Bayesian computation with Plug & Play priors for inverse problems in imaging sciences

Rémi Laumont

Université Paris Cité

École doctorale de Sciences Mathématiques de Paris Centre (ED 386)

Laboratoire : Mathématiques Appliquées à Paris 5, MAP5-UMR 8145 CNRS

THÈSE DE DOCTORAT

Spécialité : Mathématiques appliquées

par

Rémi Laumont

Bayesian computation with Plug & Play priors for inverse problems in imaging sciences

dirigée par

Andrés ALMANSA & Julie DELON

Présentée et soutenue publiquement le 30 novembre 2022 devant le jury composé de :

Andrés ALMANSA	DR, CNRS-Université Paris Cité	Co-directeur
Pierre CHAINAIS	PU, Ecole Centrale de Lille	Examinateur
Emilie CHOUZENOUX	CR, INRIA Saclay	Examinatrice
Valentin De Bortoli	CR, CNRS-ENS	Examinateur
Julie DELON	PU, Université Paris Cité	Co-directrice
Marcelo PEREYRA	MdC, Heriot Watt University	Co-encadrant
Gabriele STEIDL	PU, Technische Universität Berlin	Rapporteure
Pierre WEISS	CR, CNRS-Institut de Mathématiques de Toulouse	Rapporteur

Président du jury : Pierre CHAINAIS

Après avis des rapporteurs : Gabriele STEIDL
Pierre WEISS

MAP5 - Université de Paris Cité
45 rue des Saints Pères
75006 Paris

Abstract

This thesis manuscript is devoted to the study of Plug & Play methods applied to inverse problems encountered in image restoration. Since the work of Venkatakrishnan et al. (2013) in 2013, Plug & Play (PnP) methods are often applied for image restoration in a Bayesian context. These methods aim at computing Minimum Mean Square Error (MMSE) or Maximum A Posteriori (MAP) for inverse problems in imaging by combining an explicit likelihood and an implicit a-priori defined by a denoising algorithm. In the literature, PnP methods differ mainly in the iterative scheme used for both optimization and sampling. In the case of optimization algorithms, recent works guarantee the convergence to a fixed point of a certain operator, fixed point which is not necessarily the MAP. In the case of sampling algorithms in the literature, there is no evidence of convergence. Moreover, there are still important open questions concerning the correct definition of the underlying Bayesian models or the computed estimators, as well as their regularity properties, necessary to ensure the stability of the numerical scheme. The aim of this thesis is to develop simple but efficient restoration methods while answering some of these questions. The existence and nature of MAP and MMSE estimators for PnP prior is therefore a first line of study. Three methods with convergence results are then presented, PnP-SGD for MAP estimation and PnP-ULA and PPnP-ULA for sampling. A particular interest is given to denoisers encoded by deep neural networks. The efficiency of these methods is demonstrated on classical image restoration problems such as denoising, deblurring or interpolation. In addition to allowing the estimation of MMSE, sampling makes possible the quantification of uncertainties, which is crucial in domains such as biomedical imaging. Lastly, the influence of the denoiser on the posterior is investigated and a comparison between the Bayesian probabilities reported by the model and the frequentist probabilities arising from a large number of repetitions of an experiment is drawn.

Keywords: inverse problems, Plug & Play methods, Langevin algorithms, Markov chain, Monte-Carlo methods, stochastic gradient descent, denoising, deblurring, interpolation.

Résumé

Ce manuscrit de thèse est consacré à l'étude des méthodes Plug & Play appliquées à des problèmes inverses rencontrés en restauration d'images. Depuis les travaux de Venkatakrishnan et al. (2013) en 2013, les méthodes Plug & Play (PnP) sont souvent appliquées pour la restauration d'image dans un contexte Bayésien. Ces méthodes visent à calculer les estimateurs Minimum Mean Square Error (MMSE) ou Maximum A Posteriori (MAP) pour des problèmes inverses en imagerie en combinant une vraisemblance explicite et un a-priori implicite défini par un algorithme de débruitage. Dans la littérature, les méthodes PnP diffèrent principalement par le schéma itératif utilisé que cela soit pour l'optimisation ou l'échantillonnage. Dans le cas des algorithmes d'optimisation, des travaux récents garantissent la convergence vers un point fixe d'un certain opérateur, point fixe qui n'est pas nécessairement le MAP. Dans le cas des algorithmes d'échantillonnage de la littérature, il n'existe pas de preuves de convergence. Par ailleurs, il reste d'importantes questions ouvertes portant sur la bonne définition des modèles Bayésiens sous-jacents ou encore des estimateurs calculés, ainsi que leurs propriétés de régularité, nécessaires pour assurer la stabilité du schéma numérique. Le but de cette thèse est de développer des méthodes de restauration simples mais efficaces tout en répondant à ces interrogations. L'existence et la nature des estimateurs MAP et MMSE pour des a-priori PnP constitue donc un premier axe d'étude. Deux méthodes avec des résultats de convergence sont alors présentées, PnP-SGD pour l'estimation du MAP et PnP-ULA pour l'échantillonnage. Un intérêt particulier est porté aux débruiteurs encodés par des réseaux de neurones profonds. L'efficacité de ces méthodes est démontrée sur des problèmes classiques de restauration d'image tels le débruitage, le défloutage ou l'interpolation. En plus de permettre l'estimation du MMSE, l'échantillonnage rend possible la quantification d'incertitudes, ce qui est crucial dans des domaines tels que l'imagerie biomédicale. Enfin, l'influence du débruiteur sur l'a-posteriori estimée est questionnée et une comparaison entre les probabilités données par notre modèle et les probabilités fréquentistes provenant d'un grand nombre d'expériences est faite.

Mots-Clefs : problèmes inverses, méthodes Plug & Play, algorithmes de Langevin, chaîne de Markov, méthodes de Monte-Carlo, descente de gradient stochastique, débruitage, défloutage, interpolation.

Remerciements

Je me revois encore, moi le jeune stagiaire, parcourant les différents manuscrits du bureau. En bon scientifique, j'allais directement à cette section. Qu'y trouver ? Des émotions plus que de raison, l'esprit d'un bureau, l'âme d'un labo. A l'heure de son écriture je la trouve d'autant plus savoureuse que c'est elle que j'écris en dernier. Pourtant chaque jour j'y ai toujours un peu pensé.

Je tiens tout d'abord à remercier mes directeurs et ma directrice. Andrés, Julie et Marcelo, avant de parler de travail, j'aimerais vous dire que vous rencontrer, partager des moments avec vous, ça a été très chouette. Andrés, si j'ai aimé parler science avec toi, je me souviendrai aussi de ces réunions parfois interminables où tu prenais le temps de m'aider à surmonter les difficultés rencontrés (les problèmes d'instabilités numériques, Jean-Zay, etc, ...). Même si parfois seul mon estomac était encore capable de s'exprimer, ces moments témoignent à mes yeux de ton investissement pour ce projet et de ta profonde humanité. Car au-delà de la science, c'est aussi des moments où j'avais besoin d'être rassuré. Et si je repartais parfois de ton bureau sans savoir quoi faire, c'était toujours avec le sentiment d'avoir été écouté. Andrés ne change pas.

Julie, sans toi je n'aurais jamais effectué cette thèse. C'est en assistant à ton cours d'Imagerie Numérique "histoire de réunir les crédits en Modélisation" que j'ai compris que les sciences de l'imagerie ça pouvait être cool. On ne va pas se mentir je t'ai trouvée encore plus cool. Cette manière simple et humble d'expliquer des trucs compliqués (cf le transport optimal pour la colorisation, la descente proximale)... J'adore cette manière que tu as de prendre le stylo et de revenir aux bases pour mieux comprendre. Au-delà de tout ça, il y a aussi ton sens de l'écoute, ta patience. Je me souviendrai toujours de cette conversation avec Andrés sur ta terrasse : cet air frais, ce soleil léger et cette atmosphère pesante, surtout. Puis il y avait tes mots, simples et justes. Les regards compréhensifs d'Andrés. Ce jour-là j'ai appris à ne pas avoir honte. "La thèse c'est dur, la vie aussi". Je suis ému rien qu'en lisant cette phrase. Merci.

Marcelo, même si une mer nous sépare, tu as toujours su être là quand j'avais besoin de toi même si c'était l'affaire de cinq minutes. Puis, tu as cette manière d'expliquer les choses sans te prendre la tête. En outre je te trouve très créatif et me demande combien de papiers tu lis par semaine. Sinon j'adore t'entendre raconter des histoires. J'ai l'impression de connaître un peu l'Argentine sans jamais y avoir été. Je me dis qu'elle est là la solution à l'avion. J'espère continuer à collaborer avec toi.

Je tiens aussi à remercier Valentin, un super mentor, qui démystifie les mathématiques les plus complexes. En parler avec toi les rend presque humaines.

Je voulais remercier Gabriele Steidl et Pierre Weiss d'avoir accepté d'être mes rapporteurs et d'avoir été si justes et bienveillants. Vos mots m'ont fait du bien. Je remercie par ailleurs Pierre Chainais et Emilie Chouzenoux pour avoir accepté d'être dans mon jury. Nos conversations m'ont permis d'avoir un oeil presque nouveau sur mes travaux. J'espère pou-

voir échanger à nouveau avec vous dans un future proche et sinon je peux vous assurer que je vais garder un oeil sur vos papiers !

J'aimerais remercier tous les membres du MAP5. Claire L. me disait que le MAP5 c'était un peu comme une grande famille et je crois qu'elle a raison. Je tenais à remercier les permanents avec qui j'ai souvent pu échanger au cours des déjeuners, dans les couloirs ou encore à la machine à café. Comme je suis pas très organisé et que ce n'est pas aujourd'hui que ça va commencer, j'ai notamment une pensée particulière pour Joan aka l'affreux Jojo, l'homme bronzé toujours prêt à "cloper", Lionel, qui m'a appris à me raser, Antoine M. et nos discussions sur les sports de combat, Antoine C. et nos conversations "matinales" sur le cinéma et la montagne, Rapahël et nos conversations posées, Manon et sa charlotte "minute", Rémy et sa gentillesse, Sébastien, toujours prêt à déconner, Marie qui perd tout sauf sa bonne humeur légendaire, Sara et ses conseils en matière de séries et Nathalie et son son attitude enjouée et positive même au cours des sales journées. J'ai encore plus scpécifiquement quelques mots pour la fine équipe de MC2 avec qui je me suis parfois arraché les cheveux. Merci à Marcela, Georges, Sylvain, merci pour votre gentillesse et votre compréhension. Merci à Irene, une super cheffe d'équipe qui a souvent su me rassurer quand le bateau tanguait et m'a beaucoup apporté. Sur le plan de la rigueur, t'es la meilleure. Merci à Quentin aussi. D'ici 5 ans on se fait une cyclo ensemble ! Merci à Flora (et Claire) pour l'animation lors des réunions Zoom parfois longues. Je suis sur qu'avec Elise, ça sera encore mieux !

J'aimerais prendre le temps de remercier Fabienne et Anne, deux incroyables directrices de ce laboratoire, toujours prêtes à monter au front pour nous sortir de périlleuses situations. La Corrèze peut être fière !

Je voulais avoir quelques mots pour toi, Marie-Hélène, pour te dire que j'avais adoré nos discussions régulières de bon matin. Me plaindre avec toi a toujours fait partie de mes petits plaisirs. Toujours rassurante, toujours compréhensive, toujours de bons conseils. T'es la meilleure ! Un jour on se retrouvera pour prendre un verre au Cap-Vert.

Bon, bon, bon... Je ne sais pas trop comment continuer. Peut-être par là où tout a commencé. Le 725-C1. Plus qu'un bureau, un foyer où j'ai fait de belles rencontres qui vont durer encore après ces trois années. C'est ici que j'ai pu me faire une petite place (au soleil) au sein de ce (petit) labo. Mes premiers mots iront à Claire L., ma voisine de bureau, toujours prête à apporter sa pierre à l'édifice de mes questions existentielles. Claire, plus qu'une co-bureau, une amie, tu as été une raison de venir au labo, un rayon de soleil dans l'obscurité. Et une super présidente de labo aussi ! Merci à toi Vinc' The Prince, pas le moins bavard, pas le moins sympa non plus. Toujours prêt à me filer un coup de main quand je n'y comprenais rien, merci vraiment. Mais arrête de te toucher la nouille stp ! Merci à Newton, le chavant. Je ne sais pas comment tu fais pour vivre avec Pierre R. dont le véritable mérite aura été de pouvoir acheter des places pour un match de Totenham sur le réseau du labo (en fait je ne suis pas sûr mais je l'aime bien). Merci à Pierre C., mon camarade de clavier le temps d'un simple stage. Mec, on ira au sommet, c'est sûr ! Une pensée aussi à Anton, mon camarade de promotion qui va bientôt soutenir un truc très beau, Antoine S fier représentant du "M-A-P 5" en soirée et all over the world (jte kiff), Loïc le businessman si dur en affaire, Mario G. et son maté, Alexandre S-D. qui saura peut-être un jour écrire mon prénom, Noura la killeuse du laser-game, Thaïs pain d'épice et Alasadair "mon prof" sans qui je n'aurais jamais pu installer cuda sans doute. Merci à Sonia, qui n'a fait que passer mais qui m'a montré ce que c'était roter. J'adore ton honneteté bien intentionnée. Une pensée aussi aux petits nouveaux Eloi, Guillaume et Alexander. C'est maintenant votre bureau ! Je voulais clôturer ce paragraphe en évoquant Zoé. Pouvoir parler aussi librement de Top Chef, Koh-Lanta, tricot et crochet, gastronomie, de la vie en somme, montre à quel

point t'es une super amie. Merci.

Si je me suis toujours senti bien dans le 725-C1, je ne peux pas nier que j'ai toujours adoré fouiner dans le 725-A1 aka la salle de jeu ou encore la cour de récréation/des miracles. Mon pote Ousmane Sacko, meilleur Footix du labo, pouvait y mettre un joyeux bordel. Du 725-C1, impossible de se concentrer. En même temps je dois dire qu'il avait été engrainé par une bonne de jeunes dévergondés tels que Alessandro, Andréa, Arthur et Juliana. Alessandro j'ai aimé pouvoir parler de tout avec toi quand ça allait et quand ça n'allait pas. Arthur, mec, ce que j'ai aimé pouvoir parler Tour de France avec toi. Toi t'es un vrai gars. Je me demande comment tu as fait pour calculer tes espérances dans tout ce bazar. Au-delà de ça, j'ai aimé pouvoir échangé sur tout et son contraire sans jugement tout en se marrant. Hâte qu'on se revoie. Juliana, la gentillesse et la douceur incarnées. Je me sens privilégié de vous avoir pour amis et je vous aime vraiment. Sonia, Mehdi B., Diala Ivan, Laurent et Sinda je ne vous oublie pas. Je suis sur que vous façonnerez le bureau que vous désirez. Mehdi B. encore merci pour ton aide sur la fin de thèse, ça m'a vraiment retiré un poids. Tes conseils sur les animés sont toujours les bienvenus.

Je n'oublie pas non plus le bureau 750, sans contest le plus exotique de tous les bureaux car le plus éloigné géographiquement. J'ai toujours été si bien accueilli chez vous Florian, Antoine M., Charlie (Claude est moins fort que Ugo, ouvre les yeux !), Safa, Yen, Chabane et Adrien, alors merci. Antoine, c'est toujours un petit plaisir de pouvoir parler soirée et râler avec toi.

Enfin, j'aimerais remercier mon équipe de pionniers (Rémi B., Ariane et Mariem), celle avec qui je suis allé explorer la faculté pour défrîcher le bureau 814-D. Mariem, j'ai toujours aimé tes conseils, ta douceur, tes observations, tes histoires. Ariane aka Mémére, qu'est-ce qu'on se marre avec toi. T'es la plus grosse gamine du labo, un élément moteur des sorties (et des conneries) et une super copine. Rémi B. je te trouve sympa mais un peu limité. Du coup, c'est cool parce que je me sens intelligent à côté. J'ai aussi une pensée pour Keanu, Angie et Léonard (très cool de te revoir après 10 ans).

A ce stade, j'ai aussi une pensée pour La Fine Equipe des Pieds Sous la Table, évidemment. Je me demande si vous avez cru que j'allais vous oublier. Commençons par Pierre-Louis, le rédacteur en chef, à l'origine de ce projet un peu fou. Pierre-Louis, l'impact player des troisièmes mi-temps, le siesteur le plus connu de tout le bâtiment. C'est aussi un vrai copain avec qui on peut parler de tout et qui toujours prend le temps même débordé (par l'ANRT surtout). Ta visite pour ma soutenance m'a tellement touché. J'ai tellement hâte d'aller vous rendre visite avec Chloé dans le sud. Puis il y a Rémi B., aka Magic, sans toi mon gars c'est sûr je ne serais jamais venu à bout de ce manuscrit. Même si cet été aura été pénible, sans toi il aurait été affreux. J'ai le sentiment d'avoir bravé les tempêtes, de pouvoir passer à travers n'importe quoi avec toi. J'aime toujours autant nos appels et promis j'essaierai d'être là pour toi en Juillet. Merci à toi. Les gars on se retrouve à Lyon pour notre premier enregistrement !

J'ai aussi une pensée émue pour mes amis/amies d'école d'ingénieur pour qui j'ai une grande admiration. Jamais dans le jugement moralisateur, toujours dans l'expérience libératrice. Des piliers de bar qui ne se font jamais dépasser par la marée. Je pense bien sûr à Simon, Emma, Julien, Matthieu, Mareva, Cécile, Boupi, Alice, Théo, Joran. Charley, aussi, un aventurier ordinaire parti se battre pour ce qui compte vraiment. Merci de me faire relativiser tout ce que je fais. Même si on se fait pas beaucoup vu, Khaduche, je pense grave à toi. Je me rappelle nos soirées de révisions avec Maxime dans ton appartement. On se marrait tellement. Alexandre M., la force tranquille, le moine shaolin un peu débile. Je suis tellement heureux qu'on arrive à cultiver notre relation après tant d'années. Je la trouve belle et j'en suis fier. Mehdi, juste merci d'être pas là et pas que pour les mangas. Que tu décroches alors

que je suis au bout du rouleau et que t'es en plein boulot compte beaucoup pour moi. T'es mon roc. Avec Adva, un bel ami, votre appartement est une deuxième maison, un pied à terre pour la famille. J'ai aussi une pensée particulière pour Elliot avec qui tout a commencé et Thomas F. un altruiste forcené de la route.

J'ai une pensée toute particulière pour Maxime. Ca va faire 10 ans qu'on se connaît et depuis 10 ans, q'on ne se lâche pas. Quand je pense à nous, je pense à de deux gros tocards toujours prêts à inventer de nouvelles histoires, à s'attaquer aux questions les plus existentielles qui soient (la réponse est souvent dans la question en fait). Quand je pense à toi, je vois un ami fidèle et sûr. Un vrai ami qui me pousse toujours à me dépasser, gentiment, patiemment. On trouvera notre voie, c'est sûr !

J'ai encore une pensée pour mes amis/amies de Munich. Vous êtes une merveilleuse bande de copains/copines et nous voir encore actifs/actives à Paris et ailleurs me fait tellement plaisir. Avec vous je me sens vraiment bien. J'adore nos barbecues à la poêle, nos raclettes aux pates, nos découvertes auvergnates. Merci à Manon C. (quelle humanité !), Manon G. (quel humour !), Coralie, Hélène (quelle coloc !), Romain (que de profondeur dans nos conversations !), Thibaud (quelle empathie !). J'ai une pensée particulière pour mon amie Josépha. Tu me manques j'avoue et parfois c'est frustrant de ne pas te voir plus. Mais j'adore nos appels impromptus. Et que dire de Clément D., le dresseur de chats/chattes aux converses rouges et aux t-shirts de prisonnier ? Vivement nos soirées charentaises (au McDo ?) avec Eléonore (que j'adore) et Rooney (!!!!!). Hâte de rencontrer Poppy ! J'ai aussi une pensée pour Pierre que j'aimerais voir plus aujourd'hui. Merci pour l'astuce de l'éclairage. Et Anne, ma partenaire de bibliothèque à Munich, mais surtout une amie pour la vie. J'aimerais qu'on se voie plus, tellement plus. Et Nini, Madame jamais en avance ! Ta paire de ciseaux retrouvée n'a pas aidé. T'es trop drôle et grâce à toi je vais parfois à la MEP (cher lecteur si tu ne connais pas, tu as gagné le mépris de Nina). Coucou Hadrien !

Anne-Katharina, j'espère que tu crois que je t'ai oubliée parce qu'il est évident que non. Parce que tu dois faire partie des noms auxquels je pense depuis le début, ou plutôt toujours (comme Dédé avec Mbappé). On s'aide à se réaliser mutuellement, à tatons. C'est beau. On trouve quand même le temps d'aborder des questions de fond avec le sourire (Quels sont les meilleurs raviolis Giovanni Rana ?). Pour tout ça, je te suis sincérement reconnaissant. Alors merci. Et à bientôt !

Un petit mot aussi pour mes camarades de randonnée Jean-Baptiste et Clément P., amateurs de croquettes. Avec vous souffrir n'a jamais été aussi beau. Hâte de partir avec vous pour de nouvelles aventures. C'est quoi le prochain plan ? La traversée du Vercors ? la descente de la Loire en canoë ? Une bière dans le 18$^{\text{ème}}$?

Nous approchons de la fin et j'ai l'impression que cette section n'aboutira malheureusement pas à un petit article de conférence... A défaut, je continue sur ma lancée et en profite pour remercier mes amis de lycée. Alexandre K., Edouard, Gauthier, Louise, Tom et Violette, c'est vrai qu'on sait poser le cerveau et s'amuser tous ensemble. A votre contact, je me vois évoluer. J'ai une pensée pour Quiz avec qui je partage des "déjeuners" d'une rare intensité (on devrait plutôt parler de demi-journées je pense). On a encore tant de choses à analyser, décortiquer ensemble, ça promet ! J'ai une pensée pour Adélie, ma fidèle partenaire de TP, qui a toujours su m'accompagner, me rassurer. Parfois je me dis que c'est moi qui ai deux ans de moins... Merci aussi à Edith, que je vois peu mais qui sait être là quand j'en ai besoin. Même si on a bien évolué, je suis heureux de t'avoir pour amie après toutes ces années. Léa, je ne sais pas trop quoi dire tellement j'ai à raconter. Te retrouver après toutes ces années m'a fait tellement de bien. J'avais oublié. Avec toi, je peux être moi et ça, ça n'a pas de prix. Merci. Je n'oublie pas Lucie, mon petit biscuit, avec qui j'aime partager un thé, prendre le goûter et parfois le petit-déjeuner. Une amie qui chaque jour me fait

prendre conscience qui je suis. J'ai aussi une pensée pour Aurélie, Lorène et A-P Gignac. Nos retrouvailles sont trop rares mais toujours mémorables. Un coucou à Clara Zaza parce que je ne sais pas où parler de toi.

Je tiens à remercier la famille Paetzold pour son chaleureux accueil et ma famille bien sûr. Sans votre soutien, tout ça n'aurait jamais été possible. Merci à mon père pour son goût de l'effort dans le sport. C'est une passion, parfois dévorante, mais qui me permet de m'évader. Aujourd'hui je crois avoir trouver la manière de m'exprimer dedans, mais sans toi papa, cela n'aurait pas été possible. Merci à ma soeur, Céline, que j'admire. Si j'ai fait une thèse, tu n'y es pas étrangère. J'aime nos rares discussions au cours desquelles on refait le monde (et la famille). Un océan et un continent ne sont pas assez pour nous séparer. Merci à ma mère. Pour tout, pour rien, partout, pour toujours. Aimante, attentionnée, inquiète, sensible, déterminée, courageuse. La meilleure part de moi. On en a fait du chemin depuis mon exposé de troisième sur le tungstène. Je vous aime. J'ai une pensée pour mon grand-père (forcément) parti trop tôt et ma cousine Héloïse que je suis heureux de découvrir un peu chaque jour. Heureux aussi pour François et le petit Samuel.

Et puis et puis et puis il y a Clara qui est belle comme un soleil et qui m'aime pareil que moi j'aime Clara même qu´on se dit souvent qu'on aura une maison avec des tas de fenêtres avec presque pas de murs et qu'on vivra dedans et qu´il fera bon y être et que si c'est pas sûr c'est quand même peut-être. Clara, c'est une ode à l'amour un peu tous les jours. Clara ce sont ces rencontres ambiguës si plaisantes, ces escapades romantiques décidées sur un coup de tête, ces promesses lancées sur un morceau d'oreiller, ce sentiment d'immortalité, ces rires qui prouvent le silence et ces jeux toujours plus déjantés. Clara ce sont aussi ces réveils matinaux difficiles, ces corvées de ménage partagées, ces discussions parfois pénibles une fois la nuit tombée, des pleurs parfois. C'est cette envie de rester avec toi pour et malgré tout ça. C'est ce que j'appelle Amour. Merci à toi, qui me soutiens et m'aides à traverser les épreuves de la vie comme personne. Parce que, oui la vie est dure mais elle est aussi très belle. Surtout avec toi. Je t'aime. Je te rappelle aussi que tu es Bombur au cas où tu l'aurais oublié. Je te le rappellerai sans doute ce soir en rentrant ou demain au réveil.

Ce manuscrit est dédié à ma mère, qui a toujours su m'accompagner. J'aimerais que cela ne s'arrête jamais.

Il est aussi dédié à mon ami Baptiste L. Ce manuscrit, c'est une sorte de mausolée, ma manière de ne jamais t'oublier.

Contents

List of Figures

List of Tables

1

Introduction (en français)

1.1 Contexte

De nombreux problèmes nécessitent d'inférer un signal à partir de données partiellement observées et souvent bruitées. Typiquement, ces problèmes apparaissent en ingénierie biomédicale, en géophysique, en astronomie ou encore en finance. Ces problèmes sont appelés *problèmes inverses* car les résoudre revient à inverser le processus d'observation de la quantité d'intérêt afin de recupérer un signal à partir des données observées. De tels problèmes sont souvent modélisés par l'Equation (1.1)

$$y = \mathrm{A}x + n \,, \tag{1.1}$$

où y correspond aux données observées, x à la quantité que nous souhaitons inférer, A à l'opérateur de dégradation, qui modélise le processus d'observation en l'absence de bruit et n au bruit de mesure.

Les problèmes inverses rencontrés sont souvent *mal-posés*, c'est-à-dire qu'une petite pertubation au moment de l'observation peut produire de grandes erreurs dans le signal que l'on souhaite retrouver ou qu'il existe plusieurs signaux possibles consistants avec les données observées. Tenir compte de ce caractère mal-posé est crucial, spécifiquement dans des domaines où des décisions sont prises à partir du signal restauré, comme en imagerie médicale. Par ailleurs, les problèmes inverses en imagerie sont souvent des problèmes en haute dimension, ce qui rend leur résolution d'autant plus difficile.

1.2 Position du problème

Dans beaucoup d'applications, les signaux admissibles appartiennent à un petit sous-ensemble de l'espace ambiant. Par exemple en imagerie, il est communément supposé que les images naturelles sont concentrées sur une variété de l'espace ambiant (Fefferman et al., 2016). Ceci a motivé le développement de régularisateurs qui apportent une connaissance a priori sur le signal que l'on souhaite restaurer.

Les approches variationnelles sont très populaires pour résoudre les problèmes inverses en imagerie. Elles commencent par construire une fonction objectif que l'on cherche à minimiser. Cette fonction objectif résulte généralement de la somme de deux termes, le *terme d'attache aux données* qui mesure la distance entre l'observation y et la solution à notre problème et le *terme de régularisation* qui promeut des images avec certaines propriétés recherchées ou qui se trouvent dans le voisinage d'une variété. Comme exemples de régularisateurs classiques, on peut citer la variation totale (TV) (Rudin et al., 1992) qui favorise des images avec des gradients parcimonieux (en particulier les images constantes par morceaux), ou encore les régularisateurs favorisant des solutions parcimonieuses dans des domaines transformés tels que les bases d'ondelettes (Donoho and Johnstone, 1994). Bien que beaucoup de progrés aient été réalisés dans ce domaine, les régularisateurs explicites ne parviennent généralement pas à capturer la complexité des images naturelles. De plus, ces régularisateurs sont souvent convexes afin de tirer profit des méthodes d'optimisation idoines, ce qui peut limiter l'expressivité des modèles associés.

Les problèmes inverses mal posés peuvent également être abordés dans un cadre bayésien. Dans ce cadre, les données observées et le signal à reconstruire sont considérés comme des réalisations de variables aléatoires. Dans le paradigme Bayésien, nous cherchons à déterminer la distribution a posteriori, *i.e.* la distribution de x sachant les données observées y en appliquant la loi de Bayes

$$p(x|y) = p(x)p(y|x) / \int p(z)p(y|z)\mathrm{d}z \ , \tag{1.2}$$

où $p(y|x)$ est appelée la *vraisemblance* et exprime notre connaissance de la dégradation subie par l'image x pour aboutir à l'observation y, et où $p(x)$ est appelé la distribution *a priori* et encode l'information dont nous disposons à propos de x avant d'observer y.

La distribution a posteriori décrit toutes les solutions possibles au problème inverse considéré, compte tenu des données observées. Elle représente une solution bien plus complète que celle proposée dans le contexte variationnel qui fournit une simple estimation de la quantité d'intérêt x. En effet, connaître la distribution a posteriori permet tout d'abord de proposer différentes solutions au problème inverse. De plus, il est possible de quantifier l'incertitude des solutions proposées, ce qui est un avantage indéniable, notamment dans les applications où la prise de décision est basée sur la restauration proposée. Par exemple, la Figure 1.1 montre les limites du paradigme variationnel puisque deux solutions à

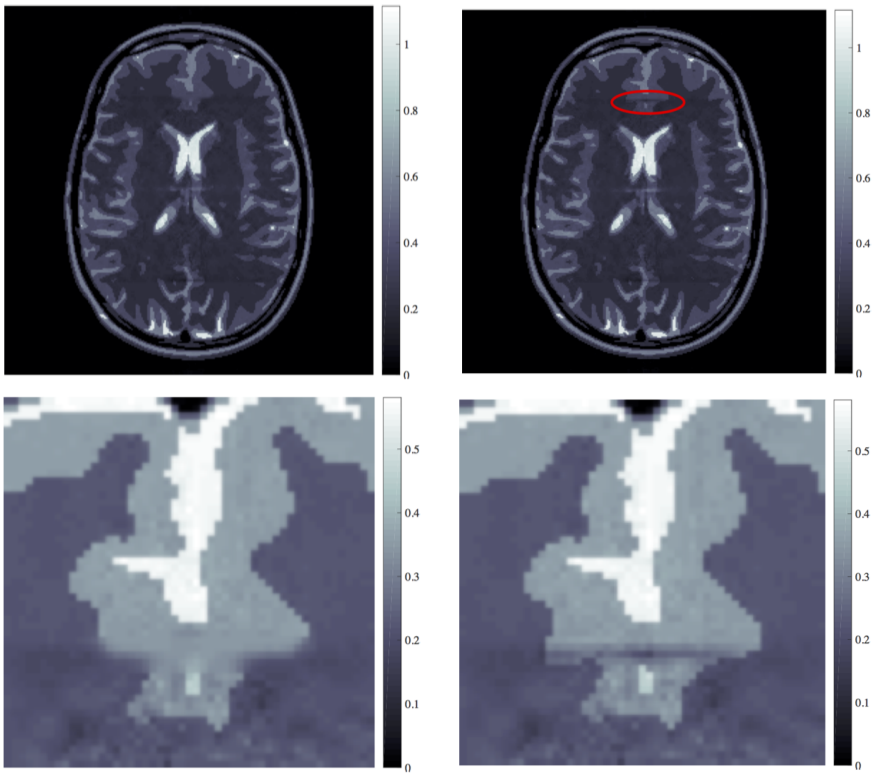


Figure 1.1: *Illustration des limites du paradigme variationnel.* Deux solutions possibles à un problème inverse en imagerie médicale délivrent 2 diagnostics différents. En effet, la solution de droite présente une lésion contrairement à celle de gauche. Quelle solution doit-on choisir ? Quelle confiance avons-nous dans la lésion ? Ces questions peuvent être répondues dans le cadre bayésien. Images tirées de (Repetti et al., 2019).

un problème inverse en imagerie médicale conduisent à deux diagnostics distincts, l'une des restaurations présentant une lésion et l'autre pas. Cependant, les images étant des objets vivant dans des espaces de haute dimension, il est impossible d'un point de vue computationnel d'estimer cette distribution a posteriori. Par ailleurs, se pose également la question du choix de la distribution a priori. En pratique, elle doit encoder des informations significatives sur la quantité d'intérêt x et ne pas impliquer des opérations de calcul trop intensives. C'est pourquoi les a priroris explicites non-appris ont longtemps été limités à des modèles log-concaves simples (Bardsley, 2012; Louchet and Moisan, 2013; Durmus et al., 2018).

Le développement récent des réseaux de neurones constitue une véritable avancée en imagerie. Les méthodes basées sur l'apprentissage profond permettent aujourd'hui d'obtenir des résultats de pointe dans de nombreux domaines, tels que la vision par ordinateur, la reconnaissance du langage ou encore les prévision météorologiques, et cela pour une grande variété de tâches. Ces approches consistent à adapter un modèle générique à un problème spécifique en entraînant notre modèle sur des données d'entraînement et cela de manière agnostique, c'est-à-dire sans connaissance sur la manière dont sont générées les données. Grâce aux ressources informatiques dont nous disposons, notre modèle apprend automatiquement une structure au sein des données fournies. Le succès des méthodes par apprentissage profond est fondé sur l'abondance des données d'apprentissage. Cependant, les données relatives à un problème inverse ne sont pas toujours disponibles en abondance. En effet, elles peu-

vent coûter très cher à générer. En outre, ces méthodes purement axées sur les données sont très spécifiques au problème. Un petit changement dans le processus d'observation implique un nouveau processus d'apprentissage qui est coûteux en temps et en calcul. Ceci limite l'utilisation de ces approches pour la résolution de problèmes inverses. Cependant, depuis les années 2010, de nombreux efforts ont été faits pour développer des méthodes combinant les approches basées sur les données et celles exploitant les informations que nous avons sur le problème inverse pour résoudre les problèmes inverses.

Un première famille de méthodes consiste à apprendre un régularisateur/a-priori à partir des données. Dans ce qui suit nous décrivons certaines de ces approches.

- Méthodes Plug & Play (PnP):

  (Venkatakrishnan et al., 2013) proposent d'utiliser des réseaux de neurones afin de définir un régularisateur implicite via un algorithme de débruitage tout en conservant une vraisemblance explicite. Cette dernière est généralement supposée connue et calibrée (Arridge et al., 2019). L'idée vient du fait que de nombreux algorithmes classiques d'optimisation font intervenir l'*opérateur proximal* du potentiel a-priori qui agit comme un débruiteur. Les approches Plug & Play mettent en relation un algorithme de débruitage avec un opérateur proximal ou un gradient associé à la densité a-priori. Elles sont principalement utilisées pour de l'estimation ponctuelle (Ryu et al., 2019; Sun et al., 2019, 2020; Xu et al., 2020; Zhang et al., 2021; Hurault et al., 2022a,b). Elles obtiennent souvent des résultats de pointe sur une grande variété de tâches (Zhang et al., 2021). Les méthodes Plug & Play sont également appliquées pour l'échantillonnage comme dans (Kadkhodaie and Simoncelli, 2020; Guo et al., 2019). Ces méthodes sont flexibles et ne nécessitent que l'entraînement d'un réseau de neurons de débruitage, qui est léger par rapport à d'autres réseaux neurones. Leurs fondements théoriques sont un domaine de recherche actif et si nous commenccons à mieux les comprendre pour l'estimation ponctuelle, à notre connaissance, aucun résultat de convergence n'a été donné pour l'échantillonnage avant les travaux présentés dans ce manuscrit.

- Méthodes basées sur le score-matching:

  Le score-matching a été conçu à l'origine pour l'apprentissage de modèles statistiques non normalisés basés sur des échantillons i.i.d. et provenant d'une distribution de données inconnue (Hyvärinen, 2005). Cependant, dans sa forme originale, le score-matching n'est pas adapté aux problèmes de haute dimension. C'est pourquoi (Bengio et al., 2013) propose une variante du score-matching original pour estimer le score d'une densité cible légèrement bruitée. Pour éviter les gradients mal définis, (Song and Ermon, 2019) entraîne un réseau de neurones qui approxime le score de différentes densités cibles bruitées et l'incorpore dans un schéma de Langevin recuit afin de générer de nouveaux échantillons à partir de la distribution empirique associée aux données d'entraînement. (Kawar et al., 2021b,a) adaptent cette méthode afin de résoudre des problèmes inverses classiques d'imagerie en échantillonnant la distribution a-posteriori. Bien que ces méthodes produisent des résultats impressionnants, le réseau de neurones utilisé pour approximer les scores est souvent très lourd et exigeant en termes de calcul. En effet, il vise à approximer directement le score de la densité a-priori induite par un ensemble de données. Il s'agit d'une tâche plus complexe que le simple débruitage. De plus, dans les méthodes actuelles, le score de l'a-priori est associé à un jeu de données spécifique (comme les chambres CelebA-HQ ou LSUN). Ainsi, l'image que nous souhaitons estimer pour résoudre le problème inverse considéré doit appartenir à la même classe d'images que les images de l'ensemble d'entraînement. Enfin, il n'est toujours pas clair quelles distributions sont échantillonnées.

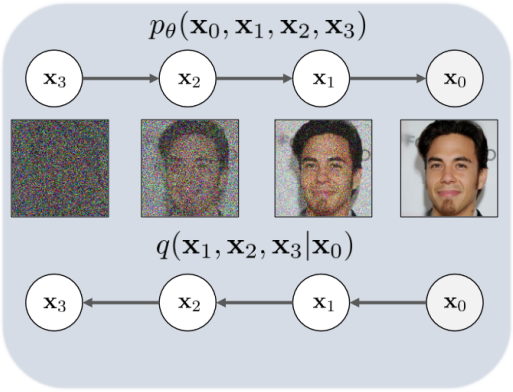


Figure 1.2: Principe d'un DPPM. p_θ correspond à la distribution de débruitage que nous apprenons par inférence variationnelle et q à la distribution de bruitage. Image issue de (Kawar et al., 2022).

- Modèles probabilistes de diffusion du bruit (DDPMs) (Ho et al., 2020) :

  Ils ont été développés dans un contexte génératif. Leur objectif est de produire de nouveaux échantillons à partir d'un jeu de données. Ces échantillons doivent être issus de la même distribution que celle des échantillons du jeu de données. Un DDPM est une chaîne de Markov paramétrée et entraînée par inférence variationnelle pour produire des échantillons correspondant aux données d'entraînement. Les transitions de cette chaîne sont apprises pour inverser un processus de diffusion de bruit. Ce processus de bruitage est généralement une chaîne de Markov qui ajoute progressivement du bruit aux données d'apprentissage jusqu'à ce que celles-ci ressemblent à du bruit pur. Ensuite, après avoir appris le processus de débruitage, nous pouvons, à partir du bruit pur, générer de nouveaux échantillons. Dans le cas le plus simple, le processus de bruitage consiste à ajouter de petites quantités de bruit gaussien et les noyaux de transitions conditionnelles sont également gaussiens. Cela permet un paramétrage particulièrement simple du réseau de neurones.

  Fort du succès de ces approches pour la génration d'images, (Kawar et al., 2022; Saharia et al., 2021) adaptent les méthodes DDPM en ajoutant des informations sur les données observées dans le modèle afin de résoudre différents problèmes inverses. Ils obtiennent d'excellents résultats pour ces différents problèmes inverses. (Kawar et al., 2022) réussissent même à résoudre des problèmes inverses avec des images très différentes de l'ensemble d'entraînement. Cependant, comme pour les méthodes basées sur le score-matching, ces approches ne présentent aucune garantie de convergence. Cela signifie que nous ne savons pas à quelle distribution appartiennent les échantillons générés. En outre, ces méthodes sont souvent exigeantes en termes de calcul et difficiles à entraîner car elles comportent de nombreux hyper paramètres.

Un autre type d'approches, appelé *Deep-Unrolling*, consiste à entraîner un réseau de neurones, dont l'architecture est inspirée d'un schéma optimisation, afin d'approcher un opérateur qui résoud un problème inverse.

- Deep unrolling:

  L'opérateur est généralement défini via un schéma itératif avec un nombre fini d'itérations N. Ce type de méthodes est associé au paradigme variationnel puisque l'opéra-

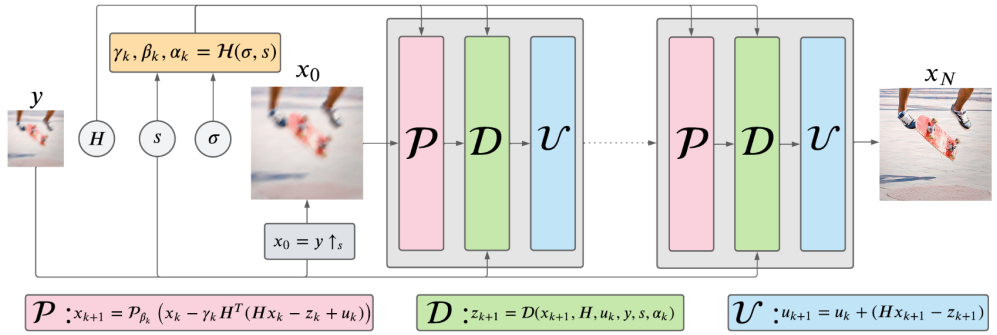


Figure 1.3: Illustration du principe des méthodes d'unrolling. Dans ce cas, le réseau de neurones vise à résoudre un problème inverse de super-résolution via un schéma ADMM linéarisé. Il existe trois modules constitutifs : le module a-priori $\mathcal{P}$ qui est appris pendant l'apprentissage, le module relatif aux données $\mathcal{D}$ comprend des informations sur le processus d'observation et le module de mise à jour $\mathcal{U}$. Image tirée de (Laroche et al., 2022).

teur est conçu pour minimiser une fonction objectif. Les méthodes basées sur l'*unrolling* optimisent les paramètres de cet algorithme itératif et apprennent une régularisation de bout en bout en minimisant cette fonctionnelle sur un ensemble de données d'apprentissage. Elles produisent d'excellents résultats en moins d'itérations que la simple application de l'algorithme d'optimisation. Les schémas d'optimisation classiques tels que *Quadratic Qplitting* (Afonso et al., 2010) ou *Alternating Direction Method of Multipliers* (ADMM) (Glowinski and Marroco, 1975; Boyd et al., 2011) sont souvent appliqués. Figure 1.3 illustre le principe de l'unrolling.

Chaque couche du réseau de neurones correspond à une opération de l'algorithme d'optimisation. Les méthodes d'unrolling incorporent directement le modèle de dégradation dans le processus d'apprentissage. Ainsi celui-ci n'est pas agnostique. Cela permet au réseau de neurones d'extraire plus d'informations des données. Ces méthodes ont prouvé leur efficacité sur de nombreuses applications : (Gregor and LeCun, 2010; Chen and Pock, 2017; Diamond et al., 2017; Adler and Öktem, 2018a; Gilton et al., 2019; Zhang et al., 2020; Laroche et al., 2022). Cependant, comme les couches du réseau de neurones comprennent des informations sur le processus de dégradations, elles sont spécifiques au problème.

Dans le cadre de cette thèse, nous nous concentrons sur les méthodes Plug & Play. Ces méthodes présentent de nombreux avantages. Elles sont flexibles et facilement adaptables à tout problème inverse car elles découplent le termes associé au processus d'observation de celui-ci du régularisateur,ce qui les rend très génériques. En outre, elles ne nécessitent pas de ressources informatiques aussi importantes que pour les autres méthodes, car il est possible d'intégrer presque n'importe quel algorithme de débruitage pré-entraîné dans leurs schémas. Pour l'informatique embarquée dans les petits appareils, la possibilité de résoudre plusieurs problèmes de restauration à l'aide d'un seul réseau de débruitage est par exemple d'un grand intérêt. Par ailleurs, il est également possible de concevoir et d'entraîner son propre débruiteur, ce qui reste moins coûteux que d'entraîner un réseau neurones pour apprendre le score d'une densité.

L'objectif de cette thèse est de mieux comprendre les méthodes Plug & Play aussi bien d'un point de vue théorique que pratique. D'un point de vue théorique, il est important

de savoir dans quelles conditions les modèles Bayésiens cadre Plug & Play sont bien définis et bien posés par exemple. D'importantes questions sur la convergence de ces algorithmes doivent également être abordées. Par exemple, les sous-ensembles vers lesquels ces algorithmes convergent lors de l'estimation de ponctuelle ne sont pas toujours clairement définis sous des hypothèses réalistes. Lors de l'échantillonnage à partir de la distribution a-posteriori, il n'existe tout simplement aucune garantie de convergence.

1.3 Contenu

Cette thèse est divisée en cinq chapitres.

Dans le Chapitre 3, nous présentons les principaux concepts apparaissant dans cette thèse. Nous introduisons tout d'abord les *problèmes inverses* et leur nature mal posée qui les rend difficiles à résoudre. Après avoir passé en revue les méthodes historiques permettant de les résoudre, nous nous concentrons sur les approches Plug & Play pour l'estimation de ponctuelle et l'échantillonnage.

Le Chapitre 4 présente PnP-SGD, un schéma basé sur la descente de gradient qui vise à estimer la MAP avec un a priori Plug & Play. Après avoir prouvé l'existence et la stabilité de l'estimateur MAP, nous montrons que ce problème est bien posé sous des hypothèses réalistes. PnP-SGD converge vers un point au voisinage de l'ensemble des points stationnaires de la distribution a-posteriori.

Le Chapitre 5 présente deux algorithmes d'échantillonnage avec un a priori Plug & Play, PnP-ULA et PPnP-ULA. Les questions théoriques pour le problème d'estimation MMSE sont abordées sous des hypothèses réalistes. Ensuite, des résultats de convergence et sur les bornes d'erreurs non-asymptotiques sont présentés. Enfin, l'efficacité de ces méthodes est prouvée sur des problèmes inverses classiques et une première étude de quantification de l'incertitude est effectuée.

Le Chapitre 6 étudie l'influence des a-prioris profonds sur la distribution a-posteriori générée par PnP-ULA. Dans un premier temps, nous étudions les solutions proposées par les distributions a-posteriori induites par différents débruiteurs d'un point de vue quantitatif. Ensuite, nous cherchons à interpréter ces résultats en termes de potentiel. Cela nous permet de mieux comprendre les solutions promues par notre débruiteur. Enfin, nous étudions la précision de la distribution échantillonnée d'un point de vue fréquentiste, afin de déterminer si la distribution que nous avons échantillonnée est réaliste et modélise correctement la réalité.

Le Chapitre 7 conclut cette thèse et propose des perspectives pour de futures études.

1.4 Contributions

Dans cette section, nous détaillons les différentes contributions de cette thèse.

1.4.1 Sur l'estimation du Maximum A-Posteriori avec des a-prioris Plug-&-Play pour la descente de gradient stochastique

Nous abordons la résolution d'un problème inverse en calculant l'estimateur MAP avec un a-priori Plug & Play et sous un schéma de descente de gradient stochastique (SGD). Le *réseau de neurones convolutionnels* (Ryu et al., 2019) est utilisé comme a-priori. L'algorithme développé est appelé Plug & Play SGD (PnP-SGD). Nous soulignons le fait qu'il s'agit d'un

problème d'optimisation hautement non convexe. Alors que les approches concurrentes produisent d'excellents résultats, elles manquent souvent de preuves de convergence (Zhang et al., 2021) ou alors celles-ci sont dérivées sous des hypothèses irréalistes (Sun et al., 2019, 2020; Ryu et al., 2019) ou du moins difficilement vérifiables (Cohen et al., 2020). Dans ce chapitre, nous abordons tout d'abord des questions théoriques importantes liées notamment à l'existence de l'estimateur MAP, sa stabilité et son caractère bien posé. Nous montrons qu'il s'agit d'un problème (faiblement) bien posé. De plus, des résultats de convergence sont présentés sous des hypothèses réalistes sur le débruiteur utilisé. Asymptotiquement, PnP-SGD converge vers des points dans le voisinage de l'ensemble des points stationnaires de la distribution a-posteriori. Nous rapportons enfin une série d'expériences démontrant l'efficacité de PnP-SGD et comparant cet algorithme avec d'autres schémas PnP. Nous montrons que PnP-SGD fournit de bons résultats par rapport aux méthodes PnP.

Ce travail va être publié dans le Journal of Mathematical Imaging and Vision (JMIV) dans un numéro spécial.

1.4.2 Méthodes Bayésiennes utilisant des a-prioris Plug & Play: quand Lagevin rencontre Tweedie

Les a-prioris Plug & Play incorporés dans les schémas d'échantillonnage de Monte Carlo ne sont pas très courants en imagerie, bien qu'ils commencent à être étudié. À notre connaissance, il n'existe aucune preuve de convergence pour de tels schémas. En plus des problèmes de convergence de ces algorithmes, comme pour le problème de l'estimation MAP, d'importantes questions restent ouvertes quant à savoir si les modèles et estimateurs bayésiens sous-jacents sont bien définis, bien posés et possèdent les propriétés de régularité de base requises pour effectuer de calculs dans le cadre Bayésien. Ce chapitre développe la théorie de l'analyse et du calcul bayésiens avec des a-prioris PnP. Nous présentons l'algorithme Plug & Play Unadjusted Langevin Algorithm (PnP-ULA) pour l'échantillonnage de Monte-Carlo et l'estimation de l'erreur quadratique moyenne minimale (MMSE). En utilisant des résultats récents sur la convergence quantitative des chaînes de Markov, nous établissons des garanties de convergence détaillées pour cet algorithme sous des hypothèses réalistes sur les opérateurs de débruitage utilisés. Une attention particulière aux débruiteurs basés sur les réseaux de neurones profonds est portée. Nous montrons également que ces algorithmes ciblent approximativement un modèle bayésien optimal en théorie de la décision et bien posé. L'efficacité de PnP-ULA est démontrée sur plusieurs problèmes inverses classiques en imagerie tels que le défloutage et l'interpolation.

Ce travail a été publié dans SIAM Journal on Imaging Science.

1.4.3 Etude approfondie des a-prioris PnP pour l'échantillonnage

Si le cadre Bayésien donne accès à la distribution a-posteriori, qui sous-tend toute inférence sur le signal que nous souhaitons récupérer à partir des données observées, il est intéressant de déterminer quelles solutions sont promues par des débruiteurs aux propriétés différentes. En outre, vérifier si le modèle bayésien estimé est significatif d'un point de vue fréquentiste est également une question de première importance. En effet, nous cherchons à vérifier si le modèle Bayésien est exact et modélise correctement la distribution a-posterior empirique. D'un point de vue pratique, nous vérifions si les probabilités calculées avec notre modèle bayésien correspondent aux probabilités empiriques. Nous effectuons cette tâche avec trois débruiteurs MMSE différents induisant des a-prioris d'image différents.

1.4.4 Publications et Pré-publications

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. Bayesian imaging using plug & play priors: When langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022. doi: 10.1137/21M1406349. URL https://doi.org/10.1137/21M1406349

- R. Laumont, V. de Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. On maximum-a-posteriori estimation with plug & play priors and stochastic gradient descent. 2021. URL https://hal.archives-ouvertes.fr/hal-03348735/document

1.4.5 Liste des présentations

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **Centre de Mathématiques et Leurs Applications (CMLA)**, ENS Paris-Saclay, le 10 Février 2020.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **Groupe de Travail des Jeunes Doctorants (GTTJD)**, Université Paris-Cité, le 14 Février 2020.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **GTTI-Centre Borelli**, ENS Paris-Saclay, le 10 Mars 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **Groupe de Travail des Jeunes Doctorants (GTTJD)**, Université Paris-Cité, le 21 Mai 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation à la 10^{eme} **Biénnale de Mathématiques Appliquées et Industrielles**, La Grande Motte, le 24 Juin 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation à la **Journées du groupe Modélisation Aléatoire et Statistique (MAS) 2020 de la Société de Mathématiques Appliquées et Industrielles (SMAI)**, Université d'Orléans, le 27 Août 27 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **Congrès des Jeunes Chercheuses et Chercheurs en Mathématiques Appliquées (CJC-MA)**, Ecole Polytechnique, le 29 Octobre 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation à l' 3^{rd} **IMA Conference on Inverse Problems from Theory to Application**, Bayes Center (Ecosse), le 5 Mai 2022.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Présentation au **Workshop: Imaging With Uncertainty Quantification (IUQ)**, Konventum Conference Center (Denmark), le 27 September 2022.

2

Introduction

2.1 Context

In several areas of science and industry, there is a need to reliably infer a signal from noisy observed data. Typical examples are encountered in biomedical engineering, geophysics, astronomy or finance. These problems are called *inverse problems* because they consist in inverting the observation process, to retrieve a signal given observed data. Such problems can often be modelled by Equation (2.1)

$$y = \mathrm{A}x + n \,, \tag{2.1}$$

where y corresponds to the observed data, x the quantity we wish to recover/to infer, A to the forward operator that models the physics behind the observation process in the absence of noise and n to the measurement noise.

These inverse problems encountered in real life are often *ill-posed*. It means that a little perturbation in the measurements can lead to large errors in the signal we wish to reconstruct

or that there exist several possible signal values that are consistent with the observed data. It is a question of prime importance especially in fields where decisions are taken based on this reconstruction like for image-guided diagnosis in medicine. Furthermore, imaging inverse problems involve high-dimensional objects, making the task more challenging.

2.2 Problem statement

In many applications, the admissible signals belong to a small subset of the ambient space. For example, in modern imaging science we tend to believe that natural images are concentrated on a low-dimensional manifold of the ambient space (Fefferman et al., 2016). It has motivated the development of regularizers/priors that incorporate prior knowledge on the data we wish to retrieve.

Variational approaches are very popular to deal with inverse problems in imaging science. They consist in building an objective function we seek to minimize. This objective function is generally the sum of two terms, the *data-fidelity term* that measures the discrepancy between the observation y and the proposed restoration and the *regularization term* which promotes images with desired properties or which are located in the neighbourhood of some sub-manifold. As classical regularizers, we can cite the Total Variation (TV) semi-norm (Rudin et al., 1992) that promotes images with sparse gradients, hence favors images that are piecewise-constant, or regularizers enforcing sparsity in transformed domains as wavelet basis (Donoho and Johnstone, 1994). Although a lot of progress has been made, hand-crafted regularizers do not succeed in capturing the whole complexity in natural images. In addition, these regularizers are often designed to be convex in order to take advantage of convex optimization tools, which might limit their effectiveness. For instance, solutions provided by the TV regularizer within the variational framework can suffer from staircasing (Louchet and Moisan, 2013).

Ill-posed inverse problems can also be tackled in a Bayesian framework. In this framework both the observed data and the signal to be reconstructed are the result of random phenomena. Both are seen as the realizations of random variables. In the Bayesian paradigm, we seek to determine the posterior distribution, *ie* the distribution of the model parameter x given the observed data y by applying the Bayes' rule

$$p(x|y) = p(x)p(y|x)/\int p(z)p(y|z)\mathrm{d}z\ , \tag{2.2}$$

where $p(y|x)$ is called the *likelihood* and is directly related to the noise distribution and $p(x)$ is the *prior* which encodes the information we have about x.

The posterior distribution describes all possible solutions to the inverse problem considered, given observed data. It represents a far more complete solution to the inverse problem than retrieving the quantity of interest x as in the variational paradigm, and offers a lot of possibilities. First of all, knowing the posterior distribution allows to propose different estimates of the posterior distribution as a solution to the inverse problem. In addition, it allows to perform uncertainty quantification studies on the proposed solutions. It is a huge advantage, especially in applications where decision making is based on a proposed restoration. For example, Figure 2.1 shows the limitations of the variational paradigm as two solutions to an inverse problem in medical imaging lead to two distinct diagnoses as one of the restorations presents a lesion. However, recovering the whole posterior in imaging science is rarely doable from a computational point of view, as we have to deal with high-dimensional objects. It also raises the question of the choice of the prior distribution. To

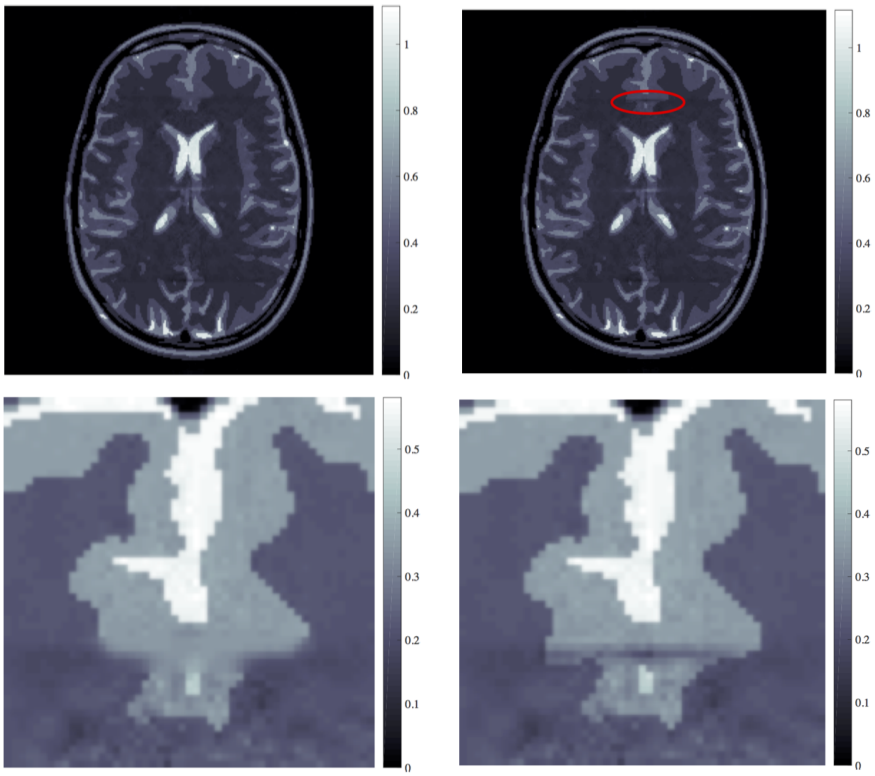


Figure 2.1: *Illustration of the advantage of the Bayesian framework over the variational paradigm.* Two possible solutions to an inverse problem in medical imaging leading to two possible different diagnoses. Indeed, the solution on the right presents a lesion whereas the one on the left does not. Which solution should we select ? How confident are we on the lesion ? These questions can be answered in the Bayesian framework. Illustrations from (Repetti et al., 2019).

be useable, it has to contain meaningful information about the quantity of interest x and not to imply too intensive computational operations. In this framework, hand-crafted priors were restricted for a long time to simple log-concave models (Bardsley, 2012; Louchet and Moisan, 2013; Durmus et al., 2018).

The recent development of neural networks is a real breakthrough in imaging science. Deep Learning based methods achieve today state-of-the-art results in a lot of different fields such as computer vision, speech recognition, weather forecasting and playing games and for a large variety of tasks. These approaches consist in adapting a generic model to a specific problem through learning against training data. Thanks to the computational resources we have at our disposal, they automatically learn a structure within the data they are fed with. Their success is based on the abundance of training data and the agnosticism from a prior knowledge of how these data are generated. However, data related to an inverse problem are not always available in abundance. Because they can cost too much to produce. Besides, these purely data-driven methods are very problem specific. A small change in the observation process implies a new time-consuming and computationally expensive learning process. It limits the use of these approaches to tackle inverse problems. However, since the 2010's, a lot of effort has been made to develop methods combining data- and knowledge-driven approaches for solving inverse problems.

A first type of methods consists in learning a regularizer/prior from the data. In the following, we describe some of these approaches.

- Plug & Play (PnP) methods: (Venkatakrishnan et al., 2013) propose to use neural networks in order to define an implicit regularizer via a denoising algorithm while keeping an explicit likelihood density, which is usually assumed to be known and calibrated (Arridge et al., 2019). The idea comes from the fact that many classical optimization algorithms involve the *proximal operator* of the prior potential which acts as a denoiser. Plug & Play approaches either relate a denoising algorithm to a proximal operator or a gradient associated with the prior density. They are mostly used to perform point estimation (Ryu et al., 2019; Sun et al., 2019, 2020; Xu et al., 2020; Zhang et al., 2021; Hurault et al., 2022a,b) where they often achieve state-of-the-art results on a large variety of tasks (Zhang et al., 2021) but are also applied for sampling as in (Kadkhodaie and Simoncelli, 2020; Guo et al., 2019). These methods are flexible and only require to train a denoising neural network which is light in comparison to other neural networks. Their theoretical foundations are an active field of research and if we begin to better understand them for point estimation, to the best of our knowledge, no convergence result has been given for sampling before the work presented in this manuscript.

- Score-matching based methods: Score-matching is originally designed for learning non-normalized statistical models based on i.i.d. samples from an unknown data distribution (Hyvärinen, 2005). However, in its original form, it scales poorly with the dimension. That is why (Bengio et al., 2013) propose a variant of the original score matching to estimate the score of a slightly perturbed target density. To avoid ill-defined gradients, (Song and Ermon, 2019) train a neural network that approximated the score of different perturbed target densities and plug it into an annealed Langevin scheme to generate new samples from the empirical distribution associated with the training data. (Kawar et al., 2021b,a) adapt this method in order to solve classical imaging inverse problems by sampling from the posterior distribution. Although, it delivers impressive results, the neural network used to approximate the scores is often very heavy and computationally demanding. Indeed, it aims to directly approximate the score of the prior density induced by a dataset. It is a more complex task than simple denoising. Furthermore, in the current methods the score of the prior is associated with a specific dataset (such as CelebA-HQ or LSUN bedrooms). Then, the image we wish to retrieve to solve the considered inverse problem must be in the same class as images of the training set. Finally, it is still not clear which distributions are sampled exactly.

- Denoising Diffusion Probabilistic Models (DDPMs) (Ho et al., 2020): were developed in a generative context. Their goal is to produce new samples from a distribution given a dataset. A DDPM is a parameterized Markov chain trained using variational inference to produce samples matching the training data. Transitions of this chain are learned to reverse a noising diffusion process. This noising process is usually a Markov chain that gradually adds noise to the training data until the training data looks like pure noise. Then, after learning the denoising process we are able given pure noise to generate new samples. In the simplest case, the noising process consists in adding small amounts of Gaussian noise and the conditional transition kernels are Gaussian too. It allows for a particularly simple neural network parameterization.

  Based on the success of these approaches in generative modelling, (Kawar et al., 2022;

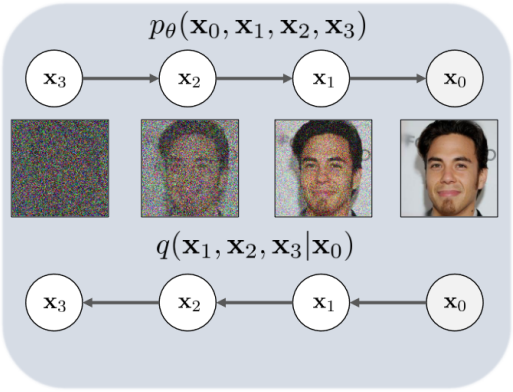


Figure 2.2: Principle of a DPPM. p_θ corresponds to the denoising diffusion distribution we learn by variational inference and q to the noising process distribution. Image taken from (Kawar et al., 2022).

