atmospheric sciences*, vol. 20, *2*, pp. 130–141, 1963.

[105] C. Snyder, « Introduction to the kalman filter », *in Advanced data assimilation for geosciences: Lecture notes of the LES Houches School of Physics: Special issue, June 2012*, É. Blayo, M. Bocquet, E. Cosme, and L. F. Cugliandolo, Eds. Oxford University Press, 2015, pp. 75–120, ISBN: 978-0-19-872384-4.

[106] A. H. Jazwinski, *Stochastic processes and filtering theory*. Courier Corporation, 2007.

[107] A. C. Lorenc, « Four-dimensional data assimilation », *in Advanced data assimilation for geosciences: Lecture notes of the LES Houches School of Physics: Special issue, June 2012*, É. Blayo, M. Bocquet, E. Cosme, and L. F. Cugliandolo, Eds. Oxford University Press, 2015, pp. 31–73, ISBN: 978-0-19-872384-4.

[108] D. Barker, W. Huang, Y.-R. Guo, and A. Bourgeois, « A three-dimensional variational (3dvar) data assimilation system for use with mm5 », *NCAR Tech Note*, vol. 68, 2003.

[109] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, « Neural ordinary differential equations », *Advances in neural information processing systems*, vol. 31, 2018.

[110] O. Talagrand, « 4d-var: four-dimensional data assimilation », *in Advanced data assimilation for geosciences: Lecture notes of the LES Houches School of Physics: Special issue, June 2012*, É. Blayo, M. Bocquet, E. Cosme, and L. F. Cugliandolo, Eds. Oxford University Press, 2015, pp. 3–30, ISBN: 978-0-19-872384-4.

[111] R. N. Bannister, « Elementary 4d-var », *Reading: University of Reading*, 2001.

[112] T. Mitchell, *Machine Learning*. McGraw Hill, 1997.

[113] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009, vol. 2.

[114] P. E. Hart, D. G. Stork, and R. O. Duda, *Pattern classification*. Wiley Hoboken, 2000.

[115] Y. LeCun, Y. Bengio, and G. Hinton, « Deep learning », *Nature*, vol. 521, *7553*, pp. 436–444, May 2015, ISSN: 1476-4687. DOI: 10.1038/nature14539. [Online]. Available: https://doi.org/10.1038/nature14539.

[116] J. Chai, H. Zeng, A. Li, and E. W. Ngai, « Deep learning in computer vision: a critical review of emerging techniques and application scenarios », *Machine Learning with Applications*, vol. 6, p. 100 134, 2021.

[117] Y. Li, « Research and application of deep learning in image recognition », *in 2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA)*, IEEE, 2022, pp. 994–999.

[118] J. Bharadiya, « A comprehensive survey of deep learning techniques natural language processing », *European Journal of Technology*, vol. 7, *1*, pp. 58–66, 2023.

[119] D. W. Otter, J. R. Medina, and J. K. Kalita, « A survey of the usages of deep learning for natural language processing », *IEEE transactions on neural networks and learning systems*, vol. 32, *2*, pp. 604–624, 2020.

[120] C. C. Aggarwal *et al.*, *Neural networks and deep learning*. Springer, 2018.

[121] J. Schmidhuber, « Deep learning in neural networks: an overview », *Neural networks*, vol. 61, pp. 85–117, 2015.

[122] D. E. Rumelhart, G. E. Hinton, J. L. McClelland, *et al.*, « A general framework for parallel distributed processing », *Parallel distributed processing: Explorations in the microstructure of cognition*, vol. 1, *45-76*, p. 26, 1986.

[123] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[124] K. Hornik, M. Stinchcombe, and H. White, « Multilayer feedforward networks are universal approximators », *Neural networks*, vol. 2, *5*, pp. 359–366, 1989.

[125] T. Chen and H. Chen, « Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems », *IEEE Transactions on Neural Networks*, vol. 6, *4*, pp. 911–917, 1995. DOI: 10.1109/72.392253.

[126] Y. Bengio, A. Courville, and P. Vincent, « Representation learning: a review and new perspectives », *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, *8*, pp. 1798–1828, 2013.

[127] Y. Bengio *et al.*, « Learning deep architectures for ai », *Foundations and trends® in Machine Learning*, vol. 2, *1*, pp. 1–127, 2009.

[128] A. M. Saxe, J. L. McClellans, and S. Ganguli, « Learning hierarchical categories in deep neural networks », *in Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 35, 2013.

[129] S. Akodad, L. Bombrun, M. Puscasu, J. Xia, C. Germain, and Y. Berthoumieu, « Deep ensemble learning model based on covariance pooling of multi-layer cnn features », *in 2022 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2022, pp. 1081–1085.

[130] J. Ba and R. Caruana, « Do deep nets really need to be deep? », *Advances in neural information processing systems*, vol. 27, 2014.

[131] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, « Learning representations by back-propagating errors », *Nature*, vol. 323, *6088*, pp. 533–536, Oct. 1986, ISSN: 1476-4687. DOI: 10.1038/323533a0. [Online]. Available: https://doi.org/10.1038/323533a0.

[132] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, « Automatic differentiation in machine learning: a survey », *Journal of Machine Learning Research*, vol. 18, *153*, pp. 1–43, 2018. [Online]. Available: http://jmlr.org/papers/v18/17-468.html.

[133] H.-J. Liao, J.-G. Liu, L. Wang, and T. Xiang, « Differentiable programming tensor networks », *Physical Review X*, vol. 9, *3*, p. 031 041, 2019.

[134] M. Bücker, *Automatic differentiation: applications, theory, and implementations*. Springer, 2006.

[135] D. A. Bezgin, A. B. Buhendwa, and N. A. Adams, « Jax-fluids: a fully-differentiable high-order computational fluid dynamics solver for compressible two-phase flows », *Computer Physics Communications*, vol. 282, p. 108 527, 2023, ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2022.108527. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010465522002466.

[136] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, « A systematic review on supervised and unsupervised machine learning algorithms for data science », *in Supervised and Unsupervised Learning for Data Science*, M. W. Berry, A. Mohamed, and B. W. Yap, Eds. Cham: Springer International

Publishing, 2020, pp. 3–21, ISBN: 978-3-030-22475-2. DOI: 10.1007/978-3-030-22475-2_1. [Online]. Available: https://doi.org/10.1007/978-3-030-22475-2_1.

[137] S. B. Kotsiantis, I. Zaharakis, P. Pintelas, *et al.*, « Supervised machine learning: a review of classification techniques », *Emerging artificial intelligence applications in computer engineering*, vol. 160, *1*, pp. 3–24, 2007.

[138] J.-C. Huang, K.-M. Ko, M.-H. Shu, and B.-M. Hsu, « Application and comparison of several machine learning algorithms and their integration models in regression problems », *Neural Computing and Applications*, vol. 32, pp. 5461–5469, 2020.

[139] Z. Ghahramani, « Unsupervised learning », *in Summer school on machine learning*, Springer, 2003, pp. 72–112.

[140] S. Ravishankar, J. C. Ye, and J. A. Fessler, « Image reconstruction: from sparsity to data-adaptive methods and machine learning », *Proceedings of the IEEE*, vol. 108, *1*, pp. 86–109, 2020. DOI: 10.1109/JPROC.2019.2936204.

[141] C. Tian, Y. Xu, L. Fei, and K. Yan, « Deep learning for image denoising: a survey », *in Genetic and Evolutionary Computing: Proceedings of the Twelfth International Conference on Genetic and Evolutionary Computing, December 14-17, Changzhou, Jiangsu, China 12*, Springer, 2019, pp. 563–572.

[142] A. E. Ezugwu, A. M. Ikotun, O. O. Oyelade, *et al.*, « A comprehensive survey of clustering algorithms: state-of-the-art machine learning applications, taxonomy, challenges, and future research prospects », *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104 743, 2022.

[143] D. Foster, *Generative deep learning*. " O'Reilly Media, Inc.", 2022.

[144] S. Bond-Taylor, A. Leach, Y. Long, and C. G. Willcocks, « Deep generative modelling: a comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models », *IEEE transactions on pattern analysis and machine intelligence*, 2021.

[145] D. P. Kingma and M. Welling, « Auto-encoding variational bayes », *arXiv preprint arXiv:1312.6114*, 2013.

[146] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, « Generative adversarial nets », *Advances in neural information processing systems*, vol. 27, 2014.

[147] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, « Deep unsupervised learning using nonequilibrium thermodynamics », *in Proceedings of the 32nd International Conference on Machine Learning*, F. Bach and D. Blei, Eds., ser. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, Jul. 2015, pp. 2256–2265. [Online]. Available: https://proceedings.mlr.press/v37/sohl-dickstein15.html.

[148] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, « Diffusion models in vision: a survey », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[149] S. Khan, H. Rahmani, S. A. A. Shah, M. Bennamoun, G. Medioni, and S. Dickinson, *A guide to convolutional neural networks for computer vision*. Springer, 2018, vol. 8.

[150] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, « Exploiting multi-channels deep convolutional neural networks for multivariate time series classification », *Frontiers of Computer Science*, vol. 10, pp. 96–112, 2016.

[151] S. Kiranyaz, T. Ince, O. Abdeljaber, O. Avci, and M. Gabbouj, « 1-d convolutional neural networks for signal processing applications », *in ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 8360–8364. DOI: 10.1109/ICASSP.2019.8682194.

[152] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, « Dive into deep learning », *in* 2020, ch. Padding and Stride, https://d2l.ai/chapter_convolutional-neural-networks/padding-and-strides.html.

[153] J. Wu, « Introduction to convolutional neural networks », *National Key Lab for Novel Software Technology. Nanjing University. China*, vol. 5, *23*, p. 495, 2017.

[154] L. Stanković and D. Mandic, « Convolutional neural networks demystified: a matched filtering perspective-based tutorial », *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2023.

[155] V. Dumoulin and F. Visin, « A guide to convolution arithmetic for deep learning », *arXiv preprint arXiv:1603.07285*, 2016.

[156] G. E. Hinton and R. R. Salakhutdinov, « Reducing the dimensionality of data with neural networks », *science*, vol. 313, *5786*, pp. 504–507, 2006.

[157] D. Bank, N. Koenigstein, and R. Giryes, « Autoencoders », *arXiv preprint arXiv:2003.05991*, 2020.

[158] S. Wold, K. Esbensen, and P. Geladi, « Principal component analysis », *Chemometrics and intelligent laboratory systems*, vol. 2, *1-3*, pp. 37–52, 1987.

[159] O. Ronneberger, P. Fischer, and T. Brox, « U-net: convolutional networks for biomedical image segmentation », *in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, Springer, 2015, pp. 234–241.

[160] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, « Dive into deep learning », *in* 2020, ch. Long Short-Term Memory (LSTM), `https://d2l.ai/chapter_recurrent-modern/lstm.html`.

[161] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, « Dive into deep learning », *in* 2020, ch. Residual Networks (ResNet) and ResNeXt, `https://d2l.ai/chapter_convolutional-modern/resnet.html`.

[162] S. A. Marhon, C. J. F. Cameron, and S. C. Kremer, « Recurrent neural networks », *in Handbook on Neural Information Processing*, M. Bianchini, M. Maggini, and L. C. Jain, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 29–65, ISBN: 978-3-642-36657-4. DOI: `10.1007/978-3-642-36657-4_2`. [Online]. Available: `https://doi.org/10.1007/978-3-642-36657-4_2`.

[163] S. Hochreiter and J. Schmidhuber, « Long short-term memory », *Neural computation*, vol. 9, *8*, pp. 1735–1780, 1997.

[164] G. Van Houdt, C. Mosquera, and G. Nápoles, « A review on the long short-term memory model », *Artificial Intelligence Review*, vol. 53, pp. 5929–5955, 2020.

[165] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, « Attention is all you need », *Advances in neural information processing systems*, vol. 30, 2017.

[166] Z. Niu, G. Zhong, and H. ²Yu, « A review on the attention mechanism of deep learning », *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[167] K. He, X. Zhang, S. Ren, and J. Sun, « Deep residual learning for image recognition », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[168] S. Ouala, A. Pascual, and R. Fablet, « Residual integration neural network », *in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 3622–3626.

[169] N. Dridi, L. Drumetz, and R. Fablet, « Learning stochastic dynamical systems with neural networks mimicking the euler-maruyama scheme », *in 2021 29th European Signal Processing Conference (EUSIPCO)*, IEEE, 2021, pp. 1990–1994.

[170] F. Rousseau, L. Drumetz, and R. Fablet, « Residual networks as flows of diffeomorphisms », *Journal of Mathematical Imaging and Vision*, vol. 62, pp. 365–375, 2020.

[171] M. Sander, P. Ablin, and G. Peyré, « Do residual neural networks discretize neural ordinary differential equations? », *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 520–36 532, 2022.

[172] C. M. Hyun, S. H. Baek, M. Lee, S. M. Lee, and J. K. Seo, « Deep learning-based solvability of underdetermined inverse problems in medical imaging », *Medical Image Analysis*, vol. 69, p. 101 967, 2021.

[173] T. A. Bubba, G. Kutyniok, M. Lassas, *et al.*, « Learning the invisible: a hybrid deep learning-shearlet framework for limited angle computed tomography », *Inverse Problems*, vol. 35, *6*, p. 064 002, 2019.

[174] C. Chen, Q. Chen, J. Xu, and V. Koltun, « Learning to see in the dark », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3291–3300.

[175] A. Gastineau, J.-F. Aujol, Y. Berthoumieu, and C. Germain, « Generative adversarial network for pansharpening with spectral and spatial discriminators », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2021.

[176] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, « Using deep neural networks for inverse problems in imaging: beyond analytical methods », *IEEE Signal Processing Magazine*, vol. 35, *1*, pp. 20–36, 2018.

[177] Y. Liu, H. Chen, Y. Chen, W. Yin, and C. Shen, « Generic perceptual loss for modeling structured output dependencies », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5424–5432.

[178] Y. Zhang, C. Fang, Y. Wang, *et al.*, « Multimodal style transfer via graph cuts », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5943–5951.

[179] J. Scarlett, R. Heckel, M. R. Rodrigues, P. Hand, and Y. C. Eldar, « Theoretical perspectives on deep learning methods in inverse problems », *IEEE journal on selected areas in information theory*, vol. 3, *3*, pp. 433–453, 2022.

[180] R. Fablet, B. Chapron, L. Drumetz, E. Mémin, O. Pannekoucke, and F. Rousseau, « Learning variational data assimilation models and solvers », *Journal of Advances in Modeling Earth Systems*, vol. 13, *10*, e2021MS002572, 2021, e2021MS002572 2021MS002572. DOI: https://doi.org/10.1029/2021MS002572. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021MS002572. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021MS002572.