Saharia et al., 2021) adapt this framework in order to solve different inverse problems by adding information about the observed data in the model. They achieve great results for the different inverse problems. (Kawar et al., 2022) even succeed to solve inverse problems with images very different from the training set. However, as for score-matching based methods, these approaches do not have any convergence guarantees. It means that we do not know to which distribution the generated samples belong. In addition, theses methods are often computationally demanding and difficult to train as they have a lot of hyper parameters.

A second kind of approaches, called *Deep-Unrolling*, consists in training a neural network, whose architecture is inspired by optimization scheme, to approximate an operator that solves a given inverse problem.

- Deep unrolling: The operator is generally defined via an iterative scheme with a given finite number of iterations N. It is associated with the variational framework as the operator is designed to minimize a functional. The unrolling based methods both optimize the parameters of this iterative algorithm and learn a regularization in an end-to-end manner by minimizing this functional over a training set. It produces great results in fewer iterations than simply applying the optimization algorithm. Classical optimization schemes such as half-quadratic splitting (Afonso et al., 2010) or Alternating Direction Method of Multipliers (ADMM) (Glowinski and Marroco, 1975; Boyd et al., 2011) are considered. Figure 2.3 illustrates the principle of deep-unrolling methods.

  Each layer of the neural network corresponds to an operation of the iterative optimization algorithm. Deep unfolding methods directly incorporate the degradation model into the learning process, making it not agnostic any more. It allows the neural network to extract more information from the data. These methods have proved their efficiency over many applications (Gregor and LeCun, 2010; Chen and Pock, 2017; Diamond et al., 2017; Adler and Öktem, 2018a; Gilton et al., 2019; Zhang et al., 2020; Laroche et al., 2022). However, because the neural network layers include information about the measurement process, they are very problem-specific.

In this thesis, we focus on Plug & Play methods. These methods have numerous advantages. They are flexible and easily adaptable to any inverse problem as they decouple the

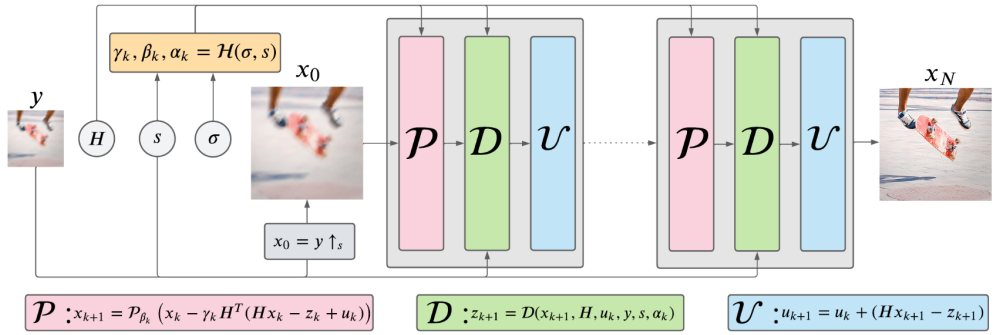


Figure 2.3: Illustration of the deep-unrolling principles. In this case, the neural network aims at solving a super-resolution inverse problem via a linearised ADMM scheme. There are three constitutive modules: the prior module $\mathcal{P}$ that is learnt during the training, the data module $\mathcal{D}$ which incorporates information about the observation process and the update module $\mathcal{U}$. Image taken from (Laroche et al., 2022).

observation related terms from the plugged regularizer. It makes them really universal. In addition, they do not require massive computational resources as we can plug almost any pre-trained denoising algorithm within their schemes. For embedded computing in small devices, being able to solve several restoration problems by using a single denoising network is of huge interest for instance. Besides, it is also possible to design and train its own denoising neural network, which is still less expensive than training a score matching neural network.

The goal of this thesis is to better understand Plug & Play methods both from a theoretical and a practical point of view and to pave the way for further studies. From a theoretical point of view, it is important to know under which conditions the Bayesian models derived within a Plug & Play framework are well defined and well posed for instance. There are also important questions about the convergence of these algorithms that need to be addressed. For instance, the subsets towards which these algorithms converge when performing point estimation is not always clearly defined under realistic assumptions. When sampling from the posterior distribution, there are simply no convergence guarantees.

2.3 Outline

This thesis is divided in 5 chapters.

In Chapter 3 we present the main concepts appearing in this thesis. We firstly introduce *inverse problems* and their ill-posed nature that makes them difficult to solve. After reviewing historical methods to tackle them, we focus on Plug & Play approaches for point estimation and sampling.

Chapter 4 introduces PnP-SGD, a gradient descent based scheme that aims at estimating the MAP with a Plug & Play prior. After proving the existence, the stability of the MAP, we show that this problem is well-posed under realistic assumptions. PnP-SGD converges towards a point in the vicinity of the set of the stationary points of the posterior distribution.

Chapter 5 presents two sampling algorithms, PnP-ULA and PPnP-ULA, with a Plug & Play prior. Theoretical questions for the MMSE estimation problem are addressed under realistic assumptions. Then, covergence results and non-asymptotic error bounds are pre-

sented. Finally, the efficiency of the methods is proved on a classical inverse problems and a first uncertainty quantification study is delivered.

Chapter 6 investigates the influence of deep priors in the posterior distribution. First, we study the solutions proposed by the posterior distributions induced by different denoisers from a quantitative point of view. Then, we seek to interpret these results in terms of potential. It allows us to better understand the solutions promoted by our denoiser. Finally, we investigate the accuracy of the sampled distribution from a frequentist point of view, in order to determine if the distribution we sampled from is realistic and properly models the reality.

Chapter 7 concludes this thesis and proposes perspectives for further studies.

2.4 Contributions

In this section, we detail the contributions of this thesis.

2.4.1 On Maximum-a-Posteriori estimation with Plug & Play priors and stochastic gradient descent

We address the problem of solving an inverse problem by computing the MAP estimator with a Plug & Play prior using a stochastic gradient descent (SGD) scheme. A state-of-the-art *Convolutionnal neural network* is used in place of the prior (Ryu et al., 2019). The algorithm developed is called Plug & Play SGD (PnP-SGD). We emphasize the fact that it is a highly non-convex optimization problem. Where previous approaches show impressive results, they either lack convergence proofs (Zhang et al., 2021) or they are derived under unrealistic (Sun et al., 2019, 2020; Ryu et al., 2019) or at least not easily checkable (Cohen et al., 2020) assumptions. In this chapter, we firstly address key theoretical questions related the existence of such an estimator, its stability and its well-posedness. We show that it is a (weakly) well-posed problem. In addition, convergence results are presented under realistic assumptions on the denoiser used. Asymptotically, PnP-SGD converges towards points in the vicinity of the set of stationary point of the posterior distribution. We finally report a range of imaging experiments demonstrating PnP-SGD as well as comparisons with other PnP schemes. We show that PnP-SGD provides good results in comparison with state-of-the-art PnP methods that are clearly interpretable.

This work will be published on Journal of Mathematical Imaging and Vision (JMIV) under a special issue.

2.4.2 Bayesian imaging using Plug & Play priors: when Langevin meets Tweedie

Plug & Play priors within Monte Carlo sampling schemes for general Bayesian computation are not very common in imaging science although they begin to be an active field of research. To the best of our knowledge, there is no proof of convergence. Algorithm convergence issues aside, as for the MAP estimation problem, there are important open questions regarding whether the underlying Bayesian models and estimators are well defined, well-posed, and have the basic regularity properties required to support efficient Bayesian computation schemes. This chapter develops theory for Bayesian analysis and computation with PnP priors. We introduce Plug & Play Unadjusted Langevin Algorithm (PnP-ULA) for Monte Carlo sampling and Minimum Mean Squared Error (MMSE) estimation. Using recent results on the quantitative convergence of Markov chains, we establish detailed

convergence guarantees for this algorithm under realistic assumptions on the denoising operators used, with special attention to denoisers based on deep neural networks. We also show that these algorithms approximately target a decision-theoretically optimal Bayesian model that is well-posed. PnP-ULA is demonstrated on several canonical problems such as image deblurring and inpainting, where it is used for point estimation as well as for uncertainty visualisation and quantification.

This work was published on SIAM Journal on Imaging Science.

2.4.3 In-depth study of data-driven priors for sampling

If the Bayesian framework gives an access to the posterior distribution, which underpins all inference about the signal we wish to recover from the observed data, it is interesting to determine what solutions are promoted by denoisers with different properties. In addition, checking if the estimated Bayesian model is meaningful from a frequentist point of view is also a question of prime importance. Indeed, we aim at verifying if the Bayesian model is accurate and correctly models the empirical posterior distribution. From a practical point of view, we check if the computed probabilities with our Bayesian model match the empirical ones. We perform this task on three different MMSE denoisers inducing different image priors.

2.4.4 Publications and Preprints

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. Bayesian imaging using plug & play priors: When langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022. doi: 10.1137/21M1406349. URL https://doi.org/10.1137/21M1406349

- R. Laumont, V. de Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. On maximum-a-posteriori estimation with plug & play priors and stochastic gradient descent. 2021. URL https://hal.archives-ouvertes.fr/hal-03348735/document

2.4.5 List of presentations

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Centre de Mathématiques et Leurs Applications (CMLA)**, ENS Paris-Saclay, February 10th 2020.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Groupe de Travail des Jeunes Doctorants (GTTJD)**, Université Paris-Cité, February 14th 2020.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **GTTI-Centre Borelli**, ENS Paris-Saclay, March 10th 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Groupe de Travail des Jeunes Doctorants (GTTJD)**, Université Paris-Cité, May 21rst 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at 10^{eme} **Biénnale de Mathématiques Appliquées et Industrielles**, La Grande Motte, June 24th 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Journées du groupe Modélisation Aléatoire et Statistique (MAS) 2020 de la Société de Mathématiques Appliquées et Industrielles (SMAI)**, Université d'Orléans, August 27th 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Congrès des Jeunes Chercheuses et Chercheurs en Mathématiques Appliquées (CJC-MA)**, Ecole Polytechnique, October 29th 2021.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at 3^{rd} **IMA Conference on Inverse Problems from Theory to Application**, Bayes Center (Scotland), May 5th 2022.

- R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. *Bayesian imaging using Plug & Play priors.* Invited presentation at **Workshop: Imaging With Uncertainty Quantification (IUQ)**, Konventum Conference Center (Denmark), September 27th 2022.

3

Background

3.1 Inverse problems

3.1.1 Inverse problems

From a practical viewpoint solving an inverse problem boils down to determine causes from an observed phenomenon. It consists of retrieving the model parameter $x \in \mathcal{X}$ from observed data $y \in \mathcal{Y}$ where x and y are generally related by the following equation *

$$y = \mathrm{A}(x) + n \tag{3.1}$$

where $\mathcal{X}$ is the parameter space and $\mathcal{Y}$ is the data space. Both spaces are vector spaces with appropriate topologies. Elements of both spaces consist in possible parameter and data vectors. $\mathrm{A} : \mathcal{X} \to \mathcal{Y}$ is an operator called the **forward operator** and maps parameters to data. Classically it is assumed to be known and continuous. $n \in \mathcal{Y}$ is the realization of a $\mathcal{Y}$-valued random variable that characterizes the observation process stochasticity.

In imaging science, the model parameter x is usually an image in $\mathbb{R}^d$. We consider two main types of representation for images:

- Pixel-wise representation $x \in \mathbb{R}^{H \times W \times C}$ where H and W are respectively the height and the width of the picture whereas C corresponds to the number of channels. For instance, $C = 1$ if we are dealing with grayscale images and $C = 3$ if it is with RGB images. Eventually, $d = HWC$.
- Coordinate-based representation $x = u_\theta(x_1, x_2) \in \mathbb{R}^C$, with $(x_1, x_2) \in \Omega^2 \subset \mathbb{R}^2$. u_θ are the intensity values of the C channels located in (x_1, x_2) and is parametrized by $\theta \in \Theta$ with Θ the set of possible parameters for u_θ.

Classically, images are encoded by the discrete pixel-wise representation. It is the representation we will use in the following. However, we point out that recent works aim at developing neural networks learning a continuous intensity map from spatial locations. They succeed in accurately modelling natural scenes and globally improve the results when dealing with high-frequency structures as proved in (Sitzmann et al., 2020) for example.

3.1.2 Examples of inverse problems in imaging science

Most inverse problems in imaging aim at reconstructing an unknown image $x \in \mathbb{R}^d$ from a degraded observation $y \in \mathbb{C}^m$ under some assumptions on their relationship. In this case A is called the degradation operator and models deterministic instrumental aspects of the observation process, and n is an unknown (stochastic) noise term taking values in $\mathbb{C}^m$. In this section, we introduce some classical inverse problems we may deal with in this manuscript. These problems differ from each other because of the degradation operator A, which is often assumed to be linear.

3.1.2.a Image denoising

In denoising, we have $\mathrm{A} = \mathrm{Id}$ and (3.1) becomes

$$y = x + n \tag{3.2}$$

*There also exist other types of inverse problems where the noise n is not additive but multiplicative for instance. However, they are out of the scope of this thesis. See (Aguerrebere, 2014, Chapter 2) or (Dunlop, 2019) for example.

Optical blur

Motion blur

Figure 3.1: Examples of the different types of blur. In both cases the blur kernel is spatially-varying, with a mix of sharp and blurry objects for both images.

In the simplest case, the noise distribution is known. However, we can face a correlated or a spatially varying noise distribution, what increases the difficulty.

From a practical point of view, a digital photography can suffer from noise if the acquisition time is too short or if the light intensity in the scene is too low.

3.1.2.b Image deblurring

In image deblurring, we have $\forall x \in \mathbb{R}^d,\ \mathrm{A}x = k * x$ where k is a blurring kernel and $*$ stands for the convolution operator.

There are two major types of blur, the *motion blur* and the *optical blur*. Motion blur arises when either the acquisition system or the photo's subject is moving during the acquisition time. Optical blur is caused by an incorrect focus on some elements of the picture, by light diffraction and by chromatic aberrations in the lens. Figure 3.1 illustrates these two types of blur. More realistic blur operators are encoded by spatially-varying kernel. In addition, the convolution kernel k is not necessarily known, making this problem even more to difficult to solve as we need to firstly or jointly estimate k. In this case we talk about *blind* deblurring.

3.1.2.c Image interpolation

The degradation operator A is a masking operator that masks certain pixels in a (random) manner. A is the identity matrix with random rows missing. Then, (3.1) becomes

$$y = x|_{mask} + n \tag{3.3}$$

3.1.2.d Image inpainting

The inpainting problem is very much related to the interpolation problem. The difference is that a whole pixel region is masked. This problem is more complicated than interpolation.

Figure 3.2 illustrates some classical imaging inverse problems.

Original Denoising Deblurring Interpolation Inpainting

Figure 3.2: Example of imaging inverse problems.

3.1.3 Ill-posedness

The inverse problems we are facing in imaging science are usually ill-posed. It means that they are sensitive to variations in the data vector y. The paternity of the term "ill-posed" is attributed to the French mathematician Jacques Hadamard who firstly defined a *well-posed problem*.

Definition 3.1.1 (Well-Posedness) *An inverse problem is called well-posed if*

1. *There exists at least one solution. (Existence)*
2. *There is at most one solution. (Uniqueness)*
3. *The solution depends continuously on data. (Stability)*

According to Hadamard, an inverse problem is said to be ill-posed, if one of these conditions does not hold. If the two first conditions appear to be normal, the third one can be easily explained from a practical point of view. If the inverse operator of A is either unbounded or discontinuous, the additive observation noise can be dramatically amplified during the inversion process. Then, two distinct but close observations $\tilde{y}$ and $\bar{y}$ can lead to very different reconstructions $\tilde{x}$ and $\bar{x}$, what is not desirable.

Ill-posedness is often encountered. One can easily prove that every compact operator between 2 infinite-dimensional Hilbert spaces with infinite range has a discontinuous inverse and then its associated inverse problem is ill-posed. If A is linear, one can also interpret instability in terms of singular-values. The faster the decay of the singular values, the more ill-posed is the inverse problem.

Let us consider some special cases where the degradation operator A is linear and the parameter and data spaces involved are Euclidean.

1. Assume $\mathrm{A} : \mathbb{R}^d \to \mathcal{I}(\mathrm{A}) \subsetneq \mathbb{R}^m$ with $d < m$ and there exists a unique inverse operator $\mathrm{A}^{-1} : \mathcal{I}(\mathrm{A}) \to \mathbb{R}^d$. As the observation noise n does not necessarily belong to $\mathcal{I}(\mathrm{A})$, we cannot simply invert A although its inverse exists.

2. Assume $\mathrm{A} : \mathbb{R}^d \to \mathbb{R}^m$ with $d > m$. In this case, we have more unknowns that equations and the system is said to be underdetermined. Consequently, there are several possible solutions to one observation y.

3. Assume $\mathrm{A} : \mathbb{R}^d \to \mathbb{R}^d$ and there exists $\mathrm{A}^{-1} : \mathbb{R}^d \to \mathbb{R}^d$. We note λ_1 and λ_d respectively the smallest and largest eigenvalue. If $\kappa = \lambda_d/\lambda_1$ is large, then the matrix is nearly singular and the problem is sensitive to small perturbations in the observation y. The problem is then said to be ill-conditioned. Thus, the naive restoration $\tilde{x} = \mathrm{A}^{-1}y = x + \mathrm{A}^{-1}n$ is dominated by $\mathrm{A}^{-1}n$ and does not constitute a good restoration.

In addition if $n = 0$ in Equation (3.1), the problem is ill-posed if A is not invertible. Subsequently, it is clear that the interpolation and inpainting inverse problems are ill-posed.

3.1.4 Regularization

When facing an ill-posed inverse problem, Hadamard suggests in (Hadamard, 1923) to model it differently instead of trying to solve it. However, major inverse problems faced in mathematical physics are usually unstable as showed in shown in (Calderon, 1958; Calderon and Zygmund, 1989; Zygmund, 2011) for instance.

The goal of regularization is to develop stable methods in order to estimate x from data y assuming the knowledge of the degradation operator A and to prove some properties of the estimated solution. In the fact, regularization aims at computing a mapping $\mathcal{R}_\theta : \mathcal{Y} \to \mathcal{X}$ which is continuous in $\mathcal{Y}$ for a fixed regularization parameter θ and such that $\mathcal{R}_\theta(y) \to x$ as $y \to \mathrm{A}x$.

There exist 4 major types of regularization methods.

- Approximate analytic inversion The idea behind these methods is to stabilize A^{-1} by smoothing the inverse operator. It is often very problem-specific. The Filtered Back-Projection introduced by (Natterer, 2001; Natterer and Wübbeling, 2001) used in CT reconstruction is a good example of such methods.

- Iterative methods with early stopping With this kind of methods, we typically aim at minimizing the functional $z \mapsto \|\mathrm{A}z - y\|_2^2$. Applying gradient based methods, this functional usually decreases before growing up. We talk about semi-convergent behaviour. The goal of these methods is to design a stopping criterion that plays the role of a regularizer (see (Natterer and Wübbeling, 2001) for example).

- Discretization The idea is to look for an approximate solution of (3.1) in a specific subspace using Projection or Galerkin methods (see (Natterer, 1977) for example).

- Variational methods This approach boils down to solve an optimization problem

$$\mathcal{R}_\theta(y) = \arg\min_{z \in \mathcal{X}} \{\mathcal{L}(\mathrm{A}z, y) + \mathcal{S}_\theta(z)\} \ .$$

with $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ a function quantifying the similarity between two elements of the data space, often called the *the data-fidelity term*, $\mathcal{S}_\theta : \mathcal{X} \to \mathbb{R}$ a regularization function associated with the regularization parameter θ. This term allows to promote solutions with desirable features. It stabilizes the algorithm by encoding a-priori information about x. Designing a good regularizer consists in a challenging task and is an active field of research. The classical Tikhonov regularization uses Hilbert-space norms to regularize the inverse problem and was introduced in (Tikhonov, 1943). The value of θ has also a huge impact on the proposed solution and it can be learned or set manually. The data-fidelity term is often chosen with respect to the observation noise distribution and generally as an affine transformation of the negative log-likelihood in order to have a statistical interpretation. Typically in digital photography or in tomography applications, the observation noise is modelled by a Poisson distribution, which leads to a Kullback-Leibler (KL) divergence for the data-fidelity term. Otherwise, the observation noise is often modelled by a Gaussian distribution.

It is a very adaptive framework with a plug-and-play structure as the degradation operator A, the data-fidelity term and the regularizer are chosen in accordance to the properties of the tackled inverse problem. It also involves terms with clearly defined roles.

3.1.5 Variational Approaches in imaging science and their regularizers

In this section we focus on variational approaches in imaging science, which can usually be written

$$\mathcal{R}_\lambda(y) = \arg\min_{z\in\mathcal{X}} \{\mathcal{L}(\mathrm{A}z, y) + \lambda\mathcal{S}(z)\} \ , \tag{3.4}$$

where $\lambda > 0$ and balances the trade-off between the regularization and data-fidelity.

As previously explained, adding a regularization term allows to add information about the properties of the solution of the tackled inverse problem. This supplementary piece of information can favour a solution over another and can turn an ill-posed inverse problem into a well-posed one. The appeal for these methods in imaging is due to two main reasons. First, they are very adaptable as explained before. Second, with the progress in (convex) optimization, they scale very well to high dimensional problems. Finding a good regularizer capturing the richness of images consists in an active field of research.

- Total variation (TV). TV was introduced in (Rudin et al., 1992) for the denoising inverse problem. It reads

$$\forall z \in \mathbb{R}^d,\ \mathcal{S}(z) = \mathrm{TV}(z) = \int_\Omega \mathrm{d}|Dz| \tag{3.5}$$

  where $\Omega \subset \mathbb{R}^d$ and corresponds to the image domain.

  Discretizing (3.5), we get

$$\forall z \in \mathbb{R}^d,\ \mathrm{TV}(z) = \sum_{(i,j)\in[0,n_{rows}]\times[0,n_{cols}]} \|\nabla_{i,j} z\|_p \tag{3.6}$$

  with $p = 1$ or 2 and $\nabla_{i,j} z = [z_{i+1,j} - z_{i,j}, z_{i,j+1} - z_{i,j}]$.

  It is a regularization widely used in imaging science as it promotes images with sparse gradients and assumes that images are piecewise constant objects. However, it struggles to restore images with high-frequency structures and tends to add jumps. We talk about *staircasing effect*. In order to alleviate this issue, other regularization terms based on TV were designed such as the Total Generalized Variation (TGV) (Bredies et al., 2010) or the Infimal-Convolution Total Variation (ICTV) (Chambolle and Lions, 1997).

- Sparsity in transformed domains. The regularizer can be written $\mathcal{S}(z) = \|Wz\|_1$ with $W : \mathbb{R}^d \to \mathcal{Z}$. This regularization favours sparsity of the image coefficients in a representation space $\mathcal{Z}$. $\mathcal{Z}$ is typically the space spanned by a learned dictionary (Elad and Aharon, 2006) or by a wavelet basis Beck and Teboulle (2009) or a wavelet frame (Donoho and Johnstone, 1994).

- Expected Patch Log-Likelihood (EPLL). This approach is intrinsically related to the Bayesian approach described in Section 3.2. It consists in learning a prior p on a set of image patches and then to minimize (3.4) with $\mathcal{S}(z) = -\sum_i \log p(\mathbf{P}_i z)$ and $\mathbf{P}_i$ is a matrix extracting the i-th path from the image in vectorized form out of all overlapping patches. The authors of (Zoran and Weiss, 2011) used a Gaussian mixture model prior p whose parameters have been learned from a patch dataset with an *Expectation-Maximization* (EM) algortihm.

The variational approach consists in solving an optimization problem. There rarely exist closed-form solutions except in simple cases like, for example, when the data-fidelity and regularization terms are both quadratic. With more complex regularizers, we have to apply iterative schemes. The recent progress in convex optimization allows to efficiently tackle high-dimensional optimization problems even if the objective function is not smooth. As iterative schemes commonly used in imaging, we can cite the *Alternating Direction Method of Multipliers (ADMM)* (Glowinski and Marroco, 1975; Boyd et al., 2011), the first-order primal-dual algorithm (Chambolle and Pock, 2011). However, the convergence guarantees only concern convex regularizers, which limits their reliability in more general contexts.

3.2 Bayesian Approach in imaging science

The Bayesian paradigm is a complete statistical inferential methodology providing a natural framework to regularise inverse problems so as to deliver accurate and well-posed solutions. It is the framework of this thesis and it includes most of the variational methods. In this context, both data and model parameters are considered as the realization of some random variables. Accordingly, the relationship between x and y is described by a statistical model with *likelihood* function $p(y|x)$ or by the potential $F(x, y) = -\log p(y|x)$. The knowledge about x before observing y is encoded by the *marginal* or *prior distribution* for x, typically specified via a density function p or by its potential $U(x) = -\log p(x)$. Unless explicitly stated otherwise, we henceforth assume that all densities are defined w.r.t. to the appropriate Lebesgue measure. The likelihood and prior define the *joint distribution* with density $p(x, y) = p(y|x)p(x)$, from which we derive the *posterior distribution* with density

$$p(x|y) = p(y|x)p(x)/p(x, y) = p(y|x)p(x)/\int_{\mathbb{R}^d} p(y|z)p(z)\mathrm{d}z \ . \tag{3.7}$$

However, the Bayesian framework proposes far more than a simple reconstruction based on an observation y, as it produces (after applying the Bayes' theorem) the posterior distribution. It describes all the possible solutions given the observation y. In addition, having access to the posterior distribution allows to estimate uncertainty. Most imaging methods seek to derive estimators reaching some kind of consensus between prior and likelihood and summarizing the posterior distribution, for instance the Minimum Mean Square Error (MMSE) or Maximum A Posteriori (MAP) estimators

$$\hat{x}_{\text{MAP}} = \arg\max_{x\in\mathbb{R}^d} p(x|y) = \arg\min_{x\in\mathbb{R}^d} \{F(x, y) + U(x)\} \ , \tag{3.8}$$

$$\hat{x}_{\text{MMSE}} = \arg\min_{u\in\mathbb{R}^d} \mathbb{E}[\|x - u\|^2|y] = \mathbb{E}[x|y] = \int_{\mathbb{R}^d} \tilde{x}p(\tilde{x}|y)\mathrm{d}\tilde{x} \ . \tag{3.9}$$

Computing the MAP is very appealing as it boils down to solve an optimization problem as in the variational case. Consequently, computing the MAP under a specific prior $p(x) \propto e^{-\lambda\mathcal{S}(x)}$ is equivalent to solve an optimization with the regularization term $\lambda\mathcal{S}(x)$.

Estimating the MMSE is on the contrary more difficult as it implies estimating high-dimensional integrals. Although computing the MAP is attractive from a computational point of view, it is still not correctly understood from a theoretical point of view in the Bayesian framework. Indeed, an estimator is a Bayes estimator if it minimizes the expected cost for a given cost function $\mathcal{C}$ under the posterior distribution $p(x|y)$

$$\mathcal{R}_{\mathcal{C}}(\tilde{x}) = \mathbb{E}_{p(y)}\mathbb{E}_{p(z|y)}[\mathcal{C}(z,\tilde{x})] = \int_{y\in\mathcal{Y}}\int_{z\in\mathcal{X}} \mathcal{C}(z,\tilde{x})p(z|y)\mathrm{d}z\ p(y)\ \mathrm{d}y\ . \tag{3.10}$$

As explained in (Pereyra, 2019), if the posterior is not log-concave, we do not know if the MAP minimizes (3.10) under a cost function $\mathcal{C}$. On the contrary, the MMSE estimator minimizes (3.10) with $\mathcal{C} : \mathcal{X}\times\mathcal{X} \to \mathbb{R}, (z_1, z_2) \mapsto \|z_1 - z_2\|_2^2$.

At this point, we point out the fact that if both the prior and the likelihood are Gaussian, then the MAP and the MMSE estimators are equal.

The quality of the inference about x given y depends on how accurately the specified prior represents the true marginal distribution for x. Most works in the Bayesian imaging literature consider relatively simple priors promoting sparsity in transformed domains or piece-wise regularity (e.g., involving the ℓ_1 norm or the total-variation pseudo-norm (Rudin et al., 1992; Chambolle, 2004; Louchet and Moisan, 2013; Pereyra, 2016)), Markov random fields (MRF, 2011), or learning-based priors like patch-based Gaussian or Gaussian mixture models (Zoran and Weiss, 2011; Yu et al., 2011; Aguerrebere et al., 2017; Teodoro et al., 2018b; Houdard et al., 2018). Special attention is given in the literature to models that have specific factorisation structures or that are log-concave, as this enables the use of Bayesian computation algorithms that scale efficiently to high-dimensions and which have detailed convergence guarantees, (Pereyra, 2016; Durmus et al., 2018; Repetti et al., 2019; Girolami and Calderhead, 2011; Chen et al., 2014).

Methods related to the Bayesian approach are described in Section 3.5.

3.3 Learning Approach with Deep Neural Networks in imaging science

With the development of computational resources, a lot of efforts have been made to develop artificial *neural networks* and their applications. They are mapping inspired by the working of human brains. They consist in interconnected nodes called *neurons*. As in the human brain, artificial neurons are trained to indirectly react to some stimuli and fire when they perceive one. Their expressivity makes them a powerful tool to solve inverse problems (Cybenko, 1989; Hornik, 1991). In this section, we expose some basics components about neural networks. We are only interested in *feedforward neural networks*.

3.3.1 Neural networks

Neural networks are oriented graphs where each node is called a neuron. Neurons are grouped into layers. The first layer is called the *input layer* and the last one the *output layer*. Every layers between theses layers are called *hidden layers*.

When using a neural network, we have to determine an *architecture*, *ie* basic components composing our neural network such as the activation functions, the connection type between layers, the number of layers, etc, ... There exist different types of architectures. In this section, we present different types of architectures used in imaging science when performing point estimation.

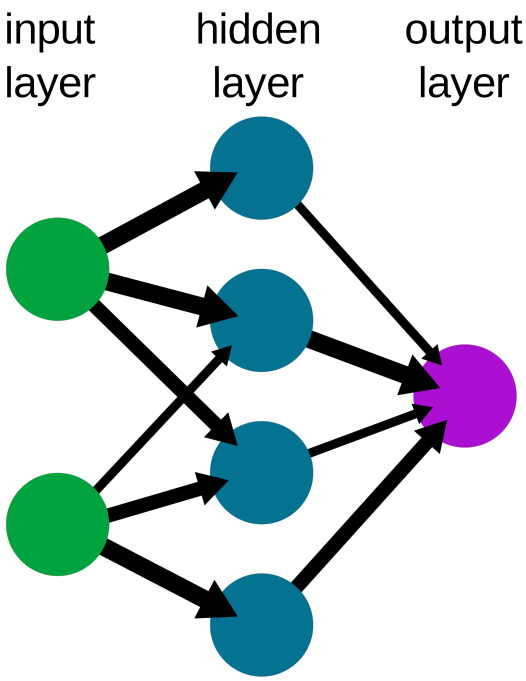


Figure 3.3: Simplified neural network. Each node corresponds to a neuron and each edge corresponds to a weight. Neurons of the same colour are grouped into layers. Image from Wikipedia.

FEEDFORWARD NEURAL NETWROK Let us consider a neural network with H layers and $H-2$ hidden layers. We denote $\{n_0, n_1, ..., n_H\} \in \mathbb{N}^{H+1}$ each layer size and $f_h(x)$ the result obtained when computing the result of the $h-$th layer for the input $x \in \mathbb{R}^{n_0}$. The mapping $f_h : \mathbb{R}^{n_0} \to \mathbb{R}^{n_0}$ is defined by

$$f_h(x) = \begin{cases} x & \text{if } h = 0 \\ \sigma(W_h f_{h-1}(x) + b_h) & \text{otherwise,} \end{cases} \tag{3.11}$$

where $W_h \in \mathbb{R}^{n_{h-1} \times n_h}$ is a matrix with entries corresponding to the weights of each edge, $b_h \in \mathbb{R}^{n_h}$ is the bias added by the h-th layer and σ is an activation function, usually non-linear, like the Rectified Linear Unit (ReLU) or the sigmoid function. $(W_h, b_h)_{h \in \{1,2,..H,\}}$ are parameters to learn.

If every neuron from a layer are connected to each neuron of the following one, we talk about *fully-connected* neural network.

The depth of neural network corresponds to its number of hidden layers. Works such as (Eldan and Shamir, 2016) and (Daniely, 2017) support the idea, the deeper is a neural network, the more expressive it is.

CONVOLUTIONAL NEURAL NETWORK (CNN) However, when dealing with images, which are high-dimensional objects, fully-connected neural networks imply too many parameters to set. We might want to opt for a different *architecture*. In this case, we prefer to use *convolutional neural networks*. They are feedforward neural networks with a specific structure. The equation (3.11) also describes their working process, except that $(W_h)_{h \in \{1,...,H\}}$ results from the concatenation of convolutional operators. Then a neuron in the h-th layer results from

a combination of only a few neurons of the $h-1$-th layer located in the same region. The weights performing these convolutions are shared between neurons that is why they involve less parameters. In addition to have a lot less parameters, CNNs are approximately shift-invariant (due to boundary effects), which is a desirable property when dealing with images. Indeed, important features can be located anywhere in the input space and CNNs can then detect feature regardless their locations. On the contrary, fully-connected neural networks for instance are not shift-invariant. Then, to be able to detect a specific feature, they have to see this feature in different locations to be sure that the neurons of each region can spot it.

Moreover, within a layer, other operations can be added. A very common operation is *max-pooling* that allows to reduce the dimension of the input signal by summarizing a zone by its largest element. Furthermore, we emphasize the use of U-Nets that were firstly developed for image segmentation in the biomedical context (see (Ronneberger et al., 2015)). These CNNs have successive layers where pooling operations are replaced by upsampling operators allowing the network to propagate context information to higher resolution layers.

RESIDUAL NEURAL-NETWORK (RES-NET) They were presented in (He et al., 2016). Skip connections are used to jump over some layers. Typical ResNet models are implemented with double- or triple- layer skips that contain nonlinearities like ReLU and batch normalization in between. The equation (3.12) describes the working process of such networks.

$$f_h(x) = \begin{cases} x & \text{if } h = 0 \\ \sigma(W_h \sigma(W_{h-1} f_{h-2}(x) + b_{h-1}) + b_h + f_{h-2}(x)) & \text{otherwise.} \end{cases} \tag{3.12}$$

Adding skip connections allows us to avoid the problem of vanishing gradients that considerably slows down the neural netowrk training phase. Then it eases the neural networks learning. In addition, skip connections mitigates the degradation problem sometimes encountered when training a neural network as reported by (He et al., 2016). Indeed, with an increasing depth, which is often associated with a better expressivity, the neural network accuracy gets saturated and finally degrades.

3.3.2 Learning process with neural networks

Training a neural network means, given a specific architecture, learning the best set of parameters $\theta = W_h, b_{h\,h\in\{1,\dots,H\}}$ to solve our inverse problem. In supervised learning, we seek to determine it over a training set $(x_i, y_i)_{\{i=1,\dots,N\}}$ consisting in input-output examples, where x_is and y_is are respectively the target solutions and their associated observations. These pairs are considered as i.i.d samples from the joint distribution with density $p(x, y)$. The goal is to find θ^* such that

$$\theta^* = \arg\min_{\theta\in\Theta} \sum_{i=1}^{N} \mathcal{L}(x_i, f_\theta(x_i)) \,, \tag{3.13}$$

where Θ is the set of admissible parameters, $\mathcal{L}$ is the cost function. This cost function is often quadratic.

The optimal parameter θ^* is often computed using stochastic optimization algorithms such as the *stochastic gradient descent* (Goodfellow et al., 2016) or *Adam* (Kingma and Ba, 2014).

If the size of the training set N is large enough, according to the law of large numbers, we have $\mathbb{E}_{(X,Y)}[\mathcal{L}(x, f_\theta(y))] \approx \sum_{i=1}^{N} \mathcal{L}(x_i, f_\theta(y_i))$.

3.3.3 Neural networks for point estimation

In the last few years, deep neural networks have become ubiquitous to solve inverse problems in imaging, showing unmatched performance for point estimation for some specific problems like image denoising. Deep networks can be trained without explicitly using the knowledge of the forward model (3.1) (Dong et al., 2014; Zhang et al., 2017, 2018; Gharbi et al., 2016; Schwartz et al., 2018; Gao et al., 2019) or on the contrary can use this model explicitly via unrolled optimization techniques (Gregor and LeCun, 2010; Chen and Pock, 2017; Diamond et al., 2017; Gilton et al., 2019). Then, the goal is to learn a deep neural network approximating an operator $\mathcal{R} : \mathcal{Y} \to \mathcal{X}$ that is implicitly defined via an iterative scheme. This scheme often aims at finding the minimum of a functional.

Also, imaging approaches based on neural networks struggle to support more advanced inference by comparison to a Bayesian treatment by Monte Carlo sampling, which can support a wide breadth of statistical analyses beyond point estimation, particularly Bayesian decision-theoretic approaches to deal with advanced forms of uncertainty quantification (e.g., hypothesis tests, p-values, model misspecification tests), as well as approaches to deal with automatic calibration of partially unknown models and objective model comparison (Robert, 2007).

3.3.4 Neural networks for sampling

MONTE-CARLO DROPOUT In order to go further than simple point estimation, (Gal and Ghahramani, 2016) develop a new theoretical framework casting dropout training in deep neural networks where dropout consists in randomly omitting each hidden unit with probability P. Then, a neural network with dropout can be interpreted as a variational Bayesian approximation and allows to perform uncertainty quantification.

GENERATIVE MODELING WITH NEURAL NETWORKS To go further than simple point estimation, generative neural networks were developed. These neural networks are trained in order to sample from a target distribution $\pi \in \mathcal{P}(\mathbb{R}^d)$ given an empirical distribution $\hat{\pi} = (1/N)\sum_{i=1}^{N} \delta_{x_i}$ where the x_i are samples of the target distribution π. These methods consist in learning a mapping g that transforms samples of a easy-to-sample distribution $\pi_0 \in \mathcal{P}(\mathbb{R}^p)$, like a Gaussian distribution, into samples from the target distribution π. There exist 3 major types of neural networks to sample from a distribution namely *Variational Auto-Encoders* (VAEs) (Kingma and Welling, 2019), *Normalizing Flows* (NFs) and *Generative Adversarial Networks* (GANs). In this section, we briefly introduce these neural networks.

VAEs have a similar structure to autoencoders which consist of two networks.

- The *Encoder* $G : \mathbb{R}^d \to \mathbb{R}^p$ maps the samples from π_0 to a latent variable in $\mathbb{R}^p$. Usually, $p < d$.
- The *Decoder* $D : \mathbb{R}^p \to \mathbb{R}^d$ maps back latent variables in $\mathbb{R}^p$ to images in $\mathbb{R}^d$.

The latent space corresponds to a space where it is easier to manipulate the data. It supports the idea that images are concentrated on a low-dimensional sub-manifold.

The idea behind VAE is to model the latent space in a probabilistic manner. Then the encoder and the decoder are both associated with distributions as shown in Figure 3.4. In

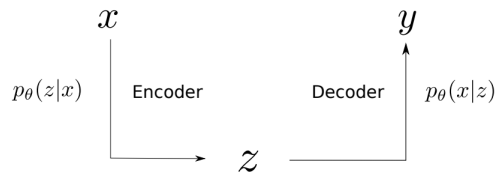


Figure 3.4: Structure of a VAE

this case we sample a latent variable z from π_0 in the latent space and use the decoder to map into an image.

We want to encourage the latent space to follow the distribution π_0 which is usually Gaussian.

GANs consist of two neural networks trained in a competitive manner (Goodfellow et al., 2014).

- The *Generator* $G : \mathbb{R}^p \to \mathbb{R}^d$ maps the samples from π_0 to images in $\mathbb{R}^d$.
- The *Discriminator* $D : \mathbb{R}^d \to [0, 1]$ tries to distinguish samples from $\hat{\pi}$ for which $D(x_i) = 1$ from artificial images generated from π_0 and for which $D(G(z)) = 0$.

The loss reads

$$\min_G \max_G \{\mathbb{E}_x \log D(x) + \mathbb{E}_z \log[1 - G(D(z))]\} .$$

NFs are such that $p = d$. The goal is to train an *invertible* neural network T that maps images to latent representation $z = T(x)$ classically normally distributed (Papamakarios et al., 2019). Then, we have

$$\pi(x) = \pi_0(T(x)) \times |\det J_T(x)| . \tag{3.14}$$

We have an explicit expression of the target density π. In order to be exploitable, $|\det J_T|$ has to be easily computable. That is why T is often composed of simple transformations with triangular Jacobians.

In addition, stochastic normalizing flows (SNFs) were introduced in (Wu et al., 2020) and consist of a sequence of deterministic flow transformations and stochastic sampling methods with tractable paths, such as Markov Chain Monte Carlo (MCMC) or overdamped Langevin dynamics.

There are several survey papers for VAEs (Kingma and Welling, 2019), GANs (Creswell et al., 2018; Wang et al., 2017) and NFs (Papamakarios et al., 2019; Kobyzev et al., 2021) providing a comprehensive review of the literature for distribution learning.

Sampling the posterior distribution Based on the success of the generative neural networks previously introduced, a lot of effort has been made to sample from the posterior distribution using a neural network in order to solve inverse problems. The basic idea is to add an input to the networks for the observation y. (Mirza and Osindero, 2014) introduced conditional GANs (cGANs), constructed by simply feeding the observation y, we wish to condition on to both the generator and discriminator. cGANS were applied to solve inverse

problems such as image reconstruction in ultra low dose 3D helical CT (Adler and Öktem, 2018b) or Partial-Differential-Equation (PDE) based inverse problems (Ray et al., 2022). (Goh et al., 2022) proposes to train a VAE to solve a PDE-based inverse problem and to perform uncertainty quantification. Eventually, (Lugmayr et al., 2020) is an example where a NF is trained to solve an ill-posed inverse problem in imaging, super-resolution. The network learns the conditional distribution of the solution to the inverse problem given the low-resolution input y. SNFs were firstly applied to tackle inverse problems in (Hagemann et al., 2022).

3.3.5 Limitations of the pure neural network based approaches

One disadvantage of using neural networks to solve imaging inverse problems is that in order to achieve state-of-the-art performance it is usually necessary to train the network for a specific problem configuration. Then, the network must be retrained if the forward model, *ie* A or the noise distribution change, or any model parameters change significantly. Solutions encoded by end-to-end neural networks are mostly problem specific and not easily adapted to reflect changes in the problem (e.g., in instrumental settings). There also exist concerns regarding the stability of such approaches for general reconstruction problems (Antun et al., 2020, 2021).

Furthermore sampling neural networks generally lack convergence proofs. Then, it is not clear which distribution we are sampling from.

In order to address these limitations, we consider Plug & Play methods for posterior sampling and Maximum-a-Posteriori estimation. Related work on PnP MAP estimation is discussed in Section 3.4 and our contributions to this subject are detailed in Chapter 4. Related work on PnP posterior sampling is discussed in Section 3.6 and our contributions to this subject are detailed in Chapters 5 and 6.

3.4 A survey of Plug & Play methods for estimating the MAP in imaging

Plug & Play methods try to combine the strengths of the Bayesian paradigm and the neural networks. It consists to specify an implicit prior defined via a denoiser. These data-driven regularisation approaches learn an implicit representation of the prior density $p(x)$ (or its potential $U(x) = -\log p(x)$) while keeping an explicit likelihood density, which is usually assumed to be known and calibrated (Arridge et al., 2019)

In the context of imaging inverse problems, *Plug & Play* methods aim at using a carefully chosen denoiser $D_\varepsilon : \mathbb{R}^d \to \mathbb{R}^d$ to implicitly define an image prior. This is achieved by relating D_ε to a proximal operator or a gradient associated with the prior density. In the first case, D_ε replaces a MAP estimator for a denoising problem. In the second case, D_ε replaces a Minimum Mean Square Error (MMSE) estimator for a denoising problem, related to the gradient of a log-prior via Tweedie's identity [†](Robbins, 1956; Miyasawa et al., 1961; Efron, 2011).

[†]Notice that although it is conceptually helpful to distinguish these two cases (in order to make a historical and practical survey of the subject), there are clear theoretical connections between the two approaches. Indeed, under regularity conditions on the Bayesian model involved, MAP denoisers can be expressed as MMSE denoisers under an alternative (albeit often unknown) Bayesian model (Gribonval, 2011). However this equivalence can not always be exploited in practice and has been mostly ignored in the literature on *Plug & Play* methods until very recently with the work of Xu *et al.* (Xu et al., 2020) to be presented later.

In what follows, we describe how these approaches have been widely used to compute the MAP estimator as a solution to the considered inverse problem. In our discussion, we pay particular attention to questions related to algorithmic convergence, and to the interpretation of the computed solutions, as this has been an important focus of the literature.

3.4.1 Plug & Play MAP estimators using proximal splitting

Let $D_\varepsilon^\dagger$ denote the MAP estimator to recover x from a noisy observation $x_\varepsilon \sim \mathcal{N}(x, \varepsilon\,\mathrm{Id})$ under the assumption that x has marginal density $p(x) \propto \exp[-U(x)]$; that is,

$$D_\varepsilon^\dagger(x_\varepsilon) = \operatorname*{arg\,min}_{x\in\mathbb{R}^d}\{\frac{1}{2}\|x_\varepsilon - x\|^2 + \varepsilon U(x)\} = \mathrm{prox}_{\varepsilon U}(x_\varepsilon)\ .$$

When we set the PnP denoiser D_ε such that $D_\varepsilon = D_\varepsilon^\dagger$, any optimization scheme making use of a proximal descent on the prior can be used to solve (3.8) via D_ε.

For instance, the alternating direction method of multipliers (ADMM) (Glowinski and Marroco, 1975; Boyd et al., 2011) writes the augmented Lagrangian of (3.8) as

$$E_\varepsilon(x, z, v) = F(x, y) + \|x - z\|^2/(2\varepsilon) + v^\top(x - z) + U(z)\ .$$

The joint optimization of the augmented Lagrangian is given by

$$(\hat{x}_{\mathrm{MAP}}, \hat{z}_{\mathrm{MAP}}) = \arg\min_{x,z\in\mathbb{R}^d} \max_{v\in\mathbb{R}^d} E_\varepsilon(x, z, v).$$

This provides the solution $\hat{x}_{\mathrm{MAP}} = \hat{z}_{\mathrm{MAP}}$ of (3.8) when $\varepsilon \to 0$.

In practice, the joint optimization is solved by an alternate minimization scheme on x and z and a gradient ascent on $u = \varepsilon v$,

$$\begin{aligned}
x_{k+1} &= \arg\min_x E_\varepsilon(x, z_k, u_k/\varepsilon) &&= \mathrm{prox}_{\varepsilon F(\cdot,y)}(z_k - u_k)\ , && (3.15)\\
z_{k+1} &= \arg\min_z E_\varepsilon(x_{k+1}, z, u_k/\varepsilon) &&= \mathrm{prox}_{\varepsilon U}(x_{k+1} + u_k) = D_\varepsilon(x_{k+1} + u_k)\ , && (3.16)\\
u_{k+1} &= u_k + x_{k+1} - z_{k+1}\ . && && (3.17)
\end{aligned}$$

Similarly, when $F(., y)$ is differentiable, the simpler Forward-backward splitting (FBS) scheme (Combettes and Pesquet, 2011), which only requires to compute ∇F, can be written in a Plug-and-Play fashion as

$$x_{k+1} = \mathrm{prox}_{\varepsilon U}(x_k - \varepsilon\nabla F(x_k, y)) = D_\varepsilon(x_k - \varepsilon\nabla F(x_k, y))\ . \tag{3.18}$$

A fully proximal version of this algorithm, called Backward-backward splitting (BBS) (Combettes and Pesquet, 2011), writes

$$x_{k+1} = \mathrm{prox}_{\varepsilon U}(\mathrm{prox}_{\varepsilon F}(x_k)) = D_\varepsilon(\mathrm{prox}_{\varepsilon F}(x_k))\ . \tag{3.19}$$

BBS aims at solving a slightly modified version of (3.8) where F is replaced by its Moreau envelope with parameter ε. The same algorithm can be derived using half-quadratic splitting to solve (3.8).

When U is convex, such splitting schemes and many variants (including primal-dual methods, ISTA or FISTA, etc.) are well understood and proved to converge to the global optimum (Boyd et al., 2011). They have also been successfully used for non-convex U like patch-based Gaussian mixture models (GMM) as pioneered for external learning by (Zoran and Weiss, 2011). The use of splitting schemes with non-convex GMM priors was later refined with convergence guarantees for scene-adapted learning (Teodoro et al., 2018a).

Following the seminal work of (Venkatakrishnan et al., 2013), this kind of splitting schemes have become ubiquitous in cases where U (and hence $D_\varepsilon^\dagger$) are unknown and unspecified, but a denoiser D_ε is available and assumed to be a good approximation of $D_\varepsilon^\dagger = \mathrm{prox}_{\varepsilon U}$. As popular and efficient these methods have become, their convergence properties have remained largely unknown. Indeed, for most denoisers D_ε, there is no guarantee that there exists a potential U such that $D_\varepsilon = \mathrm{prox}_{\varepsilon U}$. (Sreehari et al., 2016) establish some sufficient conditions for this to happen: D_ε must be differentiable, and its Jacobian J_{D_ε} should be symmetric with eigenvalues within the $[0, 1]$ interval to ensure non expansiveness. These assumptions hold for transform-domain thresholding denoisers and for variants of Non Local Means (Buades et al., 2005a) where symmetry is explicitly enforced (Sreehari et al., 2016). More recently, it has also been shown (Nair et al., 2021) that a special class of linear denoisers (including Non Local Means (Buades et al., 2005a)) are proximal operators of some closed, proper functions. This approach necessitates to work with a non standard inner product though. The previous proofs do not hold for most popular denoisers, including BM3D (Dabov et al., 2006), Non Local Bayes (Lebrun et al., 2013) and neural networks denoisers like DnCNN (Zhang et al., 2017), as observed in (Reehorst and Schniter, 2018).

CONSENSUS EQUILIBIRUM / FIXED POINT INTERPRETATION. Since it remains difficult to show that PnP schemes converge to the MAP or even a critical point of (3.8), several authors have proposed to analyse these schemes from a consensus equilibrium point of view (Buzzard et al., 2018; Ahmad et al., 2020), or similarly to consider and analyse these approaches as fixed-point algorithms (Sun et al., 2019; Ryu et al., 2019). The fixed points attained by these algorithms cannot be interpreted as MAP estimators, but should be seen as solving a set of equilibrium equations involving both the denoiser and the data term. For instance, for PnP-FBS, the idea is to show convergence to the set of points x satisfying $x = D_\varepsilon(x - \varepsilon \nabla F(x, y))$. It can easily be shown that the fixed points of several of these PnP algorithms (in particular PnP-ADMM and PnP-FBS) coincide (Meinhardt et al., 2017; Sun et al., 2019).

Assuming that such fixed points exist, Sun et al. (2019) show convergence of PnP-ISTA (which is equivalent to PnP-FBS above) under the assumptions that ∇F is $\mathtt{L}_y$-Lipschtitz, $\varepsilon \mathtt{L}_y \leqslant 1$ and D_ε is θ-averaged, see (Bauschke et al., 2011, Definition 4.33) for a definition. This assumption on the denoiser is probably too strong, since most denoisers cannot be considered as averaged operators. (Sun et al., 2020) reformulate PnP-ADMM with different convergence conditions, and still assume quite restrictive conditions on the denoiser D_ε [‡].

(Ryu et al., 2019) propose a convergence analysis of PnP-ADMM, PnP-FBS and PnP-DRS (PnP Douglas-Rachford Splitting), based on the weaker assumption that the residual operator $D_\varepsilon - \mathrm{Id}$ is $\mathtt{L}$-Lipschitz with a Lipschitz constant which depends both on the data fitting term F and the denoiser D_ε. The proof also requires F to be μ-strongly convex (which excludes all cases where A is not full rank and de facto excludes some of the applications considered in (Ryu et al., 2019)) and it imposes quite restrictive assumptions on relative values of μ, ε and $\mathtt{L}$.

In a similar direction, (Xu et al., 2020) very recently proposed a convergence study for PnP-ISTA, with the assumption that ∇F is $\mathtt{L}_y$-Lipschitz with $\varepsilon \mathtt{L}_y \leqslant 1$. However, they assume that D_ε is an exact MMSE denoiser, *i.e.* $D_\varepsilon(x_\varepsilon) = \mathbb{E}[X | X_\varepsilon = x_\varepsilon]$, where $X \sim p$ and $X_\varepsilon - X \sim \mathcal{N}(0, \varepsilon \mathrm{Id})$. Therefore their theoretical results do not carry to many classical denoisers, such that those learned from training data and implemented by neural networks.

[‡] In (Sun et al., 2020), the residual $\mathrm{Id} - D_\varepsilon$ is assumed to be firmly non expansive, which is equivalent to say that D_ε is firmly non expansive, see (Bauschke et al., 2011, Proposition 4.4).

ASSUMPTIONS ON ALGORITHM PARAMETERS. Most of the convergence proofs for PnP algorithms impose restrictive assumptions on the choice of parameters used in the iterative schemes. This may exclude interesting ranges of parameters for several inverse problems. For instance, for PnP-FBS, the parameter ε (which can be interpreted as the step of the proximal or gradient descents) and the Lipschitz parameter $\mathtt{L}_y$ of ∇F must typically be chosen such that $\mathtt{L}_y\varepsilon \leqslant C$ with $C \in [1,2]$ (see (Ryu et al., 2019; Xu et al., 2020), the exact value of C depends on the convergence proof). If $F(x,y) = \frac{1}{2\alpha\sigma^2}\|\mathrm{A}x - y\|^2$, with $\|\mathrm{A}\| \leq 1$, it implies that $\frac{1}{\alpha} \leqslant \frac{\sigma^2}{\varepsilon}$. The parameter ε is imposed by the denoiser D_ε (the denoiser is trained for a noise of variance ε), and σ is given by the quantity of noise in the forward model. If, for instance, the forward model involves a noise standard deviation σ which is 5 times smaller than the one used for the denoiser D_ε, it means that the penalty α (which balances the respective weights of the data and prior terms) should be chosen larger than 25, which implies that the algorithm will only converges for huge regularizations. We will see in Section 4.3 that for this kind of reason the PnP-FBS algorithm often fails to converge for classical imaging inverse problems, or converges only for values of α which are not interesting in practice. Fully proximal algorithms such as PnP-ADMM or PnP-BBS are much more robust in practice, even when the conditions of their theoretical convergence are not fully met. The PnP-SGD algorithm that we will introduce in the following does not suffer from the same convergence limitations.

AMP ALGORITHMS. It is worth mentioning at this point that the Plug-and-Play framework has also been shown to be very efficient with Approximate Message Passing algorithms (Ahmad et al., 2020). These algorithms have excellent convergence properties for data terms of the form $\|\mathrm{A}x-y\|^2$ with A belonging to specific classes of random matrices. This restriction on A does not hold for the inverse problems considered in this thesis so we focus instead on classical optimization scheme such as the ones described above.

3.4.2 Plug & Play MAP estimators using gradient descent

Now, assume that $D_\varepsilon = D^\star_\varepsilon$, where $D^\star_\varepsilon$ is the MMSE estimator to recover x from the noisy observation x_ε with $(X_\varepsilon|X=x) \sim \mathcal{N}(x, \varepsilon\,\mathrm{Id})$ when X has marginal density p; that is,

$$D^\star_\varepsilon(x_\varepsilon) = \mathbb{E}[X|X_\varepsilon = x_\varepsilon] = \textstyle\int_{\mathbb{R}^d} zp(z)G_\varepsilon(x_\varepsilon - z)\mathrm{d}z / \int_{\mathbb{R}^d} p(z)G_\varepsilon(x_\varepsilon - z)\mathrm{d}z\ , \tag{3.20}$$

where G_ε is a Gaussian kernel with variance ε, meaning that for all $x \in \mathbb{R}^d$,

$$G_\varepsilon(x) = (2\pi\varepsilon)^{-d/2}\exp[-\|x\|^2/(2\varepsilon)]\ .$$

We introduce the following class of smooth approximations of $p(x)$, defined for any $x \in \mathbb{R}^d$ by

$$p_\varepsilon(x) = \textstyle\int_{\mathbb{R}^d} p(\tilde{x})G_\varepsilon(x-\tilde{x})\mathrm{d}\tilde{x}\ . \tag{3.21}$$

In this case, Tweedie's identity (Efron, 2011) establishes the following relationship between the MMSE denoiser $D^\star_\varepsilon$ and (3.21), for any $x \in \mathbb{R}^d$

$$\nabla U_\varepsilon(x) = -\nabla \log p_\varepsilon(x) = (x - D^\star_\varepsilon(x))/\varepsilon\ , \tag{3.22}$$

where $U_\varepsilon = -\log(p_\varepsilon)$. This relation can be used to plug the MMSE denoiser $D^\star_\varepsilon$ in any gradient descent scheme involving ∇U_ε as follows

$$X_{k+1} = X_k - \delta\nabla F(X_k, y) - \delta\nabla U_\varepsilon(X_k) + \delta Z_{k+1}\ , \tag{3.23}$$

or

$$X_{k+1} = X_k + \delta \nabla \log p(y|X_k) + \frac{\delta}{\varepsilon}(D^\star_\varepsilon(X_k) - X_k) + \delta Z_{k+1} \,, \tag{3.24}$$

where $\{Z_k \,:\, k \in \mathbb{N}\}$ is a sequence of i.i.d Gaussian random variables with zero mean and identity covariance matrix and $\delta > 0$ is the gradient descent step-size.

It is at the core of the algorithm PnP-SGD presented in Chapter 4.

Similarly to the MAP denoiser $D^\dagger_\varepsilon$, the MMSE denoiser $D^\star_\varepsilon$ is usually not known, so PnP methods rely on other denoisers D_ε that are believed to be good approximations of $D^\star_\varepsilon$. Observe that CNN denoisers are usually trained to minimize an empirical quadratic risk on a large database of natural images. As a consequence, they naturally produce good approximations of MMSE denoisers $D^\star_\varepsilon$ for realistic image priors. This makes approaches based on Tweedie's identity particularly attractive. On the other hand, learning mechanisms to produce good approximations of MAP denoisers $D^\dagger_\varepsilon$ are much less widespread, although under some conditions, MMSE denoisers can be shown to be MAP denoisers on a different prior (see (Gribonval, 2011; Xu et al., 2020; Hurault et al., 2022b)).

A similar relation is derived by (Romano et al., 2017) where they present the Regularization by Denosing (RED) method, which proposes an insightful Bayesian formulation of denoiser-based priors as image-adaptive Laplacian regularisations. Instead of using Tweedie's identity, the RED method solves (3.8) via different optimization algorithms (including gradient descent and ADMM) with explicit regularization $U_\varepsilon(x) = (1/2)\langle x,\, x - D_\varepsilon(x)\rangle$. As shown in (Reehorst and Schniter, 2018), under the assumptions that D_ε is locally homogeneous and has symmetric Jacobian, this implies that for any $x \in \mathbb{R}^d$, $\nabla U_\varepsilon(x) = x - D_\varepsilon(x)$, which is (up to a scaling factor $1/\varepsilon$) the same expression as Tweedie's identity in (3.22). Unfortunately, as pointed out before, these assumptions on D_ε are not strictly satisfied by most commonly used denoisers (Reehorst and Schniter, 2018), although we note that Jacobian symmetry can be explicitely enforced (Milanfar, 2013). The convergence of the RED algorithms for denoisers that do not verify the above-mentioned assumption remains unproven. As an alternative interpretation the RED algorithm can be seen as a way to approximate the score ∇U_ε by $(x - D_\varepsilon(x))/\varepsilon$ in the optimality equation $\nabla F + \nabla U_\varepsilon = 0$. Here the optimal MMSE denoiser $D^\star_\varepsilon$ is again replaced by some other denoiser.

More recently, (Cohen et al., 2020) studies a projected RED estimator which seeks to minimize a data fidelity term subject to the constraint that the solution belongs to the set of fixed points $\{x \in \mathbb{R}^d : x = D_\varepsilon(x)\}$, thus sharing strong link with the consensus equilibrium interpretation of proximal-based PnP estimators. It is reported in (Cohen et al., 2020) that when D_ε is a demi-contractive mapping, its fixed points define a convex set, which allows the construction of provably convergent algorithms for this alternative RED estimator. However, as pointed out in (Pesquet et al., 2020), verifying that a given denoising operator is demi-contractive is not easy and, to be the best of our knowledge, it is not yet clear what denoisers verify this property. Furthermore, from a Bayesian inference viewpoint, additional studies would be required in order to determine when this projected RED estimator defines or approximates a MAP estimator for a suitable Bayesian model.

In a similar direction, two very recent works (Hurault et al., 2022a,b) show how to train efficiently a denoiser that explicitly satisfies $D_\varepsilon(x) = x - \nabla g_\varepsilon(x)$ for some functional g_ε. Plugging this denoiser in appropriate PnP schemes, they are able to prove convergence to stationary points of an explicit cost function.

The PnP-SGD optimisation algorithm that will be presented in Chapter 4 is very close to the gradient descent version of RED presented in (Romano et al., 2017). We will show

that it converges to the vicinity of the solution of (3.8) under much milder conditions than previously assumed. In particular, the convergence guarantees hold even when D_ε is not an exact MAP or MMSE denoiser, which is often the case in practice. Importantly, our convergence guarantees hold for the neural network denoiser used in (Ryu et al., 2019) (a variant of DnCNN (Zhang et al., 2017) with a contractive residual) and also for the native Non Local Means (Buades et al., 2005b).