[181] R. Fablet, L. Drumetz, and F. Rousseau, « Joint learning of variational representations and solvers for inverse problems with partially-observed data », 2020. arXiv: 2006.03653. [Online]. Available: https://arxiv.org/abs/2006.03653.

[182] A. Paszke, S. Gross, F. Massa, *et al.*, « Pytorch: an imperative style, high-performance deep learning library », *in Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., Curran Associates, Inc., 2019, pp. 8024–8035.

[183] Q. Febvre, R. Fablet, J. L. Sommer, and C. Ubelmann, « Joint calibration and mapping of satellite altimetry data using trainable variational models », *in ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 1536–1540. DOI: 10.1109/ICASSP43922.2022.9746889.

[184] S. Benaïchouche, C. Goff, B. Boussidi, F. Rousseau, and R. Fablet, « Learnable variational models for the reconstruction of sea surface currents using ais data streams: a case study on the sicily channel », *in IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2022, pp. 6821–6824. DOI: 10.1109/IGARSS46834.2022.9883870.

[185] M. Zambra, D. Cazau, N. Farrugia, *et al.*, « Learning-based temporal estimation of in-situ wind speed from underwater passive acoustics », *IEEE Journal of Oceanic Engineering*, vol. 48, *4*, pp. 1215–1225, 2023. DOI: 10.1109/JOE.2023.3288970.

[186] S. Dieleman and B. Schrauwen, « End-to-end learning for music audio », *in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 6964–6968. DOI: 10.1109/ICASSP.2014.6854950.

[187] R. Fablet, M. M. Amar, Q. Febvre, M. Beauchamp, and B. Chapron, « End-to-end physics-informed representation learning for satellite ocean remote sensing data: applications to satellite altimetry and sea surface currents », *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. V-3-2021, pp. 295–302, 2021. DOI: 10.5194/isprs-annals-V-3-2021-295-2021. [Online]. Available: https://isprs-annals.copernicus.org/articles/V-3-2021/295/2021/.

[188] R. Fablet, M. Beauchamp, L. Drumetz, and F. Rousseau, « Joint interpolation and representation learning for irregularly sampled satellite-derived geophysical fields », *Frontiers in Applied Mathematics and Statistics*, vol. 7, 2021, ISSN: 2297-4687. DOI: 10.3389/fams.2021.655224. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fams.2021.655224.

[189] N. Cressie and C. K. Wikle, *Statistics for spatio-temporal data*. John Wiley & Sons, 2015.

[190] A. Ancona, D. Geman, N. Ikeda, and D. Geman, « Random fields and inverse problems in imaging », *in École d'été de probabilités de Saint-Flour XVIII-1988*, Springer, 1990, pp. 115–193.

[191] M. Andrychowicz, M. Denil, S. G. Colmenarejo, *et al.*, « Learning to learn by gradient descent by gradient descent », *in Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16, Barcelona, Spain: Curran Associates Inc., 2016, pp. 3988–3996, ISBN: 9781510838819.

[192] C. Runge and H. König, *Vorlesungen über numerisches Rechnen*. Berlin: Springer, 1924.

[193] J. R. Cash, « Review paper. efficient numerical methods for the solution of stiff initial-value problems and differential algebraic equations », *Proceedings: Mathematical, Physical and Engineering Sciences*, vol. 459, *2032*, pp. 797–815, 2003, ISSN: 13645021.

[194] M. Beauchamp, R. Fablet, C. Ubelmann, M. Ballarotta, and B. Chapron, « Intercomparison of data-driven and learning-based interpolations of along-track nadir and wide-swath swot altimetry observations », *Remote Sensing*, vol. 12, Nov. 2020.

[195] S. Ouala, R. Fablet, L. Drumetz, *et al.*, « Physically informed neural networks for the simulation and data-assimilation of geophysical dynamics », *in IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 3490–3493.

[196] D. Nguyen, S. Ouala, L. Drumetz, and R. Fablet, « Assimilation-based learning of chaotic dynamical systems from noisy and partial data », *in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 3862–3866.

[197] S. Ouala, D. Nguyen, C. Herzet, *et al.*, « Learning ocean dynamical priors from noisy data using assimilation-derived neural nets », *in IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 9451–9454. DOI: 10.1109/IGARSS.2019.8900345.

[198] J. Brajard, A. Carrassi, M. Bocquet, and L. Bertino, « Combining data assimilation and machine learning to emulate a dynamical model from sparse and noisy observations: a case study with the lorenz 96 model », *Journal of Computational Science*, vol. 44, p. 101 171, 2020, ISSN: 1877-7503. DOI: https://doi.org/10.1016/j.jocs.2020.101171. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877750320304725.

[199] M. Bocquet, J. Brajard, A. Carrassi, and L. Bertino, « Bayesian inference of chaotic dynamics by merging data assimilation, machine learning and expectation-maximization », *Foundations of Data Science*, vol. 2, *1*, pp. 55–80, 2020.

[200] C.-O. 4. developing team. Starter version of the 4DVarNet implementation. Visited on 09 November 2023. (), [Online]. Available: https://github.com/CIA-Oceanix/4dvarnet-starter.

[201] C.-O. 4. developing team. Core version of the 4DVarNet implementation. Visited on 09 November 2023. (), [Online]. Available: https://github.com/CIA-Oceanix/4dvarnet-core.

[202] E. N. Lorenz, « Predictability: a problem partly solved », *in Proc. Seminar on predictability*, Reading, vol. 1, 1996.

[203] P. Best, R. Marxer, S. Paris, and H. Glotin, « Temporal evolution of the mediterranean fin whale song », *Scientific reports*, vol. 12, *1*, p. 13 565, 2022.

[204] P. Best, M. Ferrari, M. Poupard, *et al.*, « Deep learning and domain transfer for orca vocalization detection », *in 2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, pp. 1–7.

[205] M. Goldwater, D. P. Zitterbart, D. Wright, and J. Bonnel, « Machine-learning-based simultaneous detection and ranging of impulsive baleen whale vocalizations using a single hydrophone », *The Journal of the Acoustical Society of America*, vol. 153, *2*, pp. 1094–1107, 2023.

[206] P. Cauchy, K. J. Heywood, N. D. Merchant, B. Y. Queste, and P. Testor, « Wind speed measured from underwater gliders using passive acoustics », *Journal of Atmospheric and Oceanic Technology*, vol. 35, *12*, pp. 2305–2321, 2018. DOI: https://doi.org/10.1175/JTECH-D-17-0209.1. [Online]. Available: https://journals.ametsoc.org/view/journals/atot/35/12/jtech-d-17-0209.1.xml.

[207] W. O. Taylor, M. N. Anagnostou, D. Cerrai, and E. N. Anagnostou, « Machine learning methods to approximate rainfall and wind from acoustic underwater measurements (february 2020) », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, *4*, pp. 2810–2821, 2021. DOI: 10.1109/TGRS.2020.3007557.

[208] P. Berrisford, D. Dee, P. Poli, *et al.*, « The era-interim archive, version 2.0 », ECMWF, Technical Report, Nov. 2011.

[209] D. P. Dee, S. M. Uppala, A. J. Simmons, *et al.*, « The era-interim reanalysis: configuration and performance of the data assimilation system », *Quarterly Journal of the Royal Meteorological Society*, vol. 137, *656*, pp. 553–597, 2011. DOI: https://doi.org/10.1002/qj.828. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.828. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.828.

[210] D. L. Patiris, S. Pensieri, C. Tsabaris, *et al.*, « Rainfall investigation by means of marine in situ gamma-ray spectrometry in ligurian sea, mediterranean sea, italy », *Journal of Marine Science and Engineering*, vol. 9, *8*, p. 903, 2021.

[211] S. Pensieri, R. Bozzano, J. A. Nystuen, E. N. Anagnostou, M. N. Anagnostou, and R. Bechini, « Underwater acoustic measurements to estimate wind and rainfall in

the mediterranean sea », *Advances in Meteorology*, vol. 2015, p. 612 512, Apr. 2015, ISSN: 1687-9309. DOI: 10.1155/2015/612512.

[212] M. Zuba, E. Anagnostou, J. Nystuen, and M. Anagnostou, « Upal: underwater passive aquatic listener », *in OCEANS 2015 - MTS/IEEE Washington*, 2015, pp. 1–4. DOI: 10.23919/OCEANS.2015.7404450.

[213] J. A. Nystuen, M. N. Anagnostou, E. N. Anagnostou, and A. Papadopoulos, « Monitoring greek seas using passive underwater acoustics », *Journal of Atmospheric and Oceanic Technology*, vol. 32, *2*, pp. 334–349, 2015. DOI: 10.1175/JTECH-D-13-00264.1.

[214] G. Evensen, « The ensemble kalman filter for combined state and parameter estimation », *IEEE Control Systems Magazine*, vol. 29, *3*, pp. 83–104, 2009. DOI: 10.1109/MCS.2009.932223.

[215] J. Blum, F.-X. Le Dimet, and I. M. Navon, « Data Assimilation for Geophysical Fluids », *in Computational Methods for the Atmosphere and then Oceans*, ser. Handbook of Numerical Analysis, R. Temam and J. Tribbia, Eds., vol. 14, Elsevier, 2009, pp. 385–442.

[216] U. D. Mäder, *Augmented models in estimation and control*. ETH Zurich, 2010.

[217] P. Baldi, « Autoencoders, unsupervised learning, and deep architectures », *in Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds., ser. Proceedings of Machine Learning Research, vol. 27, Bellevue, Washington, USA: PMLR, Jul. 2012, pp. 37–49.

[218] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, « Stacked convolutional auto-encoders for hierarchical feature extraction », *in International conference on artificial neural networks*, Springer, 2011, pp. 52–59.

[219] S. Siami-Namini, N. Tavakoli, and A. S. Namin, « The performance of lstm and bilstm in forecasting time series », *in 2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 3285–3292.

[220] A. Paszke, S. Gross, S. Chintala, *et al.*, « Automatic differentiation in pytorch », *in NIPS 2017 Workshop on Autodiff*, 2017.

[221] D. P. Kingma and J. Ba, « Adam: A method for stochastic optimization », *in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[222] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, « Catboost: unbiased boosting with categorical features », *in Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018.

[223] J. H. Friedman, « Greedy function approximation: a gradient boosting machine », *Annals of statistics*, pp. 1189–1232, 2001.

[224] J. H. Friedman, « Stochastic gradient boosting », *Computational statistics & data analysis*, vol. 38, *4*, pp. 367–378, 2002.

[225] L. Breiman, « Random forests », *Machine Learning*, vol. 45, *1*, pp. 5–32, Oct. 2001.

[226] L. Breiman, « Bagging predictors », *Machine Learning*, vol. 24, *2*, pp. 123–140, Aug. 1996.

[227] M. Belmonte Rivas and A. Stoffelen, « Characterizing era-interim and era5 surface wind biases using ascat », *Ocean Science*, vol. 15, *3*, pp. 831–852, 2019. DOI: 10.5194/os-15-831-2019. [Online]. Available: https://os.copernicus.org/articles/15/831/2019/.

[228] S. Wager, S. Wang, and P. S. Liang, « Dropout training as adaptive regularization », *in Advances in Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., vol. 26, Curran Associates, Inc., 2013.

[229] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, « Dropout: a simple way to prevent neural networks from overfitting », *Journal of Machine Learning Research*, vol. 15, *56*, pp. 1929–1958, 2014.

[230] Q. Febvre, J. L. Sommer, C. Ubelmann, and R. Fablet, « Training neural mapping schemes for satellite altimetry with simulation data », *arXiv preprint arXiv:2309.14350*, 2023.

[231] M. Optis, A. Kumler, G. N. Scott, M. C. Debnath, and P. J. Moriarty, « Validation of ru-wrf, the custom atmospheric mesoscale model of the rutgers center for ocean observing leadership », National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2020.

[232] D. Green, « Transitioning noaa moored buoy systems from research to operations », *in OCEANS 2006*, IEEE, 2006, pp. 1–3.

[233] A. Ducournau and R. Fablet, « Deep learning for ocean remote sensing: an application of convolutional neural networks for super-resolution on satellite-derived sst data », *in 2016 9th IAPR Workshop on Pattern Recogniton in Remote Sensing (PRRS)*, IEEE, 2016, pp. 1–6.

[234] K. Stengel, A. Glaws, D. Hettinger, and R. N. King, « Adversarial super-resolution of climatological wind and solar data », *Proceedings of the National Academy of Sciences*, vol. 117, *29*, pp. 16 805–16 815, 2020. DOI: 10.1073/pnas.1918964117. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1918964117. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.1918964117.

[235] D. Lambhate and D. N. Subramani, « Super-resolution of sea surface temperature satellite images », *in Global Oceans 2020: Singapore–US Gulf Coast*, IEEE, 2020, pp. 1–7.

[236] J. Leinonen, D. Nerini, and A. Berne, « Stochastic super-resolution for downscaling time-evolving atmospheric fields with a generative adversarial network », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, *9*, pp. 7211–7223, 2020.

[237] Y. Hatanaka, Y. Glaser, G. Galgon, G. Torri, and P. Sadowski, « Diffusion models for high-resolution solar forecasts », *arXiv preprint arXiv:2302.00170*, 2023.

[238] R. Durall, A. Ghanim, M. R. Fernandez, N. Ettrich, and J. Keuper, « Deep diffusion models for seismic processing », *Computers & Geosciences*, vol. 177, p. 105 377, 2023.

[239] L. Chen, F. Du, Y. Hu, Z. Wang, and F. Wang, « Swinrdm: integrate swinrnn with diffusion model towards high-resolution and high-quality weather forecasting », *in Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 2023, pp. 322–330.

[240] M. Beauchamp, Q. Febvre, H. Georgenthum, and R. Fablet, « 4dvarnet-ssh: end-to-end learning of variational interpolation schemes for nadir and wide-swath satellite altimetry », *Geoscientific Model Development Discussions*, vol. 2022, pp. 1–37, 2022.

[241] D. Kingma and J. Ba, « Adam: a method for stochastic optimization », *International Conference on Learning Representations*, Dec. 2014.

[242] T. G. Dietterich, « Ensemble methods in machine learning », *in International workshop on multiple classifier systems*, Springer, 2000, pp. 1–15.

[243] T. N. Rincy and R. Gupta, « Ensemble learning techniques and its efficiency in machine learning: a survey », *in 2nd international conference on data, engineering and applications (IDEA)*, IEEE, 2020, pp. 1–6.

[244] D. Xu, M. L. Lee, and W. Hsu, « Classification with dynamic data augmentation », *in 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, 2021, pp. 1434–1441. DOI: 10.1109/ICTAI52525.2021.00228.