3.5 Posterior sampling in imaging

In this section we review some of the methods commonly used in imaging for sampling. There is a vast literature on Bayesian computation methodology for models related to imaging sciences (see, e.g., (Pereyra et al., 2015)). Here, we briefly summarise efficient high-dimensional Bayesian computation strategies derived from the Langevin stochastic differential equation (SDE)

$$\mathrm{d}\mathbf{X}_t = \nabla \log p(\mathbf{X}_t|y) + \sqrt{2}\mathrm{d}\mathbf{B}_t = \nabla \log p(y|\mathbf{X}_t) + \nabla \log p(\mathbf{X}_t) + \sqrt{2}\mathrm{d}\mathbf{B}_t \ , \tag{3.25}$$

where $(\mathbf{B}_t)_{t\geqslant 0}$ is a d-dimensional Brownian motion. When $p(x|y)$ is proper and smooth, with $x \mapsto \nabla \log p(x|y)$ Lipschitz continuous[§], then, for any initial condition $\mathbf{X}_0 \in \mathbb{R}^d$, the SDE (3.25) has a unique strong solution $(\mathbf{X}_t)_{t\geqslant 0}$ that admits the posterior of interest $p(x|y)$ as unique stationary density Roberts et al. (1996). In addition, for any initial condition $\mathbf{X}_0 \in \mathbb{R}^d$ the distribution of $\mathbf{X}_t$ converges towards the posterior distribution in total variation. Although solving (3.25) in continuous time is generally not possible, we can use discrete time approximations of (3.25) to generate samples that are approximately distributed according to $p(x|y)$. A natural choice is the Unadjusted Langevin algorithm (ULA) Markov chain $(X_k)_{k\geqslant 0}$ obtained from an Euler-Maruyama discretisation of (3.25), given by $X_0 \in \mathbb{R}^d$ and the following recursion for all $k \in \mathbb{N}$

$$X_{k+1} = X_k + \delta \nabla \log p(y|X_k) + \delta \nabla \log p(X_k) + \sqrt{2\delta} Z_{k+1} \ , \tag{3.26}$$

where $\{Z_k \ : \ k \in \mathbb{N}\}$ is a family of i.i.d Gaussian random variables with zero mean and identity covariance matrix and $\delta > 0$ is a step-size which controls a trade-off between asymptotic accuracy and convergence speed (Dalalyan, 2017; Durmus and Moulines, 2017). The approximation error involved in discretizing (3.25) can be asymptotically removed at the expense of additional computation by combining (3.26) with a Metropolis-Hastings correction step, leading to the so-called Metropolis-adjusted Langevin Algorithm (MALA) (Roberts et al., 1996).

It is interesting to draw comparison between ULA and SGD. As for the SGD update rule, the red term in the ULA update rule points towards the target distribution modes. However, the square root on on δ in the green term allows exploration and consists in the main difference with SGD.

When the prior density $p(x)$ is log-concave but not smooth, one can still use ULA by approximating the gradient of $U(x) = -\log p(x)$ in (3.26) by the gradient of the smooth Moreau-Yosida envelope $U_\lambda(x)$, given for any $x \in \mathbb{R}^d$ and $\lambda > 0$ by $\nabla U_\lambda(x) = \frac{1}{\lambda}(x - \mathrm{prox}_U^\lambda(x))$. [¶] For example, one could use the Moreau-Yosida ULA Durmus et al. (2018),

[§]That is, there exists $L \geqslant 0$ such that for any $x_1, x_2 \in \mathbb{R}^d$, $\|\nabla \log p(x_1|y) - \nabla \log p(x_2|y)\| \leqslant L\|x_1 - x_2\|$

[¶]Recall: The Moreau-Yosida envelope is defined as $U_\lambda(x) = \inf_{\tilde{x}} U(\tilde{x}) + \frac{1}{2\lambda}\|x - \tilde{x}\|^2$ and the proximal operator is defined as $\mathrm{prox}_U^\lambda(x) = \arg\min_{\tilde{x}\in\mathbb{R}^d} U(\tilde{x}) + \frac{1}{2\lambda}\|x - \tilde{x}\|_2^2$.

given by $X_0 \in \mathbb{R}^d$ and the following recursion for all $k \in \mathbb{N}$

$$X_{k+1} = X_k + \delta \nabla \log p(y|X_k) + \frac{\delta}{\lambda}\left[\text{prox}_U^\lambda(X_k) - X_k\right] + \sqrt{2\delta} Z_{k+1} \,. \tag{3.27}$$

Notice that prox_U^λ is equivalent to MAP denoising under the prior $p(x)$, for additive white Gaussian noise with noise variance λ. The *Plug & Play* ULA methods studied in Chapter 5 are closely related to (3.27), with a state-of-the-art Gaussian denoiser "plugged" in lieu of prox_U^λ. However, instead of approximating ∇U via a Moreau-Yosida envelope as above, we use Tweedie's identity (3.22) relating ∇U to an MMSE denoiser (see Section 5.2).

3.6 A survey of Plug & Play methods for sampling the posterior distribution

A natural strategy to reconcile the strengths of the Bayesian paradigm and neural networks is provided by *Plug & Play* approaches. As explained in Section 3.4, *Plug & Play* approaches seek to derive an approximation of the gradient ∇U (called the Stein score) Bigdeli et al. (2017); Bigdeli and Zwicker (2017) based on the Tweedie's formula using a denoising algorithm D_ε within a Monte Carlo sampling scheme. To the best of our knowledge, the idea of leveraging a denoising algorithm to approximate the score ∇U within an iterative Monte Carlo scheme was first proposed in the seminal paper (Alain and Bengio, 2014) in the context of generative modelling with denoising auto-encoders, where the authors present a Monte Carlo scheme that can be viewed as an approximate *Plug & Play* MALA. This scheme was recently combined with an expectation maximisation approach and applied to Bayesian inference for inverse problems in imaging in (Guo et al., 2019) where the goal was to tackle blind imaging inverse problems. Similarly, the recent work (Kadkhodaie and Simoncelli, 2020) proposes to solve imaging inverse problems by using a *Plug & Play* stochastic gradient strategy that has close connections to an unadjusted version of the MALA scheme of (Alain and Bengio, 2014). However, (Kadkhodaie and Simoncelli, 2020) only deals with noiseless inverse problems. (Kawar et al., 2021b,a) also suggest to plug an MMSE denoiser into an annealed Langevin scheme instead of a score-matching neural network as in (Song and Ermon, 2019) in order to sample from the posterior distribution. However, they only show results obtained with NCSNv2 network proposed by (Song and Ermon, 2019).

While these approaches have shown some remarkable empirical performance, they rely on hybrid algorithms that are not always well understood and that in some cases fail to converge. Indeed, their convergence properties remain an important open question, especially when D_ε is implemented as a neural network that is not a gradient mapping. Consequently, the generated samples do not necessarily represent the posterior of interest $p(x|y)$. In contrast, PnP-ULA and PPnP-ULA that are presented in Chapter 5 allow to sample from the posterior distribution of an inverse problem with a deep Plug & Play prior also providing convergence guarantees and non-asymptotic error bounds.

4

On Maximum-a-Posteriori estimation with Plug & Play priors and stochastic gradient descent

4.1 Introduction

As explained in Section 3.4, several recent works have proposed and studied the use of PnP methods in order to tackle inverse problems. Most of the literature focuses on MAP estimation. If these methods can deliver accurate results, particularly when combined with state-of-the art denoisers, their theoretical analysis stayed for a long time poorly understood. Moreover, they often relied on unrealistic assumptions on the properties of the image denoiser. This chapter is mostly inspired by the preprint article (Laumont et al., 2021)

and aims at developing efficient PnP algorithms with convergence guarantees under reasonable assumptions in order to perform MAP estimation for Bayesian models with PnP priors. First, we address key questions related to the existence, the stability and the well-posedness of the inverse problem in Sections 4.2 and 4.2.1. Then, a convergence proof for MAP computation by PnP stochastic Gradient Descent (PnP-SGD) under realistic assumptions is presented in Section 4.2.2. Finally, the efficiency of the algorithm is demonstrated over a range of classical inverse problems such as denoising, deblurring and interpolation in Section 4.3.

Proofs ans convergence results were derived by Valentin De Bortoli and are exposed in Appendix A for the sake of completeness.

4.2 PnP maximum-a-posteriori estimation: analysis and computation

4.2.1 Analysis of maximum-a-posteriori estimation with PnP priors

We are interested in MAP estimation for Bayesian models involving PnP priors that are defined implicitly by an image denoising algorithm D_ε. We pay special attention to the highly practically relevant case in which D_ε approximates the optimal MMSE denoiser $D_\varepsilon^\star$ associated to p, i.e., $D_\varepsilon^\star = \mathbb{E}[X|X_\varepsilon = x_\varepsilon]$ for $(X_\varepsilon|X = x) \sim \mathcal{N}(x, \varepsilon\,\mathrm{Id})$ when X has marginal density p. As mentioned in Section 3.4.2, state-of-the-art denoisers based on neural networks are often trained to approximate $D_\varepsilon^\star$ by using a sample of clean images $\{x_i\}_{i=1}^N$ from p, corresponding noisy samples $\{x_i'\}_{i=1}^N$ with $X_i'|X_i = x_i \sim \mathcal{N}(x_i, \varepsilon\,\mathrm{Id})$, and choosing D_ε to approximately minimize the empirical MSE loss $\sum_{i=1}^N \|D_\varepsilon(x_i') - x_i\|^2$. Similarly, many state-of-the-art patch-based image denoisers are also designed to approximate $D_\varepsilon^\star$.

The fact that D_ε is only an approximation of $D_\varepsilon^\star$ leads to several complications in the analysis and computation of MAP solutions. For example, unlike $D_\varepsilon^\star$, D_ε does not define a gradient mapping in general, and key results such as Tweedie's identity (Efron, 2011) do not hold. Moreover, in the case of neural network denoisers trained from samples $\{x_i\}_{i=1}^N$ from p, the model is unknown as it is only available through $\{x_i\}_{i=1}^N$, making it difficult to check that basic regularity properties required for MAP estimation are satisfied.

Rather than imposing strong assumptions on D_ε, we address these difficulties by formulating our analysis in the *M-complete* Bayesian framework, in which we assume that the posterior $p(x|y)$ associated with the true prior $p(x)$ exists but remains largely unknown, and all inference on $x|y$ are conducted by using operational approximations of this true model (Bernardo and Smith, 2000). In particular, we focus on the class of smooth approximations of $p(x|y)$ given for any $\varepsilon > 0$ and $x \in \mathbb{R}^d$ by

$$p_\varepsilon(x|y) = \frac{p_\varepsilon(x)p(y|x)}{\int_{\mathbb{R}^d} p_\varepsilon(\tilde{x})p(y|\tilde{x})\mathrm{d}\tilde{x}}\,, \tag{4.1}$$

where $p_\varepsilon(x)$ is the smooth approximation of the prior $p(x)$ defined in (3.21). We will study MAP estimation for $p_\varepsilon(x|y)$ to establish that the procedure is well defined, well posed, amenable to efficient computation, and that it provides a useful approximation to MAP estimation with the true posterior $p(x|y)$. Following on from this, Section 4.2.2 will study the computation of MAP solutions for $p_\varepsilon(x|y)$ by using PnP SGD with a generic denoiser D_ε that approximates $D_\varepsilon^\star$, where we will pay particular attention to the conditions on D_ε required to ensure convergence, as well as to the bias introduced by using D_ε instead of $D_\varepsilon^\star$.

It is established in Chapter 5 that, under basic assumptions on the likelihood function $p(y|x)$ detailed in H1 below, the posterior approximation $p_\varepsilon(x|y)$ is well defined, proper, and can be made as close to the true posterior $p(x|y)$ as desired by reducing the value of ε, with the approximation error vanishing as $\varepsilon \to 0$. Crucially, Chapter 5 also establishes that, under H1 and mild assumptions on the optimal MMSE denoiser $D_\varepsilon^\star$ (essentially, that the denoising problem underlying $D_\varepsilon^\star$ is well posed in the sense of Hadamard), then $x \mapsto p_\varepsilon(x|y)$ is differentiable with $x \mapsto \nabla \log p_\varepsilon(x|y)$ Lipschitz continuous. We conclude that the approximation $p_\varepsilon(x|y)$ is well defined and amenable to computation by first-order schemes, such as SGD to compute critical points of $p_\varepsilon(x|y)$ and perform MAP estimation.

H1 *For any $y \in \mathbb{R}^m$, $\sup_{x\in\mathbb{R}^d} p(y|x) < +\infty$, $p(y|\cdot) \in \mathrm{C}^1(\mathbb{R}^d, (0, +\infty))$. In addition, there exists $\mathsf{L}_y > 0$ such that $\nabla \log p(y|\cdot)$ is L_y Lipschitz continuous.*

With the above-mentioned properties of $p_\varepsilon(x|y)$ in mind, we wonder if computing a MAP solution for $p_\varepsilon(x|y)$ provides useful information about a MAP solution for $p(x|y)$. More precisely, we study if critical points for $p_\varepsilon(x|y)$ are stable w.r.t. variations in ε, and if they converge to critical points of $p(x|y)$ as $\varepsilon \to 0$. Proposition 4.2.1 below establishes that this is indeed the case. In words, MAP solutions computed with $p_\varepsilon(x|y)$ are in the neighbourhood of MAP solutions for $p(x|y)$, with ε controlling a trade-off between the computational efficiency of first-order schemes and the accuracy of the delivered solutions w.r.t. $p(x|y)$. When ε is large, the approximation of the posterior is smoother so gradient descent can be used with larger steps to improve convergence speed (as the gradients have a smaller Lipschitz constant).

Formally, we investigate the stability of the set of stationary points $\mathsf{S}_{\varepsilon,\mathsf{K}} = \{x \in \mathsf{K} : \nabla \log p_\varepsilon(x|y) = 0\}$ w.r.t. $\varepsilon > 0$, where K is a compact set. We show that for any sequence $(\varepsilon_n, x_n)_{n\in\mathbb{N}}$ such that $\lim_{n\to+\infty} \varepsilon_n = 0$ and for any $n \in \mathbb{N}$, $x_n \in \mathsf{S}_{\varepsilon_n,\mathsf{K}}$ every cluster point of $(x_n)_{n\in\mathbb{N}}$ belongs to $\mathsf{S}_\mathsf{K} = \{x \in \mathsf{K} : \nabla \log p(x|y) = 0\}$. In other words, we show that the stationary points of the approximate posterior are close to the ones of the true posterior.

Proposition 4.2.1 *Assume* H1 *and that $p \in \mathrm{C}^1(\mathbb{R}^d, (0, +\infty))$ with $\|p\|_\infty + \|\nabla p\|_\infty < +\infty$. Then for any compact set $\mathsf{K} \subset \mathbb{R}^d$ and $(x_{\varepsilon_n})_{n\in\mathbb{N}}$ such that $\lim_{n\to+\infty} \varepsilon_n = 0$ and for any $n \in \mathbb{N}$, $x_{\varepsilon_n} \in \mathsf{S}_{\varepsilon_n,\mathsf{K}}$, we have that any cluster point $x^\star$ of x_{ε_n} satisfies $x^\star \in \mathsf{S}_\mathsf{K}$.*

Proof: The proof is postponed to Appendix A.1. ■

Note that the above result can be strengthened to show the convergence at the levels of sets. More precisely, we can show that any cluster point of $\{\mathsf{S}_{\mathsf{K},\varepsilon}\}_{\varepsilon>0}$ is a subset of S_K in the sense of the Haussdorff distance, see (Munkres, 2000) for a definition.

As a third and final point in our analysis, we study if MAP estimation for $p_\varepsilon(x|y)$ is a well-posed estimation procedure, which is an essential requirement for meaningful inference. One would ideally seek to establish the existence of a unique global maximiser that is Lipschitz continuous w.r.t. perturbations of the observed data y. Unfortunately, this is not possible without imposing very strong assumptions on the model. Instead, Proposition 4.2.2 below shows that, under some assumptions on the likelihood $p(y|x)$, the set of critical points of $p_\varepsilon(x|y)$ is locally Lipschitz continuous w.r.t. perturbations of y, which is a weaker form of well-posedness. Notice that the assumptions on the likelihood can be relaxed when $D_\varepsilon^\star$ is contractive, but this is usually unrealistic. This highlights a limitation of MAP estimation by comparison to other Bayesian estimators, namely MMSE estimation, which is shown in Chapter 5 to be well-posed under significantly weaker assumptions.

Proposition 4.2.2 *Assume* H1 *and that* $(x, y) \mapsto p(y|x) \in \mathrm{C}^2(\mathbb{R}^d \times \mathbb{R}^m, \mathbb{R})$. *Let* $\varepsilon > 0$, *we have that* $(x, y) \mapsto p_\varepsilon(x|y) \in \mathrm{C}^2(\mathbb{R}^d \times \mathbb{R}^m, \mathbb{R}_+)$. *Let* $y_0 \in \mathbb{R}^m$ *denote some observed data and* $x^\star_{y_0} \in \mathbb{R}^d$ *a local maximiser of the posterior* $x \mapsto p_\varepsilon(x|y_0)$ *with* $\nabla^2 \log p_\varepsilon(x^\star_{y_0}|y_0)$ *negative. Then there exists an open set* $\mathsf{V}_0 \subset \mathbb{R}^m$ *and a function* $x^\star(y) \in \mathrm{C}^1(\mathsf{V}_0, \mathbb{R}^d)$ *such that* $y_0 \in \mathsf{V}_0$ *and for any* $y \in \mathsf{V}_0$, $x^\star(y)$ *is a strict local maximizer of* $x \mapsto p_\varepsilon(x|y)$.

Proof: The proof is postponed to Appendix A.2. ■

To conclude, a major challenge in understanding Bayesian inference with PnP priors and providing guarantees for the delivered solutions is that the underlying prior and posterior densities $p(x)$ and $p(x|y)$ are unknown. Also, the image denoiser D_ε used to construct PnP schemes is not usually directly related to the model. Instead, when it approximates the optimal MMSE denoiser $D^\star_\varepsilon$, it is indirectly related to the model via Tweedie's identity and the smooth approximations $p_\varepsilon(x)$ and $p_\varepsilon(x|y)$. We establish that these operational approximations are useful for MAP inference for $x|y$, in the sense that they are well defined, proper, and MAP solutions for $p_\varepsilon(x|y)$ can be made arbitrarily close to the true MAP solutions through the choice of ε. Importantly, under some assumptions, MAP solutions for $p_\varepsilon(x|y)$ are well posed and amenable to efficient computation by first order optimisation methodology.

4.2.2 PnP-SGD and convergence

We are now ready to study the computation of MAP solutions for $p_\varepsilon(x|y)$ by using PnP SGD with a generic denoiser D_ε that approximates $D^\star_\varepsilon$. We pay particular attention to the conditions on D_ε required to ensure convergence, and to the bias introduced by using D_ε instead of $D^\star_\varepsilon$.

We begin by using Tweedie's identity to express SGD to compute critical points of $p_\varepsilon(x|y)$ as the following sequence: $X_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$

$$X_{k+1} = X_k - \delta_k \nabla F(X_k, y) - \delta_k/\varepsilon (X_k - D^\star_\varepsilon(X_k)) + \delta_k Z_{k+1} \,, \tag{4.2}$$

where $(\delta_k)_{k\in\mathbb{N}} \in (\mathbb{R}_+)^{\mathbb{N}}$ is a sequence of step-sizes, $\varepsilon > 0$, and $\{Z_k \,:\, k \in \mathbb{N}\}$ a family of i.i.d. Gaussian random variables with zero mean and identity covariance matrix. We recall that the sequences $(X_k)_{k\in\mathbb{N}}$ and $(Z_k)_{k\in\mathbb{N}}$ are defined on an underlying probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

As mentioned previously, in most practically relevant cases $D^\star_\varepsilon$ is an abstract quantity that cannot be computed. Instead, we have a different denoiser D_ε that can be assumed to be a good approximation of $D^\star_\varepsilon$. For example, when we have access to samples $\{x_i\}_{i=1}^N$ from p we can consider a noisy version of these samples $\{x'_i\}_{i=1}^N$ with level $\varepsilon > 0$ and train a neural network based denoiser D_ε to minimize the loss $\sum_{i=1}^N \|D_\varepsilon(x'_i) - x_i\|^2$. This loss corresponds to the empirical version of $\mathbb{E}[\|D_\varepsilon(x_\varepsilon) - x\|^2]$ (with $x \sim p$ and $X_\varepsilon \sim \mathcal{N}(x, \varepsilon\,\mathrm{Id})$ condtionally to x) whose minimizer is the MMSE $D^\star_\varepsilon$.

Using a generic denoiser D_ε in our SGD scheme in lieu of $D^\star_\varepsilon$ we obtain the Plug & Play SGD algorithm associated with following recursion: $X_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$

$$X_{k+1} = X_k + \delta_k (b_\varepsilon(X_k) + Z_{k+1}) \,, \tag{4.3}$$

$$b_\varepsilon(x) = \nabla \log(p(y|x)) + \alpha (D_\varepsilon(x) - x)/\varepsilon \,, \tag{4.4}$$

where we note that we have introduced a regularization parameter $\alpha > 0$ that controls the amount of regularisation enforced by D_ε. The original SGD algorithm is recovered by setting $\alpha = 1$ and $D_\varepsilon = D^\star_\varepsilon$.

Algorithm 1 PnP-SGD

Require: $n, n_{\texttt{burnin}} \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon, \alpha, \delta > 0$
  Initialization: Set $X_0 = \tilde{x}$ and $k = 0$.
  for $k = 0 : N$ **do**
    $Z_{k+1} \sim \mathcal{N}(0, \mathrm{Id})$
    if $k \leqslant n_{\texttt{burnin}}$ **then**
      $X_{k+1} = X_k + \delta_0 \nabla \log(p(y|X_k)) + (\delta_0 \alpha/\varepsilon)(D_\varepsilon(X_k) - X_k) + \delta_0 Z_{k+1}$
    end if
    if $k > n_{\texttt{burnin}}$ **then**
      $X_{k+1} = X_k + \delta_k \nabla \log(p(y|X_k)) + (\delta_k \alpha/\varepsilon)(D_\varepsilon(X_k) - X_k) + \delta_k Z_{k+1}$
      $\delta_{k+1} = \delta_0 (k + 1 - n_{\texttt{burnin}})^{-0.8}$
    end if
  end for
  return X_N

We now turn to the proof of convergence of PnP-SGD.

The asymptotic estimates we derive in this chapter are only valid for sequences which remain in a compact set K, which is a classical assumption in stochastic approximation (Tadić et al., 2017; Delyon et al., 1999; Delyon, 1996; Metivier and Priouret, 1984). Under tighter conditions on $x \mapsto \log p_\varepsilon(x|y)$ this limitation can be circumvented using the global asymptotic results of (Tadić et al., 2017, Theorem A1.1). Another way to remove this restriction would be to consider an additive term of the form $x \mapsto (x - \Pi_{\mathsf{C}}(x))/\lambda$ in b_ε (where Π_{C} is the projection onto some compact convex set C and $\lambda > 0$ some hyperparameter) which ensures the stability of the numerical scheme. We leave this analysis for future work. In practice, we have not observed any stability issues for PnP-SGD provided that the stepsize is chosen appropriately see Section 4.3.3.

In what follows, we show that the bias of PnP-SGD depends on the distance between D_ε and the MMSE estimator $D^\star_\varepsilon$, using recent results from (Tadić et al., 2017).

H2 *Assume that there exist $\varepsilon_0 > 0$, $\mathtt{L} \geqslant 0$ and a function $\mathtt{M} : \mathbb{R}^+ \to \mathbb{R}^+$ such that for any $\varepsilon \in (0, \varepsilon_0]$, $R \geqslant 0$, $x_1, x_2 \in \mathbb{R}^d$ and $x \in \overline{\mathrm{B}}(0, R)$ we have*

$$\|D_\varepsilon(x_1) - D_\varepsilon(x_2)\| \leqslant \mathtt{L} \|x_1 - x_2\| \ , \qquad \|D_\varepsilon(x) - D^\star_\varepsilon(x)\| \leqslant \mathtt{M}(R) \ , \tag{4.5}$$

where we recall that

$$D^\star_\varepsilon(x_1) = \textstyle\int_{\mathbb{R}^d} \tilde{x} \ g_\varepsilon(\tilde{x}|x_1) \mathrm{d}\tilde{x} \ , \tag{4.6}$$

with $\tilde{x} \mapsto g_\varepsilon(\tilde{x}|x)$ the probability density of X given $X_\varepsilon = x$ where $X_\varepsilon \sim \mathcal{N}(X, \varepsilon \, \mathrm{Id})$ conditionally to X and $X \sim p$.

The first part of (4.5) regarding the smoothness property of the denoiser can be explicitly verified for a certain class of neural networks by adding a spectral regularization term for each layer of the neural network, see (Ryu et al., 2019; Miyato et al., 2018). The second condition follows from carefully selecting the loss of the neural network as in the Noise2Noise network introduced in (Lehtinen et al., 2018) and controlling the population error. We refer the reader to Chapter 5 for more details regarding the role of the bounding function $\mathtt{M}(R)$. In particular, for neural network denoisers, Proposition 3.1 in Chapter 5 explains how to promote low values of $\mathtt{M}(R)$ during training by using a particular loss function. In addition, Chapter 5 makes connections with universal approximation results (see e.g., (Bach, 2017, Section 4.7)).

We are now ready to state Proposition 4.2.3 which ensures that stable PnP-SGD sequences are close to the set of stationary points of $x \mapsto \log p_\varepsilon(x|y)$ where $x \mapsto \log p_\varepsilon(x|y)$ is given in (4.1). The distance to this set of stationary points is controlled by the approximation error of the D_ε.

H3 *For any $y \in \mathbb{R}^m$, $x \mapsto -\log p(y|x)$ is real-analytic* [*] [†] [‡].

In the following, d stands for the distance induced by the Euclidean norm.

Proposition 4.2.3 *Assume* H1, H2 *and* H3. *Let $\alpha > 0$ and $\varepsilon \in (0, \varepsilon_0]$. Assume that $\lim_{k\to+\infty} \delta_k = 0$, $\sum_{k\in\mathbb{N}} \delta_k = +\infty$ and $\sum_{k\in\mathbb{N}} \delta_k^2 < +\infty$. Let $R > 0$, $\mathsf{K} \subset \overline{\mathrm{B}}(0, R)$ be a compact set, $X_0 \in \mathbb{R}^d$ and $\mathsf{A}_{\varepsilon,\mathsf{K}} \in \mathcal{F}$ given by*

$$\mathsf{A}_{\varepsilon,\mathsf{K}} = \{\omega \in \Omega : \text{ there exists } k_0 \in \mathbb{N} \text{ such that for any } k \geqslant k_0,\ X_k(\omega) \in \mathsf{K}.\}\ , \tag{4.7}$$

where $(X_k)_{k\in\mathbb{N}}$ is given by (4.3). Then there exist $C_{\varepsilon,\mathsf{K}} \geqslant 0$ and $r_{\varepsilon,\mathsf{K}} \in (0,1)$ such that $\limsup_{k\to+\infty} d(X_k(\omega), \mathsf{S}_{\varepsilon,\mathsf{K}}) \leqslant C_{\varepsilon,\mathsf{K}} \mathtt{M}(R)^{r_{\varepsilon,\mathsf{K}}}$ for any $\omega \in \mathsf{A}_{\varepsilon,\mathsf{K}}$, with

$$\mathsf{S}_{\varepsilon,\mathsf{K}} = \{x \in \mathsf{K} : \nabla \log p_\varepsilon(x|y) = 0\}\ , \tag{4.8}$$

where $x \mapsto p_\varepsilon(x|y)$ is given in (4.1).

Proof: The proof is postponed to Appendix A.3. ■

The proof can be extended to the case where $Z_k = 0$ using (Tadić et al., 2017, Theorem 2.1). In this case the assumption that $\sum_{k\in\mathbb{N}} \delta_k^2 < +\infty$ can be replaced by $\lim_{k\to+\infty} \delta_k = 0$.

The following experimental section demonstrates the PnP-SGD algorithm on three canonical imaging inverse problems, namely image denoising, deblurring and interpolation, along with other standard PnP algorithms.

4.3 Experimental study

In this section, we study the behaviour of several PnP algorithms for three classical inverse problems: denoising, deblurring and interpolation. We recall that in each of these problems we consider a prior model $p(x) \propto \exp[-U(x)]$ which is unknown and that the inference $x|y$ is obtained by approximation of this model. For the deblurring and denoising problems, the log-posterior of the degradation model can be written for any $x, y \in \mathbb{R}^d$ as

$$-\log p(x|y) = \|\mathrm{A}x - y\|^2/(2\sigma^2) + \alpha U(x) + C\ , \tag{4.9}$$

[*] A function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be real-analytic if for any $x_0 = (x_0^1, \ldots, x_0^d) \in \mathbb{R}^d$ there exists $(a_{n_1,\ldots,n_d})_{n_1,\ldots,n_d\in\mathbb{N}} \in \mathbb{R}^{\mathbb{N}^d}$ and $r > 0$ such that for any $x = (x^1, \ldots, x^d) \in \mathrm{B}(x_0, r)$

$$f(x) = \textstyle\sum_{n_1\in\mathbb{N}} \cdots \sum_{n_d\in\mathbb{N}} a_{n_1,\ldots,n_d} \prod_{j=1}^d (x^j - x_0^j)^{n_j}\ .$$

[†] The assumption that $x \mapsto \log(p(y|x))$ is real-analytic is satisfied in all of our experiments since there exists $\mathrm{A} \in \mathbb{R}^{p\times d}$ and $\sigma > 0$ such that for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$, $\log p(y|x) = \|\mathrm{A}x - y\|^2/(2\sigma^2)$.

[‡] From Liouville's theorem one could think that the simultaneously verifying that $\nabla \log p(y|\cdot)$ is Lipschitz continuous and that $x \mapsto \log(p(y|x))$ is real-analytic restricts our analysis to models for which $\nabla^2 \log p(y|\cdot)$ is constant (i.e., Gaussian models), but this is not the case because Liouville's theorem applies entire functions, which are a subclass of the real-analytic class.

where A is a $d \times d$ matrix, $C \geqslant 0$ is a constant and the parameter $\alpha \geqslant 0$ balances the weights of the log-likelihood $F(x, y)$ and the log-prior U. In this case, we have for any $x, y \in \mathbb{R}^d$, $F(x, y) = \|\mathrm{A}x - y\|^2/(2\sigma^2)$. In our interpolation experiments, we change the likelihood so that pixels are either visible or hidden. In this case the log-posterior can be written for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$ as

$$-\log p(x|y) = \iota_{\mathrm{Q}x=y} + \alpha U(x) + C \ , \text{ with } \iota_{\mathsf{C}}(x) = \begin{cases} 0 & \text{if } x \in \mathsf{C} \\ +\infty & \text{otherwise,} \end{cases} \tag{4.10}$$

with Q an $m \times d$ matrix consisting of m random lines from the $d \times d$ identity matrix.

4.3.1 Image dataset

In Figures 4.1 and 4.2 we present the 6 original images used in the experiments. These images contain both geometric structures, constant areas and textured regions. On the same figures, we display degraded versions of each image for each set of experiments. For the denoising experiment, the level of the Gaussian noise is fixed to $\sigma^2 = (30/255)^2$. In the case of deblurring, the operator A correponds to a 9×9 uniform blur operator, and we add Gaussian noise with variance $\sigma^2 = (1/255)^2$. Finally, in the context of interpolation, we hide 80% of the pixels.

4.3.2 Algorithms

In this section, we evaluate PnP-SGD (Algorithm 1) along with three other classical PnP algorithms: PnP-ADMM (Algorithm 2), PnP-FBS (Algorithm 3) and PnP-BBS (Algorithm 4). Note that in the case of interpolation, the log-likelihood is not differentiable, since ι_C is not differentiable. In Section 4.3.6 we will present an extension of these PnP algorithms to this setting using proximal operators.

In order to take into account the parameter $\alpha > 0$ into Algorithms 2 to 4, we slightly modify the target function. Instead of minimizing $x \mapsto -\log p(x|y)$ we aim at minimizing $x \mapsto -\log p(x|y)/\alpha$. Doing so, the parameter $\alpha > 0$ can be included in the parameters of the log-likelihood which becomes $(x, y) \mapsto F(x, y)/\alpha$. All algorithms are implemented using Python and the PyTorch library. Our experiments are run on an Intel Xeon CPU E5-2609 server with an Nvidia Titan XP graphic card.

Algorithm 2 PnP-ADMM

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon > 0, \alpha > 0$, $x_0 \in \mathbb{R}^d$
  Initialization: Set $x_0 = z_0$, and $u_k = 0$.
  for $k = 0 : N$ **do**
    $x_{k+1} = \mathrm{prox}_{(\varepsilon/\alpha)F(\cdot,y)}(z_k - u_k)$
    $z_{k+1} = D_\varepsilon(x_{k+1} + u_k)$
    $u_{k+1} = u_k + (x_{k+1} - z_{k+1})$
  end for
  return x_{N+1}

4.3.3 Parameters settings and convergence conditions

In this section, we recall and discuss the choice of the different parameters, as well as the convergence conditions for PnP-SGD. We also discuss the convergence properties of PnP-

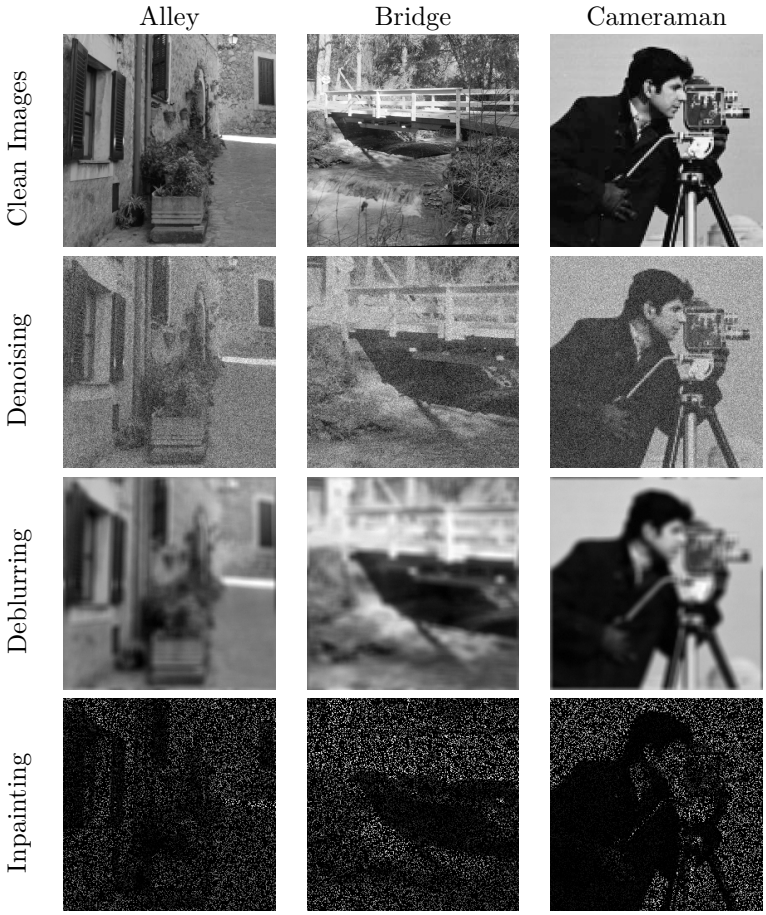


Figure 4.1: *Dataset (part 1):* First three images in our dataset, and examples of degraded images for the three inverse problems considered in this chapter. For denoising, we add a Gaussian noise with variance $\sigma^2 = (30/255)^2$. For deblurring, the operator A correponds to a 9×9 uniform blur operator, and we add Gaussian noise with variance $\sigma^2 = (1/255)^2$. For interpolation, we hide 80% of the pixels.

Algorithm 3 PnP-FBS

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon > 0$, $\alpha > 0$, $x_0 \in \mathbb{R}^d$
  for $k = 0 : N$ **do**
    $x_{k+1} = D_\varepsilon \left(x_k - (\varepsilon/\alpha) \nabla F(x_k, y) \right)$
  end for
  return x_{N+1}

Algorithm 4 PnP-BBS

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon > 0$, $\alpha > 0$, $x_0 \in \mathbb{R}^d$
  for $k = 0 : N$ **do**
    $x_{k+1} = D_\varepsilon(\mathrm{prox}_{(\varepsilon/\alpha)F(\cdot,y)}(x_k))$
  end for
  return x_{N+1}

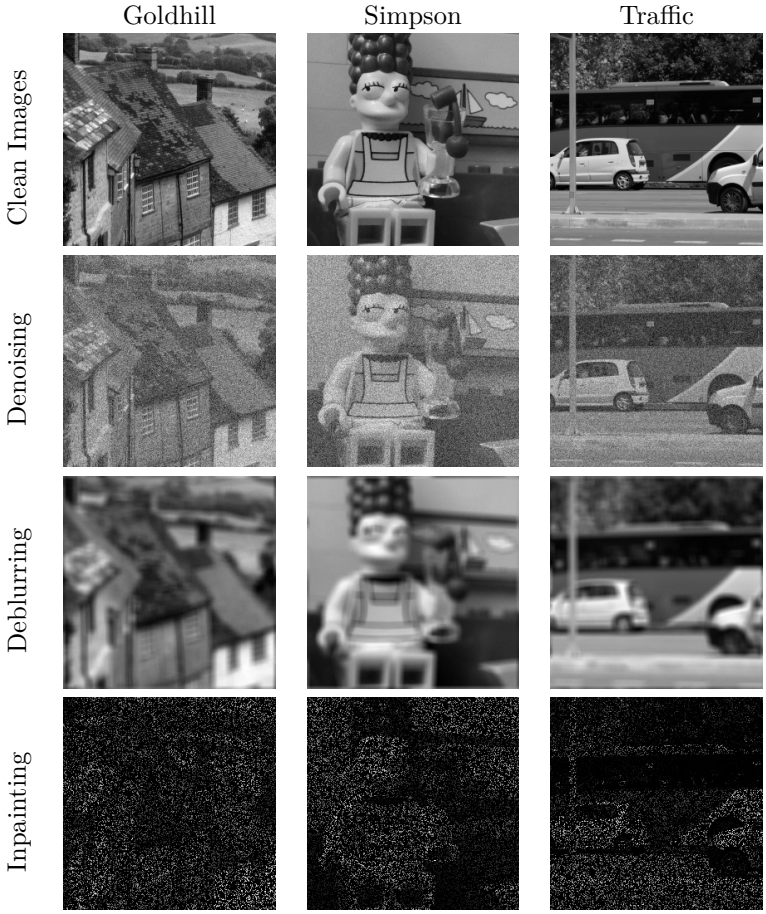


Figure 4.2: *Dataset (part 2):* Last three images in our dataset, and examples of degraded images for the three inverse problems considered in this chapter. For denoising, we add a Gaussian noise with variance $\sigma^2 = (30/255)^2$. For deblurring, the operator A correponds to a 9×9 uniform blur operator, and we add Gaussian noise with variance $\sigma^2 = (1/255)^2$. For interpolation, we hide 80% of the pixels.

ADMM and PnP-FBS following the guidelines of (Ryu et al., 2019; Xu et al., 2020).

Recall that from (4.1), we denote $\mathtt{L}_y$ the Lipschitz constant of the log-likelihood gradient $x \mapsto \nabla F(., y)$. For $F(x, y) = \|\mathrm{A}x - y\|^2/(2\sigma^2)$, $\mathtt{L}_y = \|\mathrm{A}^\star\mathrm{A}\|/\sigma^2$, with $\mathrm{A}^\star$ the adjoint of A. F is μ-strongly convex if and only if A is invertible, in which case $\mu = \lambda_{\min}(\mathrm{A})^2/\sigma^2$, where $\lambda_{\min}(\mathrm{A})$ is the smallest singular value of A. In our experiments we have $\lambda_{\min} = 1$ for denoising and $\lambda_{\min} = 0$ for deblurring and interpolation. In our experiments, the operator A is always chosen such that $\|\mathrm{A}^\star\mathrm{A}\| = 1$. Note that if F is replaced by F/α, as it the case in Algorithms 2 to 4, we have that $\mathtt{L}_y$ and μ are replaced by $\mathtt{L}_y/\alpha$ and μ/α.

DENOISER. In all experiments, the denoising operator D_ε is chosen as the pretrained denoising neural network introduced in (Ryu et al., 2019). This denoiser is trained so that $\mathrm{Id} - D_\varepsilon$ is L-Lipschitz with $\mathtt{L} < 1$. Note that this corresponds to the first part of (4.5) in H2. In (Ryu et al., 2019) three pretrained denoisers, at noise level $\varepsilon = (5/255)^2, (15/255)^2, (40/255)^2$ are proposed. In this work, we only use the first one in our denoising and deblurring experiments. The interpolation problem requires a more subtle strategy relying on a coarse to fine approach, described in Section 4.3.6.

PNP-SGD. In Algorithm 1, we consider a burn-in regime with a constant step δ_0 until some iteration n_{burnin}. After this initial phase, we set $(\delta_k)_{k\in\mathbb{N}}$ to be a decreasing sequence satisfying the conditions of Proposition 4.2.3. In the case of denoising or deblurring, δ_0 is given by

$$\delta_0 = \delta_{\mathtt{stable}}/6, \text{ where } \delta_{\mathtt{stable}} := 2/\mathtt{L}_{\text{tot}} , \quad \mathtt{L}_{\text{tot}} = \alpha\mathtt{L}/\varepsilon + \|\mathrm{A}^*\mathrm{A}\|/\sigma^2 , \tag{4.11}$$

where $\mathtt{L}_{\text{tot}}$ is the Lipschitz constant of $\nabla \log p(.|y)$. Note that setting $\delta_0 = \delta_{\mathtt{stable}}$ ensures that the deterministic scheme: $x_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$, $x_{k+1} = x_k + \delta_0 \nabla \log p(x_k|y)$, satisfies that $(\log p(x_k|y))_{k\in\mathbb{N}}$ is non-decreasing. After the burn-in period, we use a decreasing sequence of step-sizes $(\delta_k)_{k\in\mathbb{N}}$ such that for any $k \in \mathbb{N}$ we have

$$\delta_k := \delta_0 \times (k - n_{\mathtt{burnin}})^{-0.8} , \tag{4.12}$$

which satisfies the conditions required in Proposition 4.2.3 for convergence. Note that contrary to existing work, any value of $\alpha > 0$ can be used in Algorithm 1 provided that δ_0 is defined accordingly using (4.11).

PNP-ADMM. The convergence results of (Ryu et al., 2019) for PnP-ADMM require the strong convexity of F. In our experiments, this condition is met for denoising experiments (since $\mathrm{A} = \mathrm{Id}$), but not for interpolation nor deblurring if the blur operator is not invertible (which is the case for a 9×9 uniform blur). In the denoising case, following (Ryu et al., 2019), PnP-ADMM converges to a fixed point if $\mathtt{L} \in [0, 1)$ and $\mathtt{L}/(1 + \mathtt{L}(1 - 2\mathtt{L})) < \varepsilon/(\alpha\sigma^2)$. In practice, L and ε being set, this condition can only be satisfied for small values of the regularisation parameter α, which often lead to poor-quality results. However, Algorithm 2 experimentally converges to a fixed point with interesting visual properties for larger values of α. This suggests that it might be possible to prove the convergence of PnP-ADMM under weaker conditions than the ones of (Ryu et al., 2019).

PNP-FBS. Similarly to PnP-ADMM the convergence results obtained by (Ryu et al., 2019) for PnP-FBS are only valid in a strongly convex setting. In our case this corresponds to the denoising experiment here. The condition on the Lipschitz constant of the denoiser D_ε

is $\mathtt{L}/(1+\mathtt{L}) < \varepsilon/(\alpha\sigma^2) < (\mathtt{L}+2)/(\mathtt{L}+1)$. In Section 4.3.4, we show that these conditions are not met in our experiments. In practice, we still observe convergence of the algorithm for the denoising experiments. This is no longer case in non-strongly convex problems, see Section 4.3.5 and Section 4.3.6. In (Xu et al., 2020), convergence towards the set of stationary points of the log-posterior is established for PnP-FBS provided that $D_\varepsilon = D^\star_\varepsilon$, *i.e.* D_ε is the optimal MMSE. In addition, (Xu et al., 2020) require that $\varepsilon \mathtt{L}_y \leqslant 1$. This condition implies that $\varepsilon\|\mathrm{A}^\star\mathrm{A}\| \leqslant \alpha\sigma^2$. Since $\|\mathrm{A}^\star\mathrm{A}\| = 1$ for all our experiments, this implies $\alpha \geqslant \varepsilon/\sigma^2$. In experiments with large noise level (as it is the case for our denoising setting), this leads to acceptable values of α. However, when σ is small in comparison to ε (which is the case for deblurring), the regularisation parameter α for which the convergence is ensured is too highlighted in Section 4.3.3.

4.3.4 Denoising

For these denoising experiments, we add a Gaussian noise of variance $\sigma^2 = (30/255)^2$ (see the second row of Figures 4.1 and 4.2 for examples of degraded images). In this experiment we use a denoiser D_ε trained for a noise level $\varepsilon = (5/255)^2$ on a dataset $\{x_i, x_i'\}_{i=1}^N$ with $x_i \sim p$ and $x_i' \sim \mathcal{N}(x_i, \varepsilon\,\mathrm{Id})$ for any $i \in \{1, \dots, N\}$. Using this denoiser in Algorithms 1 to 4, we aim at denoising y with noise level σ^2.

We run all algorithms for several values of the regularization parameter α and for two different initializations: first a TV-L_2 initialization, *i.e.* applying a simple TV-L_2 restoration to the noisy image following (Rudin et al., 1992; Chambolle and Pock, 2011), and second an oracle initialization (using the original image without degradation). Although the noisy observation y is a natural initialization, we observed that initializing at y usually leads to unsatisfactory results for all PnP schemes with a small value of ε. We believe that this arises from the high non-convexity of the problem. Our goal here is to assess the dependency of the algorithm on initialization, since the log-posterior we study is highly non-convex.

For PnP-SGD, the initial step-size δ_0 and the sequence $(\delta_k)_{k\in\mathbb{N}}$ are defined as explained in Section 4.3.3. For these denoising experiments, the resulting value of δ_0 is already quite small, such that decreasing δ_k after the burn-in phase effectively stops the search for a better optimum and does not change the result. The number of iterations $n_{\texttt{burnin}}$ for the burn-in was set between 5000 and 25000 for SGD. Within that range, we stop this phase as soon as $|\mathrm{PSNR}(X_{k+1}) - \mathrm{PSNR}(X_k)| < 0.1 \times \delta_0$. This conservative choice allows to make sure that the algorithm reaches its steady state, so that the oracle initialization (starting from an overestimated value of PSNR) does not overestimate the global maximum and the non-oracle initializations (starting from an underestimated value of PSNR) do not under-estimate it. In practice, convergence is reached after a few hundreds of iterations in most cases and only rarely did the algorithm iterate beyond 5000. Increasing δ_0 to $\delta_0 = 0.9 \times \delta_{\texttt{stable}}$ also permits to achieve faster convergence, but in this case adding a decreasing phase for $(\delta_k)_{k\in\mathbb{N}}$ after the burn-in regime is important to achieve the same asymptotic results.

For the splitting-based algorithms (ADMM, BBS, FBS), practical convergence is very fast and 100 iterations are largely sufficient in all cases. Observe that since we use a denoiser trained for a noise level $\varepsilon = (5/255)^2$, and our denoising experiments are run for $\sigma^2 = (30/255)^2$, theoretical convergence of PnP-ADMM following (Ryu et al., 2019) requires that $\alpha < (1+\mathtt{L}(1-2\mathtt{L}))/36\mathtt{L}$. The exact value of $\mathtt{L}$ for the denoising considered in (Ryu et al., 2019) is not available, but our experiments suggest that $\mathtt{L} \approx 1$. This implies that only drastically small values of α meet the previous condition. As a result, this condition is not satisfied with the choices of α that are experimentally optimal but does not prevent the algorithm to converge in practice. In the same way, provided that $\mathtt{L} \in [0, 1)$, convergence of PnP-FBS

following (Ryu et al., 2019) implies that α is at least larger than 18, see Section 4.3.3. Yet, interesting values of α for this denoising experiment are far smaller. The condition provided in (Xu et al., 2020), $\alpha \geqslant \varepsilon/\sigma^2 = 1/3$ gives more realistic values for α but we remind that in this case we must assume that $D_\varepsilon = D_\varepsilon^\star$.

Figure 4.3 summarizes the results of this denoising experiment on 10 independent random noise realizations on each of the 6 images in the dataset, for PnP-SGD, PnP-ADMM and PnP-BBS (PnP-FBS is not shown here for the sake of clarity, but it shows a very similar behavior). We first observe that initialization seems to play a very minor role for all the algorithms considered in this problem. A TV-L_2 initialization is sufficient to reach virtually the same reconstruction quality as the oracle initialization. This might be explained by the fact that denoising is a relatively simple inverse problem. Second, all algorithms produce very similar results, with an optimal value of α around 0.25, see Figure 4.3. Table 4.1 summarizes the denoising results of all algorithms (including PnP-FBS) obtained for this nearly optimal setting of $\alpha = 0.25$. In Figure 4.4 we display the results of the different algorithms for this denoising experiment. If the PSNR values are quite close, it seems that the algorithms make different compromises in terms of visual results. For example, the estimator obtained with PnP-ADMM seems to exhibit sharper edges. However, it also seems to hallucinate more false structures than other algorithms.

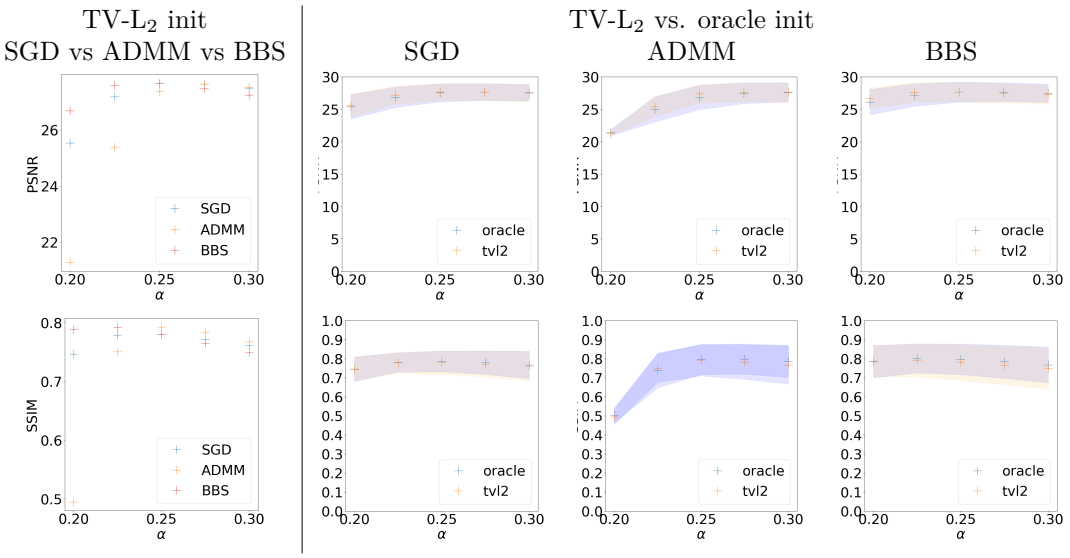


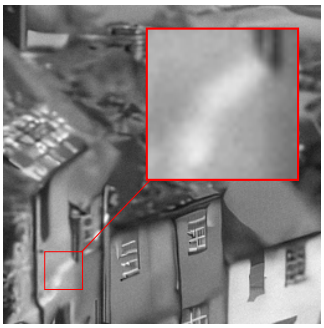
Figure 4.3: Plug & Play denoising for $\sigma^2 = (30/255)^2$ with the prior implicit in D_ε for $\varepsilon = (5/255)^2$ and different values of the regularization parameter α. This table shows means and standard deviations for PSNR and SSIM values over K=10 independent noise realizations for each of the six images and different values of the regularization parameter α. Initialization plays a very minor role in this case and all algorithms achieve similar (nearly optimal) performance for $\alpha = 0.25$.

Figure 4.5 delivers a first convergence diagnosis of PnP-SGD for the denoising task. The evolution of the average PSNR computed for each image over the 10 different experiments suggests that only a thousand of iterations seem to be needed to reach the stationary regime. This impression is confirmed by the evolution of the average gradient norm of the log-posterior, as after 1000 iterations, a plateau around 0.4 is reached. The decay after 5000 iterations is due to the forced decay of the δ_k as the algorithm has left the burn-in phase during which the discretization step-size was held constant.

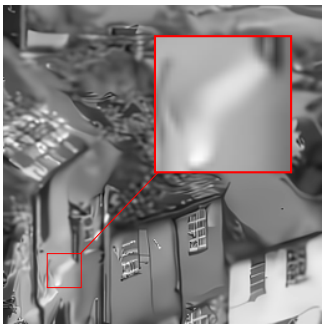
Denoising $\sigma^2 = (30/255)^2$, $\varepsilon = (5/255)^2$, TV-L_2 init, $\alpha = 0.25$				
	PnP-SGD	PnP-ADMM	PnP-BBS	PnP-FBS
Overall PSNR	27.65	27.37	27.65	27.56
Simpsons	30.04	30.10	30.41	30.35
Traffic	27.36	27.09	27.31	27.27
Cameraman	28.54	28.21	28.74	28.48
Alley	27.16	26.82	26.98	26.96
Bridge	26.28	25.83	26.18	26.03
Goldhill	26.55	26.18	26.30	26.30

Table 4.1: Plug & Play denoising for $\sigma^2 = (30/255)^2$ with the prior in D_ε for $\varepsilon = (5/255)^2$. This table shows mean PSNR values over K=10 independent noise realizations for each of the six images. The regularization parameter $\alpha = 0.25$ is nearly optimal for all algorithms.

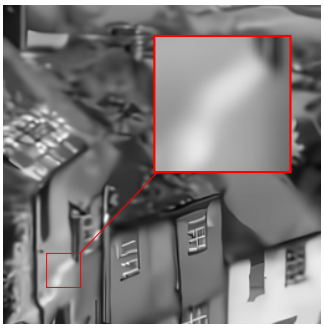
SGD (PSNR=26.58 dB, SSIM=0.69)



ADMM (PSNR=26.17 dB, SSIM=0.68)



BBS (PSNR=26.33 dB, SSIM=0.64)



FBS (PSNR=26.31 dB, SSIM=0.67)

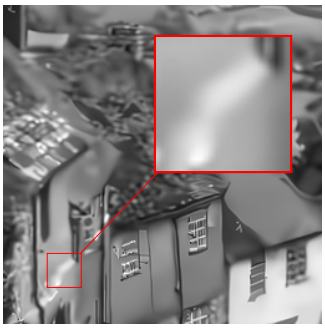
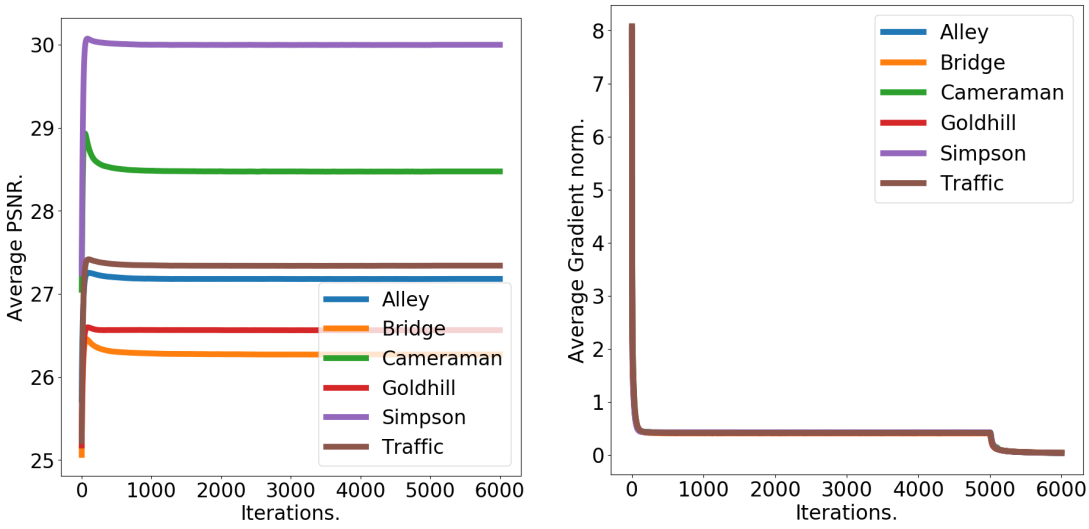


Figure 4.4: Plug & Play denoising for $\sigma^2 = (30/255)^2$, $\varepsilon = (5/255)^2$ and with $\alpha = 0.25$. Although the results obtained by the different methods are close from a quantitative point of view, they look for different compromises. For example, PnP-ADMM looks for sharper edges than PnP-SGD but tends to hallucinate structures.



Evolution of the PSNR.

Evolution of the gradient norm.

Figure 4.5: Convergence diagnosis for Plug & Play denoising for $\sigma^2 = (30/255)^2$, $\varepsilon = (5/255)^2$ with $\alpha = 0.25$ and $\mathrm{TV} - \mathrm{L}_2$ initialization. Left: Evolution of the average PSNR computed for $K = 10$ independent noise realizations for each image. A thousand of iterations seem to be sufficient to leave the burn-in phase and enter the stationary phase. The decay of the discretization step-size δ_k does not alter the results, which suggests that the algorithm has converged. Right: Evolution of the average gradient norm of the log-posterior computed over the 10 experiments for each image. In less than 500 iterations, it stabilizes around 0.4 for each image. These plots suggest that the algorithm has converged. The decrease observed after 5000 iterations is explained by the decay of the discretization step-size δ_k and does not alter the final result.

4.3.5 Deblurring

We now turn to the deblurring problem. In this section, images are blurred with a uniform 9×9 kernel, and a small Gaussian noise of standard deviation $\sigma = 1/255$ is added in order to define the degradation model. We now compare the behavior of Algorithms 1 to 3.

Experiments with PnP-SGD follow the same rules as for the denoising problem and the same observations are valid. When running PnP-ADMM we use approximately 200 iterations to ensure the convergence whereas for PnP-FBS and PnP-BBS, we use approximately 500 iterations. Except for PnP-SGD (using Proposition 4.2.3), these PnP algorithms are not guaranteed to converge according to (Ryu et al., 2019) since A is not invertible. In practice PnP-FBS indeed converges only for very large values of the regularization parameter α, whereas other PnP algorithms converge for all our experiments. As highlighted in Section 4.3.3 this suggests that convergence for PnP-ADMM and PnP-FBS occur under weaker conditions than the ones prescribed in (Ryu et al., 2019).

Figure 4.6 summarizes the results of deblurring on 10 independent random noise realizations on each of the 6 images in the dataset, for PnP-SGD, PnP-ADMM and PnP-BBS (PnP-FBS is not shown here because it does not converge most of the time), for TV-L_2 and oracle initializations. Again, initialization appears to play a minor role in the final results.

Observe that all algorithms show very similar performance (when they converge) for these deblurring experiments. While PnP-SGD is slower to converge, it is ensured to approximate the MAP theoretically. Table 4.2 summarizes the deblurring results of all algorithms (including PnP-FBS) obtained for the nearly optimal setting of $\alpha = 0.3$. In Figure 4.7 we display the results of the different algorithms for this deblurring experiment. Interestingly, we note that visual results for this deblurring problem are much more similar to each other than for denoising experiments.

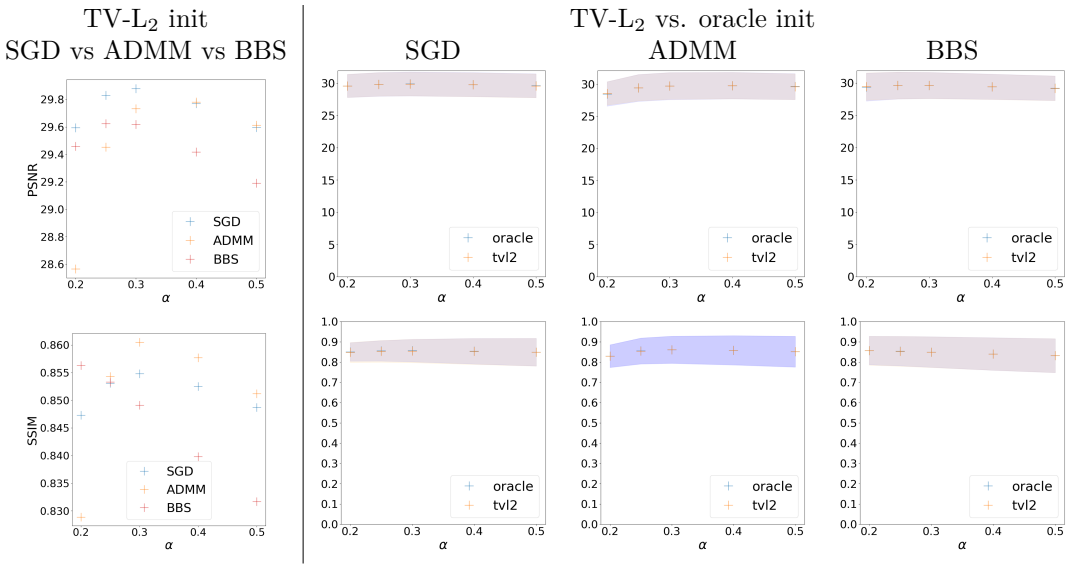


Figure 4.6: Plug & Play deblurring. Image are blurred with a 9×9 uniform kernel, a Gaussian noise of standard deviation $\sigma^2 = (1/255)^2$ is added. The denoiser D_ε is trained at $\varepsilon = (5/255)^2$. The plots shows mean and standard deviation values of PSNR and SSIM over K=10 independent noise realizations for each of the six images and different values of the regularization parameter α. Initialization plays a very minor role in this case and all algorithms achieve similar (nearly optimal) performance for $\alpha = 0.3$, except for FBS which requires a larger (sub-optimal) α to converge.

Deblurring a 9×9 kernel with $\sigma^2 = (1/255)^2$, $\varepsilon = (5/255)^2$, TV-L_2 init, $\alpha = 0.3$				
	PnP-SGD	PnP-ADMM	PnP-BBS	PnP-FBS
Overall PSNR	29.88	29.73	29.62	NaN
Simpsons	33.51	33.93	33.70	NaN
Traffic	29.41	29.27	29.10	NaN
Cameraman	30.68	30.43	30.39	NaN
Alley	29.26	28.99	28.90	NaN
Bridge	28.08	27.77	27.65	NaN
Goldhill	28.33	28.01	27.97	NaN

Table 4.2: Plug & Play deblurring. Image are blurred with a 9×9 uniform kernel, a Gaussian noise of standard deviation $\sigma = 1/255$ is added. The denoiser D_ε is trained at $\varepsilon^2 = (5/255)^2$. This table shows mean PSNR values over K=10 independent noise realizations for each of the six images. The regularization parameter $\alpha = 0.30$ is nearly optimal for all algorithms.