[245] M. Liu, G. Chowdhary, B. C. Da Silva, S.-Y. Liu, and J. P. How, « Gaussian processes for learning and control: a tutorial with examples », *IEEE Control Systems Magazine*, vol. 38, *5*, pp. 53–86, 2018.

[246] M. Cathelain, R. Husson, H. Berger, and M. Fragoso, « High-resolution satellite observations to account for coastal gradient in wind resource assessment: application to french coastal areas », *in Journal of Physics: Conference Series*, IOP Publishing, vol. 2505, 2023, p. 012 027.

[247] J. Schulz-Stellenfleth, S. Emeis, M. Dörenkämper, *et al.*, « Coastal impacts on offshore wind farms—a review focussing on the german bight area », *Meteorol. Z*, vol. 31, pp. 289–315, 2022.

[248] L. Boussioux, C. Zeng, T. Guénais, and D. Bertsimas, « Hurricane forecasting: a novel multimodal machine learning framework », *Weather and Forecasting*, vol. 37, *6*, pp. 817–831, 2022.

[249] S. Benaïchouche, C. Le Goff, B. Boussidi, F. Rousseau, and R. Fablet, « Multimodal data assimilation of sea surface currents from ais data streams and satellite altimetry using 4dvarnet », Copernicus Meetings, Tech. Rep., 2023.

[250] C. Bai, D. Zhao, M. Zhang, and J. Zhang, « Multimodal information fusion for weather systems and clouds identification from satellite images », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 7333–7345, 2022.

[251] B. Lim and S. Zohren, « Time-series forecasting with deep learning: a survey », *Philosophical Transactions of the Royal Society A*, vol. 379, *2194*, p. 20 200 209, 2021.

[252] A. G. Salman, B. Kanigoro, and Y. Heryadi, « Weather forecasting using deep learning techniques », *in 2015 international conference on advanced computer science and information systems (ICACSIS)*, Ieee, 2015, pp. 281–285.

[253] A. M. Abdalla, I. H. Ghaith, and A. A. Tamimi, « Deep learning weather forecasting techniques: literature survey », *in 2021 International Conference on Information Technology (ICIT)*, IEEE, 2021, pp. 622–626.

[254] X. Ren, X. Li, K. Ren, *et al.*, « Deep learning-based weather prediction: a survey », *Big Data Research*, vol. 23, p. 100 178, 2021.

[255] N.-H. Kim, J. H. Hwang, J. Cho, and J. S. Kim, « A framework to determine the locations of the environmental monitoring in an estuary of the yellow sea », *Environmental Pollution*, vol. 241, pp. 576–585, 2018.

[256] N.-H. Kim, D. Baek, J.-i. Kwon, J.-Y. Choi, and K.-Y. Heo, « Strategy for additional buoy array installation in operational buoy-observation network in korea », *Ocean Engineering*, vol. 266, p. 112 746, 2022.

[257] S. Liu, M. Song, S. Chen, *et al.*, « An intelligent modeling framework to optimize the spatial layout of ocean moored buoy observing networks », *Frontiers in Marine Science*, vol. 10, p. 1 134 418, 2023.

[258] T. P. Merrifield, D. P. Griffith, S. A. Zamanian, *et al.*, « Synthetic seismic data for training deep learning networks », *Interpretation*, vol. 10, *3*, SE31–SE39, 2022.

[259] W. Liu, B. Luo, and J. Liu, « Synthetic data augmentation using multiscale attention cyclegan for aircraft detection in remote sensing images », *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[260] T. Gupta, P. Zwartjes, U. Bamba, K. Ghosal, and D. K. Gupta, « Near-surface velocity estimation using shear-waves and deep-learning with a u-net trained on synthetic data », *Artificial Intelligence in Geosciences*, vol. 3, pp. 209–224, 2022.

[261] F. Zhuang, Z. Qi, K. Duan, *et al.*, « A comprehensive survey on transfer learning », *Proceedings of the IEEE*, vol. 109, *1*, pp. 43–76, 2020.

[262] K. Weiss, T. M. Khoshgoftaar, and D. Wang, « A survey of transfer learning », *Journal of Big data*, vol. 3, *1*, pp. 1–40, 2016.

[263] N. Lafon, R. Fablet, and P. Naveau, « Uncertainty quantification when learning dynamical models and solvers with variational methods », *Journal of Advances in Modeling Earth Systems*, vol. 15, *11*, e2022MS003446, 2023. DOI: https://doi.org/10.1029/2022MS003446. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2022MS003446. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003446.

[264] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, « Variational inference: a review for statisticians », *Journal of the American statistical Association*, vol. 112, *518*, pp. 859–877, 2017.

[265] M. Beauchamp, R. Fablet, and H. Georgenthum, « Neural spde solver for uncertainty quantification in high-dimensional space-time dynamics », 2023.

[266] H. Holden, B. Øksendal, J. Ubøe, *et al.*, *Stochastic partial differential equations*. Springer, 1996.

[267] J. Long, E. Shelhamer, and T. Darrell, « Fully convolutional networks for semantic segmentation », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

Part IV

Appendix

Appendix A

Annex of Chapter 4

This annex is dedicated to the supplementary information associated with the analyses presented in Chapter 4. In this appendix, we provide the following supplementary information. We argument the choice of the model architecture in Section A.1. Sections A.1.1 and A.1.2 present the different models reconstructions and the outcome of the LR bias analyses (cfr. Section 4.4.3) for different choices of the neural architecture used. We choose to perform this supplementary analysis to check if and in which measure a deeper and more complex architecture may impact the model capability to learn the correction for the biased LR data.

A.1 Model complexity

One primary point concerns the choice of the architecture of the model Φ. This model is both used to parameterize the direct data-to-state variable inversion and to parameterize the dynamical operator of the 4DVarNet-based inversion scheme. Refer to Section 4.3 for an overview on the two approaches.

The analyses presented in Chapter 4 involve a simple neural architecture to parameterize the operator Φ. Section 4.3.4 described this architecture. The choice of such a simple model Φ is motivated by some preliminary sensitivity test w.r.t. model complexity. Let Φ_α be the neural architecture described in Section 4.3.4. Let Φ_β and Φ_γ two more complex models. Figure A.1 illustrates these three models. Φ_α and Φ_β are two convolutional networks. The first is a simple linear two-layered network while the second has one more layer and Leaky Rectified Linear Unit (ReLU) non-linear activation functions after the first two layers. The hidden channels dimension of Φ_β is 128 while Φ_α shrinks the channels dimension to 32 in the connection between the two layers.

The model Φ_γ is a U-Net (cfr. Section 2.3.3). Contrarily to the first two models, Φ_γ implements downsampling and unsampling modules. After processing the input with a Conv2d layer, the downsampling module uses a max-pool and Conv2d layers to reduce

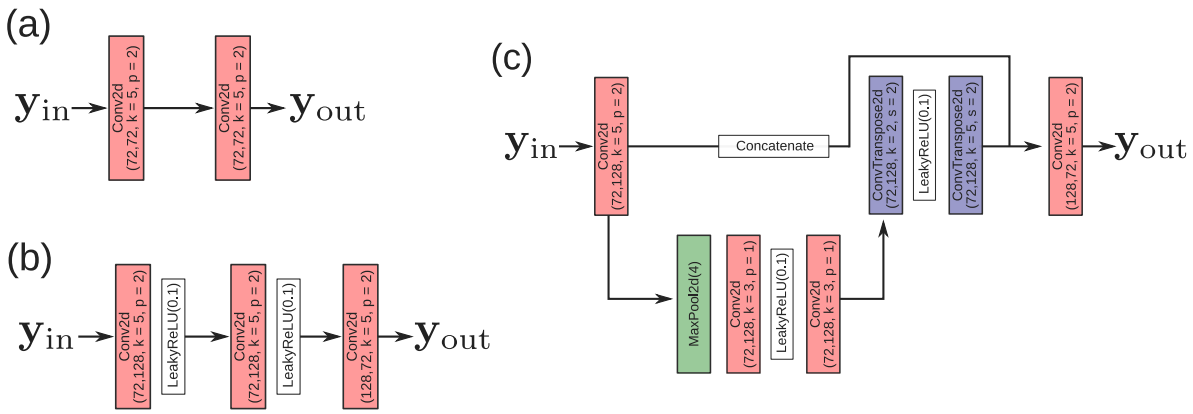


Figure A.1 – Sketch of the neural models tested. Panel (a): Model Φ_α. This is the model used for the analyses presented in Chapter 4. Panel (b): Model Φ_β. This model is similar to the simple network Φ_α but is deeper, has more parameters and is non-linear. Panel (c): Model Φ_γ. This architecture is a U-Net model. The red blocks represent Conv2d layers. Blue blocks represents Conv2d transposed layers. The symbols are the following. The first two argument of the Conv2d and Conv2d transposed functions are the input and output channels referred to one given layer. Letter k is the kernel size, p is the padding and s is the stride. The green block represents a max-pooling 2d operation. The downsampling factor is 4. The argument of the Leaky ReLU is the negative slope of the activation function.

the signal dimensions. The downsampled signal is processed by the upsampling module, which implements Conv2d transposed layers. The transposed convolutions are used to increase the dimension of the signal [267]. This upampled signal is concatenated with the convolved input to create the U-Net shortcut (cfr. Section 2.3.3). The signal is finally passed to a Conv2d layer.

Table A.1 reports the reconstruction error (the RMSE), the training times and the memory size for each one of the models introduced above. These metrics are referred to the ensemble of 10 models. So the RMSE is referred to the distance between the median aggregation and the ground truths (cfr. Section 4.4.1). The training time is the time required to train the ensemble of 10 models and the memory size is the memory required to save the ensemble of 10 models. Let

$$\Delta\text{Perf}(\Phi) = \text{RMSE}_{B_1(\Phi)}(\hat{\mathbf{x}}_{\text{median}}, \mathbf{u}) - \text{RMSE}_{M_m(\Phi)}(\hat{\mathbf{x}}_{\text{median}}, \mathbf{u}) \tag{A.1}$$

be the difference between the RMSE of the direct inversion B_1 and the RMSE of the 4DVarNet M_m, as reported in Table A.1. The symbol $\hat{\mathbf{x}}_{\text{median}}$ represents the aggregated

Model	RMSE [m s^{-1}]	Time	Memory size [MB]
		Direct inversion	
$B_1(\Phi_\alpha)$	0.9571	20 min	39
$B_1(\Phi_\beta)$	0.9217	30 min	129
$B_1(\Phi_\gamma)$	0.8862	40 min	430
		4DVarNet	
$M_m(\Phi_\alpha)$	0.8617	4 hours 40 min	157
$M_m(\Phi_\beta)$	0.8604	6 hours 40 min	248
$M_m(\Phi_\gamma)$	0.8517	7 hours 30 min	549

Table A.1 – Comparison of the three model architectures tested. The evaluation metrics reported in this table refer to 10 runs. The memory size is then the memory required to store the ensemble of the 10 models and the time is the training time of the 10 models ensemble. The models are trained under the C3 data configuration (cfr. Section 4.4.2).

output (cfr. Section 4.4.1) and $\mathbf{u}$ is the ground truth. The first remark is the following. The trend of the performance difference ΔPerf decreases with model complexity. The values for $\Delta\text{Perf}(\Phi_\alpha)$, $\Delta\text{Perf}(\Phi_\beta)$ and $\Delta\text{Perf}(\Phi_\gamma)$ are respectively 0.0954, 0.0613 and 0.0345 m s^{-1}. This means that for the 4DVarNet inversion the performance improvement is more shrunken. Interestingly, the improvement is more evident for the case of direct inversion. In addition, the computational time required to train the 4DVarNet model is relevant. A gain of 10^{-3} m s^{-1} in terms of RMSE costs 2 hours more using Φ_β instead of Φ_α. Training the 4DVarNet model with Φ_γ as dynamical operator leads to a gain of 10^{-2} m s^{-1} in terms of RMSE w.r.t. the case of Φ_α but for the price of nearly 3 more hours of training time. In light of these experiments, we choose to use the simple neural architecture Φ_α to perform the numerical simulations. This choice is motivate by the extensiveness of the simulations required for the full analyses. However, we also think that an operationalization of such framework would require some further investigation on the optimal model architecture.

A.1.1 Reconstructions of the different models

Figure A.2 depicts graphically the remark above. Panel (a) and (b) illustrate reconstructions and associated average error maps for the direct and 4DVarNet inversions backed by the three models presented above. For the reconstructions of the direct inversion case the improvement can be appreciated by naked eye. For the different 4DVarNet

reconstruction this improvement is not visible. Recalling the results in Table A.1, the difference in terms of RMSE between these reconstructions is of the order of 10^{-2} $\mathrm{m\,s^{-1}}$.

A.1.2 LR biased data sensitivity

The results on the biased LR data are reported in Figure A.3. These results corroborate the remarks provided above. The substantial differences between the model architectures are appreciable for the case of the direct inversion. In the case of the 4DVarNet framework such differences are minimal. This suggests that the overall 4DVarNet end-to-end architecture, constituted by the neural dynamical operator Φ and the gradient solver Γ (cfr. Section 2.4) attenuate the limitations of a possibly under-sized and linear model like Φ_α. A further test, in Figure A.4, aims to assess which is the difference between the performance of the two instances of the 4DVarNet model (cfr. Section 4.3.2). This test is done for the model Φ_α. The difference between the curve related to the model M_s (non-trainable data proximity term in the variational cost) and M_m (trainable data proximity term) is evident. This may suggest that a relevant part of the 4DVarNet improvement and bias correction learning is implemented by the trainable multi-modal approach to process different data sources.

A.2 Uniform buoys network

The main text reported an ablation analysis w.r.t. the in-situ sensors. This analysis showed quantitatively the impact of (i) each sensor and (ii) sensors clusters on the overall model reconstruction performance. This kind of study suggests that an optimal configuration of in-situ sensors may be sought running the simulations and evaluating *a-posteriori* the goodness of the buoys constellation. The analyses of Chapter 4 account for a real buoys network. In this section we present the result of one such a-posteriori analysis, where the buoys network is arbitrary. The overall number of sensors is still 13 but their positions are chosen to cover more uniformly the region selected.

The value of such analysis may be stated in the perspective of the the discussion of Section 1.2.1. The installation of in-situ sensors is a costly operation in terms of economical resources and field personnel safety. By this kind of observing system simulation experiment, we can emulate the effect of an alternative buoys constellation with no field effort and expense.