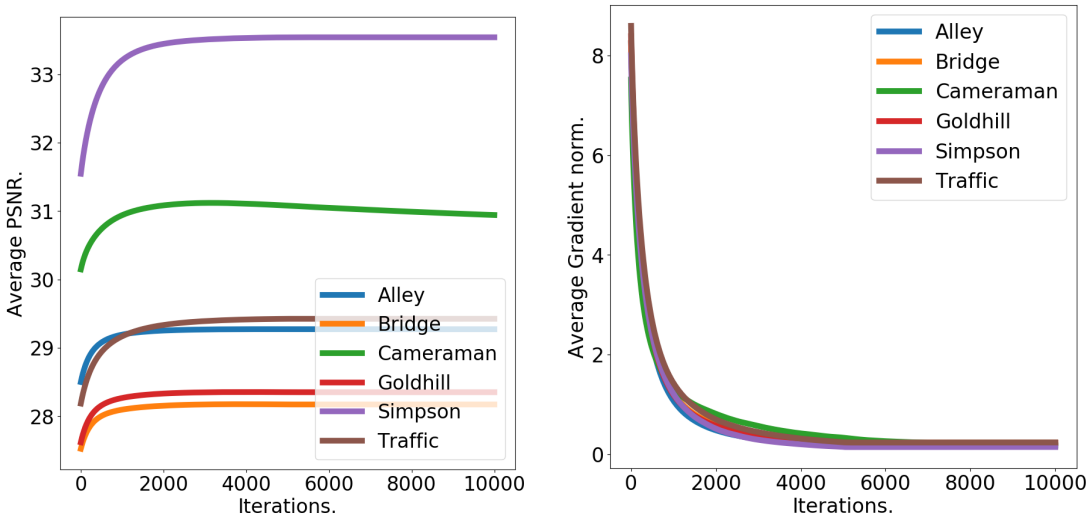
Figure 4.7: Plug & Play deblurring, for a 9×9 kernel, an additive Gaussian noise of standard deviation $\sigma^2 = (1/255)^2$, for $\varepsilon = (5/255)^2$ and for the nearly optimal value of $\alpha = 0.3$.

As for the denoising problem, we study the convergence of PnP-SGD studying the evolution of the average PSNR and gradient norm of the log-posterior over the iterations. Figure 4.8 shows the evolution of the average PSNR computed for each image over the 10 different experiments. It suggests that 4000 of iterations seem to be needed to have a stable PSNR for all images except `Cameraman`, which requires on average $1.5e4$ iterations. This difference between the other images could be explained by the fact that $\alpha = 0.3$ is a more sub-optimal regularization parameter for this image. This impression is confirmed by the evolution of the average gradient norm of the log-posterior, as after 6000 iterations, a plateau around 0.2 is reached. These plots also suggest that the convergence is slower for deblurring than for denoising.

4.3.6 Interpolation

The interpolation problem consists in trying to recover $x \in \mathbb{R}^d$ from a small proportion of its pixels, namely from the measurements vector $y = \mathrm{Q}x$, where Q is a $m \times d$ matrix consisting of m random lines from the $d \times d$ identity matrix, and $m = qd \ll d$. In our experiments we set $q = 20\%$. In this case, since measurements are not affected by noise, the data-fitting term takes the form of a hard constraint, *i.e.* for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$ we have

$$F(x, y) = \iota_{\mathsf{C}_y}(x), \quad \text{where } \mathsf{C}_y = \{x : y = \mathrm{Q}x\} .$$



Evolution of the PSNR. Evolution of the gradient norm.

Figure 4.8: Convergence diagnosis for Plug & Play deblurring with $\sigma^2 = (1/255)^2$, $\varepsilon = (5/255)^2$, $\alpha = 0.3$ and $\mathrm{TV} - \mathrm{L}_2$ initialization. Left: Evolution of the average PSNR computed for $\mathsf{K} = 10$ independent noise realizations for each of the 6 images. As expected the convergence is slower for the deblurring problem. 4000 iterations seem to be required to enter the stationary phase for all images except `Cameraman`, that needs on average $1.5e4$ iterations. Right: Evolution of the average gradient norm of the log-posterior computed over the 10 experiments for each image. For all images, the gradient norm stabilizes around 0.2.

The non-differentiability of F is not problem when using ADMM and BBS since in this case the proximal operator of $\gamma F(\cdot, y)$ is not only defined but admits a closed-form (which is independent of $\gamma = \varepsilon/\alpha$). More precisely, we have for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$, $\mathrm{prox}_{\gamma\iota_C}(x) = \mathrm{P}^\star \mathrm{P} x + \mathrm{Q}^* y$ in terms of the $(d-m) \times d$ matrix P containing all the lines of the identity matrix which are not contained in Q. However, SGD and FBS cannot be directly applied to this problem because they require F to be differentiable. Nevertheless we can apply these algorithms to an equivalent formulation in the reduced space $\mathbb{R}^{d-m}$ of unknown pixels, as shown in the following subsection.

4.3.6.a Adapting SGD to the non-differentiable inpainting problem

In what follows, we denote by $\tilde{x} := \mathrm{P}x \in \mathbb{R}^n$ the vector of $n = d - m$ unknown pixels in x. Given the unknown pixels $\tilde{x} = \mathrm{P}x$ and the measurements $y = \mathrm{Q}x$ we can reconstruct x via the affine mapping $f_y : \mathbb{R}^n \to \mathbb{R}^d$ defined for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$ by $f_y(\tilde{x}) = \mathrm{P}^* \tilde{x} + \mathrm{Q}^* y$.

The solution of the original problem $x_{\mathrm{MAP}} = \arg\min_x F(x, y) + U(x)$ can then be written as

$$x_{\mathrm{MAP}} = \arg\min_{x \in \mathsf{C}_y} U(x) = f_y(\arg\min_{\tilde{x}} U(f_y(\tilde{x}))) \;, \quad \tilde{x}_{\mathrm{MAP}} = \arg\min_{\tilde{x}} U(f_y(\tilde{x})) \;, \tag{4.13}$$

and $\tilde{x}_{\mathrm{MAP}}$ can be found by gradient descent on $\tilde{U} = U \circ f_y$. Using the chain rule and Tweedie's formula, we have that the gradient of $\tilde{U}$ is given for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$ by

$$\nabla \tilde{U}(\tilde{x}) = \mathrm{P} \nabla U(f_y(\tilde{x})) = (1/\varepsilon)\mathrm{P}(\mathrm{Id} - D_\varepsilon) \circ f_y(\tilde{x}) \;. \tag{4.14}$$

Finally, since the affine operators P and f_y are 1-Lipschitz we have that $\tilde{\mathrm{L}} \leqslant (1/\varepsilon)$, where $\tilde{\mathrm{L}}$ is the Lipschitz constant of $\nabla \tilde{U}$.

4.3.6.b Parameter settings and results

The interpolation problem we consider is extremely ill-posed since 80% of the pixels are only constrained by the image prior. Since our implicit prior $p_\varepsilon(x)$ is most likely far from log-concave, the posterior shows a particularly large number of local optima. For this reason all methods are extremely sensitive to the initial condition. The initial conditions used in the previous experiments may misguide both ADMM and SGD to a wrong local optimum.

To deal with this more difficult case, we consider a different approach, combining:

- A coarse to fine scheme where we start by solving the MAP problem for large values of ε, and then use the result of this coarse MAP as an initialization for the next smaller value of ε. In our experiments we used $\varepsilon = (40/255)^2, (15/255)^2, (5/255)^2$, both for ADMM and for SGD;
- For each value of ε, a burn-in phase of 2000 iterations with $\delta_0 = 2.5\delta_{\texttt{stable}}$, followed by a phase of 1000 decreasing steps, as defined in (4.12).

Table 4.3 summarizes the results of different algorithmic strategies to solve our inpainting problem, on our set of 6 images with $K = 4$ random realizations for each image, and Figure 4.9 shows an example of results on the *Simpsons* image.

We can observe in Table 4.3 that the coarse-to-fine scheme is beneficial to both SGD and ADMM, allowing to reach a reconstruction quality which comes very close to the oracle initialization. This benefit is also clear on the visual results shown on Figure 4.9. In the case of a random initialization, the coarse to fine strategy is needed to avoid the apparition of spurious geometric structure in the background. In the case of the $\mathrm{TV} - \mathrm{L}_2$ initialization, it yields better continuity in the fine black lines of the image. This holds both for ADMM and SGD.

In these interpolation experiments, we also observed that using larger initial step-sizes at the beginning and using the stochastic gradient descent instead of a simple gradient descent are important to obtain good MAP estimates. This could be explained by the non-convex nature of this problem: the stochastic term and the larger step sizes are required to avoid getting trapped in spurious local optima.

4.4 Conclusion

In this chapter we studied MAP estimation in Bayesian imaging models with PnP priors defined by image denoising algorithms. First, we sought to better understand, from a theoretical point of view, MAP estimation for solving imaging inverse problems. That is why we first addressed key questions about the definition, the stability and the well-posedness of this problem. We established that, under mild conditions, MAP solutions are well-posed. For computation, we proposed a PnP-SGD optimisation method that is provably convergent under mild and realistic assumptions on the denoiser. It paves the way for further studying more elaborate PnP schemes to estimate the MAP. The proposed approach was then illustrated on a range of imaging inverse problems by using a deep network denoiser that satisfied our conditions for convergence.

In future work, we would like to continue our theoretical and empirical investigation of Bayesian PnP models, methods and algorithms. First, we would like to develop provably convergent accelerated algorithms based on more elaborated optimization schemes. Furthermore, developing methods to automatically adjust the regularisation parameter α directly from the observed data y in a manner akin to (Vidal et al., 2020), is also a crucial question.

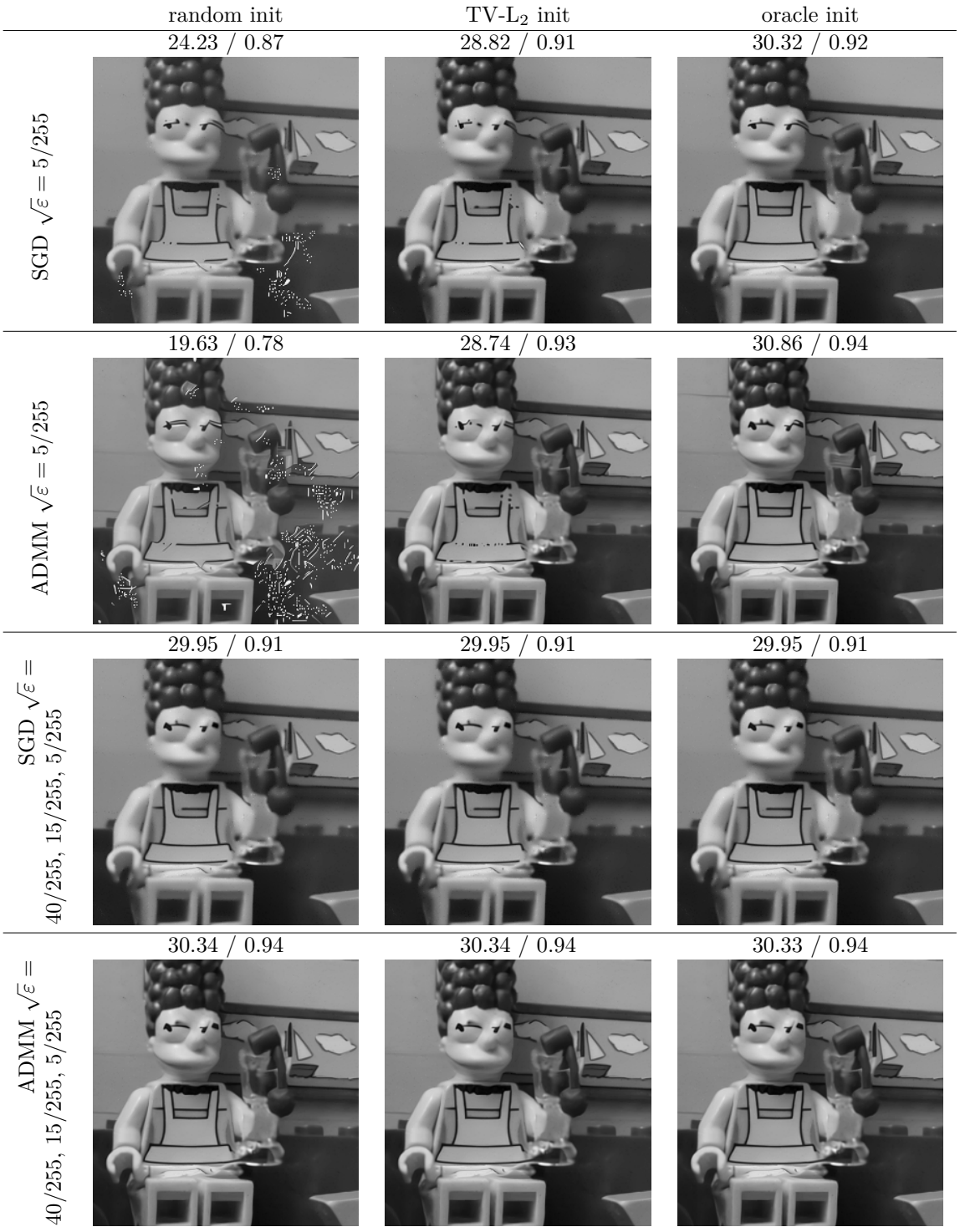


Figure 4.9: Interpolation results for the Simpson's image with $p = 0.8$, $\sigma = 0$ each column corresponds to a different initial condition.

Method	PSNR		SSIM	
	mean	std dev	mean	std dev
Random initialization				
SGD $\varepsilon = (5/255)^2$	23.43	2.75	0.7715	0.0517
SGD $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	**26.32**	1.76	0.8074	0.0702
ADMM $\varepsilon = (5/255)^2$	19.34	3.09	0.6787	0.0629
ADMM $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	25.94	2.19	**0.8292**	0.0745
TV-L_2 initialization				
SGD $\varepsilon = (5/255)^2$	26.01	1.53	0.8042	0.0684
SGD $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	**26.34**	1.80	0.8074	0.0699
ADMM $\varepsilon = (5/255)^2$	25.38	1.74	0.8216	0.0754
ADMM $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	25.87	2.13	**0.8266**	0.0764
Oracle initialization				
SGD $\varepsilon = (5/255)^2$	**26.67**	1.66	0.8116	0.0700
SGD $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	26.36	1.76	0.8079	0.0702
ADMM $\varepsilon = (5/255)^2$	26.16	2.18	**0.8330**	0.0742
ADMM $\varepsilon = (40/255)^2$, $(15/255)^2$, $(5/255)^2$	25.93	2.14	0.8269	0.0768

Table 4.3: Interpolation with $p = 0.8$, $\sigma = 0$ with random, TV-L_2 and oracle initialization. Mean and standard deviation of PSNR and SSIM measures computed on K=4 random tests for each of the 6 images. Note the effectiveness of the coarse-to-fine scheme with either random or TV-L_2 initialization: Coarse to fine SGD is only 0.33 dB away from the solution obtained with oracle init, which should be quite close to the global optimum. ADMM is only 0.22 dB away from the solution obtained with oracle init.

So far, we set this parameter by experimenting, which is time consuming, computationally demanding and not theoretically grounded. It can also lead to model miss-specification that hurts the proposed MAP restoration. It certainly explains the results obtained for `Cameraman` on the deblurring task with $\alpha = 0.25$. It would also be interesting to extend the decision-theoretic foundation of MAP estimation in log-concave models of (Pereyra, 2019) to encompass MAP estimation in (not log-concave) PnP Bayesian models. Indeed, computing the MAP to solve an inverse problem (in the general case where the prior is not necessarily log-concave) is still not understood from a Bayesian point of view, as we do not know if it minimizes any cost function under the posterior distribution (Bassett and Deride, 2019; Pereyra, 2019).

In the following chapter, we go further than simple point estimation. Indeed, as explained in Chapter 2, although it is practical to summarize the posterior distribution by one point, we have no information about the uncertainty on the proposed solution. It can be problematic, especially in contexts where a decision is based on the proposed restoration like in medicine. In the next chapter, we propose two Plug & Play sampling algorithms with detailed convergence guarantees and non-asymptotic error bounds under realistic assumptions to tackle this issue.

5

Bayesian imaging using Plug & Play priors: when Langevin meets Tweedie

5.1 Introduction

As explained in Chapter 2, although computing the MAP is appealing, it is only well-understood in log-concave cases. We may wish to compute other estimators such as the MMSE, which is theoretically well-founded within the Bayesian framework. Furthermore, we may want to better exploit the resources provided by the Bayesian framework and not

to simply summarize the posterior distribution to 1 point. For example, we could wish to quantify uncertainty on the proposed restoration or perform model calibration.

In this chapter we present our work on Monte Carlo sampling with PnP priors for general Bayesian computation. First, we answer primordial questions concerning the well-posedness, and the stability of the related Bayesian models and estimators in Section 5.2. Then, we present the Plug-and-Play Unadjusted Langevin Algorithm (PnP-ULA) and Projected Plug-and-Play Unadjusted Langevin Algorithm (PPnP-ULA), two Monte-Carlo sampling algorithms with detailed convergence guarantees under realistic assumptions on the denoising operators used. A special attention is given to deep neural network denoisers (see Section 5.3). Section 5.4 illustrates the methods on classical imaging inverse problems such as denoising, deblurring and interpolation where it is used to perform MMSE point estimation and as well as for uncertainty visualization and quantification. Finally, we propose to speed-up the state-space exploration of the proposed sampling algorithms using a state-of-the-art discretization scheme and show the results in Section 5.5.

Convergence results were derived by Valentin De Bortoli and are exposed in Appendix B for the sake of completeness.

This chapter is mostly based on the article (Laumont et al., 2022) published in SIAM Journal on Imaging Sciences. Section 5.5 is entirely new.

5.2 Bayesian inference with Plug & Play priors: theory methods and algorithms

5.2.1 Bayesian modelling and analysis with *Plug & Play* priors

This section presents the same formal framework for Bayesian analysis and computation with *Plug & Play* priors as in Section 4.2.1. In order to address ill-posedness, we also introduce prior knowledge about x by specifying an image denoising operator D_ε for recovering x from a noisy observation x_ε with $(X_\varepsilon|X=x) \sim \mathcal{N}(x, \varepsilon\,\mathrm{Id})$ with noise variance $\varepsilon > 0$. A case of particular relevance in this context is when D_ε is implemented by a neural network, trained by using a set of clean images $\{x'_i\}_{i=1}^N$. As explained in Section 4.2.1, a central challenge in the formalisation of Bayesian inference with *Plug & Play* priors is that the denoiser D_ε used is generally not directly related to a marginal distribution for x, so it is not possible to derive an explicit posterior for $x|y$ from D_ε. As a result, it is not clear that plugging D_ε into gradient-based algorithms such as ULA leads to a well-defined or convergent scheme that is targeting a meaningful Bayesian model.

To overcome this difficulty, in a manner akin to Chapter 4, we analyse in this chapter the Bayesian models through the prism of *M-complete* Bayesian modelling (Bernardo and Smith, 2000). We recall that, in this paradigm, there exists a true -albeit unknown and intractable- marginal distribution for x on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$, where $\mathcal{B}(\mathbb{R}^d)$ denotes the Borel σ-field of $\mathbb{R}^d$, noted μ, and a posterior distribution for $x|y$. If it were possible, basing inferences on these two distributions would be optimal both in terms of point estimation and in terms of delivering Bayesian probabilities accurate in a frequentist viewpoint. When μ admits a density w.r.t. the Lebesgue measure on $\mathbb{R}^d$, we denote it by $p^\star$. In the latter case, the posterior distribution for $x|y$ associated with the marginal μ also admits a density* that is

*Strictly speaking, the true likelihood $p^\star(y|x)$ may also be unknown, this is particularly relevant in the case of blind or myopic inverse imaging problems. For simplicity, we restrict our experiments and theoretical development to the case where $p(y|x)$ represents the true likelihood. Generalizations of our approach to the blind or semi-blind setting are discussed, e.g. by (Guo et al., 2019) - formalising these generalisations is an

given for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$ by

$$p^\star(x|y) = p(y|x)p^\star(x)/\textstyle\int_{\mathbb{R}^d} p(y|\tilde{x})p^\star(\tilde{x})\mathrm{d}\tilde{x}\,. \tag{5.1}$$

Unlike most Bayesian imaging approaches that operate implicitly in an *M-closed* manner and treat their postulated Bayesian models as true models (see (Bernardo and Smith, 2000) for more details), we explicitly regard $p^\star$ (or more precisely μ) as a fundamental property of the unknown x, and models used for inference as operational approximations of $p^\star$ specified by the practitioner (either analytically, algorithmically, or from training data). This distinction will be useful for using the oracle posterior (5.1) as a reference, and *Plug & Play* Bayesian algorithms based on a denoiser D_ε as approximations to reference algorithms to perform inference w.r.t. $p^\star$. The accuracy of the *Plug & Play* approximations will depend chiefly on the closeness between D_ε and an optimal denoiser $D^\star_\varepsilon$ derived form $p^\star$ that we define shortly.

In this conceptual construction, the marginal μ naturally depends on the imaging application considered. It could be the distribution of natural images of the size and resolution of x, or that of a class of images related to a specific application. And in problems where there is training data $\{x'_i\}_{i=1}^N$ available, we regard $\{x'_i\}_{i=1}^N$ as samples from μ. Lastly, we note that the posterior for $x|y$ remains well defined when μ does not admit a density; this is important to provide robustness to situations where $p^\star$ is nearly degenerate or improper. For clarity, our presentation assumes that $p^\star$ exists, although this is not strictly required [†].

Notice that because μ is unknown, we cannot verify that $p^\star(x|y)$ satisfies the basic desiderata for gradient-based Bayesian computation: i.e., $p^\star(x|y)$ need not be proper and differentiable, with $\nabla \log p^\star(x|y)$ Lipschitz continuous. To guarantee that gradient-based algorithms that target approximations of $p^\star(x|y)$ are well defined by construction, we introduce a regularised oracle μ_ε obtained via the convolution of μ with a Gaussian smoothing kernel with bandwidth $\varepsilon > 0$. Indeed, by construction, μ_ε has a smooth proper density p_ε given for any $x \in \mathbb{R}^d$ and $\varepsilon > 0$ by

$$p^\star_\varepsilon(x) = (2\pi\varepsilon)^{-d/2} \textstyle\int_{\mathbb{R}^d} \exp\left[-\|x-\tilde{x}\|_2^2/(2\varepsilon)\right] p^\star(\tilde{x})\mathrm{d}\tilde{x}\,.$$

Equipped with this regularised marginal distribution, we use Bayes' theorem to involve the likelihood $p(y|x)$ and derive the posterior density $p^\star_\varepsilon(x|y)$, given for any $\varepsilon > 0$ and $x \in \mathbb{R}^d$ by

$$p^\star_\varepsilon(x|y) = p(y|x)p^\star_\varepsilon(x)/\textstyle\int_{\mathbb{R}^d} p(y|\tilde{x})p^\star_\varepsilon(\tilde{x})\mathrm{d}\tilde{x}\,, \tag{5.2}$$

which inherits the regularity properties required for gradient-based Bayesian computation when the likelihood satisfies H1, which for presentation clarity we recall below.

H1 *For any $y \in \mathbb{R}^m$, $\sup_{x\in\mathbb{R}^d} p(y|x) < +\infty$, $p(y|\cdot) \in \mathrm{C}^1(\mathbb{R}^d, (0,+\infty))$ and there exists $\mathtt{L}_y > 0$ such that $\nabla \log(p(y|\cdot))$ is $\mathtt{L}_y$ Lipschitz continuous.*

More precisely, Proposition 5.2.1 below establishes that the regularised prior $p^\star_\varepsilon(x)$ and posterior $p^\star_\varepsilon(x|y)$ are proper, smooth, and that they can be made arbitrarily close to the original oracle models $p^\star(x)$ and $p^\star(x|y)$ by reducing ε, with the approximation error vanishing as $\varepsilon \to 0$.

Proposition 5.2.1 *Assume* H1. *Then, for any $\varepsilon > 0$ and $y \in \mathbb{R}^m$, the following hold:*

important perspective for future work.

[†]Operating without densities requires measure disintegration concepts that are technical (Schwartz).

(a) $p^\star_\varepsilon$ *and* $p^\star_\varepsilon(\cdot|y)$ *are proper.*

(b) *For any* $k \in \mathbb{N}$, $p^\star_\varepsilon \in \mathrm{C}^k(\mathbb{R}^d)$. *In addition, if* $p(y|\cdot) \in \mathrm{C}^k(\mathbb{R}^d)$ *then* $p^\star_\varepsilon(\cdot|y) \in \mathrm{C}^k(\mathbb{R}^d, \mathbb{R})$.

(c) *Let* $k \in \mathbb{N}$. *If* $\int_{\mathbb{R}^d} \|\tilde{x}\|^k \, p^\star(x)\mathrm{d}\tilde{x} < +\infty$ *then* $\int_{\mathbb{R}^d} \|\tilde{x}\|^k \, p^\star_\varepsilon(\tilde{x}|y)\mathrm{d}\tilde{x} < +\infty$.

(d) $\lim_{\varepsilon \to 0} \|p^\star_\varepsilon(\cdot|y) - p^\star(\cdot|y)\|_1 = 0$.

(e) *In addition, if there exist* $\kappa, \beta \geqslant 0$ *such that for any* $x \in \mathbb{R}^d$, $\|p^\star - p^\star(\cdot - x)\|_1 \leqslant \|x\|^\beta$, *then there exists* $C \geqslant 0$ *such that* $\|p^\star_\varepsilon(\cdot|y) - p^\star(\cdot|y)\|_1 \leqslant C\varepsilon^{\beta/2}$.

Proof: The proof is postponed to Appendix B.8.2. ■

Under H1 and $p(y|\cdot) \in \mathrm{C}^1(\mathbb{R}^d)$, $x \mapsto \nabla \log p^\star_\varepsilon(x|y)$ is well-defined and continuous. However, $x \mapsto \nabla \log p^\star_\varepsilon(x|y)$ might not be Lipschitz continuous and hence the Langevin SDE (3.25) might not have a strong solution. This requires an additional assumption on μ.

To study the Lipschitz continuity of $x \mapsto \nabla \log p^\star_\varepsilon(x|y)$, as well as to set the grounds for *Plug & Play* methods that define priors implicitly through a denoising algorithm, we use again the oracle MMSE denoiser $D^\star_\varepsilon$ defined in (3.20). For presentation clarity, we recall it below.

$$\forall \ (x, \varepsilon) \in \mathbb{R}^d \times \mathbb{R}^*_+, \ D^\star_\varepsilon(x) = (2\pi\varepsilon)^{-d/2} \int_{\mathbb{R}^d} \tilde{x} \exp\left[-\|x - \tilde{x}\|^2/(2\varepsilon)\right] p^\star(\tilde{x})\mathrm{d}\tilde{x} \ .$$

Under the assumption that the expected mean square error (MSE) is finite, $D^\star_\varepsilon$ is the MMSE estimator to recover an image $x \sim \mu$ from a noisy observation $x_\varepsilon \sim \mathcal{N}(x, \varepsilon \, \mathrm{Id})$ (Robert, 2007). Again, this optimal denoiser is a fundamental property of x and it is generally intractable. Motivated by the fact that state-of-the-art image denoisers are close-to-optimal in terms of MSE, in Section 5.3 we will characterise the accuracy of *Plug & Play* Bayesian methods for approximate inference w.r.t. $p^\star_\varepsilon(x|y)$ and $p^\star(x|y)$ as a function of the closeness between the denoiser D_ε used and the reference $D^\star_\varepsilon$.

To relate the gradient $x \mapsto \nabla \log p^\star_\varepsilon(x)$ and $D^\star_\varepsilon$, we use Tweedie's identity (Efron, 2011) which states that for all $x \in \mathbb{R}^d$

$$\varepsilon \nabla \log p^\star_\varepsilon(x) = D^\star_\varepsilon(x) - x \ , \tag{5.3}$$

and hence $x \mapsto \nabla \log p^\star_\varepsilon(x|y)$ is Lipschitz continuous if and only if $D^\star_\varepsilon$ has this property. We argue that this is a natural assumption on $D^\star_\varepsilon$, as it is essentially equivalent to assuming that the denoising problem underpinning $D^\star_\varepsilon$ is well-posed in the sense of Hadamard (recall that an inverse problem is said to be well posed if its solution is unique and Lipschitz continuous w.r.t to the observation (Stuart, 2010)). As established in Proposition 5.2.2 below, this happens when the expected MSE involved in using $D^\star_\varepsilon$ to recover x from $x_\varepsilon \sim \mathcal{N}(x, \varepsilon \, \mathrm{Id})$, where x has marginal μ, is finite and uniformly upper bounded for all $x_\varepsilon \in \mathbb{R}^d$.

Proposition 5.2.2 *Assume* H1. *Let* $\varepsilon > 0$. $\nabla \log p^\star_\varepsilon$ *is Lipschitz continuous if and only if there exists* $C \geqslant 0$ *such that for any* $x_\varepsilon \in \mathbb{R}^d$

$$\int_{\mathbb{R}^d} \left\|x - D^\star_\varepsilon(x_\varepsilon)\right\|^2 g_\varepsilon(x|x_\varepsilon)\mathrm{d}x \leqslant C \ , \tag{5.4}$$

where $g_\varepsilon(\cdot|x_\varepsilon)$ *is the density of the conditional distribution of the unknown image* $x \in \mathbb{R}^d$ *with marginal* μ, *given a noisy observation* $x_\varepsilon \sim \mathcal{N}(x, \varepsilon \, \mathrm{Id})$. *See Section 5.3.2 for details.*

Proof: The proof is postponed to Lemma B.6.2. ■

These results can be generalised to hold under the weaker assumption that the expected MSE for $D_\varepsilon^\star$ is finite but not uniformly bounded, as in this case $x \mapsto \nabla \log p_\varepsilon^\star(x|y)$ is locally instead of globally Lipschitz continuous (this technical extension is postponed to future work). The pathological case where $D_\varepsilon^\star$ does not have a finite MSE arises when μ is such that the denoising problem does not admit a Bayesian estimator w.r.t. to the MSE loss. In summary, the gradient $x \mapsto \nabla \log p_\varepsilon^\star(x|y)$ is Lipschitz continuous when μ carries enough information to make the problem of Bayesian image denoising under Gaussian additive noise well posed.

Notice that by using Tweedie's identity, we can express a ULA recursion for sampling approximately from $p_\varepsilon^\star(x|y)$ as follows:

$$X_{k+1} = X_k + \delta \nabla \log p(y|X_k) + (\delta/\varepsilon)\left(D_\varepsilon^\star(X_k) - X_k\right) + \sqrt{2\delta} Z_{k+1} \,. \tag{5.5}$$

where we recall that $\{Z_k : k \in \mathbb{N}\}$ are i.i.d standard Gaussian random variables on $\mathbb{R}^d$ and $\delta > 0$ is a positive step-size. Under standard assumptions on δ, the sequence generated by (5.5) is a Markov chain which admits an invariant probability distribution with a density provably close to $p_\varepsilon^\star(x|y)$, with δ controlling a trade-off between asymptotic accuracy and convergence speed. In the following section we present *Plug & Play* ULAs that arise from replacing $D_\varepsilon^\star$ in (5.5) with a denoiser D_ε that is tractable.

Before concluding this section, we study whether the oracle $p^\star(x|y)$ is itself well-posed, i.e., if $p^\star(x|y)$ changes continuously w.r.t. y under a suitable probability metric (see (Latz, 2020)). We answer positively to this question in Proposition 5.2.3 which states that, under mild assumptions on the likelihood, $p^\star(x|y)$ is locally Lipschitz continuous w.r.t. y for an appropriate metric. This stability result implies, for example, that the MMSE estimator derived from $p^\star(x|y)$ is locally Lipschitz continuous w.r.t. y, and hence stable w.r.t. small perturbations of y. Note that a similar property holds for the regularised posterior $p_\varepsilon^\star(x|y)$. In particular, Proposition 5.2.3 holds for Gaussian likelihoods (see Section 5.3 for details).

Proposition 5.2.3 *Assume that there exist* $\Phi_1 : \mathbb{R}^d \to [0, +\infty)$ *and* $\Phi_2 : \mathbb{R}^m \to [0, +\infty)$ *such that for any* $x \in \mathbb{R}^d$ *and* $y_1, y_2 \in \mathbb{R}^m$

$$\|\log(p(y_1|x)) - \log(p(y_2|x))\| \leqslant (\Phi_1(x) + \Phi_2(y_1) + \Phi_2(y_2)) \|y_1 - y_2\| \,, \tag{5.6}$$

and for any $c > 0$, $\int_{\mathbb{R}^d} (1 + \Phi_1(\tilde{x})) \exp[c\Phi_1(\tilde{x})] p^\star(x) \mathrm{d}\tilde{x} < +\infty$. *Then* $y \mapsto p^\star(\cdot|y)$ *is locally Lipschitz w.r.t* $\|\cdot\|_1$, i.e. *, for any compact set* K *there exists* $C_\mathsf{K} \geqslant 0$ *such that for any* $y_1, y_2 \in \mathsf{K}$, $\|p^\star(\cdot|y_1) - p^\star(\cdot|y_2)\|_1 \leqslant C_\mathsf{K} \|y_1 - y_2\|$.

Proof: The proof is a straightforward application of Proposition B.5.1. ■

To conclude, starting from the decision-theoretically optimal model $p^\star(x|y)$, we have constructed a regularised approximation $p_\varepsilon^\star(x|y)$ that is proper and smooth by construction, with gradients that are explicitely related to denoising operators by Tweedie's formula. Under mild assumptions on $p(y|x)$, the approximation $p_\varepsilon^\star(x|y)$ is well-posed and can be made arbitrarily close to the oracle $p^\star(x|y)$ by controlling ε. Moreover, we established that $x \mapsto \nabla \log p_\varepsilon^\star(x)$ is Lipschitz continuous when the problem of Gaussian image denoising for μ under the MSE loss is well posed. This allows imagining convergent gradient-based algorithms for performing Bayesian computation for $p_\varepsilon^\star(x|y)$, setting the basis for *Plug & Play* ULA schemes that mimic these idealised algorithms by using a tractable denoiser D_ϵ such as neural network, trained to optimise MSE performance and hence to approximate the oracle MSE denoiser $D_\epsilon^\star$.

5.2.2 Bayesian computation with *Plug & Play* priors

We are now ready to study *Plug & Play* ULA schemes to perform approximate inference w.r.t. $p^\star_\varepsilon(x|y)$ (and hence indirectly w.r.t. $p^\star(x|y)$). We use (5.5) as starting point, with $D^\star_\varepsilon$ replaced by a surrogate denoiser D_ε, but also modify (5.5) to guarantee geometrically fast convergence[‡] to a neighbourhood of $p^\star_\varepsilon(x|y)$. In particular, geometrically fast convergence is achieved here by modifying far-tail probabilities to prevent the Markov chain from becoming too diffusive as it explores the tails of $p^\star_\varepsilon(x|y)$. We consider two alternatives to guarantee geometric convergence with markedly different bias-variance trade-offs: one with excellent accuracy guarantees but that requires using a small step-size δ and hence has a higher computational cost, and another one that allows taking a larger step-size δ to improve convergence speed at the expense of weaker guarantees in terms of estimation bias.

First, in the spirit of Moreau-Yosida regularised ULA (Durmus et al., 2018), we define *Plug & Play* ULA (PnP-ULA) as the following recursion: given an initial state $X_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$,

$$\begin{aligned}\text{(PnP-ULA)} \quad X_{k+1} =& X_k + \delta \nabla \log p(y|X_k) + (\delta/\varepsilon)\,(D_\varepsilon(X_k) - X_k) \\ &+ (\delta/\lambda)(\Pi_{\mathsf{C}}(X_k) - X_k) + \sqrt{2\delta} Z_{k+1}\,,\end{aligned} \tag{5.7}$$

where $\mathsf{C} \subset \mathbb{R}^d$ is some large compact convex set that contains most of the prior probability mass of x, Π_{C} is the projection operator onto C w.r.t the Euclidean scalar product on $\mathbb{R}^d$, and $\lambda > 0$ is a tail regularisation parameter that is set such that the drift in PnP-ULA satisfies a certain growth condition as $\|x\| \to \infty$ (see Section 5.3 for details).

An alternative strategy (which we call Projected PnP-ULA, *i.e.* PPnP-ULA, see Algorithm 6) is to modify PnP-ULA to include a hard projection onto C, *i.e.* $(X_k)_{k\in\mathbb{N}}$ is defined by $X_0 \in \mathsf{C}$ and the following recursion for any $k \in \mathbb{N}$

$$X_{k+1} = \Pi_{\mathsf{C}}\left[X_k + \delta \nabla \log p(y|X_k) + (\delta/\varepsilon)(D_\varepsilon(X_k) - X_k) + \sqrt{2\delta} Z_{k+1}\right]\,, \tag{5.8}$$

where we notice that, by construction, the chain cannot exit C because of the action of the projection operator Π_{C}. The hard projection guarantees geometric convergence with weaker restrictions on δ and hence PPnP-ULA can be tuned to converge significantly faster than PnP-ULA, albeit with a potentially larger bias. These two schemes are summarised in Algorithm 5 and Algorithm 6 below. Note the presence of a regularisation parameter α in these algorithms, which permits to balance the weights between the prior and data terms. For the sake of simplicity, this parameter is set to $\alpha = 1$ in Section 5.3 and Section 5.4 but will be taken into account in the supplementary material Appendix B. Section 5.3.2 and Section 5.3.3 present detailed convergence results for PnP-ULA and PPnP-ULA. Implementation guidelines, including suggestions for how to set the algorithm parameters of PnP-ULA and PPnP-ULA are provided in Section 5.4.

Lastly, it is worth mentioning that Algorithm 5 and Algorithm 6 can be straightforwardly modified to incorporate additional regularisation terms. More precisely, one could consider a prior defined as the (normalised) product of a *Plug & Play* term and an explicit analytical term. In that case, one should simply modify the recursion defining the Markov chain by adding the gradient associated with the analytical term. In a manner akin to (Durmus et al., 2018), analytical terms that are not smooth are involved via their proximal operator.

Before concluding this section, it is worth emphasising that, in addition to being important in their own right, Algorithm 5 and Algorithm 6 and the associated theoretical results

[‡]Geometric convergence is highly desirable property in large-scale problems and guarantees that the generated Markov chains can be used for Monte Carlo integration.

Algorithm 5 PnP-ULA

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon, \lambda, \alpha, \delta > 0$, $\mathsf{C} \subset \mathbb{R}^d$ convex and compact
Ensure: $2\lambda(2\mathrm{L}_y + \alpha\mathrm{L}/\varepsilon) \leqslant 1$ and $\delta < (1/3)(\mathrm{L}_y + 1/\lambda + \alpha\mathrm{L}/\varepsilon)^{-1}$
  Initialization: Set $X_0 \in \mathbb{R}^d$ and $k = 0$.
  for $k = 0 : N$ **do**
    $Z_{k+1} \sim \mathcal{N}(0, \mathrm{Id})$
    $X_{k+1} = X_k + \delta\nabla \log(p(y|X_k)) + (\alpha\delta/\varepsilon)(D_\varepsilon(X_k) - X_k) + (\delta/\lambda)(\Pi_{\mathsf{C}}(X_k) - X_k) + \sqrt{2\delta} Z_{k+1}$
  end for
  return $\{X_k \,:\, k \in \{0, \ldots, N+1\}\}$

Algorithm 6 PPnP-ULA

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon, \lambda, \alpha, \delta > 0$, $\mathsf{C} \subset \mathbb{R}^d$ convex and compact
  Initialization: Set $X_0 \in \mathsf{C}$ and $k = 0$.
  for $k = 0 : N$ **do**
    $Z_{k+1} \sim \mathcal{N}(0, \mathrm{Id})$
    $X_{k+1} = \Pi_{\mathsf{C}}\left(X_k + \delta\nabla \log(p(y|X_k)) + (\alpha\delta/\varepsilon)(D_\varepsilon(X_k) - X_k) + \sqrt{2\delta} Z_{k+1}\right)$
  end for
  return $\{X_k \,:\, k \in \{0, \ldots, N+1\}\}$

set the grounds for analysing more advanced stochastic simulation and optimisation schemes for performing Bayesian inference with *Plug & Play* priors, in particular accelerated optimisation and sampling algorithms (Pereyra et al., 2020). This is an important perspective for future work.

5.3 Theoretical analysis

In this section, we provide a theoretical study of the long-time behaviour of PnP-ULA, see Algorithm 5 and PPnP-ULA, see Algorithm 6. For any $\varepsilon > 0$ we recall that $p^\star_\varepsilon$ is given by the Gaussian smoothing of p with level ε, for any $x \in \mathbb{R}^d$ by

$$p^\star_\varepsilon(x) = (2\pi\varepsilon)^{-d/2} \int_{\mathbb{R}^d} \exp[-\|x - \tilde{x}\|^2 / (2\varepsilon)]\, p^\star(\tilde{x}) \mathrm{d}\tilde{x} \,. \tag{5.9}$$

One typical example of likelihood function that we consider in our numerical illustration, see Section 5.4, is $p(y|x) \propto \exp[-\|\mathrm{A}x - y\|^2 / (2\sigma^2)]$ for any $x \in \mathbb{R}^d$ with $\sigma > 0$ and $\mathrm{A} \in \mathbb{R}^{m \times d}$. We define π the target posterior distribution given for any $x \in \mathbb{R}^d$ by $(\mathrm{d}\pi/\mathrm{dLeb})(x) = p^\star(x|y)$. We also consider the family of probability distributions $\{\pi_\varepsilon \,:\, \varepsilon > 0\}$ given for any $\varepsilon > 0$ and $x \in \mathbb{R}^d$ by

$$(\mathrm{d}\pi_\varepsilon/\mathrm{dLeb})(x) = p(y|x) p^\star_\varepsilon(x) \Big/ \int_{\mathbb{R}^d} p(y|\tilde{x}) p^\star_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} \,. \tag{5.10}$$

Note that in the supplementary material Appendix B we investigate the general setting where $p^\star_\varepsilon$ is replaced by $(p^\star_\varepsilon)^\alpha$ for some $\alpha > 0$ that acts as a regularisation parameter. We divide our study into two parts. We recall that π_ε is well-defined for any $\varepsilon > 0$ under H1, see Proposition 5.2.1. We start with some notation in Section 5.3.1. We then establish non-asymptotic bounds between the iterates of PnP-ULA and π_ε with respect to the total variation distance for any $\varepsilon > 0$, in Section 5.3.2. Finally, in Section 5.3.3 we establish similar results for PPnP-ULA.

5.3.1 Notation

Denote by $\mathcal{B}(\mathbb{R}^d)$ the Borel σ-field of $\mathbb{R}^d$, and for $f : \mathbb{R}^d \to \mathbb{R}$ measurable, $\|f\|_\infty = \sup_{\tilde{x}\in\mathbb{R}^d} |f(\tilde{x})|$. For μ a probability measure on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$ and f a μ-integrable function, denote by $\mu(f)$ the integral of f w.r.t. μ. For $f : \mathbb{R}^d \to \mathbb{R}$ measurable and $V : \mathbb{R}^d \to [1, \infty)$ measurable, the V-norm of f is given by $\|f\|_V = \sup_{\tilde{x}\in\mathbb{R}^d} |f(\tilde{x})|/V(\tilde{x})$. Let ξ be a finite signed measure on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$. The V-total variation distance of ξ is defined as

$$\|\xi\|_V = \sup_{\|f\|_V \leqslant 1} \left| \int_{\mathbb{R}^d} f(\tilde{x}) \mathrm{d}\xi(\tilde{x}) \right| . \tag{5.11}$$

If $V = 1$, then $\|\cdot\|_V$ is the total variation denoted by $\|\cdot\|_{\mathrm{TV}}$. Let U be an open set of $\mathbb{R}^d$. For any pair of measurable spaces $(\mathsf{X}, \mathcal{X})$ and $(\mathsf{Y}, \mathcal{Y})$, measurable function $f : (\mathsf{X}, \mathcal{X}) \to (\mathsf{Y}, \mathcal{Y})$ and measure μ on $(\mathsf{X}, \mathcal{X})$ we denote by $f_\# \mu$ the pushforward measure of μ on $(\mathsf{Y}, \mathcal{Y})$ given for any $\mathsf{A} \in \mathcal{Y}$ by $f_\#\mu(\mathsf{A}) = \mu(f^{-1}(\mathsf{A}))$. We denote $\mathscr{P}(\mathbb{R}^d)$ the set of probability measures over $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$ and for any $m \in \mathbb{N}$, $\mathscr{P}_m(\mathbb{R}^d) = \{\nu \in \mathscr{P}(\mathbb{R}^d) : \int_{\mathbb{R}^d} \|\tilde{x}\|^m \mathrm{d}\nu(\tilde{x}) < +\infty\}$.

We denote by $\mathrm{C}^k(\mathsf{U}, \mathbb{R}^m)$ and $\mathrm{C}^k_c(\mathsf{U}, \mathbb{R}^m)$ the set of $\mathbb{R}^m$-valued k-differentiable functions, respectively the set of compactly supported $\mathbb{R}^m$-valued and k-differentiable functions. Let $f : \mathsf{U} \to \mathbb{R}$, we denote by ∇f, the gradient of f if it exists. f is said to be $\mathtt{m}$-convex with $\mathtt{m} \geqslant 0$ if for all $x_1, x_2 \in \mathbb{R}^d$ and $t \in [0, 1]$,

$$f(tx_1 + (1-t)x_2) \leqslant tf(x_1) + (1-t)f(x_2) - \mathtt{m}t(1-t) \|x_1 - x_2\|^2 /2 . \tag{5.12}$$

For any $a \in \mathbb{R}^d$ and $R > 0$, denote $\mathrm{B}(a, R)$ the open ball centered at a with radius R. Let $(\mathsf{X}, \mathcal{X})$ and $(\mathsf{Y}, \mathcal{Y})$ be two measurable spaces. A Markov kernel K is a mapping $\mathrm{K} : \mathsf{X} \times \mathcal{Y} \to [0, 1]$ such that for any $\tilde{x} \in \mathsf{X}$, $\mathrm{P}(\tilde{x}, \cdot)$ is a probability measure and for any $\mathsf{A} \in \mathcal{Y}$, $\mathrm{P}(\cdot, \mathsf{A})$ is measurable. For any probability measure μ on $(\mathsf{X}, \mathcal{X})$ and measurable function $f : \mathsf{Y} \to \mathbb{R}_+$ we denote $\mu\mathrm{P} = \int_{\mathsf{X}} \mathrm{P}(x, \cdot)\mathrm{d}\mu(x)$ and $\mathrm{P}f = \int_{\mathsf{Y}} f(y)\mathrm{P}(\cdot, \mathrm{d}y)$. In what follows the Dirac mass at $\tilde{x} \in \mathbb{R}^d$ is denoted by $\delta_{\tilde{x}}$. For any $\tilde{x} \in \mathbb{R}^d$, we denote $\tau_{\tilde{x}} : \mathbb{R}^d \to \mathbb{R}^d$ the translation operator given for any $\tilde{x}' \in \mathbb{R}^d$ by $\tau_{\tilde{x}}(\tilde{x}') = \tilde{x}' - \tilde{x}$. The complement of a set $\mathsf{A} \subset \mathbb{R}^d$, is denoted by A^{c}. All densities are w.r.t. the Lebesgue measure (denoted Leb) unless stated otherwise. For all convex and closed set $\mathsf{C} \subset \mathbb{R}^d$, we define Π_{C} the projection operator onto C w.r.t the Euclidean scalar product on $\mathbb{R}^d$. For any matrix $\mathrm{a} \in \mathbb{R}^{d_1 \times d_2}$ with $d_1, d_2 \in \mathbb{N}$, we denote $\mathrm{a}^\top \in \mathbb{R}^{d_2 \times d_1}$ its adjoint.

5.3.2 Convergence of PnP-ULA

In this section, we fix $\varepsilon > 0$ and derive quantitative bounds between the iterates of PnP-ULA and π_ε with respect to the total variation distance. To address this issue, we first show that PnP-ULA is geometrically ergodic and establish non-asymptotic bounds between the corresponding Markov kernel and its invariant distribution. Second, we analyse the distance between this stationary distribution and π_ε.

For any $\varepsilon > 0$ we define $g_\varepsilon : \mathbb{R}^d \times \mathbb{R}^d \to [0, +\infty)$ for any $x_1, x_2 \in \mathbb{R}^d$ by

$$g_\varepsilon(x_1|x_2) = p^\star(x_1) \exp[-\|x_2 - x_1\|^2 /(2\varepsilon)] \Big/ \int_{\mathbb{R}^d} p^\star(\tilde{x}) \exp[-\|x_2 - \tilde{x}\|^2 /(2\varepsilon)] \mathrm{d}\tilde{x} . \tag{5.13}$$

Note that $g(\cdot|X_\varepsilon)$ is the density with respect to the Lebesgue measure of the distribution of X given X_ε, where X is sampled according to the prior distribution μ (with density $p^\star$) and $X_\varepsilon = X + \varepsilon^{1/2} Z$ where Z is a Gaussian random variable with zero mean and identity covariance matrix. Throughout, this section, we consider the following assumption on the family of denoising operators $\{D_\varepsilon : \varepsilon > 0\}$ which will ensure that PnP-ULA approximately targets π_ε.

H4 (R) *We have that* $\int_{\mathbb{R}^d} \|\tilde{x}\|^2 p^\star(\tilde{x})\mathrm{d}\tilde{x} < +\infty$. *In addition, there exist* $\varepsilon_0 > 0$, $\mathtt{M}_R \geqslant 0$ *and* $\mathtt{L} \geqslant 0$ *such that for any* $\varepsilon \in (0, \varepsilon_0]$, $x_1, x_2 \in \mathbb{R}^d$ *and* $x \in \overline{\mathrm{B}}(0, R)$ *we have*

$$\|(\mathrm{Id} - D_\varepsilon)(x_1) - (\mathrm{Id} - D_\varepsilon)(x_2)\| \leqslant \mathtt{L}\,\|x_1 - x_2\| \;, \qquad \|D_\varepsilon(x) - D_\varepsilon^\star(x)\| \leqslant \mathtt{M}_R \;, \tag{5.14}$$

where we recall that

$$D_\varepsilon^\star(x_1) = \textstyle\int_{\mathbb{R}^d} \tilde{x}\; g_\varepsilon(\tilde{x}|x_1)\mathrm{d}\tilde{x} \;. \tag{5.15}$$

H2 and H4 are very similar, except that the Lipschitz continuity condition concerns $\mathrm{Id} - D_\varepsilon$ in H4 and D_ε in H2. The Lipschitz continuity condition in (5.14) will be useful for establishing the stability and geometric convergence of the Markov chain generated by PnP-ULA. This condition can be explicitly enforced during training by using an appropriate regularization of the neural network weights (Ryu et al., 2019; Miyato et al., 2018). Regarding the second condition in (5.14), $\mathtt{M}_R$ is a bound on the error involved in using D_ε as an approximation of $D_\varepsilon^\star$ for images of magnitude R (i.e., for any $x \in \overline{\mathrm{B}}(0, R)$), and it will be useful for bounding the bias resulting from using PnP-ULA for inference w.r.t. π_ε (recall that the bias vanishes as $\mathtt{M}_R \to 0$ and $\delta \to 0$). For denoisers represented by neural networks, one can promote a small value of $\mathtt{M}_R$ during training by using an appropriate loss function. More precisely, consider a neural network $f_w : \mathbb{R}^d \to \mathbb{R}^d$, parameterized by its weights and bias gathered in $w \in \mathcal{W}$ where $\mathcal{W}$ is some measurable space, for any $\varepsilon > 0$, one could target empirical approximation of a loss of the form $\ell_\varepsilon : \mathcal{W} \to [0, +\infty)$ given for any $w \in \mathcal{W}$ by $\ell_\varepsilon(w) = \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - f_w(x_\varepsilon)\|^2 p_\varepsilon^\star(x_\varepsilon) g_\varepsilon(x|x_\varepsilon)\mathrm{d}x_\varepsilon\mathrm{d}x$. Note that such a loss is considered in the Noise2Noise network introduced in (Lehtinen et al., 2018).

With regards to the theoretical limitations stemming from representing D_ε by a deep neural network, universal approximation theorems (see e.g., (Bach, 2017, Section 4.7)) suggest that $\mathtt{M}_R$ could be arbitrarily low in principle. For a given architecture and training strategy, and if there exists $\tilde{\mathtt{M}}_R \geqslant 0$ such that $\inf_{w \in \mathcal{W}} \sup_{x \in \overline{\mathrm{B}}(0,R)} \tilde{\mathtt{M}}_R^{-1} \|f_w(x) - D_\varepsilon^\star(x)\|\} \leqslant 1$ then the second condition in (5.14) holds upon letting $D_\varepsilon = f_{w^\dagger}$ for an appropriate choice of weights $w^\dagger \in \mathcal{W}$. This last inequality can be established using universal approximation theorems such as (Bach, 2017, Section 4.7). Moreover, for any other $w \in \mathcal{W}$, $\ell_\varepsilon(w) \geqslant \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - D_\varepsilon^\star(x_\varepsilon)\|^2 p_\varepsilon^\star(x_\varepsilon) g_\varepsilon(x|x_\varepsilon)\mathrm{d}x\mathrm{d}x_\varepsilon = \ell_\varepsilon^\star$, since for any $x_\varepsilon \in \mathbb{R}^d$, $D_\varepsilon^\star(x_\varepsilon) = \int_{\mathbb{R}^d} \tilde{x}\; g_\varepsilon(\tilde{x}|x_\varepsilon)\mathrm{d}\tilde{x}$, see (5.15). Consider $w^\dagger \in \mathcal{W}$ obtained after numerically minimizing ℓ_ε and satisfying $\ell_\varepsilon(w^\dagger) \leqslant \ell_\varepsilon^\star + \eta$ with $\eta > 0$. In this case, the following result ensures that (5.14) is satisfied with $\mathtt{M}_R$ of order $\eta^{1/(2d+2)}$ for any $R > 0$ and letting $D_\varepsilon = f_{w^\dagger}$.

Proposition 5.3.1 *Assume that for any* $w \in \mathcal{W}$

$$\textstyle\int_{\mathbb{R}^d} (\|x\|^2 + \|f_w(x_\varepsilon)\|^2) p_\varepsilon^\star(x_\varepsilon) g_\varepsilon(x|x_\varepsilon)\mathrm{d}x\mathrm{d}x_\varepsilon < +\infty \;. \tag{5.16}$$

Let $R, \eta > 0$ *and* $w^\dagger \in \mathcal{W}$ *such that* $\ell_\varepsilon(w^\star) \leqslant \ell_\varepsilon^\star + \eta$. *In addition, assume that*

$$\sup_{x_1, x_2 \in \overline{\mathrm{B}}(0,2R)} \left\{ \|x_2 - x_1\|^{-1} \left(\|f_{w^\dagger}(x_2) - f_{w^\dagger}(x_1)\| + \|D_\varepsilon^\star(x_2) - D_\varepsilon^\star(x_1)\| \right) \right\} < +\infty \;, \tag{5.17}$$

where $D_\varepsilon^\star$ *is given in* (5.15). *Then there exists* $C_R, \bar{\eta}_R \geqslant 0$ *such that if* $\eta \in (0, \bar{\eta}_R]$ *then for any* $\tilde{x} \in \overline{\mathrm{B}}(0, R)$, $\|f_{w^\dagger}(\tilde{x}) - D_\varepsilon^\star(\tilde{x})\| \leqslant C_R \eta^{1/(2d+2)}$.

Proof: The proof is postponed to Appendix B.6.1. ■

Note that (5.16) is satisfied if for any $w \in \mathcal{W}$, $\sup_{x \in \mathbb{R}^d} \|f_w(x)\|(1 + \|x\|)^{-1} < +\infty$ and H4 holds.

We recall that PnP-ULA, see Algorithm 5, is given by the following recursion: $X_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$

$$X_{k+1} = X_k + \delta b_\varepsilon(X_k) + \sqrt{2\delta} Z_{k+1} \;, \tag{5.18}$$

$$b_\varepsilon(x) = \nabla \log p(y|x) + P_\varepsilon(x) + (\mathrm{prox}_\lambda(\iota_{\mathsf{C}})(x) - x)/\lambda \;, \quad P_\varepsilon(x) = (D_\varepsilon(x) - x)/\varepsilon \;, \tag{5.19}$$

where $\delta > 0$ is a step-size, $\varepsilon, \lambda > 0$ are hyperparameters of the algorithm, $\mathsf{C} \subset \mathbb{R}^d$ is a closed convex set, $\{Z_k \;:\; k \in \mathbb{N}\}$ a family of i.i.d. Gaussian random variables with zero mean and identity covariance matrix and $\mathrm{prox}_\lambda(\iota_{\mathsf{C}})$ the proximal operator of ι_{C} with step-size λ, see (Bauschke et al., 2011, Definition 12.23), where ι_{C} is the convex indicator of C defined for $x \in \mathbb{R}^d$ by $\iota_{\mathsf{C}} = +\infty$ if $x \notin \mathsf{C}$ and 0 if $x \in \mathsf{C}$. Note that for any $x \in \mathbb{R}^d$ we have $\mathrm{prox}_\lambda(\iota_{\mathsf{C}})(x) = \Pi_{\mathsf{C}}(x)$, where Π_{C} is the projection onto C.

In what follows, for any $\delta > 0$ and $\mathsf{C} \subset \mathbb{R}^d$ closed and convex, we denote by $\mathrm{R}_{\varepsilon,\delta} : \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0,1]$ the Markov kernel associated with the recursion (5.18) and given for any $x \in \mathbb{R}^d$ and $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ by

$$\mathrm{R}_{\varepsilon,\delta}(x, \mathsf{A}) = (2\pi)^{-d/2} \int_{\mathbb{R}^d} \mathbb{1}_{\mathsf{A}}(x + \delta b_\varepsilon(x) + \sqrt{2\delta} z) \exp[-\|z\|^2/2] \mathrm{d}z \;. \tag{5.20}$$

Note that for ease of notation, we do not explicitly highlight the dependency of $\mathrm{R}_{\varepsilon,\delta}$ and b_ε with respect to the hyperparameter $\lambda > 0$ and C.

Here we consider the case where $x \mapsto \log p(y|x)$ satisfies a one-sided Lipschitz condition, *i.e.* we consider the following condition.

H5 *There exists* $\mathtt{m} \in \mathbb{R}$ *such that for any* $x_1, x_2 \in \mathbb{R}^d$ *we have*

$$\langle \nabla \log p(y|x_2) - \nabla \log p(y|x_1), x_2 - x_1 \rangle \leqslant -\mathtt{m} \|x_2 - x_1\|^2 \;. \tag{5.21}$$

We refer to the supplementary material Appendix B.3 for refined convergence rates in the case where $x \mapsto \log p(y|x)$ is strongly $\mathtt{m}$-concave. Note that if H5 is satisfied with $\mathtt{m} > 0$ then $x \mapsto \log p(y|x)$ is $\mathtt{m}$-concave. Assume H1 then H5 holds for $\mathtt{m} = -\mathtt{L}_y$. However, it is possible that $\mathtt{m} > -\mathtt{L}_y$ which leads to better convergence rates for PnP-ULA. As a result even when H1 holds we still consider H5. In order to deal with H5 in the case where $\mathtt{m} \leqslant 0$, we set $\mathsf{C} \subset \mathbb{R}^d$ to be some convex compact set fixed by the user. Doing so, we ensure the stability of the Markov chain. The choice of C in practice is discussed in Section 5.4. In our imaging experiments, we recall that for any $x \in \mathbb{R}^d$ we have, $p(y|x) \propto \exp[-\|\mathrm{A}x - y\|^2/(2\sigma^2)]$. If A is not invertible then $x \mapsto \log p(y|x)$ is not $\mathtt{m}$-concave with $\mathtt{m} > 0$. This is the case, in our deblurring experiment when the convolution kernel has zeros in the Fourier domain.

We start with the following result which ensures that the Markov chain (5.18) is geometrically ergodic under H4 for the Wasserstein metric $\mathbf{W}_1$ and in V-norm for $V : \mathbb{R}^d \to [1, +\infty)$ given for any $x \in \mathbb{R}^d$ by

$$V(x) = 1 + \|x\|^2 \;. \tag{5.22}$$

Proposition 5.3.2 *Assume* H1, H4(R) *for some* $R > 0$ *and* H5. *Let* $\lambda > 0$, $\varepsilon \in (0, \varepsilon_0]$ *such that* $2\lambda(\mathtt{L}_y + \mathtt{L}/\varepsilon - \min(\mathtt{m}, 0)) \leqslant 1$ *and* $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. *Then for any* $\mathsf{C} \subset \mathbb{R}^d$ *convex and compact with* $0 \in \mathsf{C}$, *there exist* $A_{1,\mathsf{C}} \geqslant 0$ *and* $\rho_{1,\mathsf{C}} \in [0,1)$ *such that for any* $\delta \in (0, \bar{\delta}]$, $x_1, x_2 \in \mathbb{R}^d$ *and* $k \in \mathbb{N}$ *we have*

$$\left\| \delta_{x_1} \mathrm{R}^k_{\varepsilon,\delta} - \delta_{x_2} \mathrm{R}^k_{\varepsilon,\delta} \right\|_V \leqslant A_{1,\mathsf{C}} \rho^{k\delta}_{1,\mathsf{C}} (V^2(x_1) + V^2(x_2)) \;, \tag{5.23}$$

$$\mathbf{W}_1(\delta_{x_1} \mathrm{R}^k_{\varepsilon,\delta}, \delta_{x_2} \mathrm{R}^k_{\varepsilon,\delta}) \leqslant A_{1,\mathsf{C}} \rho^{k\delta}_{1,\mathsf{C}} \|x_1 - x_2\| \;, \tag{5.24}$$

where V *is given in* (5.22).

Proof: The proof is postponed to Appendix B.6.2. ■

The constants $A_{1,\mathsf{C}}$ and $\rho_{1,\mathsf{C}}$ do not depend on the dimension d but only on the parameters $\mathtt{m}, \mathtt{L}, \mathtt{L}_y, \varepsilon$ and C. Note that a similar result can be established for $\mathbf{W}_p$ for any $p \in \mathbb{N}^*$ instead of $\mathbf{W}_1$. Under the conditions of Proposition 5.3.2 we have for any $\nu_1, \nu_2 \in \mathscr{P}_1(\mathbb{R}^d)$

$$\left\|\nu_1 \mathrm{R}_{\varepsilon,\delta}^k - \nu_2 \mathrm{R}_{\varepsilon,\delta}^k\right\|_V \leqslant A_{1,\mathsf{C}} \rho_{1,\mathsf{C}}^{k\delta} \left(\int_{\mathbb{R}^d} V^2(\tilde{x}) \mathrm{d}\nu_1(\tilde{x}) + \int_{\mathbb{R}^d} V^2(\tilde{x}) \mathrm{d}\nu_2(\tilde{x}) \right) , \tag{5.25}$$

$$\mathbf{W}_1(\nu_1 \mathrm{R}_{\varepsilon,\delta}^k, \nu_2 \mathrm{R}_{\varepsilon,\delta}^k) \leqslant A_{1,\mathsf{C}} \rho_{1,\mathsf{C}}^{k\delta} \left(\int_{\mathbb{R}^d} \|\tilde{x}\| \, \mathrm{d}\nu_1(\tilde{x}) + \int_{\mathbb{R}^d} \|\tilde{x}\| \, \mathrm{d}\nu_2(\tilde{x}) \right) . \tag{5.26}$$

First, $(\mathscr{P}_1(\mathbb{R}^d), \mathbf{W}_1)$ is a complete metric space (Villani, 2009, Theorem 6.18). Second, for any $\delta \in (0, \bar{\delta}]$, there exists $\mathrm{m} \in \mathbb{N}^*$ such that f^{m} is contractive with $\mathrm{f} : \mathscr{P}_1(\mathbb{R}^d) \to \mathscr{P}_1(\mathbb{R}^d)$ given for any $\nu \in \mathscr{P}_1(\mathbb{R}^d)$ by $\mathrm{f}(\nu) = \nu \mathrm{R}_{\varepsilon,\delta}$ using Proposition 5.3.2. Therefore we can apply the Picard fixed point theorem and we obtain that $\mathrm{R}_{\varepsilon,\delta}$ admits an invariant probability measure $\pi_{\varepsilon,\delta} \in \mathscr{P}_1(\mathbb{R}^d)$.