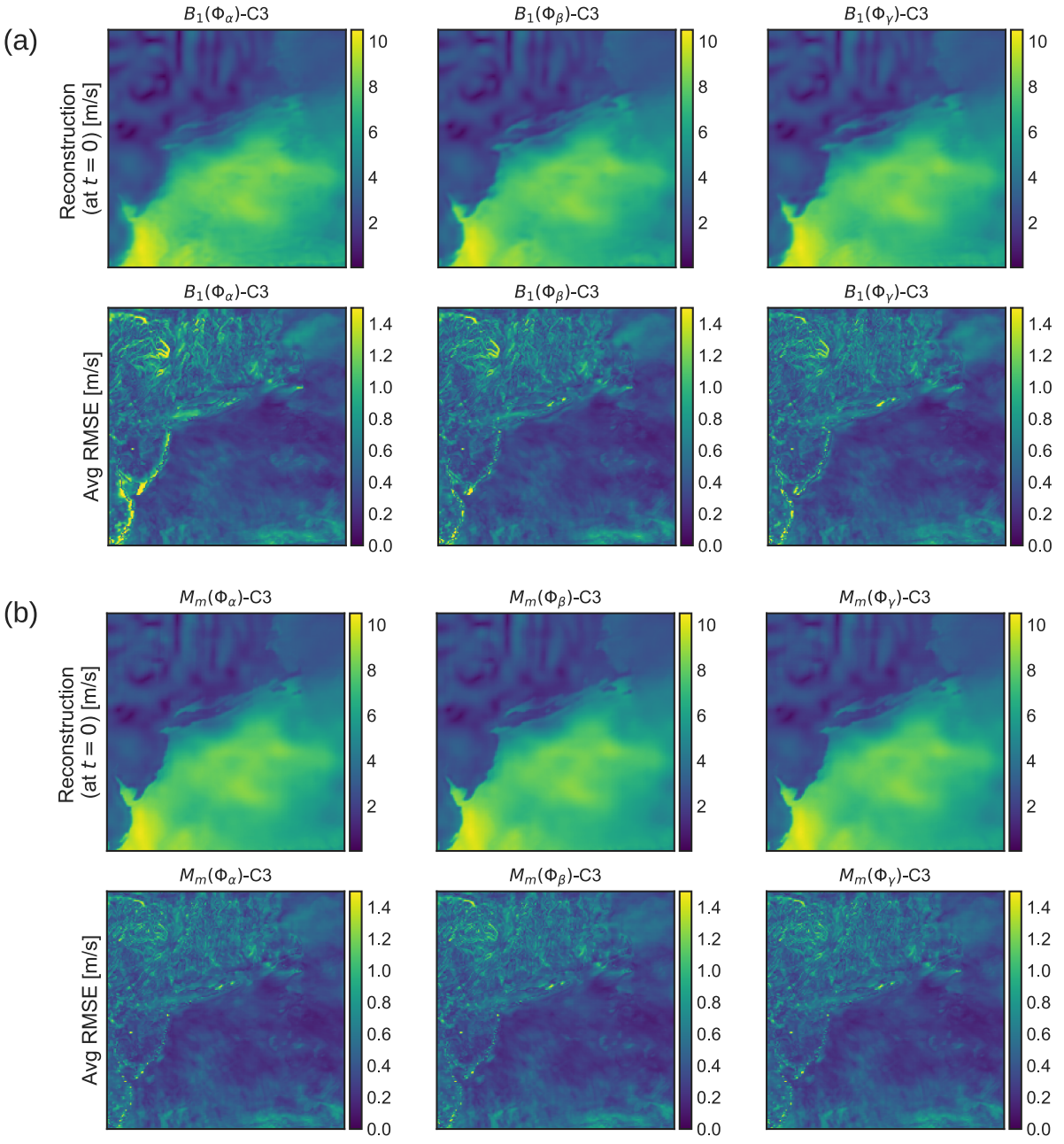


Figure A.2 – Reconstructions and RMSE visualizations for the three models discussed. Panel (a): Direct inversion. Panel (b): 4DVarNet inversion. In both panels, the top row depicts the reconstructions and the bottom row the RMSE of the model reconstructions w.r.t. the original data.

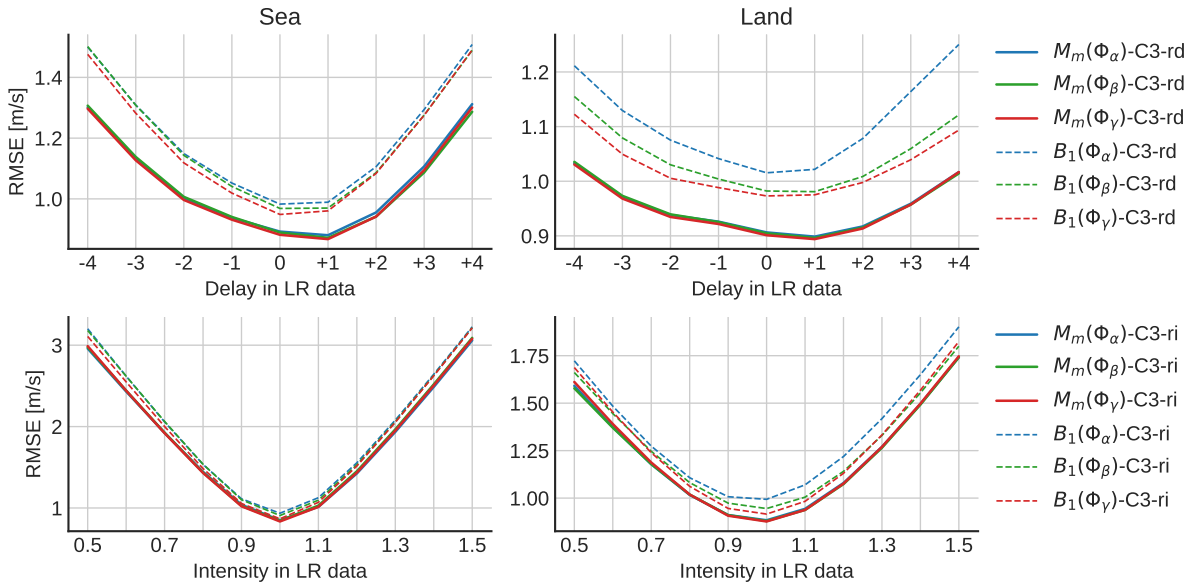


Figure A.3 – Degradation curves referred to the three models Φ_α, Φ_β, Φ_γ for both direct inversion and 4DVarNet.

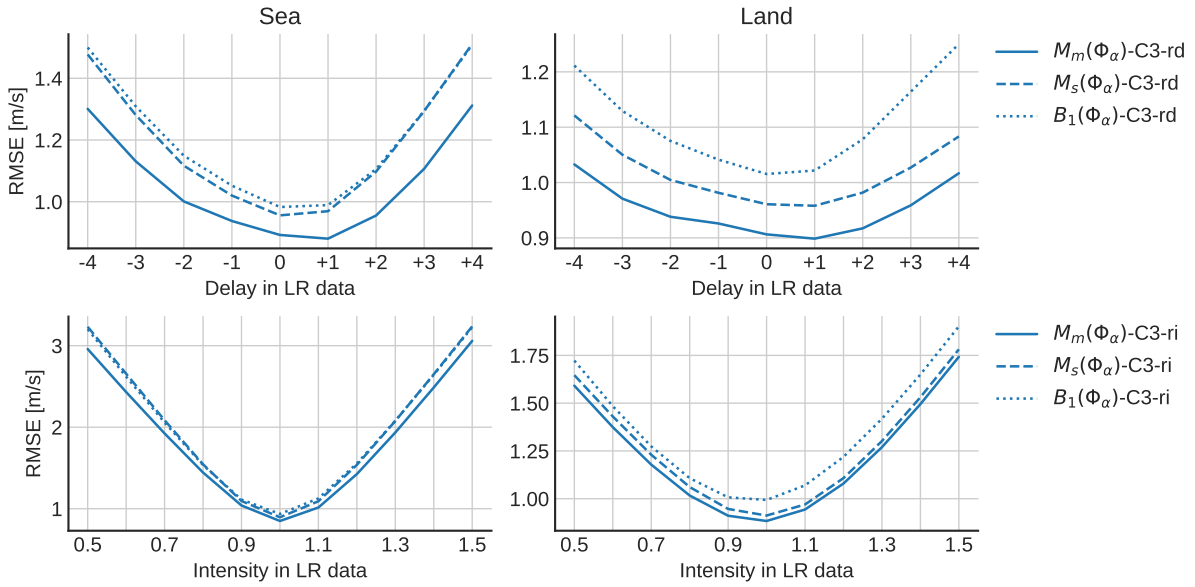


Figure A.4 – Degradation curves referred to Φ_α. This plot depicts the performance difference of direct inversion and the two instances of the 4DVarNet, that is with the simple and trainable data term in the variational cost.

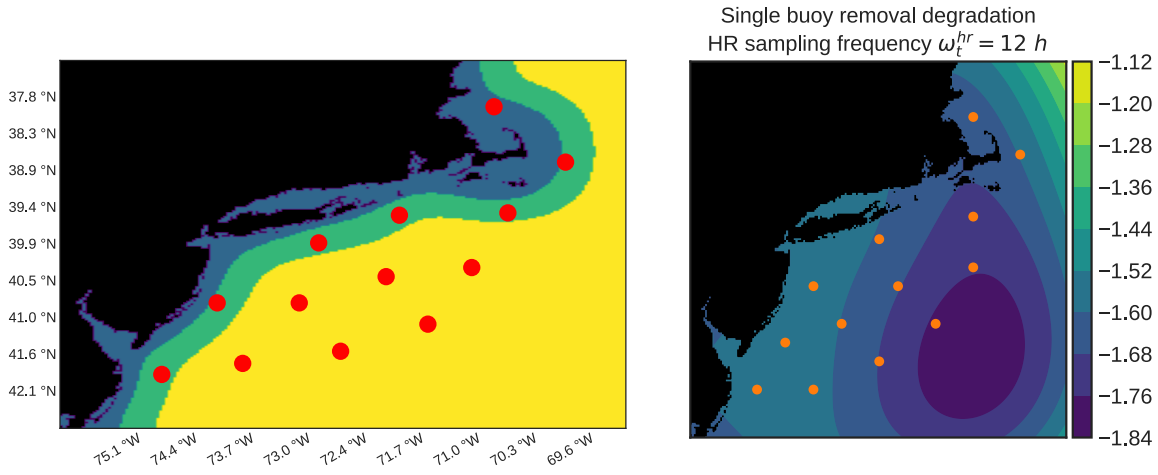


Figure A.5 – Left panel: Our arbitrary buoys network. Right panel: Degradation map.

Figure A.5 shows the results of this simulation. The overall reconstruction error of the M_m-C3 model using the buoys network as in the left panel of the figure is 0.8749 $\mathrm{m\,s^{-1}}$. The performance is slightly worse that the performance of M_m-C3 using the real sensors network (cfr. Section 4.4). This may be explained by the fact that in the real network some buoys are installed in the close proximity of the coastline and this gives the model an advantage in capturing the wind speed gradients due to the shelter effect, as observed in the main text.

However, the interesting result of this analysis is reported in the right panel of Figure A.5. While the overall reconstruction performance did not improve w.r.t. the configuration reported in Chapter 4, the ablation study on the synthetic buoys network reveals that the sensors in this configuration are more inter-dependent. The map of average per-buoy degradation attains higher degradation values. This result can be interpreted as follows. A more uniform network does not improve the overall reconstruction performance but is capable of partially observing the spatial variability of the wind speed on a larger spatial extent.

A.3 HR sampling hours

Throughout the experiments reported in Chapter 4 we consistently used the sampling scheme for the HR fields by which the HR fields are available at 06:00 and 18:00 for each 24-hours daily time series. Although realistic from the sampling frequency point of view,

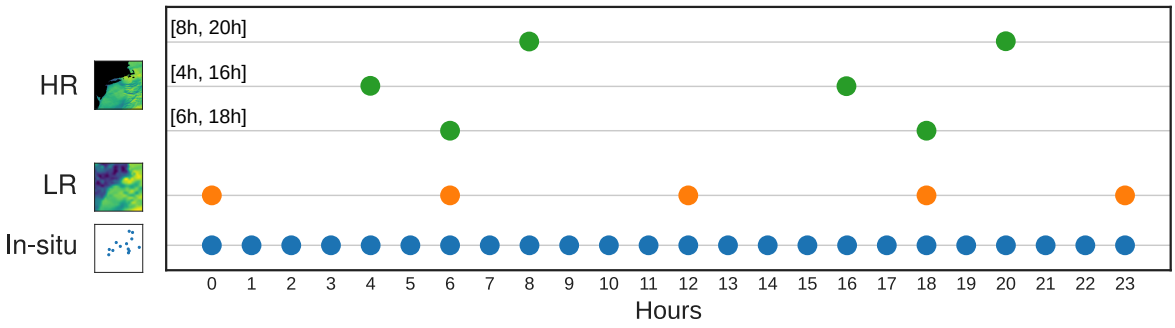


Figure A.6 – Visualization of the sampling frequencies of the HR fields in the alternative sampling schemes.

Model	RMSE $[\mathrm{m\,s^{-1}}]$
M_m-C3 HR [6h, 18h]	0.8617
M_m-C3 HR [4h, 16h]	0.8062
M_m-C3 HR [8h, 20h]	0.8114

Table A.2 – Reconstruction errors associated to different sampling schemes for the HR fields.

this scheme neglects the diurnal cycles that may impact the wind speed intensity and/or direction. We choose to perform some additional simulations using the same temporal sampling frequency but setting the HR fields at different hours in the day. Figure A.6 represents these alternative schemes.

Table A.2 reports the results of the simulations. We see that the reconstruction performances associated to the proposed alternative sampling schemes are better that the classical 06:00-18:00 scheme used in the results of the main text. These results may be interpreted as follows. In the 06:00-18:00 configuration, the two HR daily snapshots always match with the LR fields. On the other hand, if for example the HR fields are available at 04:00 and 16:00, the 24-hours daily series is richer of information. There are more hours in the day in which some information, coming from the low or high-resolution fields is available.

Appendix B

Résumé étendu

Conformément aux directives de l'Institut Mines-Télécom (IMT) Atlantique Bretagne-Pays de la Loire, la thèse de doctorat rédigée dans une langue autre que le français doit être accompagnée d'un résumé détaillé du travail en langue française. Ce chapitre contient la version longue du résumé et est organisé comme suit. La Section B.1 décrit le contexte scientifique et les objectifs du travail. La Section B.2 résume les contributions de la thèse. La Section B.3 décrit le cadre méthodologique mis en œuvre pour les analyses dans les chapitres de contribution de la thèse. Les Sections B.4 et B.5 présentent les principaux résultats des contributions. La section B.6 présente les conclusions générales.

In accordance with the guidelines of the Institut Mines-Télécom (IMT) Atlantique Bretagne-Pays de la Loire, a doctoral thesis written in a language other than French must be accompanied by a detailed summary of the work in French. This chapter contains the long version of the summary and is organized as follows. Section B.1 describes the scientific background and objectives of the work. Section B.2 summarizes the contributions of the thesis. Section B.3 describes the methodological framework implemented for the analyses in the contribution chapters of the thesis. Sections B.4 and B.5 present the main results of the contributions. Section B.6 presents the overall conclusions.

B.1 Contexte général et objectifs scientifiques

Le travail présenté dans cette thèse de doctorat vise à appliquer des méthodes récentes basées sur l'IA à des applications et des problèmes scientifiques concernant la surveillance de la surface des océans. Couvrant 70 % de la surface de la Terre, les océans jouent un rôle central dans la régulation du climat mondial et le développement de phénomènes météorologiques à grande échelle. En outre, ils constituent la principale infrastructure sur laquelle reposent le commerce mondial et un grand nombre d'autres activités humaines. Compte tenu de l'importance centrale du milieu maritime et océanique, il est facile de comprendre comment la caractérisation précise de la dynamique de surface est une condition essentielle

pour la planification et la mise en œuvre des activités mentionnées ci-dessus.

L'océan, tout comme l'atmosphère et tous les écoulements fluides, est régi par des lois physiques fortement non linéaires. La conséquence réelle de cette caractéristique est l'imprévisibilité et la nature chaotique du mouvement des fluides. Cela en empêche la reconstruction à toutes les échelles spatio-temporelles et la prédiction pour des horizons temporels moyens à longs. En outre, ces phénomènes présentent une forte variabilité spatio-temporelle. En raison des caractéristiques susmentionnées, il n'est pas possible de décrire les processus physiques à la surface à la fois à un niveau de résolution élevé et sur de grandes étendues spatiales. Dans le cadre de cette thèse, parmi tous les processus ayant lieu dans ce compartiment physique, nous traitons exclusivement de la vitesse du vent. Ce paramètre géophysique affecte l'interface entre l'atmosphère et l'océan, c'est-à-dire la surface de la mer et la couche située juste en dessous.

Compte tenu de l'importance de ce paramètre, diverses techniques d'observations ont été développées et mises en œuvre sur le terrain et à distance. Les principales techniques sont divisées en techniques **in-situ** [1], [2] et **télédétection** [3], notamment grâce à l'imagerie SAR [4], [5]. Dans le premier cas, le capteur est placé à proximité du processus à observer. Ce type de capteur peut mesurer le phénomène avec continuité temporelle mais sans extension spatiale. Les mesures in-situ doivent donc être considérées comme des observations locales. D'autre part, les techniques de télédétection impliquent l'installation du capteur sur un support aérien ou satellitaire. Cela permet d'observer, à un instant donné, une région dont l'étendue spatiale est importante. Le prix à payer est cependant la fréquence limitée d'observation d'une même région dans le temps. Ceci empêche une caractérisation continue de l'évolution temporelle du vent à la surface.

Les méthodes basées sur les mesures sont complémentaires (et constituent une partie essentielle) des méthodes de prévision et de reconstruction basées sur la **simulation** et la **prévision météorologique numérique**. Les systèmes d'**assimilation de données** [6]-[8] constituent l'état de l'art dans cette catégorie de méthodes. D'un point de vue mathématique, ces schémas sont utilisés pour identifier l'état d'un système dynamique dont le comportement évolutif est connu et dont les observations partielles sont disponibles. Les produits des schémas d'assimilation, appelés *réanalyses* [9] sont largement utilisés pour les prévisions météorologiques et la recherche sur le climat. Les réanalyses, telles que le catalogue ERA-5 du ECMWF [10], ont une couverture mondiale, une fréquence horaire et une résolution spatiale de 0.25°, environ 30 km à l'équateur. Cependant, les réanalyses présentent également des limites, notamment en ce qui concerne les résolutions spatiales.

Une résolution de 30 km peut ne pas être suffisante pour résoudre des échelles plus petites. De plus, les réanalyses sont affectées par les erreurs de simulation, liées aux hypothèses sous-jacentes à la mise en œuvre numérique. Ces erreurs sont exprimées dans les produits opérationnels comme des erreurs de synchronisation entre la prévision et l'observation réelle du phénomène, ou comme des erreurs dans l'intensité du phénomène [11].

Cet aperçu a illustré les principales techniques et méthodes de mesure et de description du vent à la surface. Il apparaît clairement qu'aucune des sources d'information mentionnées n'est complète, car aucune ne peut saisir simultanément toute la variabilité spatio-temporelle de la distribution du vent à la surface. Ces dernières années, des méthodes basées sur l'**apprentissage profond** [12], [13] ont été utilisées avec succès dans des applications géoscientifiques [14]-[16]. Par exemple, pour des problèmes d'inpainting [17], de la super-résolution [18], de l'interpolation géophysique [19] et l'identification et modélisation des systèmes dynamiques [20].

Dans notre cas précis, nous prenons l'exemple des développements récents dans le domaine de l'**apprentissage profond multi-modal** [21], [22]. L'idée derrière cette classe de méthodes est d'utiliser différentes sources de données d'entrée (modalités) pour en exploiter l'informations complémentaires. L'objectif de cette thèse est de mettre en place des méthodes multi-modales de traitement de grands volumes de données et d'informations issues de l'observation et de la modélisation du vent à la surface de la mer. En développant notre travail, nous souhaitons exploiter la flexibilité et l'efficacité de l'apprentissage profond et les fondements conceptuels et théoriques de méthodologies plus établies, telles que l'assimilation de données. Ces dernières années, de nombreux travaux se sont concentrés sur l'étude des liens entre l'assimilation de données et l'apprentissage profond [23], par exemple pour améliorer la résolution spatiale des produits de réanalyse [24] et pour caractériser les erreurs dans les modèles d'assimilation [25]. Nous considérons en particulier l'approche proposée par Fablet *et al.* [26], [27], dans laquelle le schéma d'assimilation de données *weak-constrained* 4DVar [6] est, dans une large mesure, paramétré par des opérateurs apprenables basés sur des réseaux neuronaux. Cette approche, appellé 4DVarNet, est utilisée pour apprendre simultanément le modèle sous-jacent et le solveur associé [28]. L'attrait de ce type de cadre réside dans la contrainte physique et dynamique des schémas de type 4DVar, qui est particulièrement utile pour modéliser les processus temporels, et dans le pouvoir expressif des modèles d'apprentissage profond, qui permettent de caractériser et de décrire efficacement le comportement dynamique du phénomène [26].

B.2 Contributions

Le travail de cette thèse s'inscrit dans le cadre scientifique décrit ci-dessus et se développe à travers deux contributions principales. La première concerne une étude de cas locale, où l'objectif est de reconstruire la vitesse du vent de surface à partir d'observations acoustiques sous-marines obtenues à partir d'une infrastructure d'observation installée dans le Golfe de Gênes, en Italie. Les résultats de cette analyse montrent le potentiel du cadre 4DVarNet par rapport aux approches plus traditionnelles appliquées à l'estimation du vent de surface à partir de l'acoustique sous-marine. Il est également montré comment le schéma d'inversion de l'acoustique au vent de surface peut être amélioré par l'approche multi-modale, juxtaposant l'acoustique sous-marine avec la vitesse du vent reconstruite par des produits réanalysés.

La deuxième contribution se concentre sur la reconstruction du vent à la surface de la mer sur une région spatiale plus large. Il s'agit d'une région mixte englobant à la fois une partie de la côte est des États-Unis et une partie de l'océan Atlantique tout proche. Dans cette analyse, nous utilisons des données de modèle pour simuler des observations couramment disponibles dans le monde réel, en particulier des images satellite à haute résolution, des séries temporelles locales obtenues à partir de capteurs in situ, et des produits réanalysés à basse résolution. Comme mentionné dans l'introduction, chacune de ces sources d'information n'est pas complète. L'objectif de l'analyse est la reconstruction de séries temporelles de champs de vitesse du vent à haute résolution. Les résultats de nos simulations montrent que ces sources d'entrée peuvent être utilisées efficacement pour la reconstruction.

B.3 Cadre méthodologique

Cette section est consacrée à une brève présentation de l'ensemble des outils de méthodologie utilisés pour les analyses qui seront illustrées dans les sections suivantes. Les deux cas mentionnés dans la section précédente peuvent être formalisés mathématiquement comme des *problèmes inverses* [29]. L'objectif est de reconstruire l'état d'un système dynamique à partir de ses observations partielles et d'une connaissance a priori de son comportement dynamique. Les techniques d'assimilation de données évoquées ci-dessus sont un exemple concret de résolution de problèmes inverses en océanographie et en météorologie. La simulation numérique est liée aux observations recueillies pour déterminer

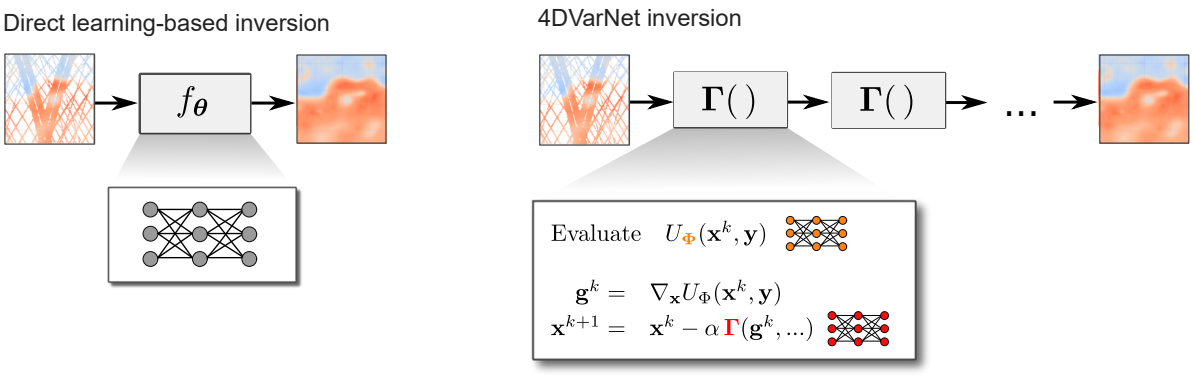


FIGURE B.1 – Direct inversion and 4DVarNet schemes.

de manière optimale l'état de l'océan ou de l'atmosphère.

D'une manière générale, l'état du système à déterminer, $\mathbf{x}$, évolue selon un processus direct, exprimé par l'opérateur $\mathcal{M}$. La variable d'état peut être partiellement échantillonnée par le processus, également direct, d'observation, via l'opérateur $\mathcal{H}$. Celui-ci renvoie les observations $\mathbf{y}$ de la variable d'état. En termes de formulation *état-espace* continue,

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathcal{M}(t, \mathbf{x}(t)) + \boldsymbol{\eta}(t) \\ \mathbf{y}(t) &= \mathcal{H}(t, \mathbf{x}(t)) + \boldsymbol{\epsilon}(t) \end{cases} \tag{B.1}$$

Les symboles $\boldsymbol{\eta}$ et $\boldsymbol{\epsilon}$ représentent les processus d'erreur de modèle et d'observation. D'une manière générale, l'objectif est de déterminer l'état $\mathbf{x}$ à partir des observations $\mathbf{y}$ et de la connaissance des operateurs directs $\mathcal{H}$ et $\mathcal{M}$. La formulation générique B.1 est adoptée, adaptée au contexte, dans les deux cas analysés dans notre travail. Nous utilisons, au cours de nos simulations, deux schémas d'inversion. Le premier, appelé inversion directe, est basé uniquement sur des modèles d'apprentissage profond. Le second est le cadre 4DVarNet évoqué plus haut. Le reste de cette section présente ces deux approches.

B.3.1 Inversion directe basée sur apprentissage

L'inversion directe implique la mise en œuvre d'un modèle basé sur un réseau neuronal entraîné à produire la reconstruction de la variable d'état à partir des observations d'entrée. Cette méthode a trouvé des applications dans de nombreux domaines [30]. L'approche de l'inversion directe a été utilisée, par exemple, pour les problèmes liés aux produits satellitaires [17], [31].

En utilisant le formalisme adopté ci-dessus, cela s'exprime comme suit

$$\hat{\mathbf{x}} = f_{\boldsymbol{\theta}}(\mathbf{y}) \tag{B.2}$$

L'objet $\hat{\mathbf{x}}$ est la reconstruction de la variable d'état, $f_{\boldsymbol{\theta}}$ est le réseau neuronal paramétré par l'ensemble de paramètres $\boldsymbol{\theta}$. Le schéma d'apprentissage de ce modèle consiste à trouver les paramètres du réseau neuronal en optimisant une fonction de coût d'apprentissage $\mathcal{L}$. Dans les cas que nous proposons, cette fonction estime la distance entre les reconstructions du modèle et les vérités de terrain, appelée $\mathbf{u}$. Formalement

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\hat{\mathbf{x}}, \mathbf{u}) \tag{B.3}$$

Souvent la choix de la fonction de coût est l'erreur quadratique moyen (MSE), pour enforcer la proximité entre les veritées terrains et la réconstruction.

Ce type d'approche permet de représenter l'inversion de manière flexible et sans contraintes physiques. Dans nos analyses, ce schéma est utilisé comme référence pour établir un niveau de performance auquel comparer les résultats d'inversion basés sur le cadre 4DVarNet.

B.3.2 Inversion basée sur le cadre 4DVarNet

Le cadre 4DVarNet s'inspire du schéma d'assimilation de données 4DVar [6]. Dans ce cas, l'inversion est actualisée en maximisant la vraisemblance de la variable d'état compte tenu des observations. Pour l'implémentation numérique, la formulation B.1 est discrétisée. L'opérateur $\mathcal{M}$ est remplacé par l'opérateur de flux Φ qui renvoie l'instance de la variable $\mathbf{x}(t)$ au prochain pas de temps $t+1$. En supposant que les erreurs $\boldsymbol{\eta}$ et $\boldsymbol{\epsilon}$ ont une distribution gaussienne, cela se réduit à la minimisation d'une fonction de coût variationnelle exprimée comme suit

$$U_{\Phi}(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_d \left\| \mathbf{x} - \mathbf{y} \right\|_{\Omega}^2 + \lambda_r \left\| \mathbf{x} - \Phi(\mathbf{x}) \right\|^2 \tag{B.4}$$

Dans cette équation, $\|\cdot\|$ représente la norme L^2 et Ω définit le domaine d'échantillonnage des observations spatio-temporelles. Dans les cas où 4DVarNet est appliqué, les observations sont des observations partielles de la variable d'état. Il suffit donc de représenter l'opérateur d'observation sous la forme d'un masque binaire. Le terme de proximité entre les observations et l'état est donc une distance entre les deux évaluée uniquement dans

les zones du domaine Ω. Les paramètres $\lambda_{\{d,r\}}$ sont des poids pour evaluer l'importance des deux termes.