Therefore, since $\pi_{\varepsilon,\delta}$ is an invariant probability measure for $\mathrm{R}_{\varepsilon,\delta}$ and $\pi_{\varepsilon,\delta} \in \mathscr{P}_1(\mathbb{R}^d)$, using (5.25), we have for any $\nu \in \mathscr{P}_1(\mathbb{R}^d)$

$$\left\|\nu \mathrm{R}_{\varepsilon,\delta}^k - \pi_{\varepsilon,\delta}\right\|_V \leqslant A_{1,\mathsf{C}} \rho_{1,\mathsf{C}}^{k\delta} \left(\int_{\mathbb{R}^d} V^2(\tilde{x}) \mathrm{d}\nu(\tilde{x}) + \int_{\mathbb{R}^d} V^2(\tilde{x}) \mathrm{d}\pi_{\varepsilon,\delta}(\tilde{x}) \right) , \tag{5.27}$$

$$\mathbf{W}_1(\nu \mathrm{R}_{\varepsilon,\delta}^k, \pi_{\varepsilon,\delta}) \leqslant A_{1,\mathsf{C}} \rho_{1,\mathsf{C}}^{k\delta} \left(\int_{\mathbb{R}^d} \|\tilde{x}\| \, \mathrm{d}\nu(\tilde{x}) + \int_{\mathbb{R}^d} \|\tilde{x}\| \, \mathrm{d}\pi_{\varepsilon,\delta}(\tilde{x}) \right) . \tag{5.28}$$

Combining this result with the fact that for any $t \geqslant 0$, $(1 - \mathrm{e}^{-t})^{-1} \leqslant 1 + t^{-1}$, we get that for any $n \in \mathbb{N}^*$ and $h : \mathbb{R}^d \to \mathbb{R}$ measurable such that $\sup_{x \in \mathbb{R}^d} \{(1 + \|x\|^2)^{-1} |h(x)|\} < +\infty$

$$\left| n^{-1} \sum_{k=1}^{n} \mathbb{E}[h(X_k)] - \int_{\mathbb{R}^d} h(\tilde{x}) \mathrm{d}\pi_{\varepsilon,\delta}(\tilde{x}) \right| \tag{5.29}$$

$$\leqslant A_{1,\mathsf{C}} (\bar{\delta} + \log^{-1}(1/\rho_{1,\mathsf{C}})) \left(V^2(x) + \int_{\mathbb{R}^d} V^2(\tilde{x}) \mathrm{d}\pi_{\varepsilon,\delta}(\tilde{x}) \right) \Big/ (n\delta) \ , \tag{5.30}$$

where $(X_k)_{k \in \mathbb{N}}$ is the Markov chain given by (5.18) with starting point $X_0 = x \in \mathbb{R}^d$.

In the rest of this section we evaluate how close the invariant measure $\pi_{\varepsilon,\delta}$ is to π_ε. Our proof will rely on the following assumption which is necessary to ensure that $x \mapsto \log p_\varepsilon^\star(x)$ has Lipschitz gradients, see Proposition 5.2.2.

H6 *For any $\varepsilon > 0$, there exists $\mathtt{K}_\varepsilon \geqslant 0$ such that for any $x \in \mathbb{R}^d$,*

$$\int_{\mathbb{R}^d} \left\| \tilde{x} - \int_{\mathbb{R}^d} \tilde{x}' g_\varepsilon(\tilde{x}'|x) \mathrm{d}\tilde{x}' \right\|^2 g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} \leqslant \mathtt{K}_\varepsilon \ , \tag{5.31}$$

with g_ε given in (5.13).

We emphasize that H6 is not needed to establish the convergence of the Markov chain. However, we impose it in order to compare the stationary distribution of PnP-ULA with the target distribution π_ε. Depending on the prior distribution density $p^\star$, H6 may be checked by hand. Finally, note that H6 can be extended to cover the case where the prior distribution μ does not admit a density with respect to the Lebesgue measure.

In the following proposition, we show that we can control the distance between $\pi_{\varepsilon,\delta}$ and π_ε based on the previous observations.

Proposition 5.3.3 *Assume* H1, H4(R) *for some* $R > 0$, H5 *and* H6. *Moreover, let* $\varepsilon \in (0, \varepsilon_0]$ *and assume that* $\int_{\mathbb{R}^d}(1 + \|\tilde{x}\|^4)p^\star_\varepsilon(\tilde{x})\mathrm{d}\tilde{x} < +\infty$. *Let* $\lambda > 0$ *such that* $2\lambda(\mathtt{L}_y + (/\varepsilon)\max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon) - \min(\mathtt{m}, 0)) \leqslant 1$ *and* $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. *Then for any* $\delta \in (0, \bar{\delta}]$ *and* C *convex and compact with* $0 \in \mathsf{C}$, $\mathrm{R}_{\varepsilon,\delta}$ *admits an invariant probability measure* $\pi_{\varepsilon,\delta}$. *In addition, there exists* $B_0 \geqslant 0$ *such that for any* C *convex compact with* $\overline{\mathrm{B}}(0, R_\mathsf{C}) \subset \mathsf{C}$ *and* $R_\mathsf{C} > 0$, *there exists* $B_{1,\mathsf{C}} \geqslant 0$ *such that for any* $\delta \in (0, \bar{\delta}]$

$$\|\pi_{\varepsilon,\delta} - \pi_\varepsilon\|_V \leqslant B_0 R_\mathsf{C}^{-1} + B_{1,\mathsf{C}}(\delta^{1/2} + \mathtt{M}_R + \exp[-R]) \,, \tag{5.32}$$

where V *is given in* (5.22).

Proof: The proof is postponed to Appendix B.6.3. ■

We now combine Proposition 5.3.2 and Proposition 5.3.3 in order to control the bias of the Monte Carlo estimator obtained using PnP-ULA. In the supplementary material Appendix B.4 we also provide bounds on $|n^{-1}\sum_{k=1}^n \mathbb{E}[h(X_k)] - \int_{\mathbb{R}^d} h(\tilde{x})\mathrm{d}\pi(\tilde{x})|$ by controlling $\|\pi - \pi_\varepsilon\|_V$.

Proposition 5.3.4 *Assume* H1, H4(R) *for some* $R > 0$, H5 *and* H6. *Moreover, let* > 0, $\varepsilon \in (0, \varepsilon_0]$ *and assume that* $\int_{\mathbb{R}^d}(1 + \|\tilde{x}\|^4)p^\star_\varepsilon(\tilde{x})\mathrm{d}\tilde{x} < +\infty$. *Let* $\lambda > 0$ *such that* $2\lambda(\mathtt{L}_y + (1/\varepsilon)\max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon) - \min(\mathtt{m}, 0)) \leqslant 1$ *and* $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. *Then there exists* $C_{1,\varepsilon} > 0$ *such that for any* C *convex compact with* $\overline{\mathrm{B}}(0, R_\mathsf{C}) \subset \mathsf{C}$ *and* $R_\mathsf{C} > 0$ *there exists* $C_{2,\varepsilon}$ *such that for any* $h : \mathbb{R}^d \to \mathbb{R}$ *measurable with* $\sup_{x\in\mathbb{R}^d}\{|h(x)|\,(1 + \|x\|^2)^{-1}\} \leqslant 1$, $n \in \mathbb{N}^*$, $\delta \in (0, \bar{\delta}]$ *we have*

$$\left|n^{-1}\sum_{k=1}^n \mathbb{E}\left[h(X_k)\right] - \int_{\mathbb{R}^d} h(\tilde{x})\mathrm{d}\pi_\varepsilon(\tilde{x})\right| \leqslant \left\{C_{1,\varepsilon}R_\mathsf{C}^{-1} + C_{2,\varepsilon,\mathsf{C}}(\delta^{1/2} + \mathtt{M}_R + \exp[-R] + (n\delta)^{-1})\right\}(1 + \|x\|^4) \,. \tag{5.33}$$

Proof: The proof is straightforward combining Proposition 5.3.2 and Proposition 5.3.3. ■

5.3.3 Convergence guarantees for PPnP-ULA

We now study the Projected *Plug & Play* Unadjusted Langevin Algorithm (PPnP-ULA). It is given by the following recursion: $X_0 \in \mathsf{C}$ and for any $k \in \mathbb{N}$

$$X_{k+1} = \Pi_\mathsf{C}(X_k + \delta b_\varepsilon(X_k) + \sqrt{2\delta}Z_{k+1}) \,, \tag{5.34}$$

$$b_\varepsilon(x) = \nabla \log p(y|x) + P_\varepsilon(x) \,, \quad P_\varepsilon(x) = (D_\varepsilon(x) - x)/\varepsilon \,, \tag{5.35}$$

where $\delta > 0$ is a step-size, $\varepsilon > 0$ is an hyperparameter of the algorithm, $\mathsf{C} \subset \mathbb{R}^d$ is a closed convex set, $\{Z_k : k \in \mathbb{N}\}$ a family of i.i.d. Gaussian random variables with zero mean and identity covariance matrix and where Π_C is the projection onto C. In what follows, for any $\delta > 0$ and $\mathsf{C} \subset \mathbb{R}^d$ closed and convex, we denote by $\mathrm{Q}_{\varepsilon,\delta} : \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0, 1]$ the Markov kernel associated with the recursion (5.34) and given for any $x \in \mathbb{R}^d$ and $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ by

$$\mathrm{Q}_{\varepsilon,\delta}(x, \mathsf{A}) = (2\pi)^{-d/2}\int_{\mathbb{R}^d} \mathbb{1}_{\Pi_\mathsf{C}^{-1}(\mathsf{A})}(x + \delta b_\varepsilon(x) + \sqrt{2\delta}z)\exp[-\|z\|^2/2]\mathrm{d}z \,. \tag{5.36}$$

Note that for ease of notation, we do not explicitly highlight the dependency of $\mathrm{Q}_{\varepsilon,\delta}$ and b_ε with respect to the hyperparameter C.

First, we have the following result which ensures that PPnP-ULA is geometrically ergodic for all step-sizes.

Proposition 5.3.5 *Assume* H1, H4(R) *for some* $R > 0$. *Let* $\lambda, \varepsilon, \bar{\delta} > 0$. *Then for any* $\mathsf{C} \subset \mathbb{R}^d$ *convex and compact with* $0 \in \mathsf{C}$, *there exist* $\tilde{A}_{\mathsf{C}} \geqslant 0$ *and* $\tilde{\rho}_{\mathsf{C}} \in [0, 1)$ *such that for any* $\delta \in (0, \bar{\delta}]$, $x_1, x_2 \in \mathsf{C}$ *and* $k \in \mathbb{N}$ *we have*

$$\|\delta_{x_1} \mathrm{Q}_{\varepsilon,\delta}^k - \delta_{x_2} \mathrm{Q}_{\varepsilon,\delta}^k\|_{\mathrm{TV}} \leqslant \tilde{A}_{\mathsf{C}} \tilde{\rho}_{\mathsf{C}}^{k\delta} \,. \tag{5.37}$$

Proof: The proof is postponed to Appendix B.7.1. ■

In particular $\mathrm{Q}_{\varepsilon,\delta}$ admits an invariant probability measure $\pi_{\varepsilon,\delta}^{\mathsf{C}}$. The next proposition ensures that for small enough step-size δ the invariant measures of PnP-ULA and PPnP-ULA are close if the compact convex set C has a large diameter.

Proposition 5.3.6 *Assume* H1, H4(R) *for some* $R > 0$ *and* H5. *In addition, assume that there exists* $\tilde{\mathtt{m}}, c > 0$ *such that for* $\mathsf{C} = \mathbb{R}^d$ *and for any* $\varepsilon > 0$ *and* $x \in \mathbb{R}^d$, $\langle b_\varepsilon(x), x \rangle \leqslant -\tilde{\mathtt{m}} \|x\|^2 + c$. *Let* $\lambda > 0$, $\varepsilon \in (0, \varepsilon_0]$ *such that* $2\lambda(\mathtt{L}_y + \mathtt{L}/\varepsilon - \min(\mathtt{m}, 0)) \leqslant 1$. *Then there exist* $\bar{A} \geqslant 0$ *and* $\eta, \bar{\delta} > 0$ *such that for any* $\mathsf{C} \subset \mathbb{R}^d$ *convex and compact with* $0 \in \mathsf{C}$ *and* $\overline{\mathrm{B}}(0, R_{\mathsf{C}}/2) \subset \mathsf{C} \subset \overline{\mathrm{B}}(0, R_{\mathsf{C}})$ *and* $\delta \in (0, \bar{\delta}]$ *we have*

$$\|\pi_{\varepsilon,\delta} - \pi_{\varepsilon,\delta}^{\mathsf{C}}\|_{\mathrm{TV}} \leqslant \bar{A} \exp[-\eta R_{\mathsf{C}}] \,, \tag{5.38}$$

where $\pi_{\varepsilon,\delta}$ *is the invariant measure of* $\mathrm{R}_{\varepsilon,\delta}$ *and* $\pi_{\varepsilon,\delta}^{\mathsf{C}}$ *is the invariant measure of* $\mathrm{Q}_{\varepsilon,\delta}$.

Proof: The proof is postponed to Appendix B.7.2. ■

It is worth mentioning at this point that in our experiments, see Section 5.4, the probability of the iterates $(X_n)_{n\in\mathbb{N}}$ leaving C with PnP-ULA or with PPnP-ULA is so low that the projection constraint is not activated. As a result, if implemented with the same step-size both algorithms produce the same results. We do not suggest completely removing the constraints as this is important to theoretically guarantee the geometric ergodicity of the algorithms.

Regarding the choice of the step-size, we observe that the bound $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$ used in PnP-ULA is conservative and our experiments suggest that PnP-ULA is stable for larger step-sizes.

5.4 Experimental study

This section illustrates the behaviour of PnP-ULA and PPnP-ULA with two classical imaging inverse problems: non-blind image deblurring and interpolation. For these two problems, we first analyse in detail the convergence of the Markov chain generated by PnP-ULA for different test images. This is then followed by a comparison between the MMSE Bayesian point estimator, as calculated by using PnP-ULA and PPnP-ULA and the MAP estimator provided by the recent PnP-SGD method presented in Chapter 4. To simplify comparisons, for all experiments and algorithms, the operator D_ε is chosen as the pretrained denoising neural network introduced in (Ryu et al., 2019), for which $(D_\varepsilon - \mathrm{Id})$ is $\mathtt{L}$-Lipschitz with $\mathtt{L} < 1$.

For the deblurring experiments, the observation model takes the form

$$y = \mathrm{A}x + n \,, \tag{5.39}$$

where $x \in \mathbb{R}^d$ is the unknown original image, $y \in \mathbb{R}^m$ the observed image, n is a realization of a Gaussian i.i.d. centered noise with variance $\sigma^2 \,\mathrm{Id}$ (with $\sigma^2 = (1/255)^2$), and A is a 9×9 box blur operator. The log-likelihood for this case writes $\log p(y|x) = -\|\mathrm{A}x - y\|^2/(2\sigma^2)$.

In the interpolation experiments, we seek to recover $x \in \mathbb{R}^d$ from $y = \mathrm{A}x$ where the matrix A is an $m \times d$ matrix containing m randomly selected rows of the $d \times d$ identity matrix. We focus on a case where 80% of the image pixels are hidden and the observed pixels are measured without any noise. Because the posterior density for $x|y$ is degenerate, we run PnP-ULA on the posterior $\tilde{x}|y$ where $\tilde{x} := \mathrm{P}x \in \mathbb{R}^n$ denotes the vector of $n = d - m$ unobserved pixels of x, and map samples to the pixel space by using the affine mapping $f_y : \mathbb{R}^n \to \mathbb{R}^d$ defined for any $\tilde{x} \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ by

$$f_y(\tilde{x}) = \mathrm{P}^\top \tilde{x} + \mathrm{A}^\top y.$$

Note that we can write the log-posterior $\tilde{U}_\varepsilon(\tilde{x}) = -\log p_\varepsilon(\tilde{x}|y)$ on the set $\mathbb{R}^n$ of hidden pixels in terms of f_y and the log-prior $U_\varepsilon(x) = -\log p_\varepsilon(x)$ on the set $\mathbb{R}^d$:

$$\tilde{U}_\varepsilon = U_\varepsilon \circ f_y.$$

Using the chain rule and Tweedie's formula, we have that for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$

$$b_\varepsilon(\tilde{x}) = -\nabla \tilde{U}_\varepsilon(\tilde{x}) = -\mathrm{P}\nabla U_\varepsilon(f_y(\tilde{x})) = (1/\varepsilon)\mathrm{P}(D_\varepsilon - \mathrm{Id})(f_y(\tilde{x}))\,. \tag{5.40}$$

Since P and f_y are 1-Lipschitz, $b_\varepsilon = -\nabla \tilde{U}_\varepsilon$ is also Lipschitz with constant $\tilde{\mathtt{L}} \leqslant (\mathtt{L}/\varepsilon)$.

Figure 5.1 shows the six test images of size 256×256 pixels that were used in the experiments. We have selected these six images for their diversity in composition, content and level of detail (some images are predominantly composed of piece-wise constant regions, whereas others are rich in complex textures). This diversity will highlight strengths and limitations of the chosen denoiser as an image prior. Figure 5.2 depicts the corresponding blurred images and Figure 5.3 the images to interpolate.

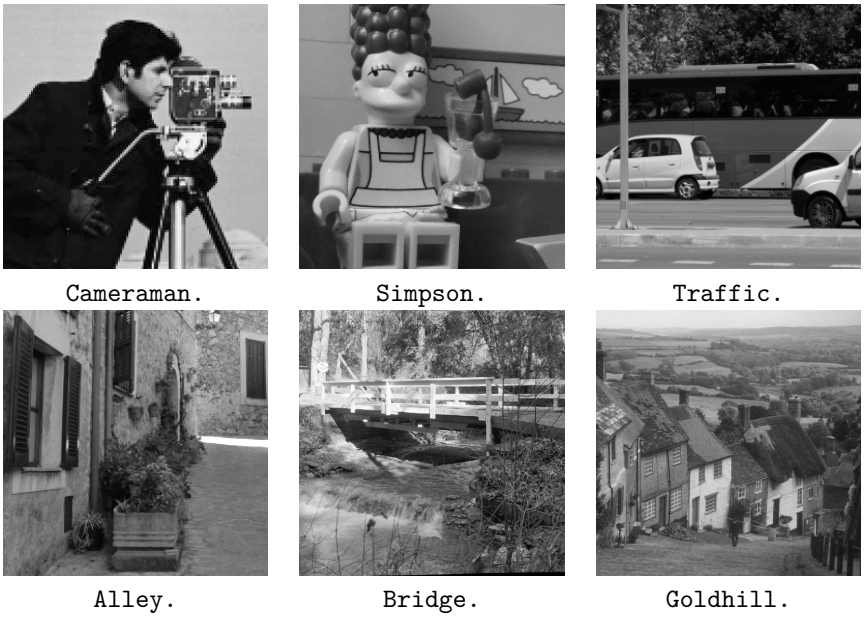
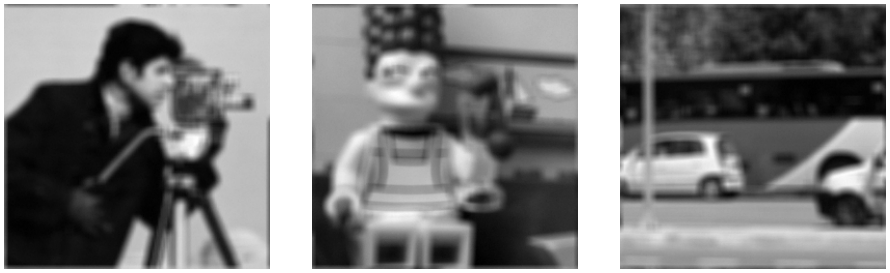


Figure 5.1: Original images used for the deblurring and interpolation experiments.



PSNR=20.30/SSIM=0.70 PSNR=22.44/SSIM=0.66 PSNR=20.34/SSIM=0.49

PSNR=22.64/SSIM=0.46 PSNR=21.84/SSIM=0.49 PSNR=22.61/SSIM=0.45

Figure 5.2: Images of Figure 5.1, blurred using a 9×9-box-filter operator and corrupted by an additive Gaussian white noise with standard deviation $\sigma = 1/255$.

5.4.1 Implementation guidelines and parameter setting

In the following, we provide some simple and robust rules in order to set the parameters of the different algorithms, in particular the discretization step-size δ and the tail regularization parameter λ.

CHOICE OF THE DENOISER The theory presented in Section 5.3 requires that D_ε satisfies H4(R). As default choice, we recommend using a pretrained denoising neural network such as the one described in (Ryu et al., 2019). The Lipschitz constant of the network is controlled during training by using spectral normalization and therefore the first condition of H4(R) holds. Moreover, the loss function used to train the network is given by ℓ_ε as introduced in Section 5.3.2. Therefore, under the conditions of Proposition 5.3.1, we get that the second condition of H4(R) holds.

STEP-SIZE δ The parameter δ controls the asymptotic accuracy of PnP-ULA and PPnP-ULA, as well as the speed of convergence to stationarity. This leads to the following bias-variance trade-off. For large values of δ, the Markov chain has low auto-correlation and converges quickly to its stationary regime. Consequently, the Monte Carlo estimates computed from the chain exhibit low asymptotic variance, at the expense of some asymptotic bias. On the contrary, small values of δ produce a Markov chain that explores the parameter space less efficiently, but more accurately. As a result, the asymptotic bias is smaller, but the variance is larger. In the context of inverse problems that are high-dimensional and ill-posed, properly exploring the solution space can take a large number of iterations. For this reason, we recommend using large values of δ, at the expense of some bias. In addition, in PnP-ULA, δ is also subject to a numerical stability constraint related to the inverse of the

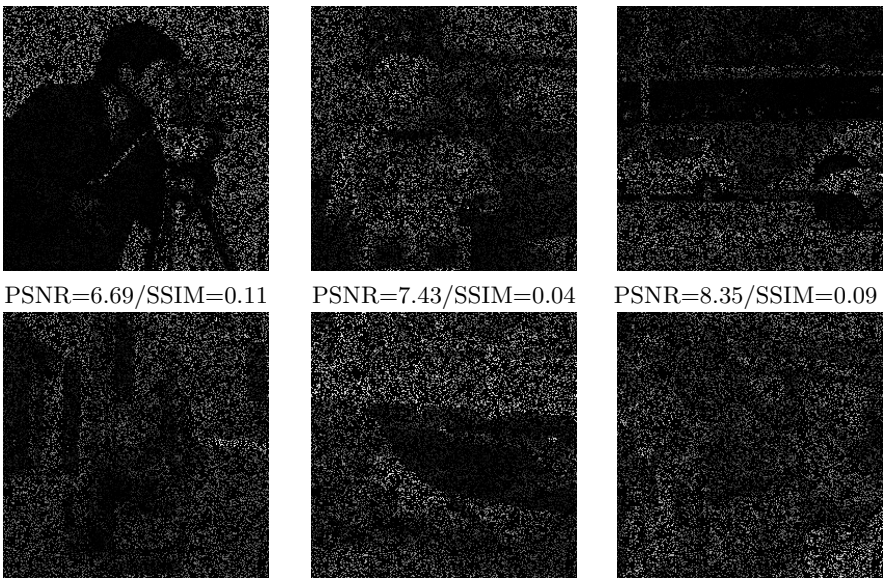


Figure 5.3: Images of Figure 5.1, with 80% missing pixels.

Lipschitz constant of $b_\varepsilon(x) = \nabla \log p_\varepsilon(x|y)$; namely, we require $\delta < (1/3)\,\mathrm{Lip}(b_\varepsilon)^{-1}$ where

$$\mathrm{Lip}(b_\varepsilon) = \begin{cases} \alpha \mathtt{L}/\varepsilon + 1/\lambda & \text{for the inpainting problem} \\ \alpha \mathtt{L}/\varepsilon + \mathtt{L}_y + 1/\lambda & \text{otherwise} \end{cases}$$

where $\mathtt{L}$ and $\mathtt{L}_y$ are respectively the Lipschitz constant of the denoiser residual $(D_\varepsilon - \mathrm{Id})$ and the Lipschitz constant of the log-likelihood gradient. In our experiments, $\mathtt{L} = 1$ and $\mathtt{L}_y = \|\mathrm{A}^\top \mathrm{A}\|/\sigma^2$, so we choose δ just below the upper bound $\delta_{th} = 1/3(\mathrm{Lip}(b_\varepsilon))^{-1}$ where $\mathrm{A}^\top$ is the adjoint of A. For PPnP-ULA, we set $\delta < (\mathtt{L}/\varepsilon + \mathtt{L}_y)^{-1}$ (resp. $\delta < (\mathtt{L}/\varepsilon)^{-1}$ for interpolation) to prevent excessive bias.

PARAMETER λ The parameter λ controls the tail behaviour of the target density. As previously explained, it must be set so that the tails of the target density decay sufficiently fast to ensure convergence at a geometric rate, a key property for guaranteeing that the Monte Carlo estimates computed from the chain are consistent and subject to a Central Limit Theorem with the standard $\mathcal{O}(\sqrt{k})$ rate. More precisely, we require $\lambda \in (0, 1/2(\mathtt{L}/\varepsilon + 2\mathtt{L}_y))$. Within this admissible range, if λ is too small this limits the maximal δ and leads to a slow Markov chain. For this reason, we recommend setting λ as large as possible below $(2\mathtt{L}/\varepsilon + 4\mathtt{L}_y)^{-1}$.

OTHER PARAMETERS The compact set C is defined as $\mathsf{C} = [-1, 2]^d$, even if in practice no samples where generated outside of C in all our experiments, which suggests that the tail decay conditions hold without explicitly enforcing them. In all our experiments, we set the noise level of the denoiser D_ε to $\varepsilon = (5/255)^2$. The initialization X_0 can be set to a random vector. In our experiments (where $m = d$), we chose $X_0 = y$ in order to reduce the number of burn-in iterations. For $m \neq d$ we could use $X_0 = \mathrm{A}^\top y$ instead. Concerning

the regularization parameter α, by default we set $\alpha = 1$, but in some cases it is possible to marginally improve the results by fine tuning it. All algorithms are implemented using Python and the PyTorch library, and run on an Intel Xeon CPU E5-2609 server with a Nvidia Titan XP graphic card or on Idris' Jean-Zay servers featuring Intel Cascade Lake 6248 CPUs with a single Nvidia Tesla V100 SXM2 GPU. Reported running times correspond to the Xeon + Titan XP configuration.

5.4.2 Convergence analysis of PnP-ULA in non-blind image deblurring and inpainting

When using a sampling algorithm such as PnP-ULA on a new problem, it is essential to check that the state space is correctly explored. In order to provide a thorough convergence study, we first run the algorithm for 25×10^6 iterations. We use a burn-in period of 2.5×10^6 iterations, and consider only the samples computed after this burn-in period to study the Markov chain in close-to-stationary regime. In Section 5.4.3, we will see that much less iterations are required if the goal is only to compute point estimators with PnP-ULA. For simplicity, the algorithm is always initialized with the observation y in our experiments with PnP-ULA (for interpolation, this means that unknown pixels are initialized to the value 0).

There is no fully comprehensive way to empirically characterise the convergence properties of a high-dimensional Markov chain, as different statistics computed from the same chain align differently with the eigenfunctions of the Markov kernel and hence exhibit different convergence speeds. In problems of small dimension, we would calculate and analyse the d-dimensional multivariate autocorrelation function (ACF) of the Markov chain, but this is not feasible in imaging problems. In problems of moderate dimension, one could characterise the range of convergence speeds by first estimating the posterior covariance matrix (which, for 256×256 images, would be a $256^2 \times 256^2$ matrix) and then performing a principal component analysis on this matrix to identify the directions with smallest and largest uncertainty, as these would provide a good indication of the subspaces where the chain converges the fastest and the slowest. However, computing the posterior covariance matrix is also not possible in imaging problems because of the dimensionality involved. Here we focus on approximations of the posterior covariance which make sense for the particular inverse problem we study. More precisely, we use the diagonalization basis of the inverse operator, *i.e.* the Fourier basis for the deblurring experiments, and the basis formed by the unknown pixels for the inpainting experiments. Under the assumption that the posterior covariance is mostly determined by the likelihood, this strategy allows broadly identifying the linear statistics that converge fastest and slowest, without requiring the estimation and manipulation of prohibitively large matrices.

INTERPOLATION We first focus on the interpolation problem. Figure 5.4 shows a map of the pixel-wise marginal standard deviations, for all images. We observe that pixels in homogeneous regions have low uncertainty, while pixels on textured regions, edges, or complex structures (a reflection on the window shutter in the Alley image for instance) are the most uncertain.

For the same experiments, Figure 5.5 shows the Euclidean distance between the final MMSE estimate (computed using all samples) and the samples of the chain, every 2500 samples (after the burn-in period, and hence in what is considered to be a close-to-stationary regime). Fluctuations around the posterior mean and the absence of temporal structure in the plots of `Alley` or `Goldhill` are a first indication that the chain explores the solution space with ease. However, in some other cases such as the `Simpson` image, we observe metastability, where the chain stays in a region of the space for millions of iterations and then

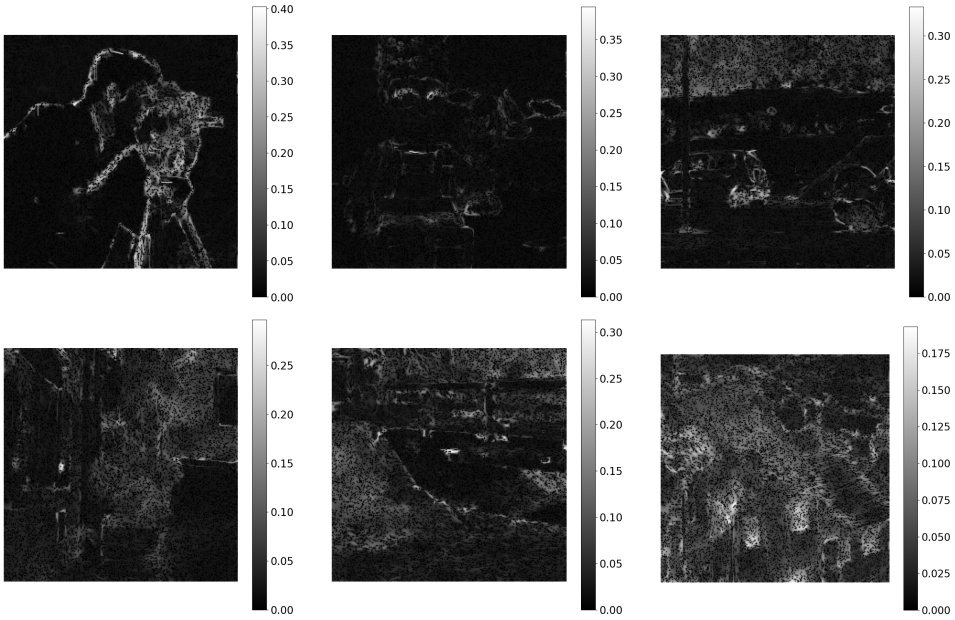


Figure 5.4: Marginal posterior standard deviation of the unobserved pixels for the interpolation problem. Uncertainty is located around edges and in textured areas.

jumps to a different region, again for millions of iterations. This is one of the drawbacks of operating with a posterior distribution that is not log-concave and that may exhibit several modes.

Lastly, Figure 5.6 displays the sample ACFs of the fastest and slowest converging statistics associated with the interpolation experiments (as estimated by identifying, for each image, the unknown pixels with lowest and highest uncertainty). These ACF plots measure how fast samples become uncorrelated. A fast decay of the ACF is associated with good Markov chain mixing, which in turn implies accurate Monte Carlo estimates. On the contrary, a slow decay of the ACF indicates that the Markov chain is moving slowly, which leads to Monte Carlo estimates with high variance. As mentioned previously, because computing and visualising a multivariate ACF is difficult, here we show the ACF of the chain along the slowest and the fastest directions in the spatial domain (for completeness, we also show the ACF for a pixel with median uncertainty). We see that independence is reached very fast in the subspaces of low or median uncertainty, and is much slower for the few very uncertain pixels.

DEBLURRING We now focus on the non-blind image deblurring experiments, where, as explained previously, we perform our convergence analysis by using statistics associated with the Fourier domain. Figure 5.7 depicts the marginal standard deviation of the Fourier coefficients (in absolute value), for all images. For the three images `Cameraman`, `Simpsons` and `Traffic`, all the standard deviations have a similar range of values, and the largest values are observed around frequencies in the kernel of the blur filter (shown on the right of the same figure) and for high frequencies. Conversely, for the three images `Alley`, `Bridge` and `Goldhill`, very high uncertainty is observed in the vicinity of four specific frequencies. This suggests that the denoiser used is struggling to regularise these specific frequencies, and

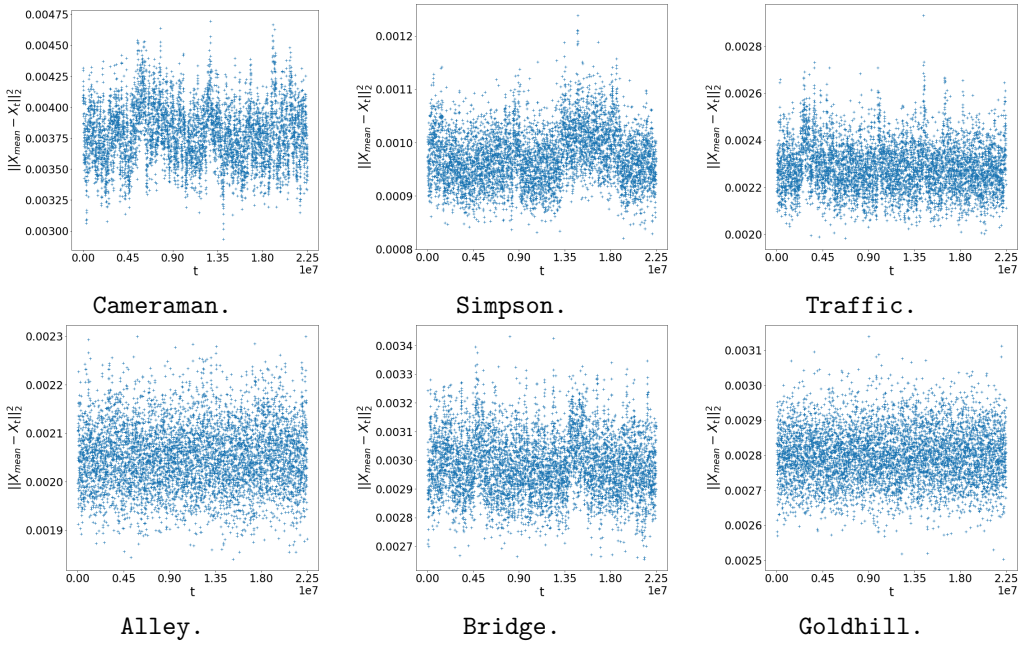


Figure 5.5: Evolution of the L_2 distance between the final MMSE estimate and the samples generated by PnP-ULA for the interpolation problem after the burn-in phase. Samples randomly oscillate around the MMSE. It means that they are uncorrelated. For the images `Cameraman`, `Simpson` or `Bridge`, we note a change of range for the L_2 distance. It could be interpreted as a mode switching as our posterior is likely not log-concave.

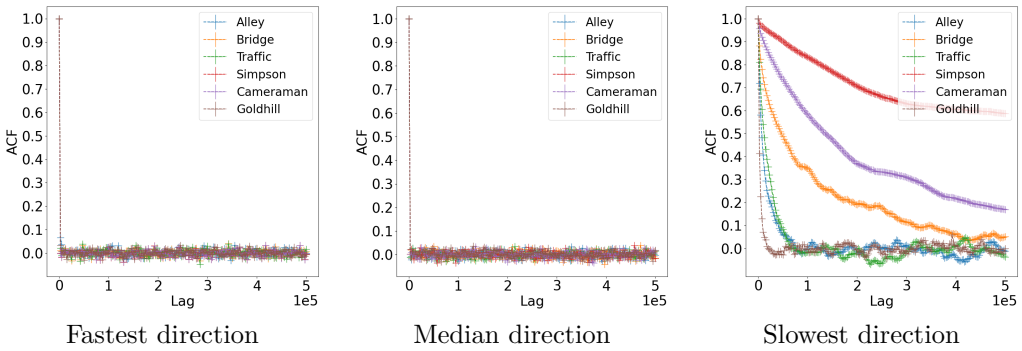


Figure 5.6: ACF for the interpolation problem. The ACF are shown for lags up to 5e5 for all images in the pixel domain. After 5e5 iterations, sample pixels are nearly uncorrelated in all spatial directions for the images `Traffic`, `Alley`, `Bridge` and `Goldhill`. For the images `Cameraman` and `Simpson`, in the slowest direction, samples need more iterations to become uncorrelated.

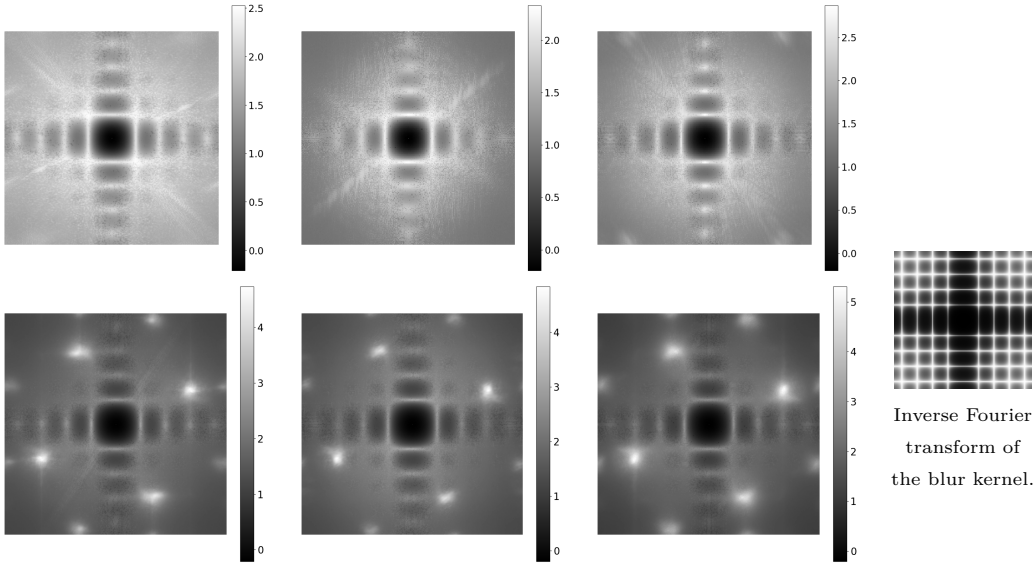


Figure 5.7: Log-standard deviation maps in the Fourier domain for the Markov chains defined by PnP-ULA for the deblurring problem. First line: images `Cameraman`, `Simpson`, `Traffic`. Second line: images `Alley`, `Bridge` and `Goldhill`. For the first three images, we clearly see that uncertainty is observed on frequencies that are near the kernel of the blur filter (shown on the right), and is also higher around high frequencies (*i.e.* around edges and textured areas in images). For the last three images, very high uncertainty is observed around some specific frequencies. In the direction of these frequencies, the Markov chain is moving very slowly and the mixing time of the chain is particularly slow, as shown on Figure 5.9.

consequently the posterior distribution is very spread along these directions and difficult to explore by Markov chain sampling as a result. Interestingly, this phenomenon is only observed in the images that are rich in texture content.

Moreover, Figure 5.8 depicts the Euclidean distance between the MMSE estimator computed from entire chain (*i.e.* all samples) and each sample (we show one point every 2500 samples). We notice that many of the images exhibit some degree of meta-stability or slow convergence because of the presence of directions in the solution space with very high uncertainty. Again, this is consistent with our convergence theory, which identifies posterior multimodality and anisotropy as key challenges that future work should seek to overcome.

Lastly, we show on Figure 5.9 the sample ACFs for the slowest and the fastest directions in the Fourier domain[§]. Again, in all experiments, independence is achieved quickly in the fastest direction. The behaviour of the slowest direction for the three images `Alley`, `Bridge` and `Goldhill` suggests that the Markov chain is close to the stability limit and exhibits highly oscillatory behaviour as well as poor mixing.

5.4.3 Point estimation for non-blind image deblurring and interpolation

We are now ready to study the quality of the MMSE estimators delivered by PnP-ULA and PPnP-ULA and report comparisons with MAP estimation by PnP-SGD introduced in Chapter 4.

[§]The slowest direction corresponds to the Fourier coefficient with the highest (real or imaginary) variance.

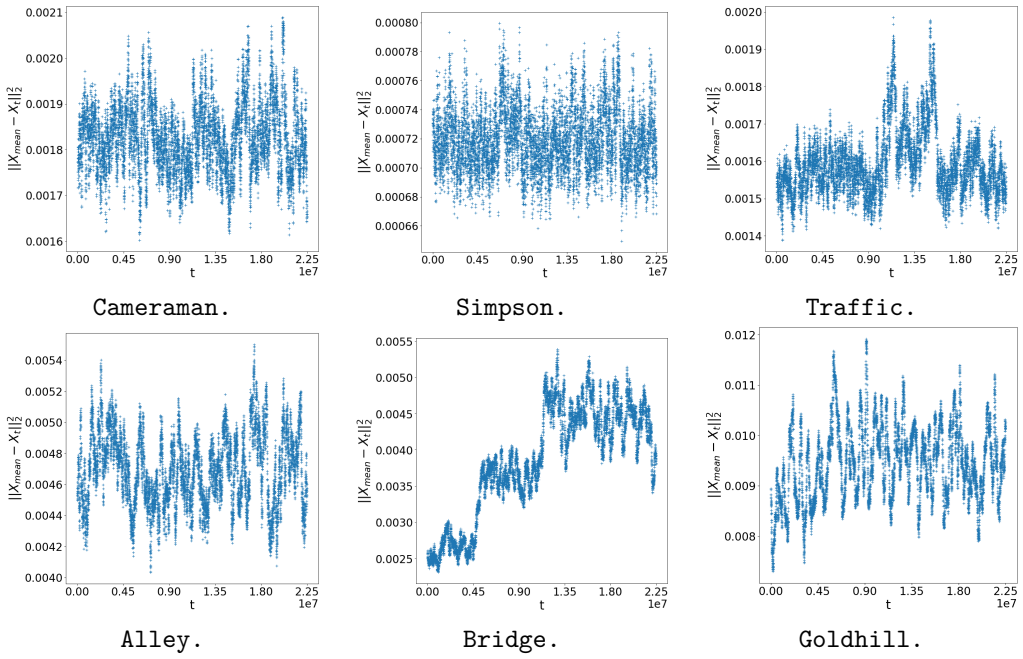


Figure 5.8: Evolution of the L_2 distance between the final MMSE estimate and the samples generated by PnP-ULA for the deblurring problem after the burn-in phase. For images as `Cameraman` or `Simpson`, samples randomly oscillate around the MMSE. On the contrary, for images as `Bridge` or `Goldhill`, the plot is structured, meaning that samples are still correlated.

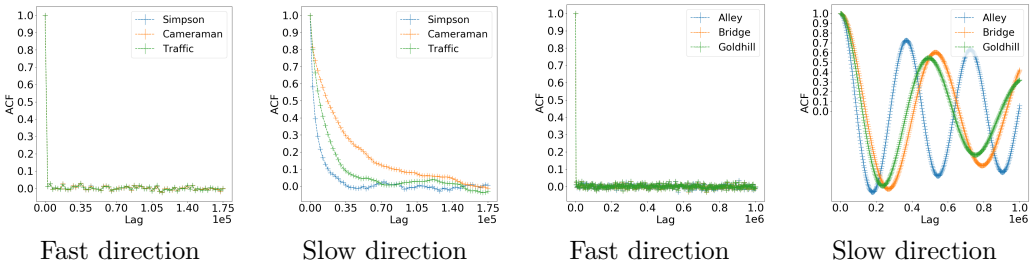


Figure 5.9: ACF for the deblurring problem. The ACF are shown for lags up to 1.75e5 for the three images `Cameraman`, `Simpson` and `Traffic` (see the two plots to the left) and independence seems to be achieved in all directions. For the three other images, independence is not achieved in the slowest direction (corresponding to the most uncertain frequency of the samples in the Fourier domain) even after 1e6 iterations.

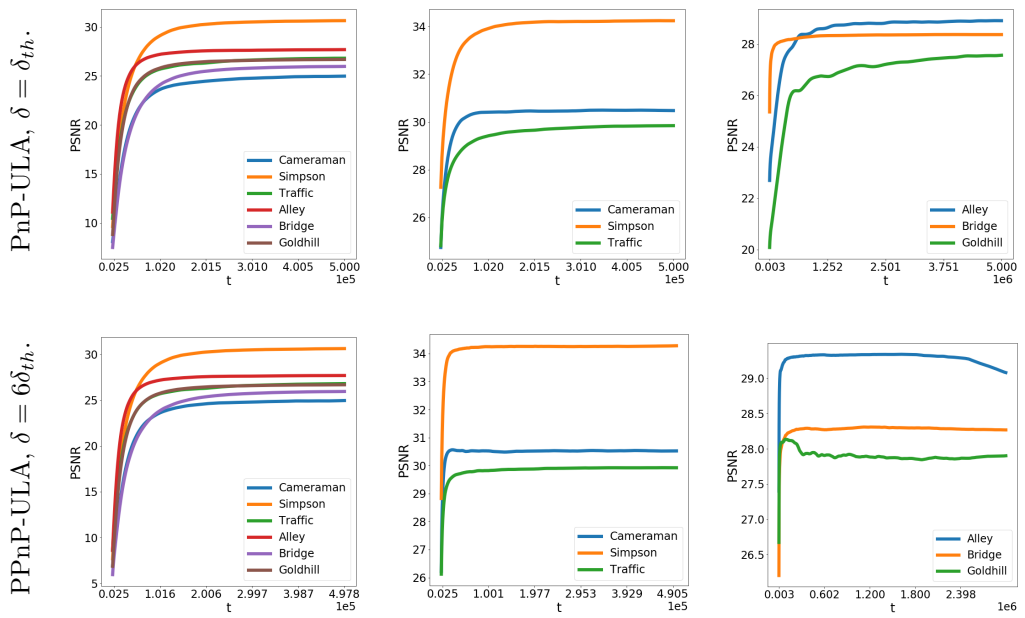


Figure 5.10: Left: PSNR evolution of the estimated MMSE for the interpolation problem. After $5e5$ iterations, the convergence of the first order moment of the posterior distribution seems to be achieved for all images. Middle and right: PSNR evolution of the estimated MMSE for the deblurring problem. The convergence for the posterior mean can be fast for simple images such as `Cameraman`, `Simpson`, and `Traffic` (for these images the PSNR evolution is shown for the first 5e5 iterations). Increasing δ increases the convergence speed for these images by a factor close to 2. For more complex images, such as `Alley` or `Goldhill`, the convergence is much slower and is still not achieved after 3e6 iterations with PPnP-ULA for $\delta = 6\delta_{th}$.

QUANTITATIVE RESULTS Figure 5.10 illustrates the evolution of the PSNR of the mean of the Markov chain (the Monte Carlo estimate of the MMSE solution), as a function of the number of iterations, for the six images of Figure 5.1. These plots have been computed by using a step-size $\delta = \delta_{th}$ that is just below the stability limit and a 1-in-2500 thinning. We observe that the PSNR between the MMSE solution as computed by the Markov chain and the truth stabilises in approximately 10^5 iterations in the experiments where the chain exhibits fast convergence, whereas over 10^6 are required in experiments that suffer from slow convergence (e.g., deblurring of `Alley`, `Bridge` and `Goldhill`). Moreover, we observe that using PPnP-ULA with a larger step-size can noticeably reduce the number of iterations required to obtain a stable estimate of the posterior mean, particularly in the image deblurring experiments.

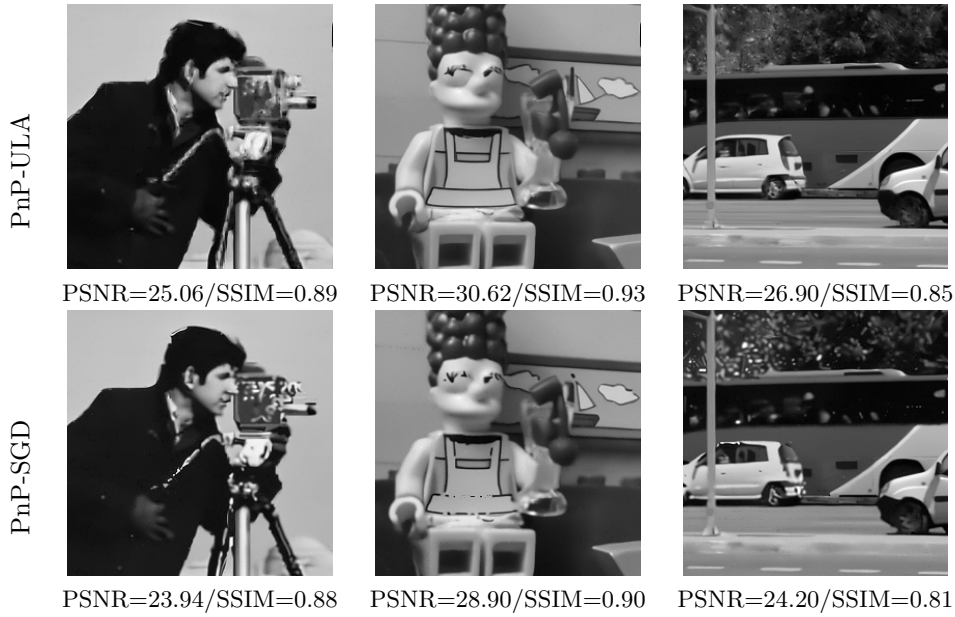


Figure 5.11: Results comparison for the interpolation task of the images presented in Figure 5.3 using PnP-ULA (first row) and PnP-SGD initialized with a TVL2 restoration (second row).

VISUAL RESULTS Figures 5.11 to 5.14 show the MMSE estimate computed by PnP-ULA on the whole chain including the burn-in for the 6 images, for the interpolation and deblurring experiments. We also provide the MAP estimation results computed by using PnP-SGD (see Chapter 4), which targets the same posterior distributions. We report the *Peak Signal-To Noise Ratio* (PSNR) and the *Structural Similarity Index* (SSIM) (Wang and Bovik, 2009; Wang et al., 2004) for all these experiments.

For the interpolation experiments, PnP-SGD struggles to converge when initialized with the observed image (see Section 4.3). For this reason, we warm start PnP-SGD by using an estimate of x obtained by minimizing the Total Variation pseudo-norm under the constraint of the known pixels. For simplicity, PnP-ULA is initialized with the observation y. We observe in Figure 5.11 and Figure 5.12 that the results obtained by computing the MMSE Bayesian estimator with PnP-ULA are visually and quantitatively superior to the ones delivered by MAP estimation with PnP-SGD. In particular, the sampling approach seems to better recover the continuity of fine structures and lines in the different images.

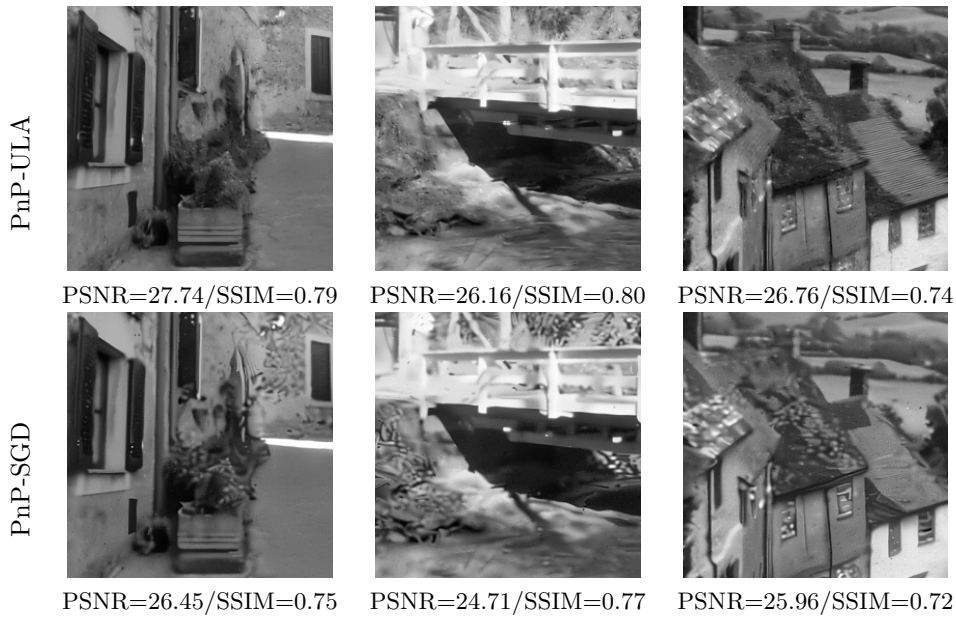


Figure 5.12: Results comparison for the interpolation task of the images presented in Figure 5.3 using PnP-ULA (first row) and PnP-SGD initialized with a TVL2 restoration (second row).

For the deblurring experiments, the results of PnP-SGD are provided by using a regularisation parameter $\alpha = 0.3$ (which was shown to yield optimal results on this set of images in Section 4.3) and for $\alpha = 1$, which recovers the model used by PnP-ULA. Observe that for the three first images (shown on Figure 5.13), the MMSE result is much sharper than the best MAP result, and the PSNR / SSIM results also show a clear advantage for the MMSE. For the other three images (results are shown on Figure 5.14), the quality of the MMSE solutions delivered is slightly deteriorated by the slow convergence of the Markov chain and the poor regularisation of some specific frequencies, which leads to a common visual artefact (a rotated rectangular pattern). Using a different denoiser more suitable for handling textures, or combining a learnt denoiser with an analytic regularisation term, might correct this behaviour and will be the topic of future work.

A partial conclusion from this set of comparisons is that the sampling approach of PnP-ULA, when it samples the space correctly, seems to provide much better results than the MAP estimator for the same posterior. Of course, this increase in quality comes at the cost of a much higher computation time.

5.4.4 Deblurring and interpolation: uncertainty visualisation study

One of the benefits of sampling from the posterior distribution with PnP-ULA is that we can probe the uncertainty in the delivered solutions. In the following, we present an uncertainty visualisation analysis that is useful for displaying the uncertainty related to image structures of different sizes and located in different regions of the image (see (Cai et al., 2018) for more details). The analysis proceeds as follows. First, Figure 5.4 and Figure 5.15 show the marginal posterior standard deviation associated with each image pixel, as computed by PnP-ULA over all samples, for the interpolation and deblurring problems. As could be expected, we observe for both problems that highly uncertain pixels are concentrated around

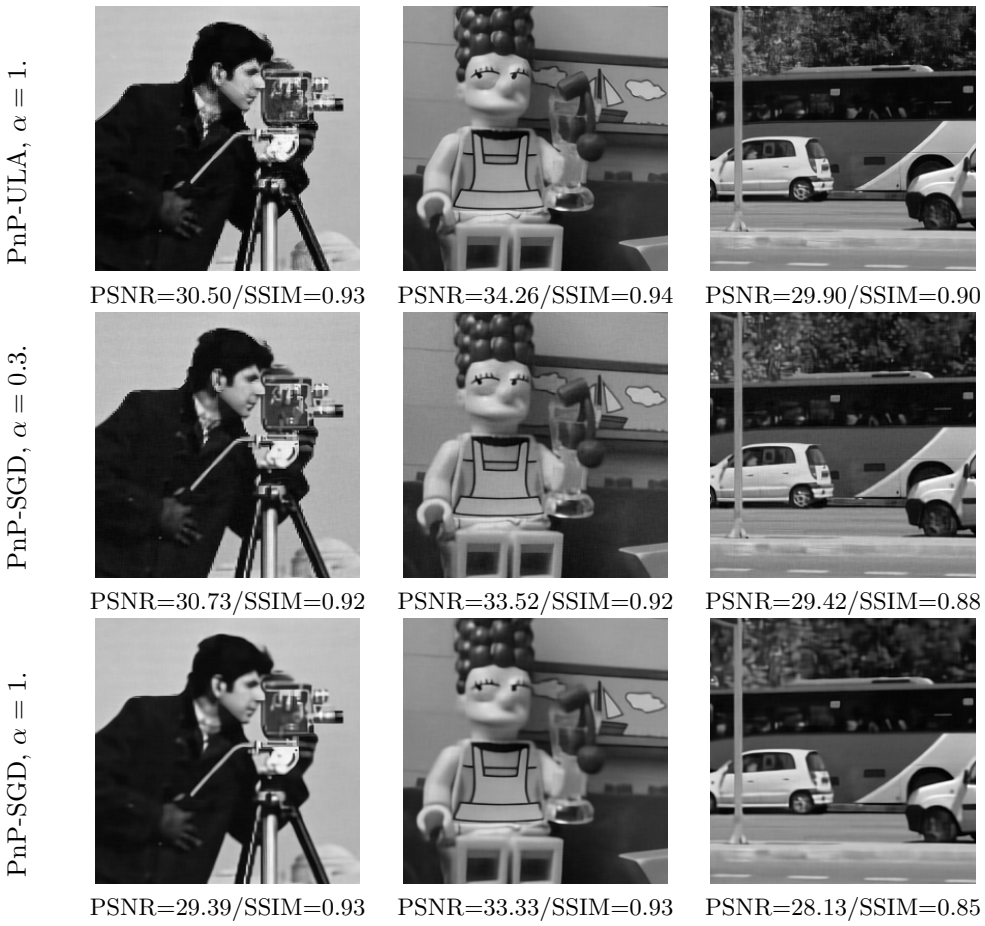


Figure 5.13: Results comparison for the deblurring task of the images presented in Figure 5.2 using PnP-ULA with $\alpha = 1$ (first row), PnP-SGD with $\alpha = 0.3$ (second row) and $\alpha = 1$ (third row). PnP-ULA was initialized with the observation y (see Figure 5.2) whereas PnP-SGD was initialised with a TVL2 restoration.

the edges of the reconstructed images, but also on textured areas. The dynamic range of the pixel standard deviations is larger for the interpolation problem than for deblurring, which suggests that the problem has a higher level of intrinsic uncertainty.

Figure 5.16 shows the evolution of the RMSE between the standard deviation computed along the samples and its asymptotic value, respectively for the interpolation and deblurring problems. Estimating these standard deviation maps necessitates to run the chain longer than to estimate the MMSE, as could be expected for second order statistical moment.

Following on from this, to explore the uncertainty for structures that are larger than one pixel, Figure 5.17 and Figure 5.18 report the marginal standard deviation associated with higher scales. More precisely, for different values of the scale i, we downsample the stored samples by a factor $2i$ before computing the standard deviation. This downsampling step permits quantifying the uncertainty of larger or lower-frequency structures, such as the bottom of the glass in `Simpson` for the deblurring experiment. At each scale, we see that the uncertainty of the estimate is much more localized for the interpolation problem (resulting in higher uncertainty values in some specific regions) and more spread out for deblurring,

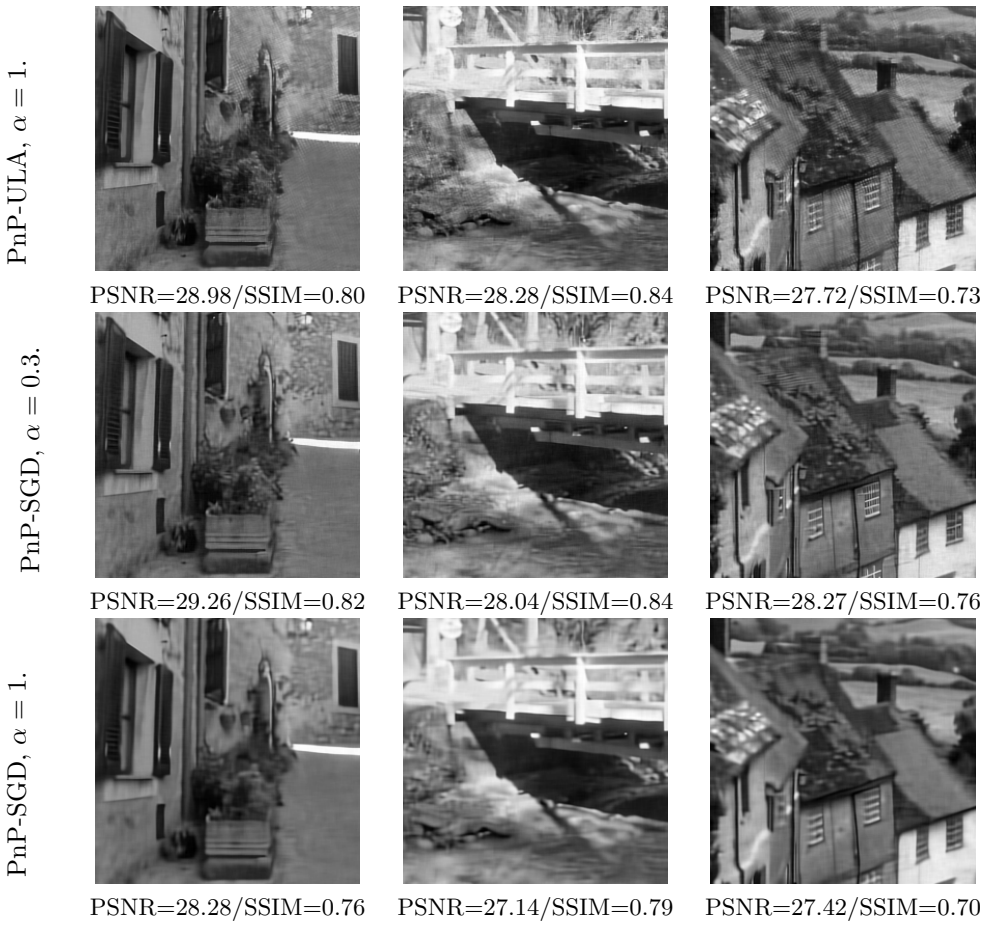


Figure 5.14: Results comparison for the deblurring task of the images presented in Figure 5.2 using PnP-ULA with $\alpha = 1$ (first row), PnP-SGD with $\alpha = 0.3$ (second row) and $\alpha = 1$ (third row). PnP-ULA was initialized with the observation y (see Figure 5.2) whereas PnP-SGD was initialised with a TVL2 restoration.

certainly because of the different nature of the degradations involves.