Dans les schémas classiques de type 4DVar, l'opérateur Φ est implémenté comme un intégrateur ODE, par exemple d'Euler ou de Runge-Kutta [32]. Dans le cas du 4DVar-Net, l'opérateur en question est paramétré avec un réseau neuronal. La fonction de coût variationnel est donc également une fonction qui peut être apprise. En outre, le schéma 4DVarNet implique l'utilisation d'un opérateur neuronal supplémentaire Γ pour paramétrer le solveur qui minimisera, via une procédure de descente de gradient, le coût variationnel B.4 [26], [27]. Cela implique la présence de deux procédures de minimisation basées sur la descente de gradient. La première, pour obtenir la variable d'état $\mathbf{x}$, est exprimée par les relations récursives suivantes

$$\begin{cases} \mathbf{g}^k &= \nabla_{\mathbf{x}} U_{\Phi}(\mathbf{x}^k, \mathbf{y}; \Omega) \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - \alpha\, \Gamma(\mathbf{g}^k) \end{cases} \tag{B.5}$$

où $k = 0, \ldots, K$ désigne l'indice d'itération. Le gradient de coût variationnel $\mathbf{g}$ est facilement obtenu grâce aux capacités de différenciation automatique mises à disposition par les bibliothèques modernes d'apprentissage profond, par exemple Pytorch [33]. Le paramètre α est le pas de mise à jour itératif de $\mathbf{x}$. La deuxième minimisation concerne les paramètres des modèles Φ et Γ. L'expression est analogue à l'équation B.3, où la fonction de coût est évaluée avec la sortie du modèle 4DVarNet $\hat{\mathbf{x}}^K$.

B.4 Estimation temporelle de la vitesse du vent in-situ basée sur l'apprentissage à partir de l'acoustique passive sous-marine

La première contribution concerne la reconstruction locale du vent à la surface de la mer à l'aide de données acoustiques sous-marines passives. Dans la pratique, les capteurs in situ sont exposés à l'environnement dans lequel ils sont installés. Dans le milieu marin, cet environnement est agressif, lié aux conditions météorologiques, au risque de collision avec des véhicules navals et au vandalisme. Un hydrophone placé sous la surface est mieux protégé contre certains de ces facteurs. Il est donc logique de se demander comment ce type de capteur et d'observations sous-marines peut être utilisé au mieux pour la reconstruction du vent à la surface. L'observation du paysage acoustique sous-marin permet de

surveiller les écosystèmes, les activités humaines et les phénomènes géophysiques. Contrairement à la pratique habituelle, les procédures d'imagerie dans le milieu sous-marin se font par propagation d'ondes acoustiques, et non d'ondes électromagnétiques. La météorologie acoustique, issue des travaux pionniers de Nystuen [34], vise à reconstruire l'état de l'atmosphère juste au-dessus de la surface de la mer à partir des caractéristiques du paysage acoustique sous-marin.

L'objectif de cette étude de cas est de montrer qu'une approche qui fusionne les concepts d'assimilation de données et d'apprentissage profond peut être utilisée pour résoudre de manière flexible et efficace le problème de l'inversion de l'acoustique sous-marine vers le vent à la surface. Dans ce cas, la flexibilité se réfère à la capacité du cadre proposé à résoudre le problème d'inversion indépendamment des modèles empiriques reliant l'acoustique sous-marine et le vent. De tels modèles sont en effet représentés par des lois linéaires qui sont sensibles à la valeur de la vitesse du vent et nécessitent la calibration des paramètres [35]. Une approche basée uniquement sur des méthodes basées sur les données a été proposée en 2021 par Taylor *et al.* [36]. Les auteurs de cet article ont montré comment les modèles d'apprentissage automatique entraîné pour prédire la vitesse du vent à partir de données acoustiques sous-marines réduisent l'erreur de reconstruction du vent de 1.4–2 $\mathrm{m\,s^{-1}}$ dans le cas des modèles empiriques linéaires [35] à environ 1 $\mathrm{m\,s^{-1}}$.

Dans notre analyse, nous approfondissons les travaux de Taylor et de ses collègues en montrant que la prise en compte explicite de la dépendance temporelle du phénomène permet d'améliorer considérablement la reconstruction des vents. Nous montrons également le cas où les valeurs de vent de la réanalyse sont combinées avec des observations acoustiques. Le but de cette étude est de voir quel niveau de performance est attendu dans le cas d'un ensemble de données multi-modales, dans lequel il y a deux sources d'information, représentant la phénoménologie physique et l'information sur l'évolution temporelle pour l'acoustique passive et le vent réanalysé, respectivement.

B.4.1 Jeu de données

Le jeu de données se compose de séquences temporelles de spectres acoustiques et de la vitesse du vent mesurés in situ. Le travail de Pensieri *et al.* [37] détaille le processus de collecte de prétraitement des données telles qu'elles sont acquises par les capteurs. Le vent in-situ est utilisé comme vérité terrain pour notre analyse. Ces observations sont fournies par la bouée météorologique ODAS Italia-1, qui fait partie de l'observatoire W1M3A (Western 1 Mediterranean Moored Multi-sensor Array). Les données de vent et les données

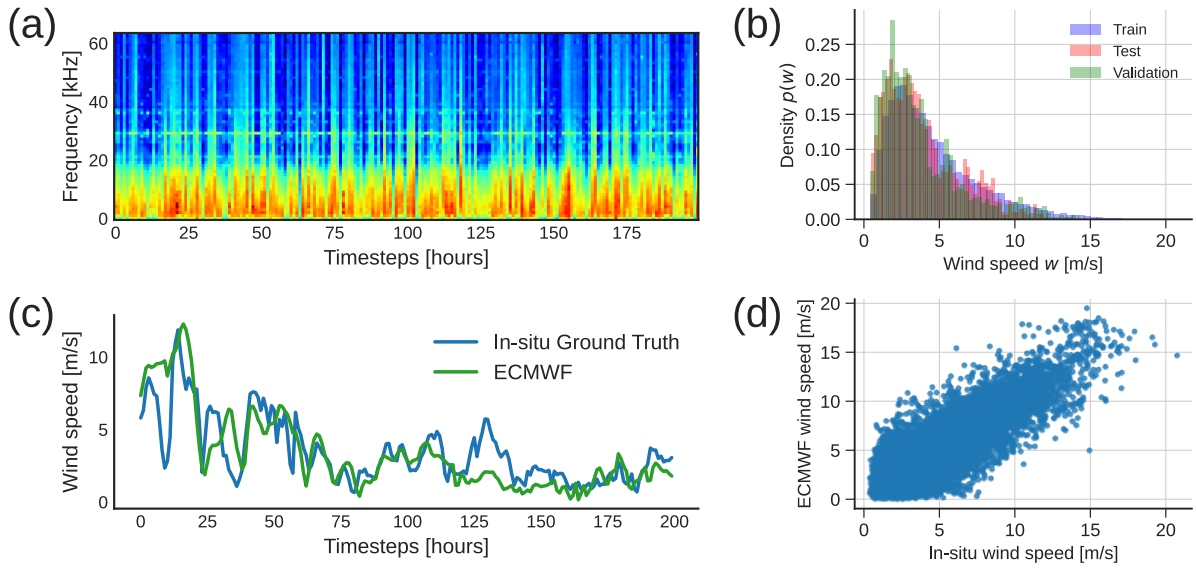


FIGURE B.2 – Représentation graphique de l'ensemble des données. Panneau (a) : série temporelle de spectres acoustiques. Panneau (b) : Isogrammes de la vitesse du vent selon la répartition de l'ensemble de données en ensembles formation-validation-test. Panneau (c) : Séries temporelles du vent in-situ et de l'ERA-interim du ECMWF. Panneau (d) : Diagramme de dispersion entre le vent in-situ et l'ERA-interim du ECMWF.

acoustiques in situ sont préparées sous forme de séries temporelles de 24 heures.

Les valeurs de vitesse de vent réanalysées sont obtenues à partir du catalogue ERA-interim, maintenu par le ECMWF [38]. La combinaison des valeurs de vent réanalysées est facultative et n'est mise en œuvre que dans l'analyse multi-modale. Les réanalyses de la base de données ERA-interim ont une résolution spatiale de 80 km. Nous supposons que la valeur de la cellule correspond à la valeur locale dans le voisinage de l'observatoire W1M3A. Toutes les observations et données ont une résolution temporelle horaire. La figure B.2 présente qualitativement l'ensemble des données utilisées.

B.4.2 Méthodes

Nous prenons les travaux de Taylor *et al.* [36] comme référence. Nous utilisons le meilleur modèle de leur analyse pour la reconstruction du vent comme référence pour mettre en perspective les résultats obtenus avec le cadre que nous proposons. Dans ce cas, le modèle apprenable est entraîné à associer une valeur de vitesse du vent à un spectre. Au lieu de cela, nous proposons l'application d'un modèle d'inversion directe basé sur l'apprentissage profond et le cadre 4DVarNet pour reconstruire les *séries temporelles* du vent

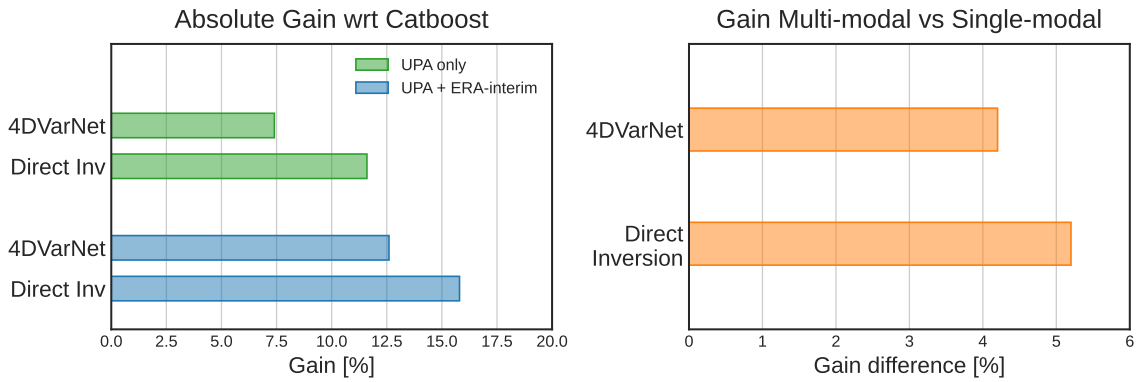


FIGURE B.3 – Le panneau gauche rapporte les gains absolus de chaque modèle utilisé par rapport à la baseline. Le panneau droite rapporte les gains imputables à l'ensemble de données multi-modales, calculés comme des différences entre les instances "UPA+ECMWF" et "UPA only".

à partir des *séries temporelles* de l'acoustique passive. Cette approche qui prend explicitement en compte la dépendance temporelle des processus étudiés est également testée avec la configuration multi-modale de l'ensemble de données. Dans ce dernier cas, l'objectif est de reconstruire des séries temporelles de vent in situ à partir de séries temporelles de vent acoustiques sous-marines et de réanalyse du ECMWF.

B.4.3 Résultats et discussion

Prenons comme référence un modèle qui ne tient pas compte de la dépendance temporelle, proposé par Taylor *et al.* [36]. Ce modèle est appelé CatBoost [39]. Notre premier test consiste à essayer de voir si un modèle basé sur un réseau neuronal peut améliorer le niveau de performance offert par ce benchmark. Le Catboost et le réseau neuronal dans la configuration non dépendante du temps offrent le même niveau de performance, 0.95 $\mathrm{m\,s^{-1}}$. Nous utilisons cette performance comme référence pour comparer les résultats obtenus avec l'inversion directe et 4DVarNet où la dépendance temporelle est prise en compte. Ces modèles sont évalués par l'erreur quadratique moyenne (RMSE) pour la cohérence physique avec les unités. Nous définissons un gagne relatif pour évaluer le pourcentage d'amélioration par rapport à une référence choisie.

La figure B.3 montre les niveaux de performance de l'analyse proposée. Le premier résultat notable (panneau à gauche) concerne la tendance à l'amélioration de la performance numérique de la métrique RMSE en fonction du choix du modèle. Le schéma d'inversion 4DVarNet est dans tous les cas plus puissant que l'inversion directe, ce qui prouve que

la prise en compte de la dépendance temporelle dans l'ensemble de données est cruciale pour une reconstruction efficace du phénomène. Le deuxième résultat (panneau à droite) concerne la multi-modalité de l'ensemble de données. De manière surprenante, le schéma d'inversion directe bénéficie mieux de la double contribution de l'acoustique et de la réanalyse des vents que le schéma 4DVarNet. Ce résultat peut s'expliquer par le choix d'une conception très simple. La multi-modalité est mise en œuvre par une concaténation des deux objets et, dans le cas du modèle 4DVarNet, dans le coût variationnel B.4, il n'y a pas d'indication explicite informant l'inversion de la présence de deux sources distinctes d'informations d'entrée. Nous verrons dans la deuxième étude de cas comment un choix de conception mieux réfléchi peut répondre à la nécessité d'informer l'inversion de sources d'entrée mixtes.

B.5 Reconstruction multi-modale basée sur l'apprentissage des champs spatiaux de vitesse du vent à haute résolution

La deuxième étude de cas analysée développe la première avec l'ambition d'évoquer le problème de l'extension de la dimension spatiale. Dans le premier cas, l'accent était mis sur l'environnement local de la bouée météorologique. Nous sommes maintenant confrontés au problème de la reconstruction du vent à haute résolution sur une zone s'étendant sur plus de 600 km. Le problème est abordé en utilisant des observations partielles de la vitesse du vent en surface comme sources d'entrée. Par exemple, dans un cas typique, pour estimer la vitesse à la surface, il y a les possibilités que nous avons mentionnées et illustrées dans l'introduction : capteurs in-situ, produits de télédétection et produits réanalysés. Pour reprendre la discussion initiale, aucune de ces sources ne décrit complètement le phénomène à toutes les échelles spatio-temporelles. L'objectif ici est de reconstruire les séries temporelles de vent à la surface à la résolution spatiale des produits satellitaires et au taux d'échantillonnage des séries temporelles in-situ. Pour faire écho à la référence faite dans la conclusion de l'étude de cas précédente, nous montrons comment c'est un choix de modèle qui touche au fonctionnement sous-jacent du schéma 4DVarNet qui lui permet de mieux profiter de l'information complémentaire apportée par l'hétérogénéité des sources d'entrée.

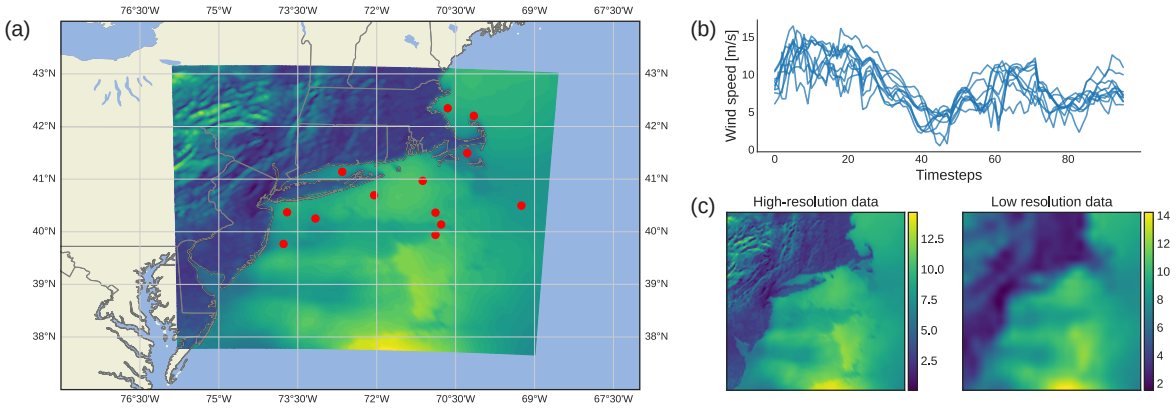


FIGURE B.4 – Description qualitative de l'ensemble des données. Panneau (a) : région géographique sélectionnée et affichage des champs à haute résolution. Panneau (b) : Série temporelles in-situ. Panneau (c) : affichage sur le terrain à basse résolution.

B.5.1 Jeu des données

Nous utilisons des données synthétiques afin de nous assurer de l'exhaustivité et de la compatibilité des informations d'entrée. L'utilisation d'un ensemble de données réelles impliquerait un déploiement de ressources inessentielles pour les analyses que nous voulons proposer dans cette partie du document.

Les données que nous utilisons proviennent du modèle RUWRF [40]. Ce modèle, utilisé pour la circulation atmosphérique à méso-échelle, résout une cellule à une résolution spatiale de 9 km et ensuite une cellule inférieure à 3 km. Cette dernière, c'est-à-dire la résolution du produit original, est supposée être la *haute résolution* du problème. À partir de ces données, nous simulons les observations mentionnées ci-dessus. Pour les produits satellitaires, les champs de vent originaux sont conservés tels qu'ils sont disponibles. Pour les produits réanalysés, les champs originaux sont sous-échantillonnés, pour obtenir la version basse résolution, et réinterpolés sur la grille pour la haute résolution. Les séries temporelles in-situ sont obtenues en conservant uniquement la valeur du pixel, dans le champ complet, à proximité des positions des bouées météorologiques de la constellation NOAA. Le jeu de données résultant est présenté dans la Figure B.4.

B.5.2 Méthodes

Dans le sillage de l'étude de cas précédente, nous utilisons les deux schémas d'inversion directe basés sur l'apprentissage profond et 4DVarNet. Pour évaluer les niveaux de

performance atteints, nous choisissons une inversion de super-résolution directe comme référence. Il est demandé au modèle de reconstruire les champs à haute résolution à partir des champs à basse résolution uniquement. Nous identifions, en plus de la configuration de "super-résolution", trois autres configurations de données d'entrée. Les champs à basse résolution sont toujours présents. Le taux d'échantillonnage de ces données est de 6 heures. Les autres configurations, appelées C1, C2 et C3, se distinguent par la disponibilité d'informations à haute résolution. La configuration C1 dispose de pseudo-observations à haute résolution toutes les 12 heures, fixées à 06h et 18h. La configuration C2 ne comporte que des séries temporelles locales, mais pas de champs spatiaux à haute résolution, et la configuration C3 est complète. Elle comporte des champs à haute résolution comme la configuration C1 et des séries temporelles in situ. Les séries in situ sont toujours horaires. De cette manière, la contribution des séries temporelles à la reconstruction peut être appréciée intuitivement.

Contrairement au cas précédent, le cadre 4DVarNet est mis en place de deux manières distinctes. La première met en œuvre un coût variationnel comme définition générale, Equation B.4. Nous appellons cette variante du 4DVarNet *single-modal*. Dans le second cas, l'objectif est d'informer explicitement l'inversion de l'hétérogénéité des données. Un terme supplémentaire est ajouté au coût variationnel qui concerne les *features maps* des observations et de la variable d'état et non les objets originaux tels qu'ils sont définis dans le sens spatio-temporel. La forme du nouveau coût variationnel est la suivante

$$U_\Phi(\mathbf{x}, \mathbf{y}; \Omega) = \lambda_d \left\|\mathbf{x} - \mathbf{y}\right\|_\Omega^2 + \lambda_d \left\|\psi_\mathbf{x}(\mathbf{x}) - \psi_\mathbf{y}(\mathbf{y})\right\|^2 + \lambda_r \left\|\mathbf{x} - \Phi(\mathbf{x})\right\|^2 \tag{B.6}$$

où $\psi_\mathbf{x}$ et $\psi_\mathbf{y}$ sont des réseaux neuronaux permettant d'extraire les informations des objets $\mathbf{x}$ et $\mathbf{y}$ à un niveau d'abstraction supérieur. Le terme supplémentaire peut être considéré comme une analogie avec les termes *perceptual loss* qui sont souvent utilisés dans l'apprentissage profond [41]. Nous appellons cette version du 4DVarNet *multi-modale*.

B.5.3 Résultats et discussion

La tendance des résultats suit celle de l'étude de cas précédente. La Figure B.5 montre les résultats sous forme de graphique. En particulier, les modèles sélectionnés sont classés par ordre croissant de performance, dans l'ordre l'inversion directe B_1, le cadre d'inversion mono-modale 4DVarNet M_s et la contrepartie multi-modale M_m. Ce résultat, bien qu'intéressant, n'est pas surprenant. L'ensemble des modèles proposés est progressivement plus

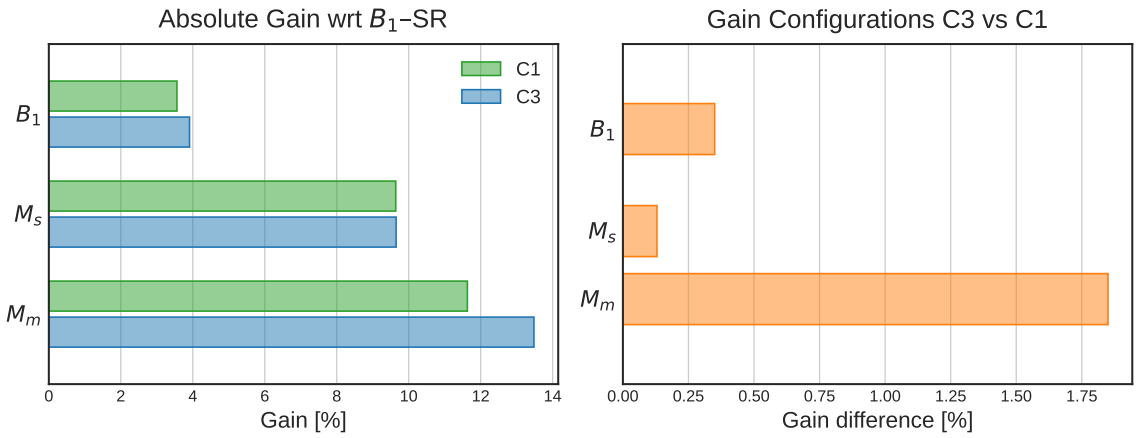


FIGURE B.5 – Panneau de gauche : performances absolues de l'inversion basée sur l'apprentissage profond (B_1), des versions mono-modales et multi-modales des schémas 4DVarNet (M_s et M_m respectivement) par rapport à la baseline de super-résolution B_1-SR. Panneau de droite : différences de gain entre les configurations C3 et C1. Pour rappel : C1 comprend champs à basse résolution chaque 6h et champs à haute résolution chaque 12h. C3, par rapport à C1, comprend également les séries temporelles in-situ.

complexe et donc capable d'exploiter les données en entrée. Le résultat le plus intéressant concerne le gain des différents modèles par rapport à la multi-modalité. Dans l'étude de cas précédente, nous avons constaté que c'est le schéma d'inversion directe qui bénéficie le plus de la multi-modalité, car le coût variationnel qui sous-tend le schéma 4DVarNet n'informe pas la solution de la diversité des données d'entrée. En revanche, dans ce cas, nous constatons que le modèle M_m, mentionné dans l'équation B.6, profite davantage de la multimodalité que l'inversion directe. Ce résultat montre que le choix du modèle est décisif pour l'exploitation efficace des informations complémentaires contenues dans les sources d'entrée. Ces résultats suggèrent comment un schéma de type 4DVarNet est capable de traiter un ensemble multi-modal pour l'inversion d'un problème manifestement mal posé.

B.6 Conclusions et perspectifs

Au cours des analyses proposées dans cette thèse, nous avons constaté deux axes selon lesquels l'approche que nous proposons apporte une amélioration au problème à résoudre. Le premier concerne le choix du modèle. Comme le résument graphiquement les Figures B.3 et B.5, un schéma d'inversion de type 4DVarNet parvient toujours à

surpasser les baselines basées sur l'inversion directe. La raison doit être recherchée dans la formulation sur laquelle le 4DVarNet est développé. Le schéma de type 4DVar est tenu explicitement, et avec lui la prise en compte explicite de la dépendance temporelle des phénomènes considérés. Cet aspect est primordial dans l'étude et la description des phénomènes géophysiques.

Le deuxième axe d'amélioration est attribuable à la mise en œuvre de l'approche multi-modale. Dans les deux études de cas présentées, cette capacité à absorber différentes sources d'information est traitée à différents niveaux de complexité. Dans le premier cas, la simplicité de la mise en œuvre multi-modale a permis au cadre 4DVarNet d'atteindre de meilleurs niveaux de performance, mais ne lui a pas permis de profiter pleinement des informations complémentaires des deux sources d'entrée. Cela était plus évident avec la baseline de l'inversion directe. Dans le second cas, nous avons montré comment le cadre 4DVarNet devait être explicitement informé de la multi-modalité des données d'entrée afin de tirer pleinement parti des informations qu'elles contiennent.

Suite aux résultats encourageants de ces analyses, nous pouvons identifier des directions de recherche futures qui permettront d'approfondir nos contributions.

- **Variables physiques**. Dans nos analyses, à l'exception de la première étude de cas, nous avons principalement travaillé avec la vitesse du vent. L'état de l'atmosphère est donné par plusieurs variables physiques en interaction, telles que la température, la pression, l'humidité, etc. Plusieurs travaux ont étudié la multi-modalité dans le sens de différentes variables [42]-[44]. Nous pensons que la mise en œuvre d'approches similaires peut conduire à une amélioration de la caractérisation de la dynamique de l'état atmosphérique à la surface de la mer.
- **Capacité de prévision**. Nos analyses se sont concentrées sur la *reconstruction* de la vitesse du vent à partir de l'acoustique sous-marine ou d'observations partielles du vent sur un domaine spatial. Ce qui n'a pas été étudié dans le cadre de cette thèse, c'est la prédiction, c'est-à-dire la détermination du vent dans le futur par rapport au dernier instant d'observation. Des exemples notables d'application de méthodes d'apprentissage profond pour la prévision [45]-[47] peuvent être trouvés dans la littérature. D'autres développements de ce qui a été présenté concerneraient également ce type d'analyse.
- **Optimisation des réseaux de capteurs in-situ**. Les mesures in situ constituent un moyen d'observation viable, mais leur déploiement est coûteux par le point de vue économique et d'effort. La configuration expérimentale de la deuxième étude

de cas nous permet d'évaluer la performance de la reconstruction pour des configurations arbitraires de bouées météorologiques ; il suffit de manipuler l'ensemble des données et les positions des bouées pour obtenir une constellation arbitraire de capteurs. Cela permet d'effectuer des analyses de sensibilité afin d'optimiser le réseau de bouées et d'installer ces capteurs de manière optimale. Plusieurs travaux ont traité ce problème [48]-[50]. Dans le sillage de ces travaux, le cadre que nous proposons se positionne comme un outil d'analyse à-priori sur ce sujet.

- **Données réelles**. La première étude de cas est basée sur des données réelles. Cependant, dans l'application des méthodes d'apprentissage profond, la disponibilité de grandes quantités de données est essentielle. Pour qu'un cadre tel que celui présenté dans la deuxième étude de cas soit pleinement opérationnel, il est nécessaire de tester le modèle sur des données réelles. L'apprentissage du modèle sur des données synthétiques est une pratique courante [51]-[53]. L'étape suivante consiste à effectuer un *apprentissage par transfert* [54] pour appliquer le modèle formé avec des données synthétiques à des données réelles. Une étude récente [55] montre comment le passage des données synthétiques aux données réelles dans la phase de test (avec le modèle entraîné) peut donner des résultats positifs.
- **Incertitudes**. Notre analyse se concentre sur la reconstruction de l'état le plus probable et ne prend pas en compte l'évaluation de la composante probabiliste. Le cadre 4DVarNet a été utilisé pour caractériser la distribution de probabilité a posteriori de l'état étant donné les observations [56] et en conjonction avec des techniques de résolution d'équations différentielles stochastiques pour caractériser simultanément les problèmes d'interpolation et de gestion des incertitudes sur les reconstructions [57]. Nous pensons que la mise en œuvre de la composante probabiliste permettrait de fournir aux reconstructions une description quantitative des fluctuations stochastiques et donc de mieux décrire le phénomène.

Liste de références essentielles

[1] J. Gould, B. Sloyan, and M. Visbeck, « In situ ocean observations: a brief history, present status, and future directions », *International Geophysics*, vol. 103, pp. 59–81, 2013.

[2] L. R. Centurioni, J. Turton, R. Lumpkin, *et al.*, « Global in situ observations of essential climate and ocean variables at the air–sea interface », *Frontiers in Marine*

Science, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00419. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00419.

[3] M. Amani, F. Mohseni, N. F. Layegh, *et al.*, « Remote sensing systems for ocean: a review (part 2: active systems) », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1421–1453, 2022.

[4] F. M. Monaldo, C. R. Jackson, and W. G. Pichel, « Seasat to radarsat-2: research to operations », *Oceanography*, vol. 26, *2*, pp. 34–45, 2013.

[5] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, « A tutorial on synthetic aperture radar », *IEEE Geoscience and remote sensing magazine*, vol. 1, *1*, pp. 6–43, 2013.