5.5 Accelerated sampling using stochastic orthogonal Runge-Kutta-Chebyshev methods with data-driven priors

If PnP-ULA shows interesting results for point estimation, we saw previously that it can have some difficulties to correctly explore the state space. In addition, samples generated by the PnP-ULA Markov chain can be highly correlated, which should be avoided if we want accurate Monte-Carlo estimates Figure 5.9.

These problems can be partially explained by a too coarse discretization scheme of the Langevin SDE. If the Euler-Maruyama scheme is straightforward to apply, it does not take into account the geometry of the posterior distribution. Indeed, the discretization step-size δ is the same in all the directions of the space and is only determined by the Lipschitz constant of the posterior score. If the posterior distribution is very anisotropic, then all the

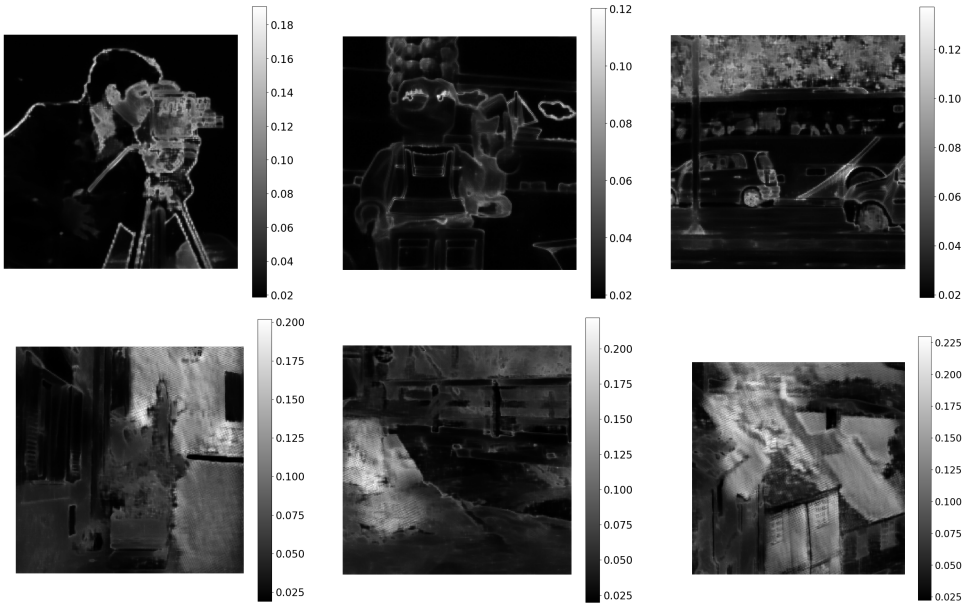


Figure 5.15: Marginal posterior standard deviation for the deblurring problem. On simple images such as Simpson (see fig. 5.1), most of the uncertainty is located around the edges. For the images `Alley`, `Bridge` and `Goldhill`, associated with a highly correlated Markov chain in some directions, some areas are very uncertain. They correspond to the zones where the rotated rectangular pattern appears in the MMSE estimate.

directions of the space are not explored with the same efficiency.

(Pereyra et al., 2020) propose to apply an orthogonal Runge-Kutta-Chebyshev based stochastic approximation presented in (Abdulle et al., 2018). This method allows us to take larger step-size δ, which leads to better exploring faculty. It requires s posterior score evaluations at specific points determined by Chebyshev polynomial extrapolation where s belongs to $\{1, ..., 15\}$. We recall that Chebyshev polynoms of first order are determined as follows:

$$\forall x \in \mathbb{R},\ T_k(x) = \begin{cases} 1 & \text{if } k = 0 \\ X & \text{if } k = 1 \\ 2xT_{k-1}(x) - T_{k-2}(x) & \text{otherwise.} \end{cases}$$

The Plug & Play SKROCK algorithm is detailed in Algorithm 7.

The more we evaluate posterior scores, the larger we can set the discretization step-size. Indeed, we have:

$$\delta_{SKROCK,s} = l_s \delta_{PnP-ULA},\ \text{with } l_s = (s - 0.5)^2(2 - 4/3\eta) \text{ and } \eta = 0.05.$$

Table 5.1 compares the different discretization step-sizes useable when applying SKROCK. Applying SKROCK allows us to take way greater step-sizes. However if increasing the number of posterior score evaluations s stabilizes the scheme, what guarantees a faster state space exploration, it also increases the asymptotic bias. In the following, we will take $\delta = 0.9\delta_{SKROCK,s}$ and let run the the algorithm during N/s iterations with $N = 1e7$ in order to guarantee the same number of forward pass into the neural network denoiser. It is less iterations than with PnP-ULA but it corresponds to a greater absolute time where the absolute time is defined as follows $T = \delta \times t$, with t the current iteration.

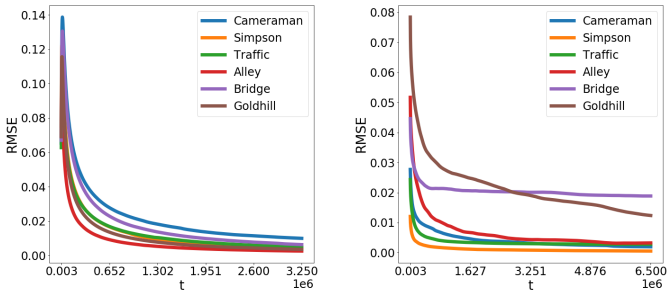


Figure 5.16: Evolution of the Root Mean Squared Error (RMSE) between the final standard deviation and the estimated current standard deviation for the interpolation and deblurring problems.

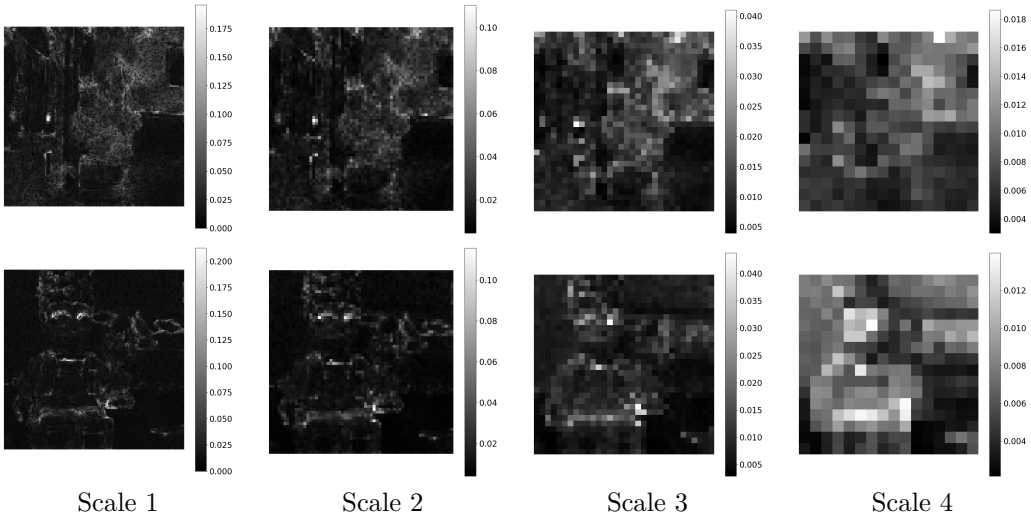


Figure 5.17: Marginal posterior standard deviation of the `Alley` and `Simpson` images for the interpolation problem at different scales. The scale i corresponds to a downsampling by a factor $2i$ of the original sample size.

$\delta_{PnP-ULA}$	$\delta_{SKROCK,s=10}$	$\delta_{SKROCK,s=15}$
$4.929e-6$	$7.673e-4$	$1.796e-3$

Table 5.1: Largest discretization step-sizes useable for PnP-ULA (without enforcing the strong convexity in the tails), SKROCK with $s = 10$ and SKROCK with $s = 15$. SKROCK allows us to take way larger step-size. It makes the the state-space exploration faster and it should guarantee a faster sample decorrelation.

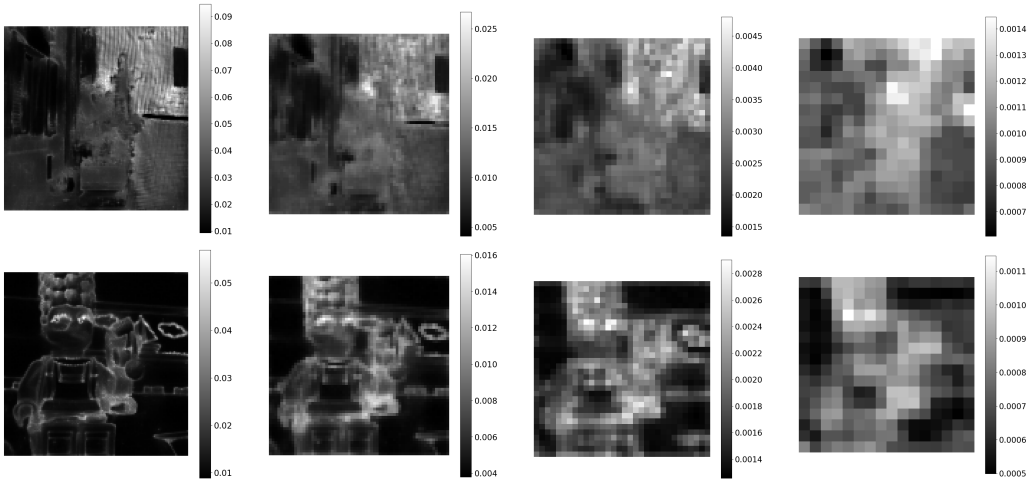


Figure 5.18: Marginal posterior standard deviation of the images `Alley` and `Simpson` for the deblurring problem at different scales. The scale i corresponds to a downsampling by a factor $2i$ of the original sample size.

Algorithm 7 SK-ROCK

Require: $n \in \mathbb{N}$, $y \in \mathbb{R}^m$, $\varepsilon, \alpha, \delta > 0, s \in \{3, ..., 15\}, \eta = 0.05$
  Compute $l_s = (s - 0.5)^2(2 - 4/3\eta)$
Ensure: $\delta < (l_s/3)(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)^{-1}$
  Compute $\omega_0 = 1 + \eta/s^2$, $\omega_1 = T_s(\omega_0)/T_s^{'}(\omega_0)$, $\mu_1 = \omega_1/\omega_0$, $\nu_1 = s\omega_1/2$, $k_1 = s\omega_1/\omega_0$
  Initialization: Set $X_0 \in \mathbb{R}^d$ and $k = 0$.
  for $k = 0 : N$ **do**
    $Z_{k+1} \sim \mathcal{N}(0, \mathrm{Id})$
    $K_0 = X_k$
    $\tilde{X_k} = X_k + \mu_1\sqrt{2\delta}Z_{k+1}$
    $K_1 = X_k + \mu_1\delta[\nabla \log(p(y|\tilde{X_k})) + (\alpha/\varepsilon)(D_\varepsilon(\tilde{X_k}) - \tilde{X_k})] + k_1\sqrt{2\delta}Z_{k+1}$
    for $j = 2 : s$ **do**
      Compute $\mu_j = 2\omega_1 T_{j-1}(\omega_0)/T_j(\omega_0)$, $\mu_j = 2\omega_0 T_{j-1}(\omega_0)/T_j(\omega_0)$, $k_j = 1 - \nu_j$
      $K_j = \mu_j\delta\nabla \log(p(y|K_{j_1}) + (\alpha\delta/\varepsilon)(D_\varepsilon(K_{j-1}) - K_{j-1}) + \mu_j K_{j-1} + k_j K_{j-2}$
    end for
    $X_{i+1} = K_s$
  end for
  return $\{X_k : k \in \{0, \ldots, N + 1\}\}$

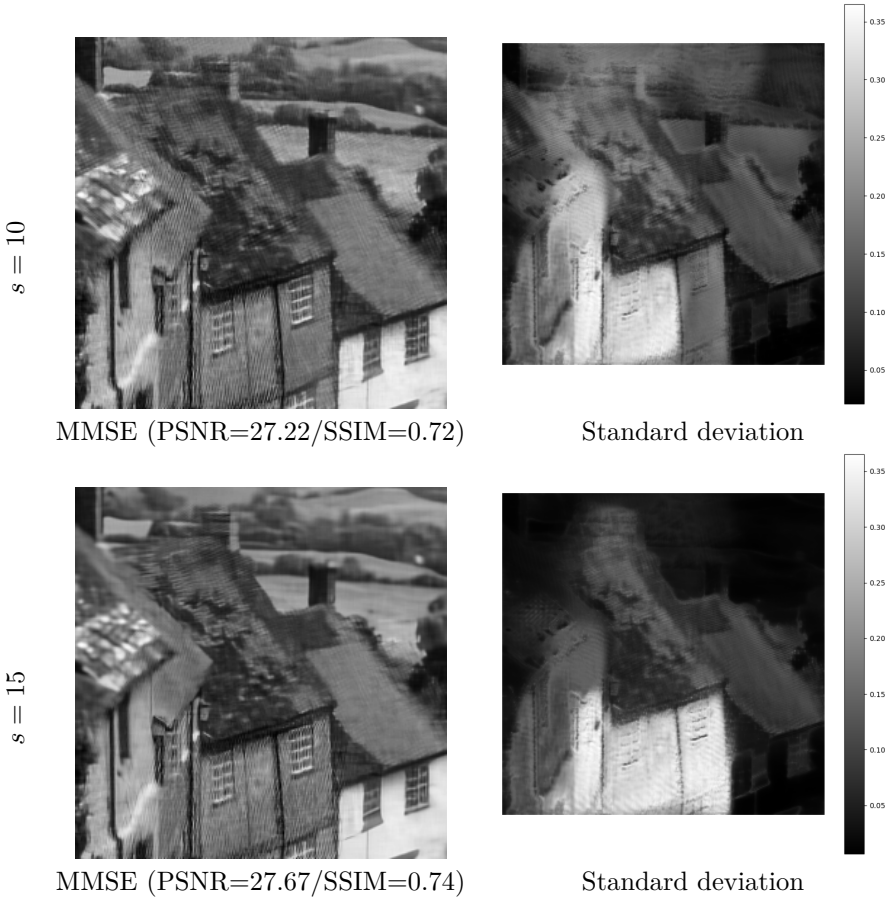


Figure 5.19: Results obtained with SKROCK for $s = 10$ and $s = 15$ gradient evaluations on a deblurring inverse problem with A a 9×9 bloc filter and with an additive Gaussian white noise with standard deviation $\sigma = 1/255$. SKROCK achieves similar results to PnP-ULA in term of MMSE and standard deviation point estimations. In order to draw a fair comparison, we let run the algorithm during n/s iterations with $n = 1e7$.

In the following, we tackle the non-blind deblurring inverse problem presented in Sections 4.3 and 5.4 on `Goldhill`. We are particularly interested in this image as PnP-ULA struggles to properly restore this picture and does not generate an ergodic Markov chain.

The point estimation results obtained with SKROCK are very similar to the ones computed with PnP-ULA as we can see on Figure 5.19. A grid pattern, ruining the visual impression, is still present for the MMSEs. The marginal posterior standard deviations estimated with SKROCK also suffer from high uncertainty on piecewise constant areas. It is due to this grid pattern that appears around high-frequency structures and which spreads out. We point out that the uncertainty magnitude is higher when using SKROCK instead of PnP-ULA. The state-space exploration is actually quicker. Then, the higher magnitude could be explained by the exploration of an area of the state-space unseen by PnP-ULA.

Figure 5.20 shows the ACF plots of the SKROCK Markov chains with $s = 10$ and 15 computed in the Fourier domain as for PnP-ULA. It is interesting to note that the meta-stable behaviour of the Markov chain observed with PnP-ULA is attenuated. Although, we do not have any convergence guarantees as for PnP-ULA, it is a positive side-effect that should urge

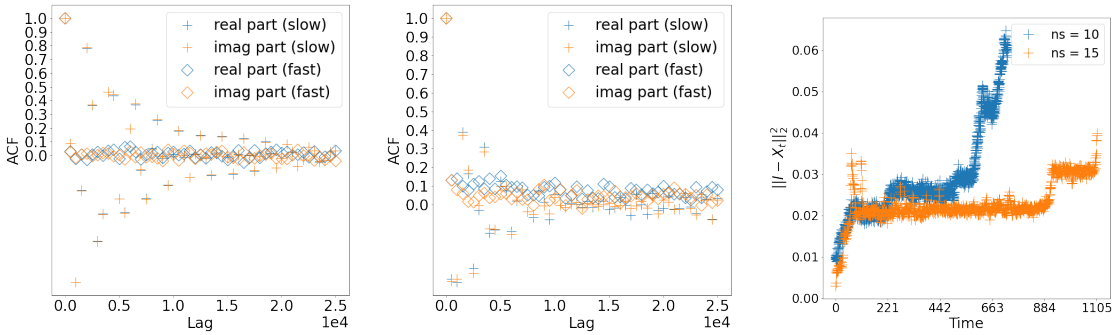


Figure 5.20: Ergodicity test of the Markov chain generated by SKROCK for the deblurring inverse problem. The 2 plots on the left correspond to ACF computed in the Fourier domain with $s = 10$ and $s = 15$. A similar meta-stability behaviour is observed as with PnP-ULA. However, it is less pronounced and the number of posterior score evaluation seems to decrease this phenomenon. The plot on the right corresponds to the evolution of the Euclidean distance between the samples generated by the SKROCK Markov chain and the original image over time $\delta \times t$. None of these Markov chains is ergodic.

us to apply this algorithm. In addition as SKROCK covers the state-space more rapidly than PnP-ULA, it can also highlight more quickly undesirable properties of the Markov chain. For instance, Figure 5.20 shows the evolution of the Euclidean distance between the original image and the samples generated by SKROCK. After $1e7$ neural network applications, the Markov chain is still not ergodic. It highlights the difficulties encountered with the prior plugged into our scheme.

5.6 Conclusion

This chapter presented theory, methods, and computation algorithms for performing Bayesian inference with *Plug & Play* priors. This mathematical and computational framework is rooted in the Bayesian *M-complete* paradigm and adopts the view that *Plug & Play* models approximate a regularised oracle model. We established clear conditions ensuring that the involved models and quantities of interest are well defined and well posed. Following on from this, we studied two Bayesian computation algorithms related to biased approximations of a Langevin diffusion process, for which we provide detailed convergence guarantees under easily verifiable and realistic conditions. For example, our theory does not require the denoising algorithms representing the prior to be gradient or proximal operators. We also studied the estimation error involved in using these algorithms and models instead of the oracle model, which is decision-theoretically optimal but intractable. To the best of our knowledge, this is the first Bayesian *Plug & Play* framework with this level of insight and guarantees on the delivered solutions. We illustrated the proposed framework with two Bayesian image restoration experiments - deblurring and interpolation - where we computed point estimates as well as uncertainty visualisation and quantification analyses and highlighted how the limitations of the chosen denoiser manifest in the resulting Bayesian model and estimates. From a Bayesian computation viewpoint, we also test the efficiency of accelerated algorithm with SKROCK (Pereyra et al., 2020) and prove their use although no convergence proofs are currently available. It is still an active field of research.

In future work, we would like to continue our theoretical and empirical investigation of

Bayesian *Plug & Play* models, methods and algorithms. From a modelling viewpoint, it would be interesting to consider priors that combine a denoiser with an analytic regularisation term. It could allow us to promote samples with some desired properties and enrich the prior. Furthermore, it could compensate for a lack of regularization by the neural network. In the same direction, we would like to consider other neural network based priors. This problematic is at the core of Chapter 6. But we could also consider generative priors as in (Bora et al., 2017) or autoencoder-based priors as in (González et al., 2021). Another field of research concerns the possible generalization to other smoothings and their properties in the context of Bayesian inverse problems. We are also very interested in strategies for training denoisers that automatically verify the conditions required for exponentially fast convergence of the Langevin SDE, for example by using the framework recently proposed in (Pesquet et al., 2020) to learn maximally monotone operators, or the data-driven regularizers described in (Kobler et al., 2020; Mukherjee et al., 2021). The recent works of (Hurault et al., 2022a,b) also offer interesting perspectives from a theoretical point of view. (Hurault et al., 2022a,b) learn a denoiser which results from the gradient of some known non-convex functional. It could be interesting to see how it affects the convergence results. Besides, with regards to experimental work, we intend to study the application of this framework to uncertainty quantification problems, e.g., in the context of medical imaging. The idea would be to train a denoising neural network on a specific dataset related to the inverse problem considered and to apply SKROCK in order to sample from the posterior distribution and perform detailed uncertainty quantification study.

If PnP-ULA and PPnP-ULA allow to sample from the posterior distribution and consequently to estimate any posterior probabilities, it is interesting to check if these probabilities coincide with the empirical probabilities we can compute. The idea behind is to check if the posterior model is accurate and models the true unavailable model. These questions are at the core of the last chapter of this thesis.

6

In-depth study of data-driven priors for sampling

6.1 Introduction

In this chapter, we are interested in the role of the prior p and its associated denoiser D_ε in sampling algorithms. We focus on deep denoisers trained on a dataset in order to automatically learn a-priori information about the data we deal with. Eventually, we wish to investigate if there are denoisers more likely to model the true albeit unknown posterior model

This chapter is motivated by the analysis of the results exposed in Section 5.4 for the deblurring problem, where a meta-stability behaviour was observed for images with high-frequency structures. Some PnP-ULA restorations present artefacts like a grid-pattern. Visualizing the standard deviation of the marginal posterior at each pixel, we note these areas are highly uncertain, meaning that the algorithm is not confident in the proposed structures. The goal of this chapter is to test the efficiency of different denoisers and to infer information about the posterior sampled distribution. To do so, we are going further than simply performing point estimation. A closer look will be given at *posterior credible sets* or *credible regions*. They are the regions of the state space in which most of the posterior distribution mass lies (Robert, 2007). A set C_β is a posterior credible region with confidence $1-\beta$ if $P(x \in C_\beta | y) = 1-\beta$. For every $\beta \in (0,1)$, there exist an infinite number of credible regions associated with the confidence level $1-\beta$. The region C_β^* that has the minimum

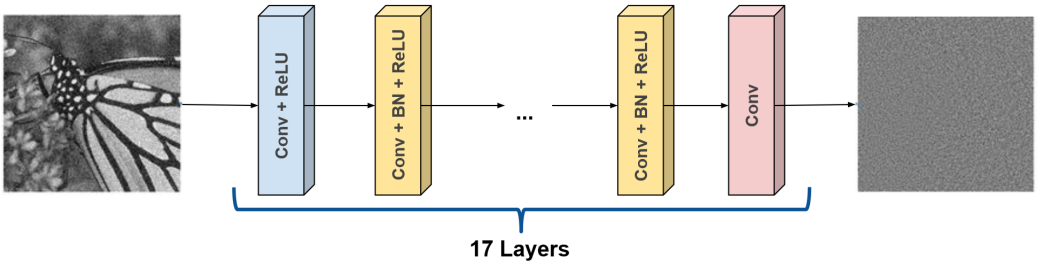


Figure 6.1: Architecture of the SN-DnCNN. Figure taken from (Ryu et al., 2019).

volume is called the *highest posterior density region* (HPD) (Robert, 2007) and is associated with a positive scalar M_β such that

$$C_\beta^* = \{x \in \mathbb{R}^d, p(x|y) \geqslant M_\beta\} \,. \tag{6.1}$$

In the case where the target posterior distribution $p(x|y)$ admits a positive density w.r.t the Lebesgue measure $x \mapsto \exp[-U(x)]$, the HPD is given by

$$C_\beta^* = \{x \in \mathbb{R}^d, U(x) \leqslant \eta_\beta\} \,, \tag{6.2}$$

where $\eta_\beta = \log(M_\beta)$. The quantity U is generally called the potential.

Consequently, the HPD region with confidence $(1-\beta)$ is only characterised by the threshold η_β. Let us point out that when we want to estimate posterior credible sets and HPD regions we have to solve very high-dimensional integrals of the form $\int_{C_\beta} p(x|y)dx$. They will be estimated using Monte-Carlo methods. A first interrogation concerns the presence or not of the original image in the posterior credible sets.

In this chapter, we consider three different denoisers:

- The Spectral Normalized Deep Convolutional Neural Network (SN-DnCNN) (Ryu et al., 2019): It is the denoiser used in the experiments of Chapters 4 and 5. It is a 17-layers convolutional neural network with ReLU activation functions and batch normalization that learns the residual mapping. It is trained with L_2-loss so that the residual mapping is contractive, ie with a Lipschitz constant smaller than 1. To do so, at each forward pass of the neural network during the training, the spectral norm of each layer operator is estimated with the power method and then the layers are normalized by their estimated spectral norm.

  The code of SN-DnCNN is available at `https://github.com/uclaopt/Provable_Plug_and_Play/`.

- The proximal gradient-step denoiser (Prox-GSD) (Hurault et al., 2022b): This denoiser D_ε is trained to act like the gradient-step of an explicit functional g_ε. We have then for all $x \in \mathbb{R}^d$ $D_\varepsilon(x) = x - \nabla g_\varepsilon(x)$. g_ε corresponds to the potential of the approximate prior p_ε introduced in Section 5.2. In (Hurault et al., 2022b), g_ε is defined such that $\forall x \in \mathbb{R}^d$, $g_\varepsilon(x) = \frac{1}{2}\|x - N(x,\varepsilon)\|_2^2$ with $N(.,\varepsilon)$ a neural network. This potential was previously introduced in (Romano et al., 2017; Bigdeli and Zwicker, 2017). Although considered but disregarded in (Romano et al., 2017), it has the advantage to be sufficiently general without requiring too restrictive hypotheses on $N(.,\varepsilon)$ to get convergent PnP-optimization schemes. In (Hurault et al., 2022b), $N(.,\varepsilon)$ is a *light*

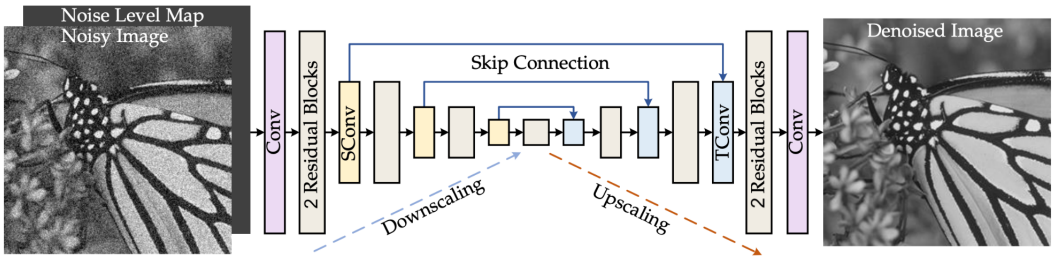


Figure 6.2: Architecture of the light DRUNet. Figure taken from (Hurault et al., 2022a)

Dilated-Residual U-Net (DRUNet) with only two residual blocks as described in Figure 6.2. DRUNet was introduced in (Zhang et al., 2021) and achieves state-of-the-art results in terms of denoising by combining U-Net and Res-Net structures.

In order to be differentiable, it has softplus activation functions. Let us point out that this denoiser, contrary to SN-DnCNN takes the noisy image and the noise level map as inputs.

The training is performed in two steps. The first step consists in training the denoiser with the classical L_2-loss. The second step enforces the non-expansivity of the residual denoiser with the following loss function

$$L(\varepsilon) = \mathbb{E}_{x\sim p, n_\varepsilon \sim \mathcal{N}(0,\varepsilon)}[\|D_\varepsilon(x+n_\varepsilon) - x\|^2 + \mu \max(\|J_{Id-D\varepsilon}(x+n_\varepsilon)\|_S, 1-\nu)] \quad (6.3)$$

where $\|\|_S$ denotes the spectral norm, $\nu = 0.1$ and $\mu = 0.001$.

It is worth mentioning at this point that this denoiser is also the proximal operator of some non-convex potential Φ_ε as shown in (Hurault et al., 2022b).

The main advantage of using this denoiser is that it is the gradient of some known potential g_ε. Consequently, it would allow us to estimate exact HPD regions or to perform automatic model calibration using maximum likelihood maximization as in (Vidal et al., 2020).

The code of Prox-GSD is available at `https://github.com/samuro95/Prox-PnP`.

- The Firmly Non-Expansive (FINE) network (Pesquet et al., 2020):

  This network is inspired by the DnCNN architecture detailed in (Zhang et al., 2017). The batch normalization layers have been removed and the ReLU activation functions have been replaced by LeakyReLUs.

  This FINE network is incorporated within the framework of Maximally Monotone Operators (MMO). We consider a multivalued operator B defined on a Hilbert space H, $B : H \to 2^H$, with 2^H the family of all subsets of H. 2^H is called the *power set of* H.The notation $B : H \to 2^H$ means that H maps every point $x \in H$ to a set $Bx \subset H$. B is characterized by its *graph*

$$\text{gra } B = \{(x,u) \in H \times H | u \in Bx\}.$$

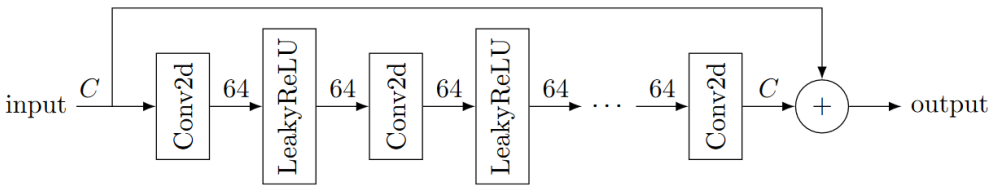


Figure 6.3: Architecture of the FINE network. Here C corresponds to the number of channels. $C = 1$ for grayscale images, $C = 3$ for color images. Figure taken from (Pesquet et al., 2020).

B is maximally monotone if and only if for every $(x, u) \in H \times H$,

$$u \in Bx \Leftrightarrow (\forall y \in H), (\forall v \in By) \ \langle x - y, u - v \rangle \geqslant 0. \tag{6.4}$$

It concretely means that there is no maximal operator $\tilde{B}$ which contains gra B. For example, if B is not maximal, then there exists $(x, u) \notin B$ such that $\tilde{B} = B \cup \{(x, u)\}$ is monotone. Furthermore, a multivalued operator B is fully characterized by its resolvent defined as $J_B = (Id - B)^{-1}$ where the inverse corresponds to the inversion of a graph. The authors of (Pesquet et al., 2020) exploit the fact that an operator B is maximally monotone if and only if there exists a non-expansive operator, ie 1−Lipschitz, Q such that $J_B = \frac{1}{2}(Id + Q)$. In the case where B corresponds to the subdifferential of a certain convex potential g, the resolvent is the proximal operator of g. The authors of (Pesquet et al., 2020) learn a non-expansive operator Q so that the operator $\tilde{J} = \frac{1}{2}((Id + Q))$ corresponds to the resolvent of a maximally monotone operator B^*. Finally, the training loss of this neural network reads

$$L = \mathbb{E}_{x \sim p, n_\varepsilon \sim \mathcal{N}(0,\varepsilon), \rho \sim U_[0,1]}[\|\tilde{J}(y_\varepsilon) - x\|_2^2 + \lambda \max\{\|\nabla Q(z_{\varepsilon,\rho})\|_S^2, 1 - \nu\}] \tag{6.5}$$

with $y_\varepsilon = x + n_\varepsilon$, $z_{\varepsilon,\rho} = \rho x + (1 - \rho)\tilde{J}(x + n_\varepsilon)$, $\lambda = 0.002$ and $\nu = 5e - 2$.

The code of the FINE denoiser is available at `https://github.com/basp-group/PnP-MMO-imaging`.

We consider these three denoisers because they all allow to estimate the Lipschitz constant `L` of the residual operator $D_\varepsilon - \mathrm{Id}$.

We did not perform any training and only considered pretrained neural networks. That is why we will show in the following color images for Prox-GSD, which is only trained on a color base.

If not specified, the compact ensuring the strong convexity in the tails is $\mathsf{C} = [-1, 2]^d$.

In a first part, we draw a comparison of the point estimators computed using the different denoisers with PnP-ULA in Section 6.2. Then, we look at results obtained in terms of potential using an approximated potential when we do not know if the denoiser is associated with a potential in Section 6.3. Doing so, we can better understand the images promoted by the plugged prior. Finally, in Section 6.4 we perform a coverage study with the FINE and SN-DnCNN induced priors to see if they are accurate from a frequentist point of view, meaning that they deliver probabilities which are coherent with the true posterior model.

*And in this case $\tilde{J}$ is firmly non-expansive, meaning that for every $(x, y) \in H, \|Bx - By\|_2^2 \leqslant \langle x - y, Bx - By \rangle$. This assumption is stronger than ensuring the residual non-expansivity.

	SN-DnCNN	FINE	Prox-GSD
$\sqrt{\varepsilon}$	5/255	2.25/255	15/255

Table 6.1: Noise levels of the different denoisers used within the PnP-ULA framework. These denoiser noise levels were found to achieve the best results from a perceptual and quantitative point of view for the non-blind deblurring inverse problem.

6.2 Qualitative comparison

In this section, we aim at analysing the results produced by our sampling algorithms from a visual perspective on a non-blind deblurring inverse problem presented in (5.39) with different denoiser induced priors. This section is motivated by the observation of metastable phenomena for certain images such as `Goldhill` or `Alley` when dealing with this inverse problem. The proposed MMSE restorations, although achieving good PSNR scores, present a grid-pattern that harms the visual impression. It urges to look for other priors derived from other denoisers.

First, we draw a comparison between the results generated with the SN-DnCNN and FINE induced priors. In both cases, the standard deviation of the additive Gaussian white noise is $\sigma = 1/255$. The comparison is drawn on `Simpson` and `Goldhill` presented in Figure 5.1. Then, we look at the results when the prior score is derived from a Prox-GSD denoiser. In this case, we cannot draw a direct comparison with other denoiser induced priors as the Prox-GSD denoiser only works with RGB images and we did not perform training on grayscale images. In order to speed up the convergence, as an application of Prox-GSD is 4 times slower than for the other two denoisers, we set the additive Gaussian white noise standard deviation to $\sigma = 5/255$.

The PnP-ULA discretization step-size is set such that $\delta = 0.9\delta_{\texttt{stable}}$ where $\delta_{\texttt{stable}}$ is defined as in Section 5.4.

Table 6.1 gives the different denoiser noise levels applied into the PnP-ULA framework. They correspond to the noise levels achieving the best results from a perceptual and quantitative point of view for the non-blind deblurring inverse problem.

Figure 6.4 compares different restorations for the deblurring inverse problem on `Simpson` and `Goldhill`. The SN-DnCNN induced posterior distribution produces the MMSE restorations which always reach the best scores in terms of PSNR or SSIM. If these quantitative results are confirmed by the visual impression on `Simpson`, the MMSE restoration for `Goldhill` presents an abnormal grid pattern that negatively affects it. This pattern is found after around $5e5$ iterations in the PnP-ULA samples and spreads to the whole image along the iterations. It only appears for images with high-frequency structures. At this point the Markov chain generated by PnP-ULA is not ergodic and we should not compute MMSE or standard deviations. The FINE MMSEs never exhibit such pattern for either image. However, their edges look smoother, which explains the lower PSNR and SSIM scores. In addition, for `Simpson`, an almost piecewise constant picture, PnP-ULA with SN-DnCNN tends to generate more diverse samples. These differences are visible around contours such as clouds.

Figure 6.5 shows the standard deviation of the marginal posterior at each pixel computed with PnP-ULA for the SN-DnCNN and FINE denoisers. With FINE, the uncertainty magnitude is lower than with SN-DnCNN. For example, it is 3 and 6.4 times smaller for respectively `Simpson` and `Goldhill`. It means that the FINE posterior distribution generate less diverse samples and it could interpret as a distribution concentrated. Furthermore, the grid pattern appearing over time for `Goldhill` causes uncertainty that propagates even on

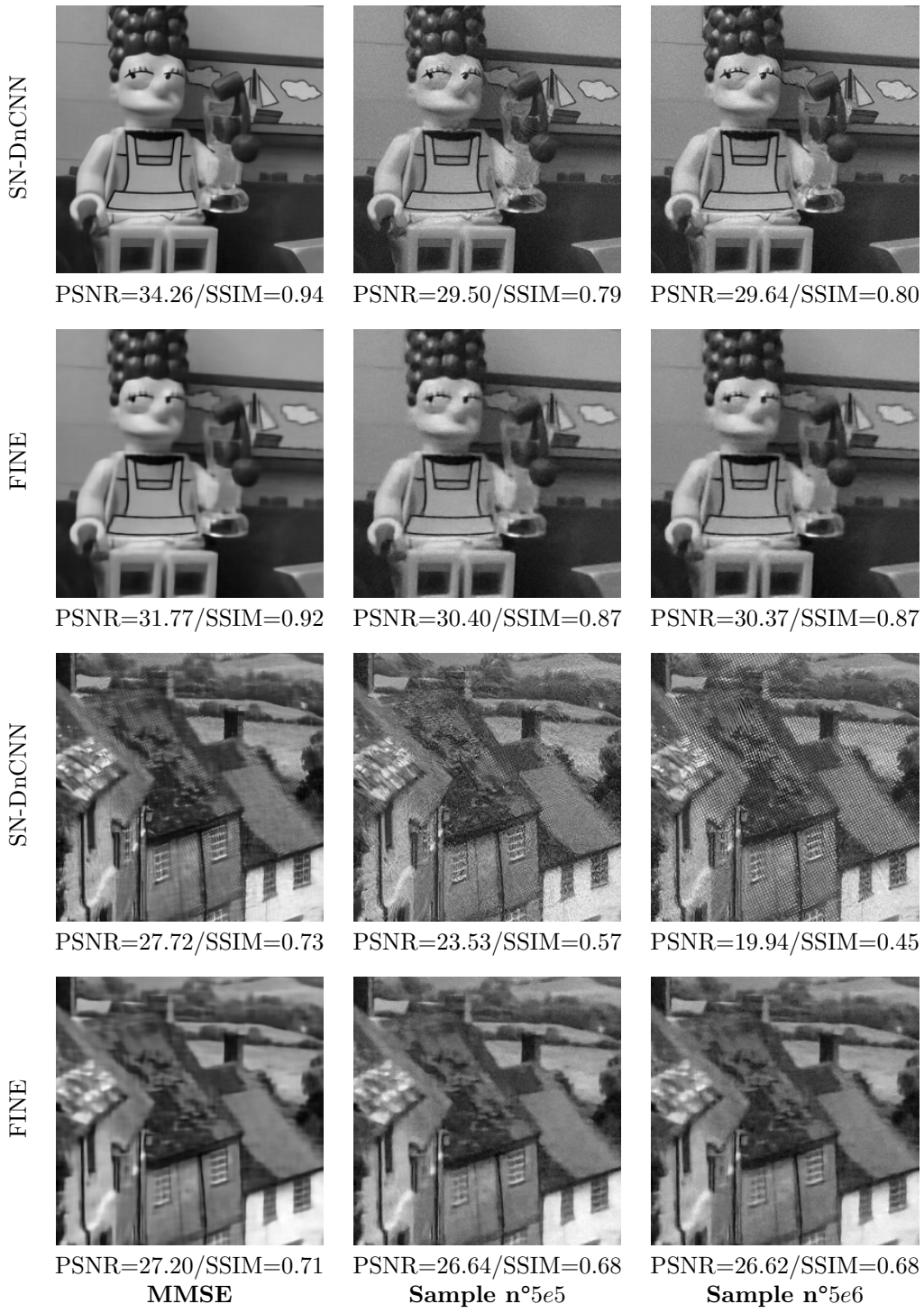


Figure 6.4: MMSE estimates and two samples obtained with PnP-ULA using SN-DnCNN and FINE induced priors for the deblurring problem for `Simpson` and `Goldhill`. The blur operator A is a 9×9 block-filter and the observation noise is an additive Gaussian white noise with standard deviation $\sigma = 1/255$. Samples generated with the FINE priors are less diverse but more robust to instabilities. The Markov chain computed with SN-DnCNN is not ergodic.

piecewise constant areas. Besides, uncertainty is more spread out with FINE than with SN-DnCNN. It seems that PnP-ULA with a FINE induced prior is more uncertain on the location of contours whereas with the SN-DnCNN prior uncertainty concerns more the edge magnitude. At this point, it is difficult to explain these differences. If the properties of SN-DnCNN limit less its expressivity in comparison with FINE and could explain the larger variety of the generated samples, we cannot explain the strange structures produced with the SN-DnCNN induced prior. It could be due to the neural network properties or a lack of training. To answer this question, we should retrain both denoisers on the same training set. This is currently under investigation.

We also test PnP-ULA with the Prox-GSD denoiser introduced by (Hurault et al., 2022b) on RGB images. The tackled inverse problem is slightly more difficult as the additive Gaussian white noise has a larger standard deviation $\sigma = 5/255$. The goal was to speed-up the convergence of the sampling algorithm as applying the Prox-GSD denoiser is computationally more expensive.

Figure 6.6 shows the MMSE and marginal posterior distribution standard deviation at each pixel estimated after $1e7$ iterations for `Color Simpson`. The MMSE is contaminated by packs of exploding pixels organized in lines that seriously ruin the restoration quality. The standard deviation plot reveals that if contours are more uncertain, piecewise constant areas are also corrupted by a lot of noise.

Figure 6.7 and Figure 6.8 allow to better understand the phenomenon that deteriorates the estimated MMSE. Figure 6.7 shows samples generated by PnP-ULA with the Prox-GSD denoiser plugged into the prior score. After $5e5$ iterations, samples are noisy deblurred solutions to the deblurring inverse problem. After $1.63e6$ iterations, the magnitude of some pixels increases and goes outside $[0, 1]$. This phenomenon arises at an edge, here the boat mast. Then, it steadily diffuses to the whole image and corrupts it. Figure 6.8 shows the cumulative histogram of the pixel values for different samples. The pixel magnitude increases over time, and so does the number of pixels greater than 1 and smaller than 0. In the end, after $1e7$ iterations, 20% of the pixels go outside $[0, 1]$.

To alleviate this pixel magnitude increase, we decide to exploit the knowledge we have about the original image that the denoiser does not seem to take into account. We know that it belongs to $[0, 1]^d$. Consequently, we apply P-PnP-ULA with $\mathsf{C} = [0, 1]^d$. Adding a projection onto the convex compact $\mathsf{C} = [0, 1]^d$ after an ULA step constrains the sample to belong to $[0, 1]^d$. Figure 6.9 exposes the estimated MMSE and the marginal posterior standard deviation derived using P-PnP-ULA. After $5e6$ iterations for the same inverse problem. The computed MMSE is a much better solution than the MMSE derived with PnP-ULA both from a visual and a quantitative point of view. It indeed achieves a great PSNR score although it is a little smooth. The uncertainty magnitude is reasonable and comparable to the one estimated for `Simpson` with PnP-ULA and SN-DnCNN. Edges are the most uncertain parts of the restoration.

Samples produced applying PPnP-ULA are presented in Figure 6.10. They appear to be good although noisy, and they do not exhibit any of the default observed with PnP-ULA. We point out here that applying PPnP-ULA allows to pick a larger step-size and consequently to more quickly and better explore the state-space. Whereas after $5e6$ iterations, samples generated by PnP-ULA with Prox-GSD were totally corrupted and did not constitute possible solutions to the deblurring inverse problem, samples with PPnP-ULA are far better.

PPnP-ULA does not solve all the problems when plugging Prox-GSD into the prior score. Figure 6.11 shows the empirical first and second order moment of the posterior distribution sampled using PPnP-ULA with Prox-GSD for `Fox` and computed after $1e6$ iterations. `Fox`

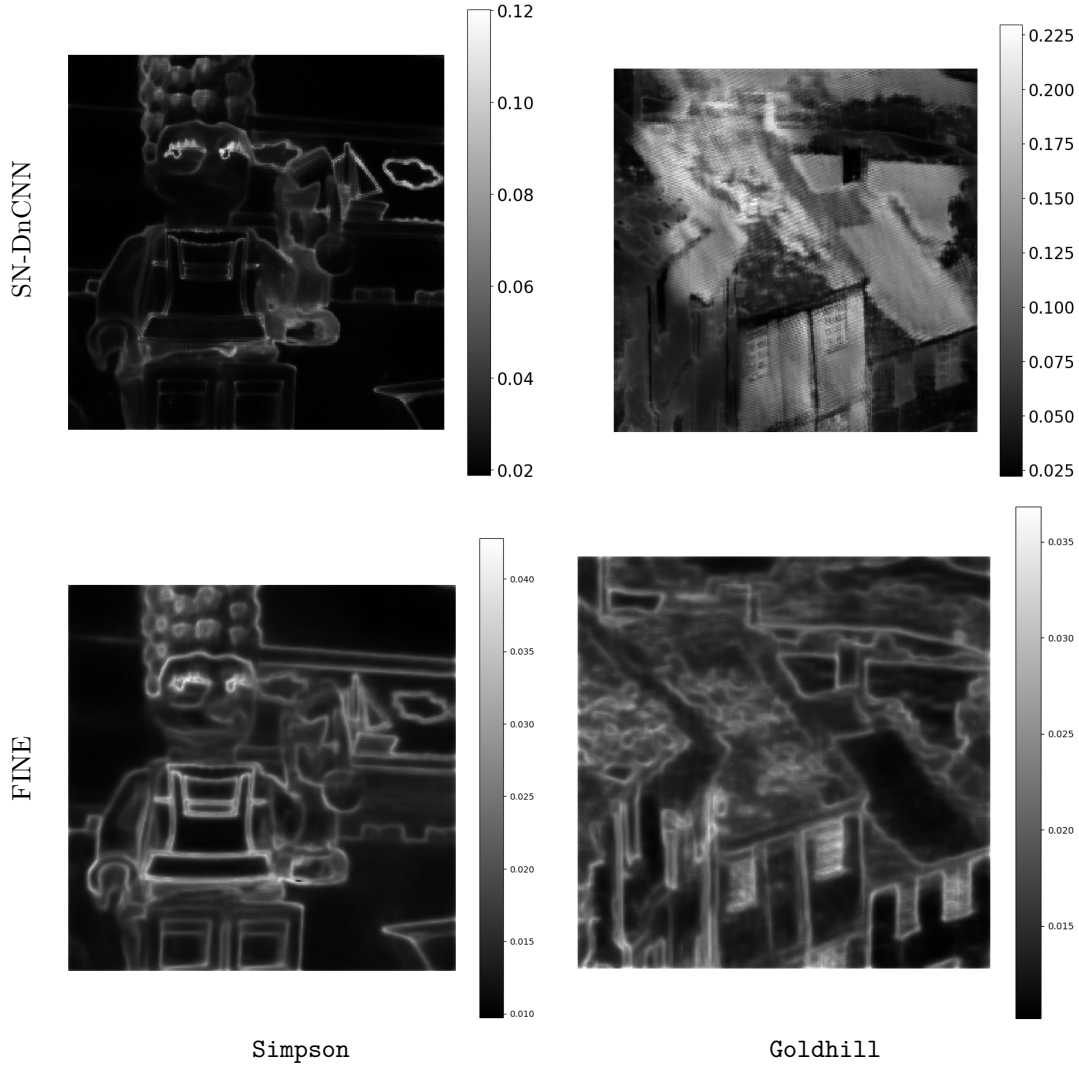
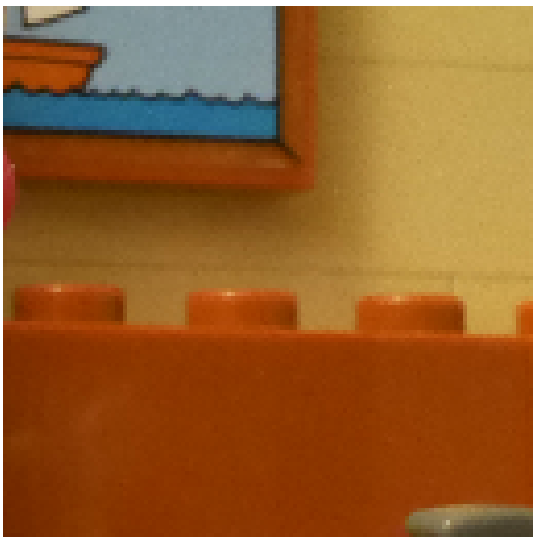
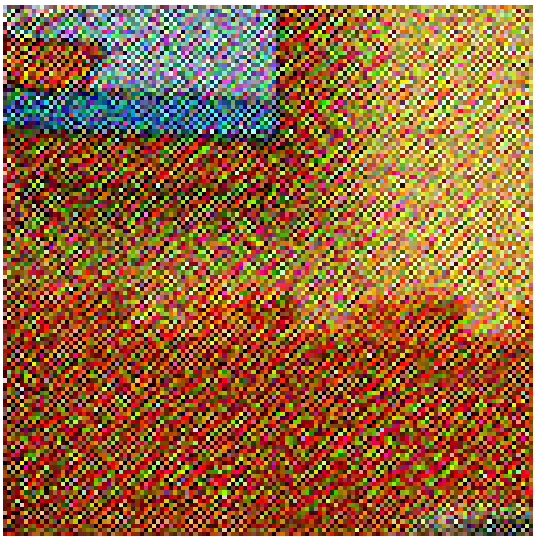


Figure 6.5: Marginal posterior standard deviation computed for the deblurring problem with the SN-DnCNN and FINE induced priors. For the Sn-DnCNN prior uncertainty is more concentrated around edges and higher, whereas for the FINE prior uncertainty is more diffuse but 4 times smaller.

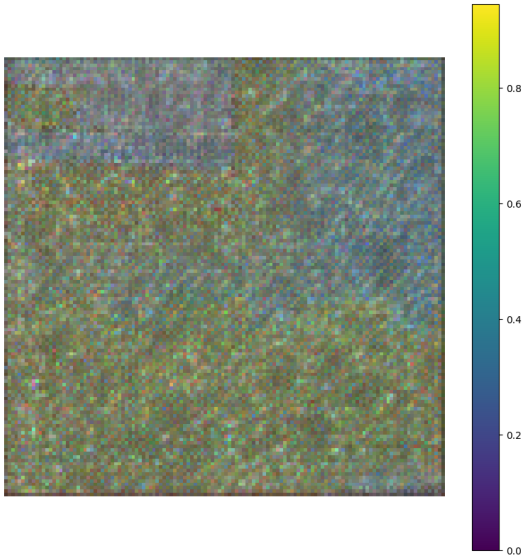


`Color Simpson`

Blurry observation (PSNR=25.18)



MMSE



Standard deviation

Figure 6.6: MMSE and marginal posterior standard deviation obtained with PnP-ULA using the Prox-GSD induced prior for the deblurring problem for `Color Simpson`. The blur operator A is a 9×9 block-filter and the observation noise is an additive Gaussian white noise with standard deviation $\sigma = 5/255$.

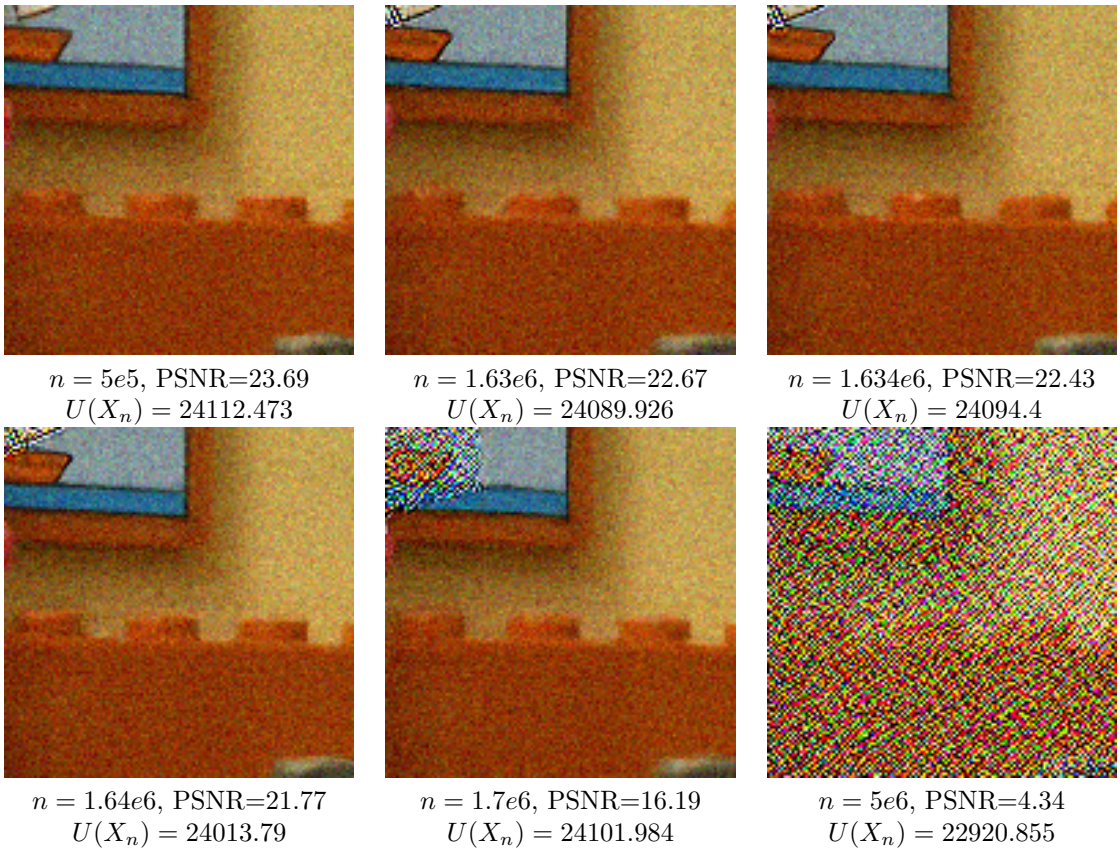
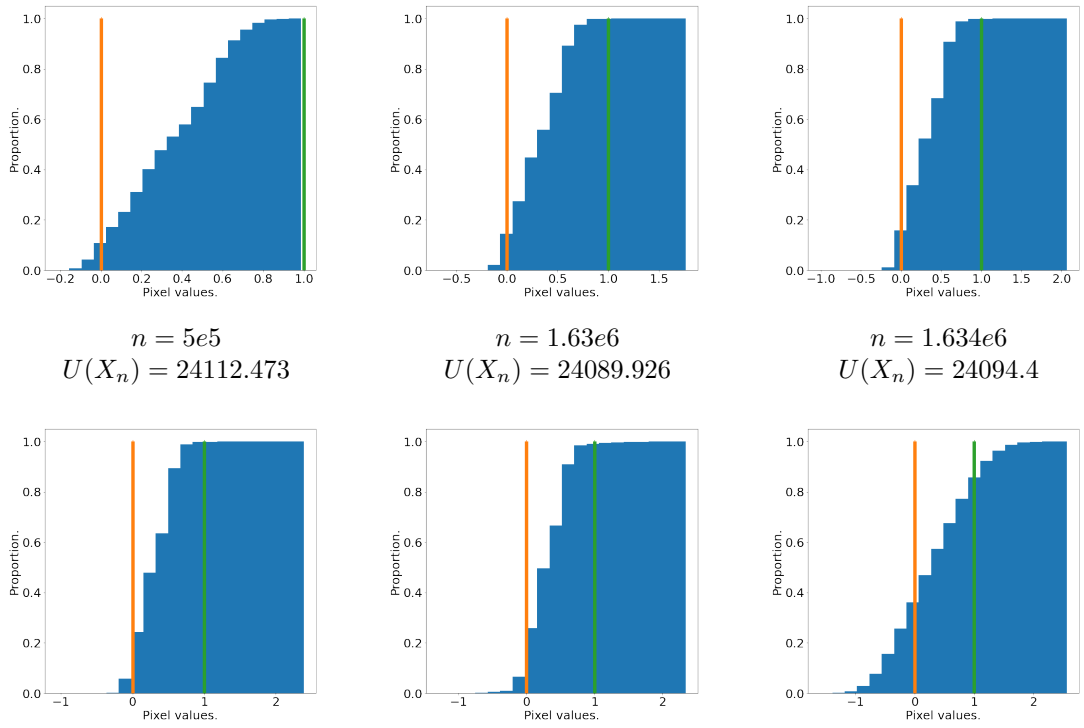


Figure 6.7: Samples generated by PnP-ULA for `Color Simpson` and their associated potential values for Prox-GSD. If produced samples look good in a first phase, they deteriorate over time. Abnormal structures appear around high-frequency areas. The Prox-GSD induced prior seems to promote images with these unnatural structures as the sample potential is at its lowest after $5e6$ iterations and the data fitting term do not penalize this evolution.



$n = 5e5$
$U(X_n) = 24112.473$

$n = 1.63e6$
$U(X_n) = 24089.926$

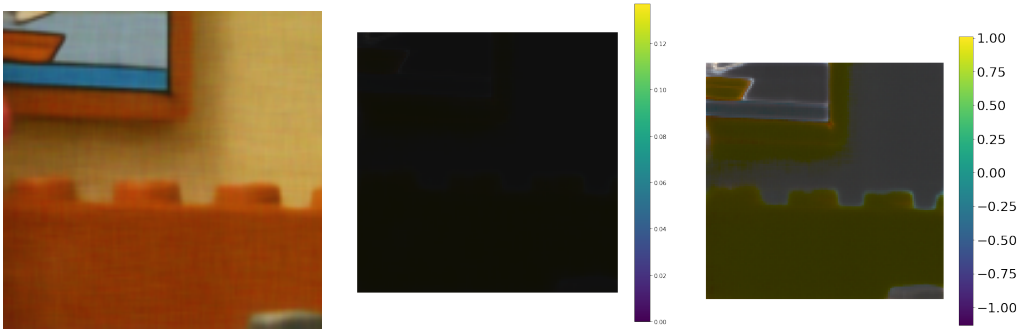
$n = 1.634e6$
$U(X_n) = 24094.4$

$n = 1.64e6$
$U(X_n) = 24013.79$

$n = 1.7e6$
$U(X_n) = 24101.984$

$n = 5e6$
$U(X_n) = 22920.855$

Figure 6.8: Cumulative histograms of the pixel values of different samples generated by PnP-ULA for `Color Simpson` with the Prox-GSD induced prior. The pixel magnitude of the generated samples tend to increase over time. After $5e6$ iterations, at least 30% of the pixels are outside $[0, 1]$. The Prox-GSD induced prior does not regularize enough the posterior distribution tails. It seems that the neural network did not learn the range of values of the images.



MMSE (PSNR=33.51)

Standard deviation

Log Standard deviation

Figure 6.9: MMSE and marginal posterior standard deviation obtained with P-PnP-ULA using the Prox-GSD induced prior for the deblurring problem inverse for `Color Simpson` with the projection convex compact set $\mathsf{C} = [0, 1]^d$. The blur operator A is 9×9 bloc-filter and the observation noise is an additive Gaussian white noise with standard deviation $\sigma = 5/255$. The hard projection onto C allows to alleviate diverging samples. Consequently the MMSE does not expose parasite structures. The log standard deviation is showed as uncertainty is very low.

$n = 1e6$, PSNR=24.25 $n = 4e6$, PSNR=24.13 $n = 5e6$, PSNR=24.25

Figure 6.10: Samples generated by P-PnP-ULA for `Color Simpson` for Prox-GSD.

is an interesting image for it has high-frequency textures. The estimated MMSE exhibits stripes that severely damages the restoration and makes it a bad solution to the inverse problem we aim at solving. The stripes mask is also perceptible on the standard deviation plot and constitutes the main source of uncertainty in the restoration. It reminds the grid pattern observed for `Goldhill, Alley` and `Bridge` when sampling from the posterior with the SN-DnCNN induced prior. As explained in Figure 5.7, high uncertainty was observed for some frequencies in the kernel of the blur operator for the samples generated by PnP-ULA. The denoiser used did not succeed in regularizing these frequencies.

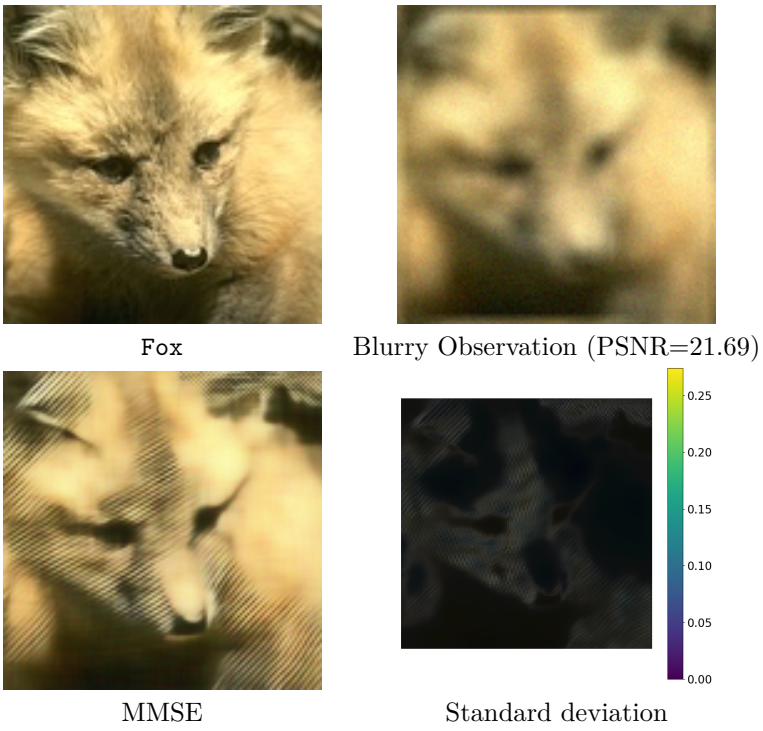
Figure 6.12 shows that these stripes come from high-frequency textures present around the fox snout and that rapidly spread over the fox coat over time. After only $1e5$ iterations samples are already corrupted by these motifs.

Although we cannot draw a direct comparison between Prox-GSD and the other denoisers previously tested as they only work on grayscale images, we apply PnP-ULA to a grayscale version of `Fox`. The idea is to see if these denoisers also struggle with this image. Figure 6.13 shows that these two denoisers induced posteriors do not struggle when dealing with this image. It would rather highlight the extreme sensitivity of the Prox-GSD induced posterior when dealing with images with high-frequency textures.

For completeness, we also show the Euclidean distance between the MMSE estimator computed from entire chain (*i.e.* all samples) and each stored sample (we show one point every 1000 samples) with both denoisers in Figure 6.14. Whereas samples generated with the FINE prior look totally uncorrelated, we observe a light meta-stability behavior for those generated with the SN-DnCNN prior. This could be explained by a very anisotropic posterior distribution induced by this denoiser.

6.3 Potential analysis

In this section we aim at understanding how the results exposed in Section 6.2 are explained in terms of potentials. As a neural network denoiser does not necessarily derive from a potential, we firstly present an approximation of the prior potential introduced in (Romano et al., 2017). It allows us to build a posterior potential U_{red} for the SN-DnCNN and FINE denoisers and to see the posterior potential evolution over time. As Prox-GSD is the gradient of some known potential g_ε, we can perform this study with the real potential U. These results are interesting as they allow to see the images promoted by our posterior distribution with our plug-and-play prior.



Fox | Blurry Observation (PSNR=21.69)

MMSE | Standard deviation

Figure 6.11: MMSE and marginal posterior standard deviation obtained with P-PnP-ULA using the Prox-GSD induced prior for the deblurring problem inverse for `Color Fox` with the projection set $\mathrm{C} = [0,1]^d$. The blur operator A is 9×9 bloc-filter and the observation noise is an additive Gaussian white noise with standard deviation $\sigma = 5/255$. A grid pattern on the MMSE ruins the restoration and increases uncertainty.

$n = 5e4$, PSNR=21.88 | $n = 7.5e4$, PSNR=21.49 | $n = 1e5$, PSNR=20.92 | $n = 1e6$, PSNR=16.60

Figure 6.12: Samples generated by P-PnP-ULA for `Fox` with Prox-GSD. A quick deterioration of the samples is observed. It seems to come from high-frequency motifs in the fox coat that eventually propagates to the whole image.

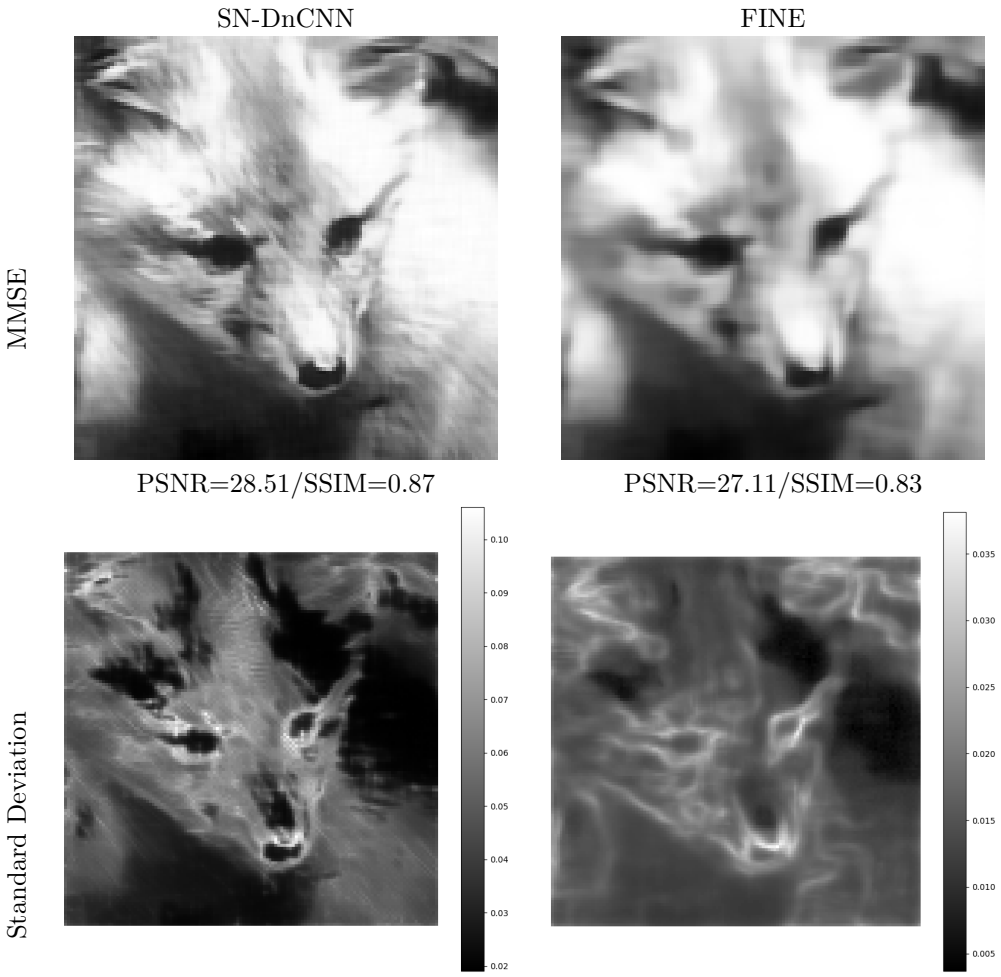
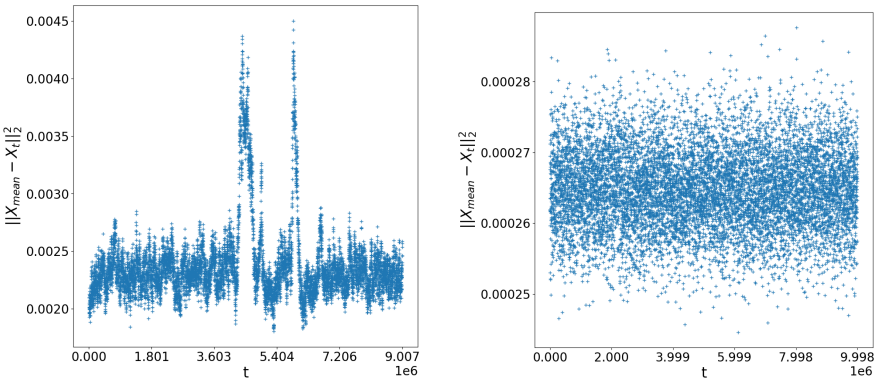


Figure 6.13: Comparison of the results generated by PnP-ULA with the SN-DnCNN and the FINE induced priors for a grayscale version of Fox after $1e7$ iterations. Both restorations do not exhibit artefacts ruining the visual impression as with Prox-GSD (see Figure 6.11). The SN-DnCNN induced posterior proposes an MMSE restoration with sharper edges and better quantitative results. The associated uncertainty is higher and less spread than with the FINE induced posterior.



SN-DnCNN FINE

Figure 6.14: Evolution of the L_2 distance between the final MMSE estimate and the samples generated by PnP-ULA with the SN-DnCNN and the FINE induced posterior for the deblurring problem after the burn-in phase for `Fox`.

When we use a classical denoiser, we do not know if there exists such a potential U associated with the posterior distribution. To alleviate this issue, we consider an approximated potential U_{red} introduced by Romano et al. in (Romano et al., 2017) and given for any $x \in \mathbb{R}^d$ by

$$\forall x \in \mathbb{R}^d,\ U_{\text{red}}(x) = \|\mathrm{A}x - y\|^2/(2\sigma^2) + (\alpha/2)x^T(x - D_\varepsilon(x)). \tag{6.6}$$

This potential is interesting because, if $\alpha \leftarrow \alpha/\varepsilon$, its gradient is equal to the gradient of the approximated log-posterior density considered in PnP-ULA and detailed in Algorithm 5. Indeed, if the denoiser D_ε is locally homogeneous, non-expansive and its Jacobian J_{D_ε} is symmetric [†], we have according to (Romano et al., 2017) and (Reehorst and Schniter, 2018) that

$$\nabla U_{\text{red}} = -\nabla \log p_{Y|X}(y|x) + \alpha/\varepsilon(x - D_\varepsilon(x)). \tag{6.7}$$

Figure 6.15 shows the evolution of the approximated potential of the samples generated by PnP-ULA with SN-DnCNN or FINE for `Simpson` and `Goldhill` on the deblurring inverse problem introduced in Section 6.2. For the FINE induced prior, the original image `Simpson` is explained by this approached potential as its value is below the potential values of the samples generated by PnP-ULA. It is not the case for `Goldhill`, as the potential values of the samples is below the potential of the original image. It can be explained by a wrong model-specification. With the SN-DnCNN induced prior, it seems that the original image does not look like an admissible solution as the potential of the original image is higher than the ones of the samples. For `Goldhill`, it seems that this posterior strongly promotes images presenting a grid-pattern.

The main advantage of using the Prox-GSD prior is that we have access to the posterior potential. We can then have an idea of the types of images that are promoted by this prior. Based on Figure 6.8, we can see that this posterior tends to promote images outside $[0, 1]^d$. The results observed in Figures 6.6 and 6.7 are not due to instabilities in the PnP-ULA

[†] In (Reehorst and Schniter, 2018), it is proved that if the Jacobian symmetry assumption does not hold, then there does not exist any potential U_{red} associated with (6.7). Furthermore the authors conducted a study to know for which denoisers this assumption holds. Interestingly, it is proved that for classical denoisers such as BM3D (Dabov et al., 2006), Non Local Means (NLM) (Buades et al., 2005a), TNRD (Chen and Pock, 2017) or DnCNN (Zhang et al., 2017), the assumption on the Jacobian symmetry does not hold.

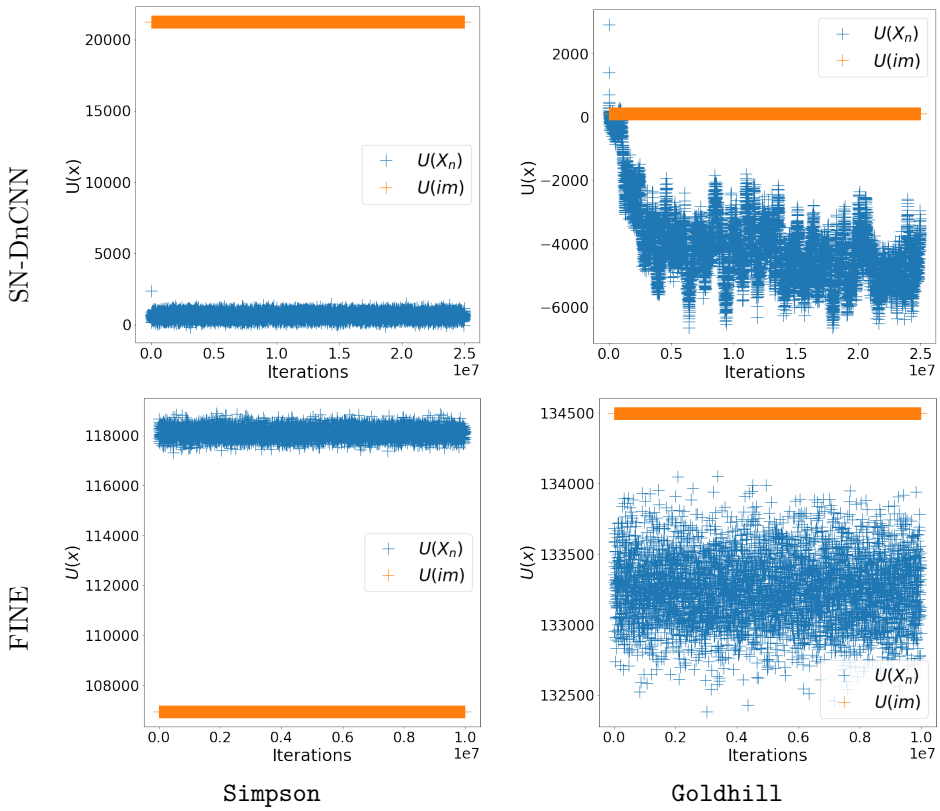
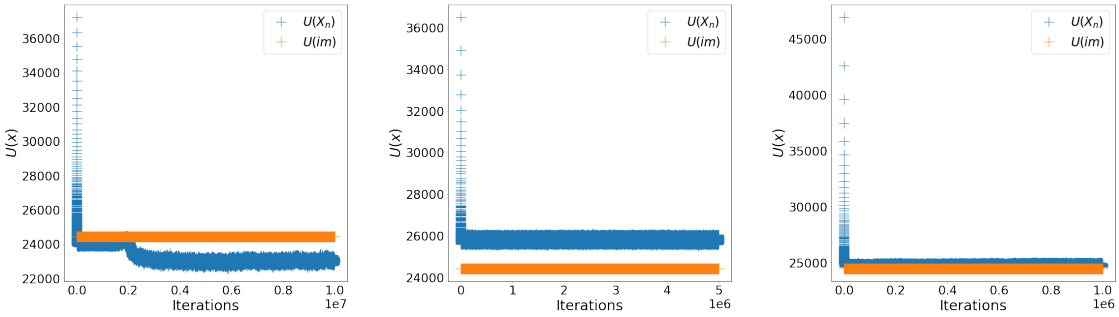


Figure 6.15: Evolution of the approximated posterior potential U_{red} for `Simpson` and `Goldhill` with the SN-DnCNN and FINE induced priors for the deblurring inverse problem. For the FINE prior, the original image potential is not always below the potential of the generated samples. For `Goldhill`, the orignal image is not a good solution for this inverse problem in terms of potential. With the SN-DnCNN induced prior, this potential does not regard the original images as a good solution. for `Goldhill`, it seems to promote images with the grid pattern.



`Color Simpson` (PnP-ULA) `Color Simpson` (P-PnP-ULA) `Fox` (P-PnP-ULA)

Figure 6.16: Evolution of the exact posterior potential U for `Color Simpson` and `Fox` for the deblurring inverse problem and PnP-ULA or P-PnP-ULA with Prox-GSD. Without the hard projection onto the convex compact set $\mathsf{C} = [0,1]^d$, the posterior induced by the Prox-GSD promotes samples as in Figure 6.7 with diverging pixel values. A hard projection onto C alleviates this issue. However, it still promotes samples with poor-perceptual quality.

scheme but by the nature of the prior. Although this denoiser was trained with images in $[0,1]^d$, it firstly seems that it has not learnt the image range of values. Consequently, it does not regularize enough the posterior tails and does not sufficiently constrain the mass probability inside $[0,1]^d$. The hard projection onto $[0,1]^d$ allows to alleviate this issue (even if it adds some bias). Adding this constraint, the original image appears to be a good solution to our inverse problem for `color Simpson`, although the sample potentials are higher. Even if we constrain the samples to belong to $[0,1]^d$, it does not necessarily produce good-looking restorations. `Fox` is a good example of this phenomenon. Of course, the original image is now an admissible solution to the inverse problem as we enforce the posterior probability mass to belong to $[0,1]^d$. However the posterior still struggles when dealing with high-frequency structures as seen in Figure 6.12 and none of the samples generated appears to be a good solution to the inverse problem considered.

6.4 Coverage ratio analysis

If the FINE and SN-DnCNN priors produce good restorations, it is interesting to question the quality of these priors from a frequentist point of view. The posterior probabilities computed are true only in the paradigm defined by the model. The aim of this section is then to check if they also are meaningful in a frequentist sense. This part is very much inspired by the work (Holden et al., 2022). As explained in (Holden et al., 2022), in a frequentist approach, the posterior probabilities should match the observed frequency over a large number of trials. It means that for m observations of the modelled quantities $\{x_1, x_2, ..., x_m\}$, we should have if m is sufficiently large

$$\frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_A(x_i) \approx \mathbb{P}(x \in A) = \int_A p(x|y)p(y)dy$$

In this section we test the coverage of the credible intervals derived by the posterior model induced by the SN-DnCNN and the FINE denoisers. To do so, we consider a dataset $D = \{x_i\}_{i=1}^m$ of clean images resulting from the concatenation of the datasets `set3c`, `CBSD10`, `CBSD68` and proceed as follows:

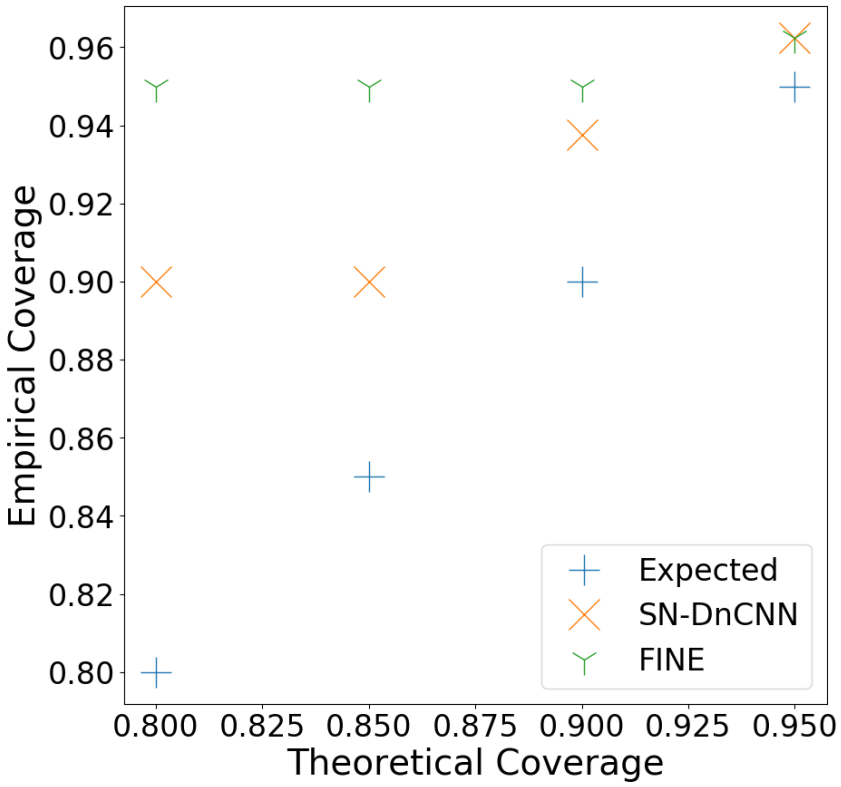


Figure 6.17: Coverage ratio analysis for the posterior induced by the SN-DnCNN and the FINE denoisers. Both posterior distributions are not accurate in the frequentist sense as the empirically estimated posterior probabilities do not match the theoretical ones. Both posterior models are over-confident.

- Considering the clean image x_i, we generate an observation $y_i \sim p(y|x = x_i)$. The observation process considered is the same as in Section 4.3.5 where the degradation operator A encodes a 9×9 bloc-filter with additive Gaussian white noise with standard deviation $\sigma = 1/255$.
- For each observation y_i, we sample from the posterior distribution using PnP-ULA over 5e5 iterations. It is initialized at the observation y_i. Then, we empirically estimate the posterior credible sets for different levels β using the RED potential U_{red}. The objective is to estimate the regions $A_{\beta,i}$ such that $\mathbb{P}(x \in A_{\beta,i}) = 1 - \beta$, *ie* the $(1 - \beta)$-quantiles of the posterior distribution for $\beta \in \{0.05, 0.10, 0.15, 0.20\}$.
- Finally, we check the proportion of clean images that belong to the credible sets of level β. The prior is considered to be true in the frequentist sense, if this proportion is close to $1 - \beta$.

Figure 6.17 shows that the estimated posterior distributions induced by both denoisers are not accurate in the frequentist sense. The observed frequencies do not match the posterior probabilities. Our posterior models are conservative and tend to overestimate uncertainty. It is partially explained by the ill-posedness of the deblurring inverse problem we are dealing with, that makes exploration difficult in some directions of the state-space. It could also

mean that the prior is misspecified. In this case, the regularization parameters used are maybe not the best to explain the true model. One way to compensate this could be to calibrate the posterior model by maximum likelihood optimization as in (Vidal et al., 2020). Finally, it is also possible that our denoisers did not properly learn the structure underlying the natural images, *ie* the sub-manifold where they live. If it is due to their structures that limit their expressivity or a lack of training is currently under investigation.

6.5 Conclusion

In this chapter, we investigated the influence of deep priors on the solutions of an ill-posed non-blind deblurring problem. We considered three denoisers with different properties. The comparison was drawn under different scopes. First, we looked at the generated results from a visual point of view. Then, we compared them in terms of potential in order to better understand which properties were promoted with the induced priors. Finally, we checked the frequentist accuracy of the FINE and SN-DnCNN induced priors.

The FINE posterior produces over-smooth samples but does not generate solutions with unrealistic artefacts. On the other hand, the SN-DnCNN posterior generates more diverse samples both from a visual and quantitative point of view in comparison to the FINE posterior. However, it can create artefacts that ruin the restoration for images with more detailed areas. Neither the SN-DnCNN posterior nor the FINE posterior is accurate in a frequentist sense and both tend to be conservative and to overestimate uncertainty. The Prox-GSD posterior really struggles with high-frequency structures.

At this point, we cannot explain these differences as the neural networks were trained on different training sets. For future work, we would like to go further in the prior analysis. First, we would like to train the FINE and SN-DnCNN denoiser on MNIST in order to see if their architectures and properties allow them to learn simple data structures. Then, we would like to train SN-DnCNN on `ImageNet` as FINE to see if their differences are due to their different properties. This is a major point that needs to be clarified as it could change the denoising neural netowrks we should use within Plug & Play sampling algorithms. Finally, we would like to test frequentist accuracy of different sampling methods such as score-matching and DDRM to better understand from which distribution they sample from.

7

Conclusion and Perspectives

In this thesis, we paved the way for a better understanding of the Bayesian Plug & Play methods for solving inverse problems. In Chapter 3 we introduced inverse problems and their inherent difficulties. We proposed a short review of the different recent strategies to combine deep neural networks and optimization schemes to solve such inverse problems. Our focus on Plug & Play methods was explained by their simplicity, their versatility, their ability to work with pre-trained networks. The mathematical and computational framework developed in this thesis is rooted in the Bayesian *M-complete* paradigm and adopts the view that *Plug & Play* models approximate a regularised oracle model. This is a major difference with the literature on Plug & Play methods that generally postulate and plug a prior assumed to be the *true* prior.

Chapter 4 exposed our first contribution on posterior maximization in Bayesian imaging with PnP priors. First, it clarified some theoretical major points about MAP estimation with PnP priors. Then it proposed an algorithm for MAP estimation, PnP-SGD (Algorithm 1), with convergence guarantees under realistic and checkable assumptions. For instance, we demonstrated that PnP-SGD converges to points in the vicinity of stationary points of the true posterior even if only an approximated MMSE denoiser is plugged into the SGD scheme. It is a major difference with most works on Plug & Play, which either proves convergence towards fixed points of some operator (Sun et al., 2019, 2020; Ryu et al., 2019) or towards stationary points of a distribution (Xu et al., 2020; Hurault et al., 2022a,b).

Chapter 5 presented our second contribution on posterior sampling in Bayesian imaging with PnP priors. It focused on MMSE restorations for inverse problems. First, it dealt with fundamental interrogations such as existence, stability and well-posedness of MMSE estimators under clear conditions. Then, two sampling algorithms were proposed, PnP-ULA (Algorithm 5) and PPnP-ULA (Algorithm 6). Convergence guarantees and non-asymptotic error bounds were derived under realistic assumptions. To the best of our knowledge, this is the first Bayesian *Plug & Play* framework with this level of insight and guarantees on the delivered solutions.

Finally, Chapter 6 investigated the role of the plugged prior when sampling from the posterior. The goal was to interrogate the importance of the denoiser properties and the solutions they promote for a given inverse problem. SN-DnCNN (Ryu et al., 2019) and FINE (Pesquet et al., 2020) induced posteriors were shown to be both over-confident and inaccurate in a frequentist sense, although they deliver excellent MMSE restorations. The differences between the results obtained with both denoisers cannot currently be precisely explained, as they were not trained on the same training set. This is an important question that we plan to investigate further in the future.

During the period of this Phd thesis, several advances have been made in the inverse problem literature both for point estimation and sampling. Among those advances, we can cite neural networks performing point estimation not trained to solve an inverse problem but a whole class of inverse problems. It allows to generalize end-to-end approaches. For example, (Zhang et al., 2018, 2021) proposed denoising neural networks taking a noise map as input in addition to the noisy image. These neural networks adapt to any Gaussian denoising inverse problem. Furthermore, (Debarnot and Weiss, 2022) proposed a neural network based method to directly estimate a spatially-variant blur kernel. Advances were also proposed in Bayesian sampling with the outbreak of score matching (Song and Ermon, 2019) and DDPM (Kawar et al., 2022) based methods. However, we still believe that Plug & Play methods remain relevant for their simplicity and their strong theoretical foundations.

One limitation of Plug & Play methods is their slowness. There are several possible directions to improve this aspect. Although, there are no convergence results available, SKROCK, presented in Section 5.5, takes a first step in this direction. To go further we could think of different discretization schemes of the Langevin SDE, which incorporate second order moment information for instance (Panloup et al., 2020). Another possibility to speed up the convergence of PnP-ULA that we considered consists in plugging an invertible denoiser such as NFs (Lugmayr et al., 2020; Gritsenko et al., 2019) or other invertible neural networks (Liu et al., 2020). We would benefit from the greater stability of implicit schemes and it would allow us to take larger step-sizes. Another possible deep prior could be the generative VAE proposed by (González et al., 2021). In this case, the log-posterior is quasi bi-concave and exhibits a more sampling-friendly geometry. It could also be applied for MAP estimation.

In Chapter 6, we saw that the Plug & Play induced posteriors were not accurate. One explanation could be that the parameters (α, ε) were not the best ones explaining the posterior model. (Vidal et al., 2020) propose a method to automatically set the regularization parameters by maximizing the likelihood in an empirical fashion. It is an axis of improvement we consider for future work.

The posterior inaccuracy raises other fundamental questions. Under which conditions can a denoiser accurately model the true posterior ? Or more, generally, can a Plug & Play posterior model accurately represent the posterior distribution ? In the literature, particular attention has been given to enforcing the Lipschitz constant of the denoiser or its residual (Ryu et al., 2019; Sun et al., 2019; Xu et al., 2020; Pesquet et al., 2020). In Chapter 6 we began to answer these questions from an experimental point of view. Building a (firmly) non-expansive denoiser does not seem to be enough. We plan to go further in this direction in future works.

As explained in Chapter 2, score-matching and DDPM based methods do not come with convergence guarantees and non-asymptotic error bounds but their results are very promising. So do neural networks based methods for sampling. However, the analysis is often limited to sample presentation. We could wish to analyse these distributions from a more statistical perspective and check if they are accurate in a frequentist sense for instance.

We may also wonder how these methods explore state-space in comparison with PnP-ULA. (Andrle et al., 2021) recently showed that invertible neural networks for sampling produced equivalent results to MCMC methods but outperformed them in terms of computational time for a specific inverse problem.

From a more practical point of view, we wish to adapt our MAP estimation and sampling methods to more realistic and concrete inverse problems. We thought for example of low-photon imaging problems where the data exhibit statistical properties that cannot be reflected by the Gaussian model. They arise when the number of photons emitted or reflected by an object or scene of interest is measured. They concern numerous areas of imaging science such as emission tomographic imaging (Hohage and Werner, 2016), fluorescence microscopy (Hohage and Werner, 2016; Bertero et al., 2018), astronomical imaging (Hohage and Werner, 2016; Starck and Murtagh, 2007), and single-photon light detection and ranging (LIDAR) (Altmann et al., 2016; Halimi et al., 2016; Rapp and Goyal, 2017; Shin et al., 2015). Mildly low-photon problems generally have Poisson statistics, whereas more challenging problems exhibit approximately Bernoulli/binomial or geometric data (Altmann et al., 2017a,b). These inverse problems are often associated with severe identifiability issues, poor stability, high uncertainty about the solution and poor regularity conditions. We also thought to tackle problems with additive spatially varying noise, which can be met in medical imaging. In parallel Magnetic Resonance Imaging (pMRI), which is an MRI allowing a faster acquisition time, we cannot assume a stationary noise model (Aja-Fernandez et al., 2015). Eventually, in RAW digital photography and satellite imaging, it is common to use a Gaussian noise model in addition to a Poisson noise and to approximate such noise model by a Gaussian distribution with a spatially-varying variance (Aguerrebere, 2014, Chapter 2).

Eventually, the *Plug & Play* models presented in this thesis approximate a regularised oracle model. The regularization is obtained by convolving the *true* but inaccessible prior p with a Gaussian kernel G_ε. However, we could think of other types of smoothing which might be more adapted to specific types of inverse problems with different measurement noise distributions.

A

Proofs of Chapter 4

In this supplementary chapter we present some extensions and gather the proofs related to Chapter 4 for completeness. The main author of this chapter is Valentin de Bortoli.

A.1 Proof of Proposition 4.2.1

Proof: Let $\mathsf{K} \subset \mathbb{R}^d$ be a compact set and $(x_n, \varepsilon_n)_{n\in\mathbb{N}}$ such that $\lim_{n\to+\infty} \varepsilon_n = 0$ and for any $n \in \mathbb{N}$, $x_n \in \mathsf{S}_{\varepsilon_n,\mathsf{K}}$. Let $x^\star \in \mathsf{S}$ a cluster point of $(x_n)_{n\in\mathbb{N}}$. Hence, for any $n \in \mathbb{N}^*$ there exist an increasing sequence $(k_n)_{n\in\mathbb{N}} \in \mathbb{N}^{\mathbb{N}}$ such that $\lim_{n\to+\infty} x_{k_n} = x^\star$.

In what follows, we show that $\lim_{n\to+\infty} \nabla \log(p_{\varepsilon_{k_n}}(x_{k_n})) = \nabla \log p(x^\star)$. First, we show that

$$\lim_{n\to+\infty} \max(|p - p_{\varepsilon_{k_n}}|_{\infty,\mathsf{K}}, \|\nabla p - \nabla p_{\varepsilon_{k_n}}\|_{\infty,\mathsf{K}}) = 0 \,. \tag{A.1}$$

Indeed, let $f \in \mathrm{C}(\mathbb{R}^d, \mathbb{R}^m)$ with $m \in \mathbb{N}$ such that $\|f\|_\infty < +\infty$ and denote $f_\varepsilon \in \mathrm{C}(\mathbb{R}^d, \mathbb{R}^m)$ given for any $x \in \mathbb{R}^d$ by

$$f_\varepsilon(x) = \textstyle\int_{\mathbb{R}^d} f(\tilde{x}) G_\varepsilon(x - \tilde{x}) \mathrm{d}\tilde{x} \,, \tag{A.2}$$

where we recall that for any $x \in \mathbb{R}^d$, $G_\varepsilon(x)$ is a Gaussian kernel with variance ε. For ease of notation, we define $G = G_1$. Let $\eta > 0$. Then, there exists $R > 0$ such that for any $\varepsilon > 0$ we have

$$\textstyle\int_{\|\tilde{x}\|>R} \|f(x - \varepsilon^{1/2}\tilde{x}) - f(x)\| G(\tilde{x}) \mathrm{d}\tilde{x} \leqslant 2\|f\|_\infty \int_{\|\tilde{x}\|>R} G(\tilde{x}) \mathrm{d}\tilde{x} < \eta/2 \,. \tag{A.3}$$

Let $\mathsf{K}' = \mathsf{K} + \overline{\mathrm{B}}(0, R)$. We have that K' is compact and therefore f is uniformly continuous on K'. Hence there exists $\xi > 0$ such that for any $x \in \mathsf{K}$, $\varepsilon \in (0, \xi]$ and $y \in \overline{\mathrm{B}}(0, R)$ we have

$$|f(x - \varepsilon^{1/2} y) - f(x)| \leqslant \eta/2 \,. \tag{A.4}$$

Hence, combining (A.3) and (A.4) we get that for any $x \in \mathsf{K}$ and $\varepsilon \in (0, \xi]$

$$\begin{aligned} \|f_\varepsilon(x) - f(x)\| &\leqslant \textstyle\int_{\mathbb{R}^d} \|f(x - \tilde{x}) - f(x)\| G_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} && \text{(A.5)} \\ &\leqslant \textstyle\int_{\mathbb{R}^d} \|f(x - \varepsilon^{1/2}\tilde{x}) - f(x)\| G(\tilde{x}) \mathrm{d}\tilde{x} && \text{(A.6)} \\ &\leqslant \textstyle\int_{\overline{\mathrm{B}}(0,R)} \|f(x - \varepsilon^{1/2}\tilde{x}) - f(x)\| G(\tilde{x}) \mathrm{d}\tilde{x} && \text{(A.7)} \\ &\quad + \textstyle\int_{\overline{\mathrm{B}}(0,R)^c} \|f(x - \varepsilon^{1/2}\tilde{x}) - f(x)\| G(\tilde{x}) \mathrm{d}\tilde{x} \leqslant \eta \,. && \text{(A.8)} \end{aligned}$$

Hence $\lim_{\varepsilon \to 0} \|f - f_\varepsilon\|_{\infty, \mathsf{K}} = 0$. Therefore using this result and that $p \in \mathrm{C}^1(\mathbb{R}^d, \mathbb{R})$ with $\|p\|_\infty + \|\nabla p\|_\infty < +\infty$ we get that

$$\lim_{n \to +\infty} \max(|p - p_{\varepsilon_{k_n}}|_{\infty, \mathsf{K}}, \|\nabla p - \nabla p_{\varepsilon_{k_n}}\|_{\infty, \mathsf{K}}) = 0 \,. \tag{A.9}$$

Combining this result, the fact that $\lim_{n \to +\infty} x_{k_n} = x^\star$ and that $p > 0$, we get that $\lim_{n \to +\infty} \nabla \log(p_{\varepsilon_{k_n}})(x_{k_n}) = \nabla \log p(x^\star)$. Indeed, we have that for any $n \in \mathbb{N}$

$$\begin{aligned} &\|\nabla \log(p_{\varepsilon_{k_n}}(x_{k_n})) - \nabla \log p(x^\star)\| && \text{(A.10)} \\ &\quad \leqslant \|\nabla \log(p_{\varepsilon_{k_n}}(x_{k_n})) - \nabla \log p(x_{k_n})\| + \|\nabla \log p(x_{k_n}) - \nabla \log p(x^\star)\| \,. && \text{(A.11)} \end{aligned}$$

We conclude using (A.9) and that $\log p \in \mathrm{C}(\mathbb{R}^d, \mathbb{R})$. Finally, we obtain that

$$0 = \lim_{n \to +\infty} \left\{ \nabla \log p(y|x_{k_n}) + \nabla \log p_{\varepsilon_{k_n}}(x_{k_n}) \right\} = \nabla \log p(y|x^\star) + \nabla \log p(x^\star) \,. \tag{A.12}$$

Hence, $x^\star \in \mathsf{S}_\mathsf{K}$. ■

A.2 Proof of Proposition 4.2.2

Proof: First, using that $p \in \mathrm{C}(\mathbb{R}^d, \mathbb{R}_+)$ we have that for any $v \in \mathbb{R}^d$ and $c \in \mathbb{R}$ there exists $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ such that $\int_\mathsf{A} |\langle x, v \rangle - c| \, p(x) \mathrm{d}x > 0$, meaning that there is no lower-dimensional affine space of $\mathbb{R}^d$ to which x belongs almost surely. Hence, we can apply (Gribonval, 2011, Lemma II.1) and $D_\varepsilon^\star \in \mathrm{C}^\infty(\mathbb{R}^d, \mathbb{R}^d)$.

We have that for any $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^m$, $\nabla \log p_\varepsilon(x|y) = \nabla \log p(y|x) + D_\varepsilon^\star(x)$. Hence, we get that $(x, y) \mapsto p_\varepsilon(x|y) \in \mathrm{C}^2(\mathbb{R}^d \times \mathbb{R}^m, \mathbb{R}_+)$. Since $-\nabla^2 \log p(x_{y_0}^\star|y_0)$ is positive there exist $\mathsf{U}_1 \subset \mathbb{R}^d$ open and $\mathsf{V}_1 \subset \mathbb{R}^m$ open such that for any $x \in \mathsf{U}_1$ and $y \in \mathsf{V}_1$, $-\nabla^2 \log p_\varepsilon(x|y)$ is positive. Hence, for any $y \in \mathsf{V}_1$, $x \in \mathsf{U}_1$ is a strict local maximizer if and only $\nabla \log p_\varepsilon(x|y) = 0$.

We have that $\nabla_x(\nabla_x \log p_\varepsilon)(x_{y_0}^\star|y_0)$ is invertible. Therefore using the implicit function theorem, there exist $\mathsf{V}_0 \subset \mathbb{R}^m$ open and $x^\star \in \mathrm{C}^1(\mathsf{V}_0, \mathsf{U}_1)$ such that for any $y \in \mathsf{V}_0$, $\nabla \log p_\varepsilon(x^\star(y)|y) = 0$, *i.e.* $x^\star(y)$ is a strict local maximizer of $x \mapsto \log p_\varepsilon(x|y)$, since $-\nabla^2 \log p_\varepsilon(x^\star(y)|y)$ is positive, which concludes the proof. ■

A.3 Proof of Proposition 4.2.3

Proof: In order to prove this theorem we are going to apply (Tadić et al., 2017, Theorem 2.1). In particular, in order to follow the notation of (Tadić et al., 2017, Theorem 2.1), we

define for $k \in \mathbb{N}$, $\zeta_k = Z_{k+1}$ and $\eta_k = b_\varepsilon(X_k) - \nabla \log p(y|X_k) - \nabla \log p_\varepsilon(X_k)$. Let $\varepsilon > 0$ and $\omega \in \mathsf{A}_{\varepsilon,\mathsf{K}}$. Using H2 we have for any $k \in \mathbb{N}$,

$$\|b_\varepsilon(X_k) - \nabla \log p(y|X_k) - \nabla \log p_\varepsilon(X_k)\| = \varepsilon^{-1}\|D_\varepsilon(X_k) - D^\star_\varepsilon(X_k)\| \leqslant \mathtt{M}(R)/\varepsilon \; . \tag{A.13}$$

Hence, we obtain that (Tadić et al., 2017, Assumption 2.1, Assumption 2.2) are satisfied. In what follows, we show that (Tadić et al., 2017, Assumption 2.3.c) holds. We have that for any $x \in \mathbb{R}^d$, $p_\varepsilon(x) = (p * G_\varepsilon)(x)$, where $*$ denotes the convolution product. Since $p, G_\varepsilon \in \mathrm{L}^1(\mathbb{R}^d)$ we get that for any $\xi \in \mathbb{R}^d$, $\widehat{p * G_\varepsilon}(\xi) = \hat{p}(\xi)\hat{G}_\varepsilon(\xi)$. Since $p \in \mathrm{L}^1(\mathbb{R}^d)$, $\|\hat{p}\|_\infty < +\infty$ using Riemann-Lebesgue theorem and in addition $\hat{G}_\varepsilon(\xi) = \exp[-\varepsilon \|\xi\|^2/2]$. Hence, $\widehat{p * G_\varepsilon} \in \mathrm{L}^1(\mathbb{R}^d)$ and we obtain that for almost every $x \in \mathbb{R}^d$

$$p_\varepsilon(x) = \textstyle\int_{\mathbb{R}^d} \hat{p}(\xi)\hat{G}_\varepsilon(\xi) \exp[\mathrm{i}\langle x, \xi\rangle] \mathrm{d}\xi \; . \tag{A.14}$$

In the rest of the proof, we denote $\bar{p}_\varepsilon : \mathbb{C}^d \to \mathbb{C}$ given for any $z = (z^1, \ldots, z^d) \in \mathbb{C}^d$ by $\bar{p}_\varepsilon(z) = \int_{\mathbb{R}^d} \hat{p}(\xi)\hat{G}_\varepsilon(\xi) \exp[\mathrm{i}\langle z, \xi\rangle] \mathrm{d}\xi$ where for any $z_1, z_2 \in \mathbb{C}^d$ we have $\langle z_1, z_2\rangle = \sum_{j=1}^d z_1^j \bar{z}_2^j$. We have that $\bar{p}_\varepsilon$ is analytic using the dominated convergence theorem. Since for any $x \in \mathbb{R}^d$, $p_\varepsilon(x) > 0$ and $\bar{p}_\varepsilon \in \mathrm{C}(\mathbb{C}^d, \mathbb{C})$, there exists an open set $\mathsf{U} \subset \mathbb{C}^d$ such that for any $z \in \mathsf{U}$, $\Re(\bar{p}_\varepsilon(z)) > 0$. Since $\log : \mathbb{C}\backslash(\{t \in \mathbb{C} : \Re(t) \leqslant 0\}) \to \mathbb{C}$ is analytic we obtain that $z \mapsto \log \bar{p}_\varepsilon(z)$ is analytic on U. Hence, $x \mapsto \log p(y|x) + \log p_\varepsilon(x)$ is real-analytic on $\mathbb{R}^d$. We conclude using (Tadić et al., 2017, Theorem 2.1).

■

B

Proofs of Chapter 5

B.1 Organization of the supplementary

In this supplementary chapter we present some extensions and gather the proofs related to Chapter 5 for completeness. The main author of this chapter is Valentin de Bortoli.

In this chapter, we first introduce a more general framework in Appendix B.2. Then in Appendix B.3 we present our improved convergence results in the case where the log-likelihood is strongly log-concave. Posterior approximation bounds in our general setting are gathered in Appendix B.4. Then we turn to the proof of these results. We first derive technical results in Appendix B.5. Proofs of Section 5.3.2 and Section 5.3.3 are presented

in Appendix B.6 and Appendix B.7 respectively. Finally, proofs of Appendix B.4 are given in Appendix B.8.

B.2 A general framework

We start by considering a slightly more general framework than the one previously introduced. More precisely, instead of $p^\star$ we consider a general distribution p and instead of considering p_ε as a prior we consider a tamed version of this density by introducing another hyperparameter $\alpha > 0$. In what follows, we describe this setting in details. We start by recalling a mild assumption on the likelihood.

H1 *For any* $y \in \mathbb{R}^m$, $\sup_{x\in\mathbb{R}^d} p(y|x) < +\infty$, $p(y|\cdot) \in \mathrm{C}^1(\mathbb{R}^d, (0,+\infty))$ *and there exists* $\mathtt{L}_y > 0$ *such that* $\nabla \log(p(y|\cdot))$ *is* $\mathtt{L}_y$ *Lipschitz continuous.*

For any $\varepsilon > 0$ we recall that p_ε is given by the Gaussian smoothing of p with level ε, for any $x \in \mathbb{R}^d$ by

$$p_\varepsilon(x) = (2\pi\varepsilon)^{-d/2} \int_{\mathbb{R}^d} \exp[-\|x-\tilde{x}\|^2/(2\varepsilon)]\, p(\tilde{x})\mathrm{d}\tilde{x} \,. \tag{B.1}$$

One typical example of likelihood function that we consider in our numerical illustration, see Section 5.4, is $p(y|x) \propto \exp[-\|\mathrm{A}x - y\|^2/(2\sigma^2)]$ for any $x \in \mathbb{R}^d$ with $\sigma > 0$ and $\mathrm{A} \in \mathbb{R}^{m\times d}$. Before turning to the analysis of the convergence of the introduced algorithms we state the following proposition which ensures the regularity of the posterior model w.r.t to the observation y.

We consider the following assumption on $x \mapsto p(y|x)$ and the prior p for some hyperparameter $\alpha > 0$ and an observation $y \in \mathbb{R}^m$.

H7 *The following hold:*

(a) $\int_{\mathbb{R}^d} p(y|\tilde{x})p^\alpha(\tilde{x})\mathrm{d}\tilde{x} < +\infty$ *and for any* $\varepsilon > 0$, $\int_{\mathbb{R}^d} p(y|\tilde{x})p_\varepsilon^\alpha(\tilde{x})\mathrm{d}\tilde{x} < +\infty$.

(b) $\int_{\mathbb{R}^d} \|\tilde{x}\|^2 p(x)\mathrm{d}x < +\infty$.

Note that if $\alpha = 1$, H7-(a) hold under H1, see Proposition 5.2.1. Under H7-(a), define π the target probability distribution for any $x \in \mathbb{R}^d$ by

$$(\mathrm{d}\pi/\mathrm{dLeb})(x) = p(y|x)p^\alpha(x) \Big/ \int_{\mathbb{R}^d} p(y|\tilde{x})p^\alpha(\tilde{x})\mathrm{d}\tilde{x} \,. \tag{B.2}$$

Note that for ease of notation, we do not explicitly highlight the dependency of the posterior distribution π with respect to the hyperparameter $\alpha > 0$, since it is fixed in the rest of this section. We also consider the family of probability distributions $\{\pi_\varepsilon : \varepsilon > 0\}$ given for any $\varepsilon > 0$ and $x \in \mathbb{R}^d$ by

$$(\mathrm{d}\pi_\varepsilon/\mathrm{dLeb})(x) = p(y|x)p_\varepsilon^\alpha(x) \Big/ \int_{\mathbb{R}^d} p(y|\tilde{x})p_\varepsilon^\alpha(\tilde{x})\mathrm{d}\tilde{x} \,. \tag{B.3}$$

We also recall the assumption on the denoiser D_ε, see Section 5.3.2 for details.

H2 *There exist* $\varepsilon_0 > 0$, $\mathtt{M}_R \geqslant 0$ *and* $\mathtt{L} \geqslant 0$ *such that for any* $\varepsilon \in (0, \varepsilon_0]$, $x_1, x_2 \in \mathbb{R}^d$ *and* $x \in \overline{\mathrm{B}}(0, R)$ *we have*

$$\|(\mathrm{Id} - D_\varepsilon)(x_1) - (\mathrm{Id} - D_\varepsilon)(x_2)\| \leqslant \mathtt{L}\,\|x_1 - x_2\| \,, \qquad \|D_\varepsilon(x) - D_\varepsilon^\star(x)\| \leqslant \mathtt{M}_R \,, \tag{B.4}$$

where we recall that

$$D_\varepsilon^\star(x_1) = \int_{\mathbb{R}^d} \tilde{x}\; g_\varepsilon(\tilde{x}|x_1)\mathrm{d}\tilde{x} \,. \tag{B.5}$$

B.3 Strongly log-concave case

We now present an improvement on the results of Section 5.3.2 in the case where the log-likelihood $x \mapsto \log p(y|x)$ is strongly concave. We recall that the Markov chain is given by the following recursion: $X_0 \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$

$$X_{k+1} = X_k + \delta b_\varepsilon(X_k) + \sqrt{2\delta} Z_{k+1} \;, \tag{B.6}$$
$$b_\varepsilon(x) = \nabla \log p(y|x) + \alpha P_\varepsilon(x) + (\mathrm{prox}_\lambda(\iota_{\mathsf{C}})(x) - x)/\lambda \;, \quad P_\varepsilon(x) = (D_\varepsilon(x) - x)/\varepsilon \;, \tag{B.7}$$

In the strongly concave setting we set $\mathsf{C} = \mathbb{R}^d$, *i.e.* $\forall x \in \mathsf{C}$, $\mathrm{prox}_\lambda(\iota_{\mathsf{C}})(x) = x$. We recall that in our image processing applications, we have that for any $x \in \mathbb{R}^d$, $p(y|x) \propto \exp[-\left\|\mathrm{A}x - y\right\|^2/(2\sigma^2)]$ and that $x \mapsto p(y|x)$ is strongly log-concave if and only if A is invertible. This is the case for denoising tasks where $\mathrm{A} = \mathrm{Id}$ and for deblurring tasks with convolution kernels which have full Fourier support.

We start with the following result which ensures that the Markov chain (B.6) is geometrically ergodic under H4 for the Wasserstein metric $\mathbf{W}_1$ and in V-norm for $V : \mathbb{R}^d \to [1, +\infty)$ given for any $x \in \mathbb{R}^d$ by

$$V(x) = 1 + \|x\|^2 \;. \tag{B.8}$$

The following proposition is the counterpart of Proposition 5.3.2.

Proposition B.3.1 *Assume* H1, H7 *and* H4(R) *for some* $R > 0$. *Let* $\alpha > 0$ *and* $\varepsilon \in (0, \varepsilon_0]$. *If there exists* $\mathtt{m} > 0$ *such that* $\log(p(y|\cdot))$ *is* $\mathtt{m}$*-concave with* $\mathtt{m} \geqslant 2\alpha\mathtt{L}/\varepsilon$ *then there exist* $A_1 \geqslant 0$ *and* $\rho_1 \in [0, 1)$ *such that for any* $\delta \in (0, \bar{\delta}]$, $x_1, x_2 \in \mathbb{R}^d$ *and* $k \in \mathbb{N}$ *we have*

$$\left\|\delta_{x_1} \mathrm{R}^k_{\varepsilon,\delta} - \delta_{x_2} \mathrm{R}^k_{\varepsilon,\delta}\right\|_V \leqslant A_1 \rho_1^{k\delta} (V^2(x_1) + V^2(x_2)) \;, \tag{B.9}$$
$$\mathbf{W}_1(\delta_{x_1} \mathrm{R}^k_{\varepsilon,\delta}, \delta_{x_2} \mathrm{R}^k_{\varepsilon,\delta}) \leqslant A_1 \rho_1^{k\delta} \left\|x_1 - x_2\right\| \;, \tag{B.10}$$

where V *is given in* (B.8) *and* $\bar{\delta} = \mathtt{m}(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)^{-2}/2$.

Proof: The proof is postponed to Appendix B.6.2. ■

We recall the assumption on g_ε which ensures that $x \mapsto \log(p_\varepsilon(x))$ has Lipschitz gradients.

H6 *For any* $\varepsilon > 0$, *there exists* $\mathtt{K}_\varepsilon \geqslant 0$ *such that for any* $x \in \mathbb{R}^d$,

$$\int_{\mathbb{R}^d} \left\| \tilde{x} - \int_{\mathbb{R}^d} \tilde{x}' g_\varepsilon(\tilde{x}'|x) \mathrm{d}\tilde{x}' \right\|^2 g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} \leqslant \mathtt{K}_\varepsilon \;, \tag{B.11}$$

with g_ε *given in* (5.13).

The following proposition is the counterpart of Proposition 5.3.3.

Proposition B.3.2 *Assume* H1, H7, H4(R) *for some* $R > 0$ *and* H6. *Moreover, let* $\alpha > 0$, $\varepsilon \in (0, \varepsilon_0]$ *and assume that* $\int_{\mathbb{R}^d} (1 + \|\tilde{x}\|^4) p_\varepsilon^\alpha(\tilde{x}) \mathrm{d}\tilde{x} < +\infty$. *In addition, if there exists* $\mathtt{m} > 0$ *such that* $\log(p(y|\cdot))$ *is* $\mathtt{m}$*-concave with* $\mathtt{m} \geqslant (2\alpha/\varepsilon)\max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon)$ *and* $\bar{\delta} = \mathtt{m}(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)^{-2}/2$, *then for any* $\delta \in (0, \bar{\delta}]$, $\mathrm{R}_{\varepsilon,\delta}$ *admits an invariant probability measure* $\pi_{\varepsilon,\delta}$ *and there exists* $B_1 \geqslant 0$ *such that for any* $\delta \in (0, \bar{\delta}]$

$$\left\|\pi_{\varepsilon,\delta} - \pi_\varepsilon\right\|_V \leqslant B_1(\delta^{1/2} + \mathtt{M}_R + \exp[-R]) \;, \tag{B.12}$$

where V *is given in* (B.8) *and* B_1 *does not depend on* R.

Proof: The proof is postponed to Appendix B.6.3. ■

The bound appearing in (B.12) depends on an extra hyperparameter $R > 0$ which may be optimized if H2(R) holds for any $R > 0$ and $\{\mathtt{M}_R : R > 0\}$ can be expressed in a closed form. In particular if there exists $\mathtt{M} \in (0, 1)$ such that for any $R > 0$, $\mathtt{M}_R = \mathtt{M} \times R$ then there exists $B_1 \geqslant 0$ such that for any $\delta \in (0, \bar{\delta}]$ and $R > 0$

$$\|\pi_{\varepsilon,\delta} - \pi_\varepsilon\|_V \leqslant B_1(\delta^{1/2} + \mathtt{M}\log(1/\mathtt{M})) \,, \tag{B.13}$$

by setting $R = \log(1/\mathtt{M})$. Similarly if there exists $\mathtt{M} > 0$ such that for any $R > 0$, $\mathtt{M}_R = \mathtt{M}$ then there exists $B_1 \geqslant 0$ such that for any $\delta \in (0, \bar{\delta}]$ and $R > 0$

$$\|\pi_{\varepsilon,\delta} - \pi_\varepsilon\|_V \leqslant B_1(\delta^{1/2} + \mathtt{M}) \,, \tag{B.14}$$

by letting $R \to +\infty$.

We now combine Proposition B.3.1 and Proposition B.3.2 in order to control the bias of the Monte Carlo estimator obtained using PnP-ULA. This proposition is the counterpart of Proposition 5.3.4.

Proposition B.3.3 *Assume* H1, H7, H2(R) *for some* $R > 0$ *and* 6. *Moreover, let* $\alpha > 0$, $\varepsilon \in (0, \varepsilon_0]$ *and assume that* $\int_{\mathbb{R}^d}(1 + \|\tilde{x}\|^4)p_\varepsilon^\alpha(\tilde{x})\mathrm{d}\tilde{x} < +\infty$. *In addition, if there exists* $\mathtt{m} > 0$ *such that* $\log(p(y|\cdot))$ *is* $\mathtt{m}$*-concave with* $\mathtt{m} \geqslant (2\alpha/\varepsilon)\max(\mathtt{L}, 1+\mathtt{K}_\varepsilon/\varepsilon)$ *and* $\bar{\delta} = \mathtt{m}(\mathtt{L}_y+\alpha\mathtt{L}/\varepsilon)^{-2}/2$, *then there exists* $C_{1,\varepsilon} \geqslant 0$ *such that for any* $h : \mathbb{R}^d \to \mathbb{R}$ *measurable with* $\sup_{x\in\mathbb{R}^d}\{|h(x)|\,(1+\|x\|^2)^{-1}\} \leqslant 1$, $n \in \mathbb{N}^*$, $\delta \in (0, \bar{\delta}]$ *we have*

$$\left|n^{-1}\sum_{k=1}^{n}\mathbb{E}\left[h(X_k)\right] - \int_{\mathbb{R}^d} h(\tilde{x})\mathrm{d}\pi_\varepsilon(\tilde{x})\right| \leqslant C_{1,\varepsilon}(\delta^{1/2} + \mathtt{M}_R + \exp[-R] + (n\delta)^{-1})(1 + \|x\|^4) \,. \tag{B.15}$$

Proof: The proof is straightforward upon combining Proposition B.3.1 and Proposition B.3.2. ■

In particular, applying Proposition B.3.3 to the family $\{h_i\}_{i=1}^d$ where for any $i \in \{1, \ldots, d\}$, $h_i(x) = x_i$ we get that

$$\left\|n^{-1}\sum_{k=1}^{n}\mathbb{E}\left[X_k\right] - \int_{\mathbb{R}^d} \tilde{x}\mathrm{d}\pi_\varepsilon(\tilde{x})\right\| \leqslant C_{1,\varepsilon}(\delta^{1/2} + \mathtt{M}_R + \exp[-R] + (n\delta)^{-1})(1 + \|x\|^4) \,, \tag{B.16}$$

and $n^{-1}\sum_{k=1}^n X_k$ is an approximation of the MMSE given by $\int_{\mathbb{R}^d}\tilde{x}\mathrm{d}\pi_\varepsilon(\tilde{x})$.

B.4 Posterior approximation

We consider the following general regularity assumption.

H8 (α) *There exist* $\kappa \geqslant 0$, $\beta > 0$ *and* $q : \mathbb{R}^d \to (0, +\infty)$ *such that* $\int_{\mathbb{R}^d} q(\tilde{x})\mathrm{d}\tilde{x} = 1$, $\|q\|_\infty < +\infty$ *and for almost every* $x \in \mathbb{R}^d$, $\int_{\mathbb{R}^d} |p(\tilde{x}) - p(x - \tilde{x})|\, q^{\min(1-1/\alpha,0)}(\tilde{x})\mathrm{d}\tilde{x} \leqslant \mathrm{e}^{\kappa(1+\|x\|^2)}\,\|x\|^\beta$.

In the case where $\alpha \geqslant 1$, H8(α) is equivalent to the following assumption: there exist $\kappa \geqslant 0$ and $\beta > 0$ such that for almost every $x \in \mathbb{R}^d$, $\|\mu - (\tau_x)_{\#}\mu\|_{\mathrm{TV}} \leqslant \mathrm{e}^{\kappa(1+\|x\|^2)} \|x\|^{\beta}$, where we recall that μ is the probability distribution with density with respect to the Lebesgue measure proportional to p and that for any $\tilde{x} \in \mathbb{R}^d$, $\tau_x(\tilde{x}) = \tilde{x} - x$. Note that since $p \in \mathrm{L}^1(\mathbb{R}^d)$ we have $\lim_{x \to 0} \|\mu - (\tau_x)_{\#}\mu\|_{\mathrm{TV}} = 0$. In H8($\alpha$) for $\alpha < 1$ we assume more regularity for $x \mapsto (\tau_x)_{\#}\mu$ in total variation in order to obtain explicit bounds between π_ε and π.

In the following proposition we provide easy-to-check conditions on the density of the prior distribution μ so that H8(α) holds.

Proposition B.4.1 *Assume that there exists $U : \mathbb{R}^d \to \mathbb{R}$ such that for any $x \in \mathbb{R}^d$, $p(x) = \mathrm{e}^{-U(x)} / \int_{\mathbb{R}^d} \mathrm{e}^{-U(\tilde{x})} \mathrm{d}\tilde{x}$. Assume that U is γ-Hölder,* i.e. *there exists $C_\gamma > 0$ such that for any $x_1, x_2 \in \mathbb{R}^d$,* i.e. *$\|U(x_1) - U(x_2)\| \leqslant C_\gamma \|x_1 - x_2\|^{\gamma}$. Then* H8($\alpha$) *is satisfied for $\alpha \geqslant 1$. In addition, assume that $\gamma \leqslant 2$ and that there exist $c_1, \varpi > 0$ and $c_2 \in \mathbb{R}$ such that for any $x \in \mathbb{R}^d$, $U(x) \geqslant c_1 \|x\|^{\varpi} + c_2$ then* H8(α) *holds for any $\alpha > 0$.*

Under H8(α) we establish the following result which ensures that π_ε is close to π in total variation for small values of ε.

Proposition B.4.2 *Assume* H1*, then the following hold:*

(a) If $\alpha = 1$, then $\lim_{\varepsilon \to 0} \|\pi_\varepsilon - \pi\|_{\mathrm{TV}} = 0$.

(b) Assume that $\|p\|_\infty < +\infty$ then for any $\alpha \geqslant 1$, $\lim_{\varepsilon \to 0} \|\pi_\varepsilon - \pi\|_{\mathrm{TV}} = 0$.

(c) Assume that $\|p\|_\infty < +\infty$ and H8(α) *then there exist $\varepsilon_1 > 0$ and $A_0 \geqslant 0$ such that for any $\varepsilon \in (0, \varepsilon_1]$ we have $\|\pi_\varepsilon - \pi\|_{\mathrm{TV}} \leqslant A_0 \varepsilon^{\beta \min(\alpha, 1)/2}$.*

Note that a related result in the case where $p(x) = \mathrm{e}^{-U(x)} / \int_{\mathbb{R}^d} \mathrm{e}^{-U(\tilde{x})} \mathrm{d}\tilde{x}$ with U Lipschitz continuous and $\alpha = 1$ can be found in (Vono et al., 2019, Corollary 1) with explicit dependency with respect to the dimension d. However, note that Proposition B.4.2 differs from (Vono et al., 2019, Corollary 1) since the Gaussian smoothing approximation is applied to the prior distribution and the estimate is given on the posterior distribution in Proposition B.4.2, whereas in (Vono et al., 2019, Corollary 1) the Gaussian smoothing approximation is applied to the posterior distribution and the estimate is given on the posterior distribution as well.

The following proposition is an extension of Proposition B.3.3 and Proposition 5.3.4. The main difference is that the approximation is expressed with respect to the true posterior π and not π_ε for some value $\varepsilon > 0$. Let $\varepsilon_1 > 0$ be given by Proposition B.4.2. In order to state this proposition, we recall the following assumption which is a relaxation of the strongly log-concave condition.

H5 *There exists $\mathtt{m} \in \mathbb{R}$ such that for any $x_1, x_2 \in \mathbb{R}^d$ we have*

$$\langle \nabla \log p(y|x_2) - \nabla \log p(y|x_1), x_2 - x_1 \rangle \leqslant -\mathtt{m} \|x_2 - x_1\|^2 \ . \tag{B.17}$$

Note that the posterior is strongly log-concave if and only if $\mathtt{m} > 0$.