[6] A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen, « Data assimilation in the geosciences: An overview of methods, issues, and perspectives », *Wiley Interdisciplinary Reviews: Climate Change*, vol. 9, *5*, e535, Sep. 2018. DOI: 10.1002/wcc.535. [Online]. Available: https://hal.science/hal-02905891.

[7] R. N. Bannister, « A review of operational methods of variational and ensemble-variational data assimilation », *Quarterly Journal of the Royal Meteorological Society*, vol. 143, *703*, pp. 607–633, 2017. DOI: https://doi.org/10.1002/qj.2982. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2982. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2982.

[8] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*. Berlin, Heidelberg: Springer-Verlag, 2006, ISBN: 354038300X.

[9] A. Valmassoi, J. D. Keller, D. T. Kleist, *et al.*, « Current challenges and future directions in data assimilation and reanalysis », *Bulletin of the American Meteorological Society*, vol. 104, *4*, E756–E767, 2023.

[10] H. Hersbach, B. Bell, P. Berrisford, *et al.*, « The era5 global reanalysis », *Quarterly Journal of the Royal Meteorological Society*, vol. 146, *730*, pp. 1999–2049, 2020. DOI: https://doi.org/10.1002/qj.3803. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803.

[11] A. Storto, A. Alvera-Azcárate, M. A. Balmaseda, *et al.*, « Ocean reanalyses: recent advances and unsolved challenges », *Frontiers in Marine Science*, vol. 6, 2019, ISSN: 2296-7745. DOI: 10.3389/fmars.2019.00418. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2019.00418.

[12] J. Schmidhuber, « Deep learning in neural networks: an overview », *Neural networks*, vol. 61, pp. 85–117, 2015.

[13] Y. LeCun, Y. Bengio, and G. Hinton, « Deep learning », *Nature*, vol. 521, *7553*, pp. 436–444, May 2015, ISSN: 1476-4687. DOI: 10.1038/nature14539. [Online]. Available: https://doi.org/10.1038/nature14539.

[14] S. Yu and J. Ma, « Deep learning for geophysics: current and future trends », *Reviews of Geophysics*, vol. 59, *3*, e2021RG000742, 2021.

[15] A. Karpatne, I. Ebert-Uphoff, S. Ravela, H. A. Babaie, and V. Kumar, « Machine learning for the geosciences: challenges and opportunities », *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, *8*, pp. 1544–1554, 2018.

[16] K. J. Bergen, P. A. Johnson, M. V. de Hoop, and G. C. Beroza, « Machine learning for data-driven discovery in solid earth geoscience », *Science*, vol. 363, *6433*, 2019.

[17] G. E. Manucharyan, L. Siegelman, and P. Klein, « A deep learning approach to spatiotemporal sea surface height interpolation and estimation of deep currents in geostrophic ocean turbulence », *Journal of Advances in Modeling Earth Systems*, vol. 13, *1*, e2019MS001965, 2021.

[18] K. Stengel, A. Glaws, D. Hettinger, and R. N. King, « Adversarial super-resolution of climatological wind and solar data », *Proceedings of the National Academy of Sciences*, vol. 117, *29*, pp. 16 805–16 815, 2020. DOI: 10.1073/pnas.1918964117. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1918964117. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.1918964117.

[19] T. Bai and P. Tahmasebi, « Accelerating geostatistical modeling using geostatistics-informed machine learning », *Computers & Geosciences*, vol. 146, p. 104 663, 2021.

[20] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

[21] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, « Multimodal machine learning: a survey and taxonomy », *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, *2*, pp. 423–443, 2018.

[22] D. Hong, L. Gao, N. Yokoya, *et al.*, « More diverse means better: multimodal deep learning meets remote-sensing imagery classification », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, *5*, pp. 4340–4354, 2020.

[23] R. Arcucci, J. Zhu, S. Hu, and Y.-K. Guo, « Deep data assimilation: integrating deep learning with data assimilation », *Applied Sciences*, vol. 11, *3*, p. 1114, 2021.

[24] S. Barthélémy, J. Brajard, L. Bertino, and F. Counillon, « Super-resolution data assimilation », *Ocean Dynamics*, vol. 72, *8*, pp. 661–678, 2022.

[25] A. Farchi, M. Bocquet, P. Laloyaux, M. Bonavita, M. Chrust, and Q. Malartic, « Model error correction with data assimilation and machine learning », *in EGU General Assembly Conference Abstracts*, 2022, EGU22–5692.

[26] R. Fablet, B. Chapron, L. Drumetz, E. Mémin, O. Pannekoucke, and F. Rousseau, « Learning variational data assimilation models and solvers », *Journal of Advances in Modeling Earth Systems*, vol. 13, *10*, e2021MS002572, 2021, e2021MS002572 2021MS002572. DOI: https://doi.org/10.1029/2021MS002572. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021MS002572. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021MS002572.

[27] R. Fablet, M. Beauchamp, L. Drumetz, and F. Rousseau, « Joint interpolation and representation learning for irregularly sampled satellite-derived geophysical fields », *Frontiers in Applied Mathematics and Statistics*, vol. 7, 2021, ISSN: 2297-4687. DOI: 10.3389/fams.2021.655224. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fams.2021.655224.

[28] R. Fablet, M. M. Amar, Q. Febvre, M. Beauchamp, and B. Chapron, « End-to-end physics-informed representation learning for satellite ocean remote sensing data: applications to satellite altimetry and sea surface currents », *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. V-3-2021, pp. 295–302, 2021. DOI: 10.5194/isprs-annals-V-3-2021-295-2021. [Online]. Available: https://isprs-annals.copernicus.org/articles/V-3-2021/295/2021/.

[29] R. Snieder and J. Trampert, « Inverse problems in geophysics », *in Wavefield Inversion*, A. Wirgin, Ed., Vienna: Springer Vienna, 1999, pp. 119–190, ISBN: 978-3-7091-2486-4.

[30] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, « Deep learning techniques for inverse problems in imaging », *IEEE Journal on Selected Areas in Information Theory*, vol. 1, *1*, pp. 39–56, 2020.

[31] A. Barth, A. Alvera Azcárate, M. Licer, and J.-M. Beckers, « A convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations (dincae) », *in EGU General Assembly Conference Abstracts*, 2020, p. 9414.

[32] J. R. Cash, « Review paper. efficient numerical methods for the solution of stiff initial-value problems and differential algebraic equations », *Proceedings: Mathematical, Physical and Engineering Sciences*, vol. 459, *2032*, pp. 797–815, 2003, ISSN: 13645021.

[33] A. Paszke, S. Gross, S. Chintala, *et al.*, « Automatic differentiation in pytorch », 2017.

[34] J. A. Nystuen, « Rainfall measurements using underwater ambient noise », *The Journal of the Acoustical Society of America*, vol. 79, *4*, pp. 972–982, 1986. DOI: 10.1121/1.393695.

[35] P. Cauchy, K. J. Heywood, N. D. Merchant, B. Y. Queste, and P. Testor, « Wind speed measured from underwater gliders using passive acoustics », *Journal of Atmospheric and Oceanic Technology*, vol. 35, *12*, pp. 2305–2321, 2018. DOI: https://doi.org/10.1175/JTECH-D-17-0209.1. [Online]. Available: https://journals.ametsoc.org/view/journals/atot/35/12/jtech-d-17-0209.1.xml.

[36] W. O. Taylor, M. N. Anagnostou, D. Cerrai, and E. N. Anagnostou, « Machine learning methods to approximate rainfall and wind from acoustic underwater measurements (february 2020) », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, *4*, pp. 2810–2821, 2021. DOI: 10.1109/TGRS.2020.3007557.

[37] S. Pensieri, R. Bozzano, J. A. Nystuen, E. N. Anagnostou, M. N. Anagnostou, and R. Bechini, « Underwater acoustic measurements to estimate wind and rainfall in the mediterranean sea », *Advances in Meteorology*, vol. 2015, p. 612 512, Apr. 2015, ISSN: 1687-9309. DOI: 10.1155/2015/612512.

[38] D. P. Dee, S. M. Uppala, A. J. Simmons, *et al.*, « The era-interim reanalysis: configuration and performance of the data assimilation system », *Quarterly Journal of the Royal Meteorological Society*, vol. 137, *656*, pp. 553–597, 2011. DOI: https://

doi.org/10.1002/qj.828. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.828. [Online]. Available: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.828.

[39] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, « Catboost: unbiased boosting with categorical features », *in Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, Curran Associates, Inc., 2018.

[40] M. Optis, A. Kumler, G. N. Scott, M. C. Debnath, and P. J. Moriarty, « Validation of ru-wrf, the custom atmospheric mesoscale model of the rutgers center for ocean observing leadership », National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2020.

[41] Y. Liu, H. Chen, Y. Chen, W. Yin, and C. Shen, « Generic perceptual loss for modeling structured output dependencies », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5424–5432.

[42] L. Boussioux, C. Zeng, T. Guénais, and D. Bertsimas, « Hurricane forecasting: a novel multimodal machine learning framework », *Weather and Forecasting*, vol. 37, *6*, pp. 817–831, 2022.

[43] S. Benaïchouche, C. Le Goff, B. Boussidi, F. Rousseau, and R. Fablet, « Multimodal data assimilation of sea surface currents from ais data streams and satellite altimetry using 4dvarnet », Copernicus Meetings, Tech. Rep., 2023.

[44] C. Bai, D. Zhao, M. Zhang, and J. Zhang, « Multimodal information fusion for weather systems and clouds identification from satellite images », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 7333–7345, 2022.

[45] A. G. Salman, B. Kanigoro, and Y. Heryadi, « Weather forecasting using deep learning techniques », *in 2015 international conference on advanced computer science and information systems (ICACSIS)*, Ieee, 2015, pp. 281–285.

[46] A. M. Abdalla, I. H. Ghaith, and A. A. Tamimi, « Deep learning weather forecasting techniques: literature survey », *in 2021 International Conference on Information Technology (ICIT)*, IEEE, 2021, pp. 622–626.

[47] X. Ren, X. Li, K. Ren, *et al.*, « Deep learning-based weather prediction: a survey », *Big Data Research*, vol. 23, p. 100 178, 2021.

[48] N.-H. Kim, J. H. Hwang, J. Cho, and J. S. Kim, « A framework to determine the locations of the environmental monitoring in an estuary of the yellow sea », *Environmental Pollution*, vol. 241, pp. 576–585, 2018.

[49] N.-H. Kim, D. Baek, J.-i. Kwon, J.-Y. Choi, and K.-Y. Heo, « Strategy for additional buoy array installation in operational buoy-observation network in korea », *Ocean Engineering*, vol. 266, p. 112 746, 2022.

[50] S. Liu, M. Song, S. Chen, *et al.*, « An intelligent modeling framework to optimize the spatial layout of ocean moored buoy observing networks », *Frontiers in Marine Science*, vol. 10, p. 1 134 418, 2023.

[51] T. P. Merrifield, D. P. Griffith, S. A. Zamanian, *et al.*, « Synthetic seismic data for training deep learning networks », *Interpretation*, vol. 10, *3*, SE31–SE39, 2022.

[52] W. Liu, B. Luo, and J. Liu, « Synthetic data augmentation using multiscale attention cyclegan for aircraft detection in remote sensing images », *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[53] T. Gupta, P. Zwartjes, U. Bamba, K. Ghosal, and D. K. Gupta, « Near-surface velocity estimation using shear-waves and deep-learning with a u-net trained on synthetic data », *Artificial Intelligence in Geosciences*, vol. 3, pp. 209–224, 2022.

[54] F. Zhuang, Z. Qi, K. Duan, *et al.*, « A comprehensive survey on transfer learning », *Proceedings of the IEEE*, vol. 109, *1*, pp. 43–76, 2020.

[55] Q. Febvre, J. L. Sommer, C. Ubelmann, and R. Fablet, « Training neural mapping schemes for satellite altimetry with simulation data », *arXiv preprint arXiv:2309.14350*, 2023.

[56] N. Lafon, R. Fablet, and P. Naveau, « Uncertainty quantification when learning dynamical models and solvers with variational methods », *Journal of Advances in Modeling Earth Systems*, vol. 15, *11*, e2022MS003446, 2023. DOI: https://doi.org/10.1029/2022MS003446. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2022MS003446. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003446.

[57] M. Beauchamp, R. Fablet, and H. Georgenthum, « Neural spde solver for uncertainty quantification in high-dimensional space-time dynamics », 2023.

COLLEGES SCIENCES
BRETAGNE POUR L'INGENIEUR
LOIRE ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Méthodes IA multimodales dans des contextes d'observation océanographique et de surveillance maritime multi-capteurs hétérogènes

Mot clés : Apprentissage Machine multi-modal, Données océaniques hétérogènes, Vitesse du vent à la surface de la mèr

Résumé : Cette thèse vise à étudier l'utilisation simultanée d'ensembles de données océaniques hétérogènes afin d'améliorer les performances des modèles prédictifs utilisés dans les domaines scientifiques et opérationnels pour la simulation et l'analyse de l'océan et du milieu marin.

Deux études de cas distinctes ont été explorées au cours des travaux de thèse. La première étude se concentre sur l'estimation locale de la vitesse du vent à la surface de la mer à partir de mesures du paysage sonore sous-marin et de produits de modèles atmosphériques. La deuxième étude considère l'extension spatiale du problème et l'utilisation d'observations à différentes échelles et résolutions spatiales, depuis les pseudo-observations simulant des images satellites jusqu'aux séries temporelles mesurées par des infrastructures in-situ.

Le thème récurrent de ces recherches est la multi-modalité des données introduites dans le modèle. En d'autres termes, dans quelle mesure et comment le modèle prédictif peut bénéficier de l'utilisation de canaux d'information spatio-temporels hétérogènes. L'outil méthodologique privilégié est un système de simulation basé sur l'assimilation variationnelle des données et les concepts d'apprentissage profond.

Title: Multi-modal AI methods in the context of heterogeneous oceanic observations and multi-sensor maritime surveillance

Keywords: Multi-modal machine learning, Heterogeneous oceanic data, Wind speed at the sea surface

Abstract: The aim of this thesis is to study the simultaneous use of heterogeneous ocean datasets to improve the performance of predictive models used in scientific and operational fields for the simulation and analysis of the ocean and marine environment.

Two distinct case studies were explored in the course of the thesis work. The first study focuses on the local estimation of wind speed at the sea surface from underwater soundscape measurements and atmospheric model products. The second study considers the spatial extension of the problem and the use of observations at different scales and spatial resolutions, from pseudo-observations simulating satellite images to time series measured by in-situ infrastructures.

The recurring theme of these investigations is the multi-modality of the data fed into the model. That is, to what extent and how the predictive model can benefit from the use of spatio-temporally heterogeneous information channels. The preferred methodological tool is a simulation system based on variational data assimilation and deep learning concepts.