Proposition B.4.3 *Assume* H1*,* H7*,* H2*,* H6 *and* H5*. Let $\alpha > 0$ and assume that for any $\varepsilon \in (0, \min(\varepsilon_0, \varepsilon_1)]$, $\int_{\mathbb{R}^d} (1 + \|\tilde{x}\|^4)(p_\varepsilon^\alpha + p^\alpha)(\tilde{x}) \mathrm{d}\tilde{x} < +\infty$ and* H8(α)*. Then there exists $C_0 \geqslant 0$ such that for any $\varepsilon > 0$ and $\lambda > 0$ such that $2\lambda(\mathtt{L}_y + (\alpha/\varepsilon) \max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon) - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \alpha \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$, there exists $C_{1,\varepsilon} \geqslant 0$ such that for any C convex compact*

with $\overline{\mathrm{B}}(0, R_{\mathsf{C}}) \subset \mathsf{C}$ *and* $R_{\mathsf{C}} > 0$*, there exists* $C_{2,\varepsilon,\mathsf{C}} \geqslant 0$ *such that for any* $h : \mathbb{R}^d \to \mathbb{R}$ *measurable with* $\sup_{x\in\mathbb{R}^d}\{|h(x)|\,(1+\|x\|^2)^{-1}\} \leqslant 1$*,* $n \in \mathbb{N}^*$*,* $\delta \in (0, \bar{\delta}]$ *and* $R > 0$ *we have*

$$\left|n^{-1}\sum_{k=1}^{n}\mathbb{E}\left[h(X_k)\right] - \int_{\mathbb{R}^d} h(\tilde{x})\mathrm{d}\pi(\tilde{x})\right| \leqslant \left\{C_0\varepsilon^{\beta\min(\alpha,1)/4} + C_{1,\varepsilon}R_{\mathsf{C}}^{-1} + C_{2,\varepsilon,\mathsf{C}}(\delta^{1/2} + \mathtt{M}_R + \exp[-R] + (n\delta)^{-1})\right\}(1+\|x\|^4)\,. \tag{B.18}$$

In addition, if there exists $\mathtt{m} > 0$ *such that* $\log(p(y|\cdot))$ *is* $\mathtt{m}$*-concave with* $\mathtt{m} \geqslant 2(\alpha/\varepsilon)\max(\mathtt{L}, 1+\mathtt{K}_\varepsilon/\varepsilon)$ *and* $\bar{\delta} = \mathtt{m}(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)^{-2}/2$*, then there exists* $C_{1,\varepsilon} \geqslant 0$ *such that for any* $h : \mathbb{R}^d \to \mathbb{R}$ *measurable with* $\sup_{x\in\mathbb{R}^d}\{|h(x)|\,(1+\|x\|^2)^{-1}\} \leqslant 1$*,* $n \in \mathbb{N}^*$*,* $\delta \in (0, \bar{\delta}]$ *and* $R > 0$ *we have*

$$\left|n^{-1}\sum_{k=1}^{n}\mathbb{E}\left[h(X_k)\right] - \int_{\mathbb{R}^d} h(\tilde{x})\mathrm{d}\pi(\tilde{x})\right| \leqslant C_0\varepsilon^{\beta\min(\alpha,1)/4} + C_{1,\varepsilon}(\delta^{1/2} + \mathtt{M}_R + \exp[-R] + (n\delta)^{-1})(1+\|x\|^4)\,. \tag{B.19}$$

Proof: In the general case where $\log(p(y|\cdot))$ is not assumed to be $\mathtt{m}$-concave with $\mathtt{m} > 0$, the proof is completed upon combining Proposition 5.3.4, Proposition B.4.2 and the fact that for any probability distribution ν_1, ν_2, $\|\nu_1 - \nu_2\|_V \leqslant \|\nu_1 - \nu_2\|_{\mathrm{TV}}^{1/2}\,(\nu_1[V^2] + \nu_2[V^2])^{1/2}$. The proof is similar in the case where $\log(p(y|\cdot))$ is $\mathtt{m}$-concave upon replacing Proposition 5.3.4 by Proposition B.3.3. ■

B.5 Technical results

In this section, we gather technical results which will be used throughout our analysis. Let $b \in \mathrm{C}(\mathbb{R}^d, \mathbb{R}^d)$ such that for any $x \in \mathbb{R}^d$, the following Stochastic Differential Equation admits a unique strong solution

$$\mathrm{d}\mathbf{X}_t = b(\mathbf{X}_t)\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbf{B}_t\,, \tag{B.20}$$

where $(\mathbf{B}_t)_{t\geqslant 0}$ is a d-dimensional Brownian motion and $\mathbf{X}_0 = x$. In this case, (B.20) defines a Markov semi-group $(\mathrm{P}_t)_{t\geqslant 0}$ for any $x \in \mathbb{R}^d$ and $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ by $\mathrm{P}_t(x, \mathsf{A}) = \mathbb{P}(\mathbf{X}_t \in \mathsf{A})$ where $(\mathbf{X}_t)_{t\geqslant 0}$ is the solution of (B.20) with $\mathbf{X}_0 = x$. Consider now the generator of $(\mathrm{P}_t)_{t\geqslant 0}$, defined for any $f \in \mathrm{C}^2(\mathbb{R}^d, \mathbb{R})$ by

$$\mathcal{A}f = \langle \nabla f, b(x)\rangle + \Delta f\,. \tag{B.21}$$

We say that a Markov semi-group $(\mathrm{P}_t)_{t\geqslant 0}$ on $\mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d)$ with extended infinitesimal generator $(\mathcal{A}, \mathrm{D}(\mathcal{A}))$ (see e.g. Meyn and Tweedie (1993) for the definition of $(\mathcal{A}, \mathrm{D}(\mathcal{A}))$) satisfies a continuous drift condition $\mathbf{D}_{\mathrm{c}}(W, \zeta, \beta)$ if there exist $\zeta > 0$, $\beta \geqslant 0$ and a measurable function $W : \mathbb{R}^d \to [1, +\infty)$ with $W \in \mathrm{D}(\mathcal{A})$ such that for all $x \in \mathbb{R}^d$

$$\mathcal{A}W(x) \leqslant -\zeta W(x) + \beta\,. \tag{B.22}$$

Similarly, we consider the Markov chain $(X_k)_{k\in\mathbb{N}}$ given by the following recursion for any $k \in \mathbb{N}$ and $x \in \mathbb{R}^d$

$$X_{k+1} = X_k + \gamma b(X_k) + \sqrt{2\gamma}Z_k\,, \tag{B.23}$$

with $X_0 = x$, $\gamma > 0$ and $\{Z_k : k \in \mathbb{N}\}$ a family of i.i.d Gaussian random variables with zero mean and identity covariance matrix. We define its associated Markov kernel $\mathrm{R}_\gamma : \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0,1]$ as follows for any $x \in \mathbb{R}^d$ and $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$

$$\mathrm{R}_\gamma(x, \mathsf{A}) = \int_{\mathbb{R}^d} \mathbb{1}_{\mathsf{A}}(x + \gamma b(x) + \sqrt{2\gamma} z) \exp[-\|z\|^2/2] \mathrm{d}z \; . \tag{B.24}$$

We say that R_γ satisfies a discrete drift condition $\mathbf{D}_{\mathrm{d}}(W, \lambda, c)$ if there exist $\lambda \in [0,1)$, $c \geqslant 0$ and a measurable function $W : \mathbb{R}^d \to [1, +\infty)$ such that for all $x \in \mathbb{R}^d$

$$\mathrm{R}_\gamma W(x) \leqslant \lambda W(x) + c \; . \tag{B.25}$$

The following two lemmas are classical, see for instance (Bortoli et al., 2020, Lemma 18, Lemma 19). We recall these results and their proofs for the sake of completeness.

Lemma B.5.1 *Assume that there exist* $\mathtt{L}, \mathtt{c} \geqslant 0$ *and* $\mathtt{m} > 0$ *such that for any* $x_1, x_2 \in \mathbb{R}^d$ *we have*

$$\langle b(x_1), x_1 \rangle \leqslant -\mathtt{m} \|x_1\|^2 + \mathtt{c} \; , \qquad \|b(x_1) - b(x_2)\| \leqslant \mathtt{L} \|x_1 - x_2\| \; . \tag{B.26}$$

Let $\bar{\gamma} = \mathtt{m}/\mathtt{L}^2$. *Then the following results hold:*

(a) For any $\varpi \in \mathbb{N}^*$ *there exist* $\lambda \in (0,1]$, $c, \beta \geqslant 0$ *and* $\zeta > 0$ *such that for any* $\gamma \in (0, \bar{\gamma}]$, R_γ *satisfies* $\mathbf{D}_{\mathrm{d}}(W, \lambda^\gamma, c\gamma)$ *and* $(\mathrm{P}_t)_{t \geqslant 0}$ *satisfies* $\mathbf{D}_{\mathrm{c}}(W, \zeta, \beta)$ *with* $W(x) = 1 + \|x\|^{2\varpi}$.

(b) For any $\varpi > 0$, *there exist* $\lambda \in (0,1]$, $c, \beta \geqslant 0$ *and* $\zeta > 0$ *such that for any* $\gamma \in (0, \bar{\gamma}]$, R_γ *satisfies* $\mathbf{D}_{\mathrm{d}}(W, \lambda^\gamma, c\gamma)$ *and* $(\mathrm{P}_t)_{t \geqslant 0}$ *satisfies* $\mathbf{D}_{\mathrm{c}}(W, \zeta, \beta)$ *with* $W(x) = \exp[\varpi \sqrt{1 + \|x\|^2}]$.

Proof: We divide the proof into two parts.

(a) Let $\varpi \in \mathbb{N}^*$ and $\gamma \in (0, \bar{\gamma}]$ with $\bar{\gamma} = \mathtt{m}/(4\mathtt{L}^2)$. Let $\mathcal{T}_\gamma(x) = x - \gamma b(x)$. In the sequel, for any $k \in \{1, \ldots, \varpi\}$, $c, \tilde{c}_k \geqslant 0$ and $\lambda, \tilde{\lambda}_k \in [0,1)$ are constants independent of γ which may take different values at each appearance. Let $\varepsilon \in (0, 1/2)$. Using (B.26), the fact that for any $a, b \geqslant 0$, $(a+b)^2 \leqslant (1+\varepsilon) a^2 + (1+\varepsilon^{-1}) b^2$ and the fact that for any $a, b \geqslant 0$ we have $(a+b)^{1/2} \leqslant a^{1/2} + b^{1/2}$, we get that for any $x \in \mathbb{R}^d$ with $\|x\| \geqslant (2\mathtt{c}/(\varepsilon \mathtt{m}))^{1/2}$

$$\begin{aligned}
\|\mathcal{T}_\gamma(x)\| &= \left(\|x\|^2 + 2\gamma \langle b(x), x \rangle + \gamma^2 \|b(x)\|^2 \right)^{1/2} && \text{(B.27)} \\
&\leqslant \left((1 - 2\gamma\mathtt{m} + (1+\varepsilon)\gamma^2 \mathtt{L}^2) \|x\|^2 + 2\gamma\mathtt{c} + (1+\varepsilon^{-1})\gamma^2 \|b(0)\|^2 \right)^{1/2} && \text{(B.28)} \\
&\leqslant \left((1 - \gamma\mathtt{m} + (1+\varepsilon)\gamma^2 \mathtt{L}^2) \|x\|^2 + (1+\varepsilon^{-1})\gamma^2 \|b(0)\|^2 \right)^{1/2} && \text{(B.29)} \\
&\leqslant \exp[-\gamma((2-\varepsilon)\mathtt{m} - (1+\varepsilon)\mathtt{L}^2\bar{\gamma})/2] \|x\| + (1 + \varepsilon^{-1/2})\gamma \|b(0)\| \; . && \text{(B.30)}
\end{aligned}$$

Note that $(2-\varepsilon)\mathtt{m} - (1+\varepsilon)\mathtt{L}^2\bar{\gamma} < 0$ since $\varepsilon \in (0, 1/2)$ and $\bar{\gamma} = \mathtt{m}/\mathtt{L}^2$. On the other hand using (B.26) and the fact that for any $a, b \geqslant 0$ with $a \geqslant b$ and $\mathrm{e}^a - \mathrm{e}^b \leqslant \mathrm{e}^a(a-b)$, we have for any $x \in \mathbb{R}^d$ with $\|x\| \leqslant (2\mathtt{c}/(\varepsilon\mathtt{m}))^{1/2}$

$$\begin{aligned}
\|\mathcal{T}_\gamma(x)\| &\leqslant (1 + \gamma\mathtt{L}) \|x\| + \gamma \|b(0)\| && \text{(B.31)} \\
&\leqslant \exp[-\gamma((2-\varepsilon)\mathtt{m} - (1+\varepsilon)\mathtt{L}^2\bar{\gamma})/2] \|x\| && \text{(B.32)} \\
&\quad + (2\mathtt{c}/(\varepsilon\mathtt{m}))^{1/2} \left\{ \exp[\gamma\mathtt{L}] - \exp[-\gamma((2-\varepsilon)\mathtt{m} - (1+\varepsilon)\mathtt{L}^2\bar{\gamma})/2] \right\} + \gamma \|b(0)\| && \text{(B.33)} \\
&\leqslant \exp[-\gamma((2-\varepsilon)\mathtt{m} - (1+\varepsilon)\mathtt{L}^2\bar{\gamma})/2] \|x\| + \gamma(2\mathtt{c}/(\varepsilon\mathtt{m}))^{1/2} \exp[\bar{\gamma}\mathtt{L}](\mathtt{L} + 2\mathtt{m}) + \gamma \|b(0)\| \; . && \text{(B.34)}
\end{aligned}$$

Combining (B.27) and (B.34), there exist $\lambda \in [0,1)$ and $c \geqslant 0$ such that for any $\gamma \in (0,\bar{\gamma}]$ and $x \in \mathbb{R}^d$,

$$\|\mathcal{T}_\gamma(x)\| \leqslant \lambda^\gamma \|x\| + \gamma c \,. \tag{B.35}$$

Note that using (B.35), for any $k \in \{1,\dots,2\varpi\}$ there exist $\tilde{\lambda}_k \in (0,1)$ and $\tilde{c}_k \geqslant 0$ such that

$$\begin{aligned}
\|\mathcal{T}_\gamma(x)\|^k &\leqslant \{\tilde{\lambda}_k^\gamma \|x\| + \gamma \tilde{c}_k\}^k && \text{(B.36)}\\
&\leqslant \tilde{\lambda}_k^{\gamma k} \|x\|^k + \gamma 2^k \max(\tilde{c}_k,1)^k \max(\bar{\gamma},1)^{k-1}\{1+\|x\|^{k-1}\} && \text{(B.37)}\\
&\leqslant \tilde{\lambda}_k^\gamma \|x\|^k + \tilde{c}_k\gamma\{1+\|x\|^{k-1}\} \leqslant (1+\|x\|^k)(1+\tilde{c}_k\gamma) \,. && \text{(B.38)}
\end{aligned}$$

Therefore, combining (B.36) and the Cauchy-Schwarz inequality we obtain that for any $\gamma \in (0,\bar{\gamma}]$ and $x \in \mathbb{R}^d$

$$\begin{aligned}
&\int_{\mathbb{R}^d} (1+\|y\|^{2\varpi}) \mathrm{R}_\gamma(x,\mathrm{d}y) = 1 + \mathbb{E}[(\|\mathcal{T}_\gamma(x)\|^2 + 2\sqrt{2\gamma}\langle \mathcal{T}_\gamma(x), Z\rangle + 2\gamma\|Z\|^2)^\varpi] && \text{(B.39)}\\
&= 1 + \sum_{k=0}^{\varpi}\sum_{\ell=0}^{k} \binom{\varpi}{k}\binom{k}{\ell} \|\mathcal{T}_\gamma(x)\|^{2(\varpi-k)} 2^{(3k-\ell)/2} \gamma^{(k+\ell)/2} \mathbb{E}[\langle \mathcal{T}_\gamma(x), Z\rangle^{k-\ell} \|Z\|^{2\ell}] && \text{(B.40)}\\
&\leqslant 1 + \|\mathcal{T}_\gamma(x)\|^{2\varpi} && \text{(B.41)}\\
&\quad + 2^{3\varpi/2} \sum_{k=1}^{\varpi}\sum_{\ell=0}^{k} \binom{\varpi}{k}\binom{k}{\ell} \|\mathcal{T}_\gamma(x)\|^{2(\varpi-k)} \gamma^{(k+\ell)/2} \mathbb{E}[\langle \mathcal{T}_\gamma(x), Z\rangle^{k-\ell} \|Z\|^{2\ell}] \mathbb{1}_{\{(1,0)\}^c}(k,\ell) && \text{(B.42)}\\
&\leqslant 1 + \|\mathcal{T}_\gamma(x)\|^{2\varpi} && \text{(B.43)}\\
&\quad + \gamma 2^{3\varpi/2} \sum_{k=1}^{\varpi}\sum_{\ell=0}^{k} \binom{\varpi}{k}\binom{k}{\ell} \|\mathcal{T}_\gamma(x)\|^{2\varpi-k-\ell} \bar{\gamma}^{(k+\ell)/2-1} \mathbb{E}[\|Z\|^{k+\ell}] \mathbb{1}_{\{(1,0)\}^c}(k,\ell) && \text{(B.44)}\\
&\leqslant 1 + \tilde{\lambda}_{2\varpi}^\gamma \|x\|^{2\varpi} + \tilde{c}_{2\varpi}\gamma\{1+\|x\|^{2\varpi-1}\} && \text{(B.45)}\\
&\quad + \gamma 2^{3\varpi/2} 2^{2\varpi} \max(\bar{\gamma},1)^{2\varpi} \sup_{k\in\{1,\dots,\varpi\}} \{(1+\tilde{c}_k\bar{\gamma})\mathbb{E}[\|Z\|^k]\}(1+\|x\|^{2\varpi-1}) && \text{(B.46)}\\
&\leqslant 1 + \lambda^\gamma \|x\|^{2\varpi} + \gamma c(1+\|x\|^{2\varpi-1}) && \text{(B.47)}\\
&\leqslant \lambda^{\gamma/2}(1+\|x\|^{2\varpi}) + \gamma c(1+\|x\|^{2\varpi-1}) + \lambda^\gamma(1+\|x\|^{2\varpi}) - \lambda^{\gamma/2}(1+\|x\|^{2\varpi}) \,. && \text{(B.48)}
\end{aligned}$$

Using that $\lambda^\gamma - \lambda^{\gamma/2} \leqslant -\log(1/\lambda)\gamma\lambda^{\gamma/2}/2$, we get that for any $\gamma \in (0,\bar{\gamma}]$, R_γ satisfies $\mathbf{D}_{\mathrm{d}}(W, \lambda^\gamma, c\gamma)$. We now show that there exist $\zeta > 0$ and $\beta \geqslant 0$ such that $(\mathrm{P}_t)_{t\geqslant 0}$ satisfies $\mathbf{D}_{\mathrm{c}}(W,\zeta,\beta)$. First, for any $x \in \mathbb{R}^d$ we have

$$\nabla W(x) = 2\varpi \|x\|^{2(\varpi-1)} x \,, \qquad \Delta W(x) = 2\varpi(2\varpi-1)\|x\|^{2(\varpi-1)} \tag{B.49}$$

Combining this result, the Cauchy-Schwarz inequality and (B.26), we obtain that for any $x \in \mathbb{R}^d$

$$\begin{aligned}
\mathcal{A}W(x) &= \langle \nabla W(x), b(x)\rangle + \Delta W(x) && \text{(B.50)}\\
&\leqslant -2\mathtt{m}\varpi \|x\|^{2\varpi} + 2\varpi c \|x\|^{2\varpi-1} + 2\varpi(2\varpi-1)\|x\|^{2(\varpi-1)} && \text{(B.51)}\\
&\leqslant -\mathtt{m}\varpi \|x\|^{2\varpi} + \sup_{x\in\mathbb{R}^d}\{2\varpi(c+2\varpi-1)\|x\|^{2\varpi-1} - \mathtt{m}\varpi\|x\|^{2\varpi}\} && \text{(B.52)}\\
&\leqslant -\mathtt{m}\varpi W(x) + \sup_{x\in\mathbb{R}^d}\{2\varpi(c+2\varpi-1)\|x\|^{2\varpi-1} - \mathtt{m}\varpi\|x\|^{2\varpi}\} + \mathtt{m}\varpi \,. && \text{(B.53)}
\end{aligned}$$

Hence letting $\zeta = \mathtt{m}\varpi$ and $\beta = \sup_{x\in\mathbb{R}^d}\{2\varpi(c+2\varpi-1)\|x\|^{2\varpi-1} - \mathtt{m}\varpi\|x\|^{2\varpi}\} + \mathtt{m}\varpi$, we obtain that $(\mathrm{P}_t)_{t\geqslant 0}$ satisfies $\mathbf{D}_{\mathrm{c}}(W,\zeta,\beta)$.

(b) First, we show that for any $\gamma \in (0,\bar{\gamma}]$, R_γ satisfies $\mathbf{D}_{\mathrm{d}}(\Phi,\lambda^\gamma,c)$, where $\Phi(x) = (1+\|x\|^2)^{1/2} = W_2^{1/2}(x)$ and $W_2(x) = 1+\|x\|^2$. Using the first part of the proof, there exist $\lambda_0 \in [0,1)$ and $c_0 \geqslant 0$ such that for any $\gamma \in (0,\bar{\gamma}]$ with $\bar{\gamma} = \mathtt{m}/(4\mathtt{L}^2)$ we have that R_γ satisfies $\mathbf{D}_{\mathrm{d}}(W_2,\lambda_0^\gamma,c_0\gamma)$. Using Jensen's inequality we obtain that for any $\gamma \in (0,\bar{\gamma}]$ and $x \in \mathbb{R}^d$ with $\|x\| \geqslant R$ and $R = \max(1,((2c_0\lambda_0^{-\bar{\gamma}})/\log(1/\lambda_0))^{1/2})$ we have

$$\mathrm{R}_\gamma\Phi(x) \leqslant (\mathrm{R}_\gamma W_2(x))^{1/2} \leqslant \exp[(\gamma/2)\{\log(\lambda_0) + \lambda_0^{-\bar{\gamma}}c_0R^{-2}\}]\Phi(x) \leqslant \lambda_0^{\gamma/4}\Phi(x)\;. \tag{B.54}$$

In addition, using that for any $a,b \geqslant 0$ with $a \geqslant b$ we have $\mathrm{e}^a - \mathrm{e}^b \leqslant \mathrm{e}^a(b-a)$, we get for any $x \in \mathbb{R}^d$ with $\|x\| \leqslant R$

$$\begin{aligned}\mathrm{R}_\gamma\Phi(x) \leqslant (\mathrm{R}_\gamma W_2(x))^{1/2} &\leqslant \exp[(\gamma/2)\{\log(\lambda_0) + \lambda_0^{-\bar{\gamma}}c_0\}]\Phi(x) &\text{(B.55)}\\ &\leqslant \exp[(\gamma/2)\{\log(\lambda_0) + \lambda_0^{-\bar{\gamma}}c_0R^{-2}\}]\Phi(x) &\text{(B.56)}\\ &\quad + \lambda_0^{-\bar{\gamma}}c_0(1-R^{-2})\exp[(\gamma/2)\{\log(\lambda_0) + \lambda_0^{-\bar{\gamma}}c_0R^{-2}\}]\Phi(R)\;. &\text{(B.57)}\end{aligned}$$

Hence, there exist $\lambda_1 \in [0,1)$ and $c_1 \geqslant 0$ such that for any $\gamma \in (0,\bar{\gamma}]$ we have that R_γ satisfies $\mathbf{D}_{\mathrm{d}}(\varpi\Phi,\lambda_1^\gamma,c_1\gamma)$. Now let $W(x) = \exp[\Phi(x)]$. Using the logarithmic Sobolev inequality (Boucheron et al., 2013, Theorem 5.5) we get for any $\gamma \in (0,\bar{\gamma}]$ and $x \in \mathbb{R}^d$ with $\|x\| \geqslant R$ and $R = 1 + (\varpi^2+c_1)^{-1}\log(1/\lambda_1)$

$$\begin{aligned}\mathrm{R}_\gamma W(x) \leqslant \exp[\mathrm{R}_\gamma\varpi\Phi(x) + \gamma\varpi^2] &\leqslant \exp[-(1-\lambda_1^\gamma)\Phi(x) + \gamma(\varpi^2+c_1)]W(x) &\text{(B.58)}\\ &\leqslant \exp[-\gamma\log(1/\lambda_1)R + \gamma(\varpi^2+c_1)]W(x) \leqslant \lambda_1^\gamma W(x)\;. &\text{(B.59)}\end{aligned}$$

In addition, using that for any $a,b \geqslant 0$ with $a \geqslant b$ we have $\mathrm{e}^a - \mathrm{e}^b \leqslant \mathrm{e}^a(b-a)$, we get for any $x \in \mathbb{R}^d$ with $\|x\| \leqslant R$

$$\begin{aligned}\mathrm{R}_\gamma W(x) &\leqslant \exp[\mathrm{R}_\gamma\varpi\Phi(x) + \gamma] \leqslant \exp[\gamma(\varpi^2+c_1)]W(x) &\text{(B.60)}\\ &\leqslant \lambda_1^\gamma W(x) + \gamma\exp[\bar{\gamma}(\varpi^2+c_1)]((1+c_1) + \log(1/\lambda_1))W(R)\;. &\text{(B.61)}\end{aligned}$$

Therefore, there exist $\lambda \in [0,1)$ and $c \geqslant 0$ such that for any $\gamma \in (0,\bar{\gamma}]$ we have that R_γ satisfies $\mathbf{D}_{\mathrm{d}}(W,\lambda^\gamma,c\gamma)$. We now show that there exist $\zeta > 0$ and $\beta \geqslant 0$ such that $(\mathrm{P}_t)_{t\geqslant 0}$ satisfies $\mathbf{D}_{\mathrm{c}}(W,\zeta,\beta)$. First, for any $x \in \mathbb{R}^d$ we have

$$\nabla W(x) = \varpi x\Phi^{-1}(x)W(x)\;, \quad \Delta W(x) = \{\varpi\Phi^{-1}(x)(1-\|x\|^2/\Phi^2(x)) + \varpi^2\|x\|^2/\Phi^2(x)\}W(x)\;. \tag{B.62}$$

Therefore using (B.26) we obtain that for any $x \in \mathbb{R}^d$ with $\|x\| \geqslant \sqrt{2}(1+(c+1+\varpi)/\mathtt{m})$

$$\mathcal{A}W(x) \leqslant \varpi(-\mathtt{m}\Phi^{-1}(x)\|x\|^2 + c + 1 + \varpi)W(x) \leqslant -(\mathtt{m}/2)W(x)\;, \tag{B.63}$$

which concludes the proof.

■

Lemma B.5.2 *Assume that there exist $\lambda \in (0,1]$, $c,\beta \geqslant 0$, $\zeta,\bar{\gamma} > 0$ such that for any $\gamma \in (0,\bar{\gamma}]$, R_γ satisfies $\mathbf{D}_{\mathrm{d}}(W,\lambda^\gamma,c\gamma)$ and $(\mathrm{P}_t)_{t\geqslant 0}$ satisfies $\mathbf{D}_{\mathrm{c}}(W,\zeta,\beta)$. Then, there exists $C \geqslant 0$ such that for any $x \in \mathbb{R}^d$, $t \geqslant 0$ and $k \in \mathbb{N}^*$ we have*

$$\mathrm{R}_\gamma^k W(x) + \mathrm{P}_t W(x) \leqslant CW(x)\;. \tag{B.64}$$

Proof: There exists $C_c \geqslant 0$ such that for any $x \in \mathbb{R}^d$ and $t \geqslant 0$, $\mathrm{P}_t W(x) \leqslant C_c W(x)$ using (Bortoli and Durmus, 2020, Lemma 25-(b)). Using that for any $t \geqslant 0$, $(1-\mathrm{e}^{-t})^{-1} \leqslant 1+1/t$ we get that for any $\gamma \in (0, \bar{\gamma}]$, $x \in \mathbb{R}^d$ and $k \in \mathbb{N}^*$

$$\mathrm{R}_\gamma^k W(x) \leqslant W(x) + c\gamma \sum_{k \in \mathbb{N}} \lambda^{k\gamma} \leqslant (1 + c(\bar{\gamma} + \log(1/\lambda)))W(x) \;, \tag{B.65}$$

which concludes the proof upon letting $C = C_c + 1 + c(\bar{\gamma} + \log(1/\lambda)$. ■

Proposition B.5.1 *Assume that there exist* $\Phi_1 : \mathbb{R}^d \to [0, +\infty)$ *and* $\Phi_2 : \mathbb{R}^m \to [0, +\infty)$ *such that for any* $x \in \mathbb{R}^d$ *and* $y_1, y_2 \in \mathbb{R}^m$

$$\|\log(q_{y_1}(x)) - \log(q_{y_2}(x))\| \leqslant (\Phi_1(x) + \Phi_2(y_1) + \Phi_2(y_2)) \, \|y_1 - y_2\| \;, \tag{B.66}$$

and for any $c > 0$, $\int_{\mathbb{R}^d} (1 + \Phi_1(\tilde{x})) \exp[c\Phi_1(\tilde{x})] p(x) \mathrm{d}x < +\infty$. *Then* $y \mapsto \pi_y$ *is locally Lipschitz w.r.t the total variation* $\|\cdot\|_{\mathrm{TV}}$, *where for any* $x \in \mathbb{R}^d, y \in \mathbb{R}^m$ *we have*

$$(\mathrm{d}\pi_y / \mathrm{dLeb})(x) = q_y(x) p(x) \Big/ \int_{\mathbb{R}^d} q_y(\tilde{x}) p(\tilde{x}) \mathrm{d}\tilde{x} \;. \tag{B.67}$$

Proof: Let $y_1, y_2 \in \mathsf{K}$ with K a compact set. Let $y_0 \in \mathsf{K}$ and D_{K} be the diameter of K. Using Lemma B.8.2 we get that

$$\|\pi_{y_1} - \pi_{y_2}\|_{\mathrm{TV}} \leqslant 2 c_{y_1} \int_{\mathbb{R}^d} |q_{y_1}(x) - q_{y_2}(x)| \, p(x) \mathrm{d}x \;, \tag{B.68}$$

with $c_{y_1} = \int_{\mathbb{R}^d} q_{y_1}(x) p(x) \mathrm{d}x$. Combining this result with the fact that for any $a, b \in \mathbb{R}$ we have $\left|\mathrm{e}^a - \mathrm{e}^b\right| \leqslant |a-b| \max(\mathrm{e}^a, \mathrm{e}^b)$ we get that

$$\begin{aligned}
\|\pi_{y_1} - \pi_{y_2}\|_{\mathrm{TV}} &\leqslant 2 c_{y_1} \int_{\mathbb{R}^d} |q_{y_1}(x) - q_{y_2}(x)| \, p(x) \mathrm{d}x & \text{(B.69)} \\
&\leqslant 2 c_{y_1} \int_{\mathbb{R}^d} (\Phi_1(x) + \Phi_2(y_1) + \Phi_2(y_2)) \, \|y_1 - y_2\| & \text{(B.70)} \\
&\qquad \times \exp[(2\Phi_1(x) + \Phi_2(y_1) + \Phi_2(y_0) + \Phi_2(y_2)) D_{\mathsf{K}}] p(x) \mathrm{d}x & \text{(B.71)} \\
&\leqslant 2 c_{y_1} (\Phi_2(y_1) + \Phi_2(y_2)) \exp[\Phi_2(y_1) + \Phi_2(y_0) + \Phi_2(y_2)] & \text{(B.72)} \\
&\qquad \times \int_{\mathbb{R}^d} (1 + \Phi_1(x)) \exp[2 D_{\mathsf{K}} \Phi_1(x)] p(x) \mathrm{d}x \times \|y_1 - y_2\| \;, & \text{(B.73)}
\end{aligned}$$

which concludes the proof. ■

B.6 Proofs of Section 5.3.2

We recall that the Markov chain $(X_k)_{k \in \mathbb{N}}$, defined in (B.6), is given by

$$\begin{aligned}
X_{k+1} &= X_k + \delta b_\varepsilon(X_k) + \sqrt{2\delta} Z_{k+1} \;, & \text{(B.74)} \\
b_\varepsilon(x) &= \nabla \log(p(y|x)) + \alpha(D_\varepsilon(x) - x)/\varepsilon + (x - \Pi_{\mathsf{C}}(x))/\lambda \;, & \text{(B.75)}
\end{aligned}$$

where $\delta > 0$ is a stepsize, $\alpha, \varepsilon, \lambda > 0$ are hyperparameters of the algorithm, $\mathsf{C} \subset \mathbb{R}^d$ is a closed convex set with $0 \in \mathsf{C}$, Π_{C} is the projection on C and $\{Z_k : k \in \mathbb{N}\}$ a family of i.i.d. Gaussian random variables with zero mean and identity covariance matrix.

In this section, we prove the convergence of PnP-ULA and control the bias of its invariant measure in the general framework introduced in Appendix B.2 (*i.e.* $\alpha \neq 1$) under two different assumptions on the posterior: either the posterior is log-concave as in Appendix B.3 or the posterior satisfies a more general one-sided Lipschitz condition as in Section 5.3.2. Note that in Section 5.3.2 the results are only stated for $\alpha = 1$. The statements of the propositions can be generalized to $\alpha > 0$ by replacing $2\lambda(\mathtt{L}_y + \mathtt{L}/\varepsilon - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$ by $2\lambda(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)^{-1}$ in Proposition 5.3.2 and $2\lambda(\mathtt{L}_y + (a/\varepsilon)\max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon) - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$ by $2\lambda(\mathtt{L}_y + (\alpha/\varepsilon)\max(\mathtt{L}, 1 + \mathtt{K}_\varepsilon/\varepsilon) - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)^{-1}$ in Proposition 5.3.3 and Proposition 5.3.4.

B.6.1 Proof of Proposition 5.3.1

Let $R > 0$. Let X and Z be random variables with distribution μ and zero mean Gaussian with identity covariance matrix. Let $X_\varepsilon = X + \varepsilon^{1/2} Z$. We recall that the distributions of X and X_ε have density with respect to the Lebesgue measure given by p and p_ε respectively. In addition, the conditional density of X given X_ε is given by g_ε. By definition $D_\varepsilon^\star(X_\varepsilon) = \mathbb{E}[X|X_\varepsilon]$ and therefore we have

$$\ell_\varepsilon(w^\dagger) = \mathbb{E}\left[\|X - f_{w^\dagger}(X_\varepsilon)\|^2\right] \tag{B.76}$$

$$= \mathbb{E}\left[\|X - D_\varepsilon^\star(X_\varepsilon)\|^2\right] + 2\mathbb{E}\left[\langle X - D_\varepsilon^\star(X_\varepsilon), D_\varepsilon^\star(X_\varepsilon) - f_{w^\dagger}(X_\varepsilon)\rangle\right] + \mathbb{E}\left[\|f_{w^\dagger}(X_\varepsilon) - D_\varepsilon^\star(X_\varepsilon)\|^2\right] \tag{B.77}$$

$$= \mathbb{E}\left[\|X - D_\varepsilon^\star(X_\varepsilon)\|^2\right] + \mathbb{E}\left[\|f_{w^\dagger}(X_\varepsilon) - D_\varepsilon^\star(X_\varepsilon)\|^2\right] = \ell_\varepsilon^\star + \mathbb{E}\left[\|f_{w^\dagger}(X_\varepsilon) - D_\varepsilon^\star(X_\varepsilon)\|^2\right] . \tag{B.78}$$

Combining this result, the condition that $\ell_\varepsilon(w^\dagger) \leqslant \ell_\varepsilon^\star + \eta$ and the Cauchy-Schwarz inequality we get that

$$\mathbb{E}[\|f_{w^\dagger}(X_\varepsilon) - D_\varepsilon^\star(X_\varepsilon)\|] \leqslant \sqrt{\eta} . \tag{B.79}$$

Since $f_{w^\dagger}$ and $D_\varepsilon^\star$ are locally Lipschitz, there exists $C_R \geqslant 0$ such that for any $x_1, x_2 \in \overline{\mathrm{B}}(0, 2R)$ we have

$$|\|f_{w^\dagger}(x_2) - D_\varepsilon^\star(x_2)\| - \|f_{w^\dagger}(x_1) - D_\varepsilon^\star(x_1)\|| \leqslant C_R \|x_2 - x_1\| . \tag{B.80}$$

Assume that $\sup_{\tilde{x} \in \overline{\mathrm{B}}(0,R)} \|f_{w^\dagger}(\tilde{x}) - D_\varepsilon^\star(\tilde{x})\| > \eta^\varpi$ with $\varpi = (2d+2)^{-1}$ and denote $x_R \in \overline{\mathrm{B}}(0, R)$ such that we have $\sup_{\tilde{x} \in \overline{\mathrm{B}}(0,R)} \|f_{w^\dagger}(\tilde{x}) - D_\varepsilon^\star(x)\| = \|f_{w^\star}(x_R) - D_\varepsilon^\star(x_R)\|$. Using (B.80) we have

$$\mathbb{E}[\|f_{w^\dagger}(X_\varepsilon) - D_\varepsilon^\star(X_\varepsilon)\|] \geqslant \int_{\overline{\mathrm{B}}(0,2R) \cap \overline{\mathrm{B}}(x_R, C_R^{-1}\eta^\varpi)} \|f_{w^\dagger}(\tilde{x}) - D_\varepsilon^\star(\tilde{x})\| p_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} \tag{B.81}$$

$$\geqslant (\|f_{w^\dagger}(x_R) - D_\varepsilon^\star(x_R)\| - \eta^\varpi) \int_{\overline{\mathrm{B}}(0,2R) \cap \overline{\mathrm{B}}(x_R, C_R^{-1}\eta^\varpi)} p_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} . \tag{B.82}$$

Combining this result and (B.79) we obtain that

$$\|f_{w^\dagger}(x_R) - D_\varepsilon^\star(x_R)\| \leqslant \eta^{1/2} \left(\int_{\overline{\mathrm{B}}(0,2R) \cap \overline{\mathrm{B}}(x_R, C_R^{-1}\eta^\varpi)} p_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} \right)^{-1} + \eta^\varpi , \tag{B.83}$$

Setting $\mathtt{M}_R = \eta^{1/2}(\int_{\overline{\mathrm{B}}(0,2R)\cap\overline{\mathrm{B}}(x_R,C_R^{-1}\eta^{\varpi})} p_\varepsilon(\tilde{x})\mathrm{d}\tilde{x})^{-1} + \eta^{\varpi}$ concludes the first part of the proof. Denote v_d the volume of the unit d-dimensional ball. We have that $\mathrm{Leb}(\overline{\mathrm{B}}(x_R, C_R^{-1}\eta^{\varpi})) = C_R^{-d}\eta^{\varpi d} v_d$. Using the Fubini theorem, the Lebesgue differentiation theorem (Bogachev, 2007, Theorem 5.6.2), the dominated convergence theorem and the fact that for $\eta \in (0, (C_R R)^{1/\varpi}]$, $\overline{\mathrm{B}}(0,2R) \cap \overline{\mathrm{B}}(x_R, C_R^{-1}\eta^{\varpi}) = \overline{\mathrm{B}}(x_R, C_R^{-1}\eta^{\varpi})$ we get that

$$\lim_{\eta\to 0} \mathrm{Leb}(\overline{\mathrm{B}}(x_R, C_R^{-1}\eta^{\varpi}))^{-1} \int_{\mathbb{R}^d} \mathbb{1}_{\overline{\mathrm{B}}(x_R,C_R^{-1}\eta^{\varpi})\cap\overline{\mathrm{B}}(x_R,C_R^{-1}\eta^{\varpi})}(x) p_\varepsilon(x)\mathrm{d}x \tag{B.84}$$

$$= \lim_{\eta\to 0} \int_{\mathbb{R}^d} |\overline{\mathrm{B}}(x_R, C_R^{-1}\eta^{\varpi})|^{-1} (2\pi\varepsilon)^{-d/2} \int_{\mathbb{R}^d} \mathbb{1}_{\overline{\mathrm{B}}(x_R,C_R^{-1}\eta^{\varpi})}(x) \exp[-\|x-\tilde{x}\|^2/(2\varepsilon)] p(\tilde{x})\mathrm{d}x\mathrm{d}\tilde{x} \tag{B.85}$$

$$= \int_{\mathbb{R}^d} (2\pi\varepsilon)^{-d/2} \exp[-\|x_R-\tilde{x}\|^2/(2\varepsilon)] p(\tilde{x})\mathrm{d}x\mathrm{d}\tilde{x} = p_\varepsilon(x_R) > 0 \,. \tag{B.86}$$

Using this result we have,

$$\limsup_{\eta\to 0} \eta^{-\varpi}\mathtt{M}_R = 1 + \limsup_{\eta\to 0} \eta^{1/2-\varpi(d+1)}\eta^{\varpi d} \left(\int_{\overline{\mathrm{B}}(0,2R)\cap\overline{\mathrm{B}}(x_R,C_R^{-1}\eta^{\varpi})} p_\varepsilon(\tilde{x})\mathrm{d}\tilde{x} \right)^{-1} \tag{B.87}$$

$$= 1 + C_R^d v_d p_\varepsilon^{-1}(x_R) < +\infty \,, \tag{B.88}$$

which concludes the proof.

B.6.2 Proof of Proposition 5.3.2 and Proposition B.3.1

We divide this section into two parts. First, we prove the general case where $\log(p(y|\cdot))$ is not assumed to be strongly concave but only satisfying a one-sided Lipschitz condition, *i.e.* Proposition 5.3.2. Then we turn to the proof of Proposition B.3.1.

(a) Let $\lambda > 0$ such that $2\lambda(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. Let C be a compact convex set with $0 \in \mathsf{C}$. Using H2, (B.6) and that $\mathrm{Id} - \Pi_{\mathsf{C}}$ is non-expansive we have for any $x_1, x_2 \in \mathbb{R}^d$

$$\|b_\varepsilon(x_1) - b_\varepsilon(x_2)\| \leqslant (\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)\|x_1 - x_2\| \,. \tag{B.89}$$

Denote $R_{\mathsf{C}} = \sup\{\|x_1 - x_2\| : x_1, x_2 \in \mathsf{C}\}$. Using (B.6), the Cauchy-Schwarz inequality and that $2\lambda(\alpha\mathtt{L}/\varepsilon - \mathtt{m}) \leqslant 1$ we have for any $x_1, x_2 \in \mathbb{R}^d$

$$\langle b_\varepsilon(x_1) - b_\varepsilon(x_2), x_1 - x_2\rangle \leqslant (-\mathtt{m} + \alpha\mathtt{L}/\varepsilon)\|x_1 - x_2\|^2 - \|x_1 - x_2\|^2/\lambda + R_{\mathsf{C}}\|x_1 - x_2\|/\lambda \tag{B.90}$$

$$\leqslant -\|x_1 - x_2\|^2/(2\lambda) + R_{\mathsf{C}}\|x_1 - x_2\|/\lambda \,. \tag{B.91}$$

Hence, for any $x_1, x_2 \in \mathbb{R}^d$ with $\|x_1 - x_2\| \geqslant 4R_{\mathsf{C}}$ we obtain that $\langle b_\varepsilon(x_1) - b_\varepsilon(x_2), x_1 - x_2\rangle \leqslant -\|x_1 - x_2\|^2/(4\lambda)$. We also have that for any $x \in \mathbb{R}^d$

$$\langle b_\varepsilon(x), x\rangle \leqslant -\|x\|^2/(4\lambda) + \sup_{\tilde{x}\in\mathbb{R}^d} \left\{ (R_{\mathsf{C}}/\lambda + \|b(0)\|)\|\tilde{x}\| - \|\tilde{x}\|^2/(4\lambda) \right\} \,. \tag{B.92}$$

We conclude the proof of Proposition 5.3.2 upon using Lemma B.5.1, Lemma B.5.2, (Bortoli and Durmus, 2020, Corollary 2) with $\bar{\gamma} \leftarrow (4\lambda)^{-1}(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)^{-2} \geqslant \bar{\delta}$ and the fact that for any probability distribution ν_1, ν_2,

$$\|\nu_1 - \nu_2\|_V \leqslant \|\nu_1 - \nu_2\|_{\mathrm{TV}}^{1/2} (\nu_1[V^2] + \nu_2[V^2])^{1/2} \,. \tag{B.93}$$

(b) Using that $\log(p(y|\cdot))$ is $\mathtt{m}$-concave with $2\alpha\mathtt{L}/(\mathtt{m}\varepsilon) \leqslant 1$, we obtain that for any $x_1, x_2 \in \mathbb{R}^d$

$$\langle b_\varepsilon(x_1) - b_\varepsilon(x_2), x_1 - x_2\rangle \leqslant -\mathtt{m}\left\|x_1 - x_2\right\|^2/2 \,, \tag{B.94}$$

$$\left\|b_\varepsilon(x_1) - b_\varepsilon(x_2)\right\| \leqslant (\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)\left\|x_1 - x_2\right\| \,. \tag{B.95}$$

This concludes the proof of Proposition B.3.1 upon using (Bortoli and Durmus, 2020, Corollary 2) with $\bar{\gamma} \leftarrow \mathtt{m}(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon)^{-2} \geqslant \bar{\delta}$ and (B.93).

B.6.3 Proof of Proposition 5.3.3 and Proposition B.3.2

Before proving Proposition 5.3.3 and Proposition B.3.2, we show the following lemma which is a straightforward consequence of Girsanov's theorem (Liptser and Shiryaev, 2001, Theorem 7.7). A similar version of this lemma can be found in the proof of (Durmus and Moulines, 2017, Proposition 2).

Lemma B.6.1 *Let $T > 0$, $b_1, b_2 : [0, +\infty) \times \mathbb{R}^d \to \mathbb{R}^d$ measurable such that for any $i \in \{1, 2\}$ and $x \in \mathbb{R}^d$, $\mathrm{d}\mathbf{X}_t^{(i)} = b_i(t, \mathbf{X}_t^{(i)})\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbf{B}_t$ admits a unique strong solution with $\mathbf{X}_0^{(i)} = x$ with Markov semigroup $(\mathrm{P}_t^{(i)})_{t\geqslant 0}$ and where $(\mathbf{B}_t)_{t\geqslant 0}$ is a d-dimensional Brownian motion. In addition, assume that for any $x \in \mathbb{R}^d$ and $\mathbb{P}(\int_0^T \{\|b_i(t, \mathbf{X}_t^{(i)})\|^2 + \|b_i(t, \mathbf{B}_t)\|^2\}\mathrm{d}t < +\infty) = 1$. Let $V : \mathbb{R}^d \to [0, +\infty)$ measurable, then for any $x \in \mathbb{R}^d$ we have*

$$\begin{aligned}
&\left\|\delta_x \mathrm{P}_T^{(1)} - \delta_x \mathrm{P}_T^{(2)}\right\|_V \\
&\quad \leqslant \left(\delta_x \mathrm{P}_t^{(1)}[V^2] + \delta_x \mathrm{P}_t^{(2)}[V^2]\right)^{1/2} \left(\int_0^T \mathbb{E}\left[\|b_1(t, \mathbf{X}_t^{(1)}) - b_2(t, \mathbf{X}_t^{(1)})\|^2\right] \mathrm{d}t\right)^{1/2} \,.
\end{aligned} \tag{B.96}$$

Proof: Let $T > 0$ and $x \in \mathbb{R}^d$. For any $i \in \{1, 2\}$, denote $\mu_{(i)}^x$ the distribution of $(\mathbf{X}_t^{(i)})_{t\in[0,T]}$ on the Wiener space $(\mathrm{C}([0, T], \mathbb{R}), \mathcal{B}(\mathrm{C}([0, T], \mathbb{R})))$ with $\mathbf{X}_0^{(i)} = x$. Similarly denote μ_B^x the distribution of $(\mathbf{B}_t)_{t\in[0,T]}$ witgh $\mathbf{B}_0 = x$. Using the generalized Pinsker inequality (Durmus and Moulines, 2017, Lemma 24) and the transfer theorem (Kullback, 1997, Theorem 4.1) we get that

$$\left\|\delta_x \mathrm{P}_T^{(1)} - \delta_x \mathrm{P}_T^{(2)}\right\|_V \leqslant \sqrt{2}\left(\delta_x \mathrm{P}_t^{(1)}[V^2] + \delta_x \mathrm{P}_t^{(2)}[V^2]\right)^{1/2} \mathrm{KL}^{1/2}(\mu_{(1)}|\mu_{(2)}) \,. \tag{B.97}$$

Since for any $i \in \{1, 2\}$ we have $\mathbb{P}(\int_0^T \{\|b_i(\mathbf{X}_t^{(i)})\|^2 + \|b_i(\mathbf{B}_t)\|^2\}\mathrm{d}t < +\infty) = 1$, we can apply Girsanov's theorem (Liptser and Shiryaev, 2001, Theorem 7.7) and μ_B-almost surely for any $w \in \mathrm{C}([0, T], \mathbb{R})$ we get

$$(\mathrm{d}\mu_{(1)}^x/\mathrm{d}\mu_B^x)((w_t)_{t\in[0,T]}) = \exp\left[(1/2)\int_0^T \langle b_1(w_t), \mathrm{d}w_t\rangle - (1/4)\int_0^T \|b_1(w_t)\|^2 \,\mathrm{d}t\right] \,, \tag{B.98}$$

$$(\mathrm{d}\mu_B^x/\mathrm{d}\mu_{(2)}^x)((w_t)_{t\in[0,T]}) = \exp\left[-(1/2)\int_0^T \langle b_2(w_t), \mathrm{d}w_t\rangle + (1/4)\int_0^T \|b_2(w_t)\|^2 \,\mathrm{d}t\right] \,. \tag{B.99}$$

Hence, we obtain that

$$\mathrm{KL}(\mu_{(1)}^x|\mu_{(2)}^x) = \mathbb{E}\left[\log((\mathrm{d}\mu_{(1)}^x/\mathrm{d}\mu_{(2)}^x)(\mathbf{X}_t^{(1)}))\right] = (1/4)\int_0^T \mathbb{E}\left[\left\|b_1(\mathbf{X}_t^{(1)}) - b_2(\mathbf{X}_t^{(2)})\right\|^2\right] \mathrm{d}t \,, \tag{B.100}$$

which concludes the proof. ■

In the following lemma, we show that under H6, $\nabla \log(p_\varepsilon)$ is Lipschitz continuous.

Lemma B.6.2 *Assume* H6. *Then for any* $x_1, x_2 \in \mathbb{R}^d$ *we have*

$$\|\nabla \log(p_\varepsilon(x_1)) - \nabla \log(p_\varepsilon(x_2))\| \leqslant (1 + \mathtt{K}_\varepsilon/\varepsilon) \|x_1 - x_2\| / \varepsilon \,. \tag{B.101}$$

Reciprocally, if there $x \mapsto \nabla \log(p_\varepsilon(x))$ *is Lipschitz-continuous then* H6.

Proof: Let $\varepsilon > 0$. We recall that for any $x \in \mathbb{R}^d$ we have

$$p_\varepsilon(x) = \int_{\mathbb{R}^d} \exp[-\|x - \tilde{x}\|^2 / (2\varepsilon)] p(\tilde{x}) \mathrm{d}\tilde{x} \,. \tag{B.102}$$

Using the dominated convergence theorem we obtain that $\log(p_\varepsilon) \in \mathrm{C}^\infty(\mathbb{R}^d, \mathbb{R})$. In particular we have for any $x \in \mathbb{R}^d$

$$\nabla^2 \log(p_\varepsilon(x)) = -\varepsilon^{-1} \operatorname{Id} + \varepsilon^{-2} \int_{\mathbb{R}^d} (x - \tilde{x})^{\otimes 2} g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} - \varepsilon^{-2} \left(\int_{\mathbb{R}^d} (x - \tilde{x}) g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} \right)^{\otimes 2} \tag{B.103}$$

$$= -\varepsilon^{-1} \operatorname{Id} + \varepsilon^{-2} \int_{\mathbb{R}^d} \left(\tilde{x} - \int_{\mathbb{R}^d} \tilde{x}' g_\varepsilon(\tilde{x}'|x) \mathrm{d}\tilde{x}' \right)^{\otimes 2} g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} \tag{B.104}$$

Therefore, using H6 we obtain that for any $x \in \mathbb{R}^d$ we have

$$\|\nabla^2 \log(p_\varepsilon(x))\|_2 \leqslant \varepsilon^{-1} + \varepsilon^{-2} \mathtt{K}_\varepsilon \,, \tag{B.105}$$

which concludes the first part of the proof. Reciprocally, since $x \mapsto \nabla \log(p_\varepsilon(x))$ is Lipschitz-continuous with constant $\mathtt{K} \geqslant 0$ we get that for any basis vector $(\mathrm{e}_i)_{i \in \{1,\dots,d\}}$ we have that $\mathrm{e}_i^\top \nabla^2 \log(p_\varepsilon(x)) \mathrm{e}_i \leqslant \mathtt{K}$. Combining this result with (B.103), we get that

$$\varepsilon^{-2} \int_{\mathbb{R}^d} \left\| \tilde{x} - \int_{\mathbb{R}^d} \tilde{x}' g_\varepsilon(\tilde{x}'|x) \mathrm{d}\tilde{x}' \right\|^2 g_\varepsilon(\tilde{x}|x) \mathrm{d}\tilde{x} \leqslant \mathtt{K} d + \varepsilon^{-1} d \,, \tag{B.106}$$

which concludes the proof. ■

In what follows we prove Proposition 5.3.3. The proof of Proposition B.3.2 is similar and left to the reader.

Proof of Proposition 5.3.3 Let $\lambda > 0$ such that $2\lambda(\mathtt{L}_y + \alpha \mathtt{L}/\varepsilon - \mathtt{m}) \leqslant 1$ and $\bar{\delta} = (1/3)(\mathtt{L}_y + \alpha \mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. We divide the proof into two parts. First, we show that for any C convex compact with $0 \in \mathsf{C}$ there exists $B_{1,\mathsf{C}} \geqslant 0$ such that for any $\delta \in (0, \bar{\delta}]$ and $R > 0$

$$\|\pi_{\varepsilon,\delta} - \tilde{\pi}_\varepsilon\|_V \leqslant B_{1,\mathsf{C}}(\delta^{1/2} + \mathtt{M}_R + \exp[-R]) \,, \tag{B.107}$$

with $\tilde{\pi}_\varepsilon$ given by

$$(\mathrm{d}\tilde{\pi}_\varepsilon / \mathrm{dLeb})(x) \propto \exp[-d^2(x, \mathsf{C})/(2\lambda)] p(y|x) p_\varepsilon^\alpha(x) \,, \tag{B.108}$$

Second, we show that there exists $B_0 \geqslant 0$ such that for any C convex compact with $0 \in \mathsf{C}$

$$\|\pi_\varepsilon - \tilde{\pi}_\varepsilon\|_V \leqslant B_0 \operatorname{diam}^{-1/4}(\mathsf{C}) \,, \tag{B.109}$$

which concludes the proof upon using the triangle inequality.

(a) Let C convex compact with $0 \in \mathsf{C}$. We introduce $(\bar{\mathbf{X}}_t)_{t\geqslant 0}$ solution of the following Stochastic Differential Equation (SDE): $\bar{\mathbf{X}}_0 = X_0$ and

$$\mathrm{d}\bar{\mathbf{X}}_t = \bar{b}_\varepsilon(\bar{\mathbf{X}}_t)\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbf{B}_t \;, \tag{B.110}$$

$$\bar{b}_\varepsilon(x) = \nabla \log(p(y|x)) + \alpha \nabla \log(p_\varepsilon(x)) + \mathrm{prox}_\lambda(\iota_{\mathsf{C}})(x) \;, \tag{B.111}$$

with $(\mathbf{B}_t)_{t\geqslant 0}$ a d-dimensional Brownian motion. $\bar{b}_\varepsilon$ is Lipschitz continuous using Lemma B.6.2, hence this SDE admits a unique strong solution for any initial condition $\mathbf{X}_0$ with $\mathbb{E}[\|\mathbf{X}_0\|^2] < +\infty$, see (Karatzas and Shreve, 1991, Chapter 5, Theorem 2.9). We denote by $(\mathrm{P}_{t,\varepsilon})_{t\geqslant 0}$ the semigroup associated with the strong solutions of (B.110). Similarly to the proof of Proposition B.3.1, replacing (Bortoli and Durmus, 2020, Corollary 2) by (Bortoli and Durmus, 2020, Corollary 22), there exist $\tilde{A}_{\mathsf{C}} \geqslant 0$ and $\tilde{\rho}_{\mathsf{C}} \in [0,1)$ such that that for any $x_1, x_2 \in \mathbb{R}^d$ and $t \geqslant 0$

$$\|\delta_{x_1}\mathrm{P}_{t,\varepsilon} - \delta_{x_2}\mathrm{P}_{t,\varepsilon}\|_V \leqslant \tilde{A}_{\mathsf{C}}\tilde{\rho}_{\mathsf{C}}^t(V^2(x_1) + V^2(x_2)) \;, \tag{B.112}$$

$$\mathbf{W}_1(\delta_{x_1}\mathrm{P}_{t,\varepsilon}, \delta_{x_2}\mathrm{P}_{t,\varepsilon}) \leqslant \tilde{A}_{\mathsf{C}}\tilde{\rho}_{\mathsf{C}}^t \|x_1 - x_2\| \;. \tag{B.113}$$

Combining (B.112), Proposition B.3.1, the fact that $(\mathscr{P}_1(\mathbb{R}^d), \mathbf{W}_1)$ is a complete metric space and the Picard fixed point theorem we obtain that for any $\delta \in (0, \bar{\delta}]$ there exist $\pi_{\varepsilon,\delta}, \tilde{\pi}_\varepsilon \in \mathscr{P}_1(\mathbb{R}^d)$ such that $\pi_{\varepsilon,\delta}\mathrm{R}_{\varepsilon,\delta,\mathsf{C}} = \pi_{\varepsilon,\delta}$ and for any $t \geqslant 0$, $\tilde{\pi}_\varepsilon \mathrm{P}_{t,\varepsilon} = \tilde{\pi}_\varepsilon$. Note that by (Roberts et al., 1996, Theorem 2.1) we have for any $x \in \mathbb{R}^d$

$$(\mathrm{d}\tilde{\pi}_\varepsilon/\mathrm{dLeb})(x) \propto \exp[-d^2(x,\mathsf{C})/(2\lambda)]p(y|x)p_\varepsilon^\alpha(x) \;, \tag{B.114}$$

since $\mathrm{prox}_\lambda(\iota_{\mathsf{C}}) = \nabla d^2(\cdot, \mathsf{C})/(2\lambda)$. Let $f : \mathbb{R}^d \to \mathbb{R}$ measurable and such that for any $x \in \mathbb{R}^d$, $|f(x)| \leqslant V(x)$. Let $m \in \mathbb{N}^*$ such that $m \geqslant \bar{\delta}^{-1}$, $x \in \mathbb{R}^d$ and $k \in \mathbb{N}$ we have

$$\left\|\delta_x \mathrm{R}_{\varepsilon,1/m}^{km}[f] - \delta_x \mathrm{P}_{km,\varepsilon}^{km}[f]\right\| = \left\|\sum_{j=0}^{k-1} \delta_x \mathrm{R}_{\varepsilon,1/m}^{jm}(\mathrm{R}_{\varepsilon,1/m}^m - \mathrm{P}_{1,\varepsilon})\mathrm{P}_{k-j-1,\varepsilon}[f]\right\| \tag{B.115}$$

Using (B.112), Lemma B.5.1 and Lemma B.5.2 there exists $B_a \geqslant 0$ such that for any $x \in \mathbb{R}^d$ and $k \in \mathbb{N}$ we have

$$\|\delta_x \mathrm{P}_{k,\varepsilon,\mathsf{C}}[f] - \tilde{\pi}_\varepsilon[f]\| \leqslant B_a \tilde{\rho}_{\mathsf{C}}^k V^2(x) \;. \tag{B.116}$$

Let $T = 1$, $b_1(t, (w_t)_{t\in[0,T]}) = \sum_{j=0}^{m-1} \mathbb{1}_{[j/m,(j+1)/m)}(t) b_\varepsilon(w_{j\delta})$ and $b_2(t, (w_t)_{t\in[0,T]} = \bar{b}_\varepsilon(w_t)$. Let $\mathbf{X}_t^{(1)}$ and $\mathbf{X}_t^{(2)}$ the unique strong solution of $\mathrm{d}\mathbf{X}_t = b(t, (\mathbf{X}_t)_{t\in[0,1]}) + \sqrt{2}\mathbf{B}_t$ with $\mathbf{X}_0 = x$ with $x \in \mathbb{R}^d$ and $b = b_1$, respectively $b = b_2$. Note that $(\mathbf{X}_t^{(2)})_{t\geqslant 0} = (\bar{\mathbf{X}}_t)_{t\geqslant 0}$ and $(\mathbf{X}_{k/m}^{(1)}) = (X_k)_{k\in\mathbb{N}}$. For any $i \in \{1,2\}$, denote $\mathrm{P}_t^{(i)}$ the Markov semigroup associated with $\mathbf{X}_t^{(i)}$. For any $x \in \mathbb{R}^d$ we have

$$\left\|\delta_x \mathrm{R}_{\varepsilon,1/m,\mathsf{C}}^m - \delta_x \mathrm{P}_{1,\varepsilon,\mathsf{C}}\right\|_{\mathrm{TV}} = \left\|\delta_x \mathrm{P}_1^{(1)} - \delta_x \mathrm{P}_1^{(2)}\right\|_{\mathrm{TV}} \;. \tag{B.117}$$

Using H2(R) and the fact that for any $a, b \geqslant 0$, $(a+b)^2 \leqslant 2(a^2+b^2)$, we have for any $t \in [j/m, (j+1)/m)$, $j \in \{0, \ldots, m-1\}$ and $(w_t)_{t\in[0,1]} \in \mathrm{C}([0,1], \mathbb{R}^d)$

$$\left\|b_1(t, (w_t)_{t\in[0,1]}) - b_2(t, (w_t)_{t\in[0,1]})\right\|^2 = \left\|b_\varepsilon(w_{j/m}) - \bar{b}_\varepsilon(w_t)\right\|^2 \tag{B.118}$$

$$\leqslant 2\left\|b_\varepsilon(w_{j/m}) - b_\varepsilon(w_t)\right\|^2 + 2\left\|\bar{b}_\varepsilon(w_t) - b_\varepsilon(w_t)\right\|^2 \tag{B.119}$$

$$\leqslant 2\mathrm{L}_b^2\left\|w_{j/m} - w_t\right\|^2 + 4\alpha^2 \mathtt{M}_R^2/\varepsilon^2 + 4\alpha^2 \mathbb{1}_{\overline{\mathrm{B}}(0,R)^c}(\|w_t\|)/\varepsilon^2 \;, \tag{B.120}$$

where $\mathtt{L}_b$ is the Lipschitz constant associated with b_ε. In addition using Itô's isometry we have for any $t \in [j/m, (j+1)/m)$

$$\mathbb{E}[\|\mathbf{X}_t^{(1)} - \mathbf{X}_{j/m}^{(1)}\|^2] = 2\mathbb{E}[\| \int_{j/m}^t \mathrm{d}\mathbf{B}_t\|^2] \leqslant 2d\delta \; . \tag{B.121}$$

Finally, using Lemma B.5.1, Lemma B.5.2, the logarithmic Sobolev inequality (Boucheron et al., 2013, Theorem 5.5), the Cauchy-Schwarz inequality and the Markov inequality, there exists $\tilde{B}_b \geqslant 0$ such that for any $t \geqslant 0$ and $x \in \mathbb{R}^d$

$$\mathbb{P}(\|\mathbf{X}_t^{(1)}\| \geqslant R) \leqslant \exp[-2R]\mathbb{E}\left[\exp[2\|\mathbf{X}_t^{(1)}\|\right] \tag{B.122}$$

$$\leqslant \exp[-2R]\mathbb{E}^{1/2}\left[\exp\left[4\sqrt{2}\| \int_{\ell_t/m}^t \mathrm{d}\mathbf{B}_t\|\right]\right] \mathbb{E}^{1/2}\left[\exp[4\|X_{\ell_t}\|\right] \tag{B.123}$$

$$\leqslant \tilde{B}_b \exp[-2R] \exp[2\Phi(x)] \; , \tag{B.124}$$

where $\ell_t = \lfloor tm \rfloor$ and $\Phi(x) = \sqrt{1 + \|x\|^2}$. Combining this result, (B.120), (B.117), (B.121) and Lemma B.6.1, we obtain that there exists $B_b \geqslant 0$ such that for any $x \in \mathbb{R}^d$ and $R > 0$

$$\left\|\delta_x \mathrm{R}_{1/m,\mathsf{C}}^m - \delta_x \mathrm{P}_{1,\mathsf{C}}\right\|_V \leqslant 2B_b(\sqrt{\delta} + \mathtt{M}_R + \exp[-R])(1 + \|x\|^4) \exp[\Phi(x)] \tag{B.125}$$

$$\leqslant 48 B_b(\sqrt{\delta} + \mathtt{M}_R + \exp[-R]) \exp[2\Phi(x)] \; , \tag{B.126}$$

Combining this result and (B.116) we obtain that for any $k \in \mathbb{N}$, $j \in \{0, \ldots, k-1\}$, $x \in \mathbb{R}^d$ and $R > 0$ we have

$$\left|(\delta_x \mathrm{R}_{1/m,\mathsf{C}}^m - \delta_x \mathrm{P}_{1,\mathsf{C}})\mathrm{P}_{k-j-1,\mathsf{C}}[f]\right| \leqslant B_a B_b(\sqrt{\delta} + \mathtt{M}_R + \exp[-R])\rho_{\mathsf{C}}^{k-j-1} \exp[2\Phi(x)] \; . \tag{B.127}$$

Using this result, Lemma B.5.1, Lemma B.5.2 and (B.115) we obtain that there exists $B_c \geqslant 0$ such that for any $m \in \mathbb{N}^*$ with $m^{-1} \geqslant \bar{\delta}$

$$\left\|\pi_{\varepsilon,1/m,\mathsf{C}} - \tilde{\pi}_\varepsilon\right\|_V \leqslant \limsup_{k \to +\infty} \left\|\delta_0 \mathrm{R}_{\varepsilon,1/m,\mathsf{C}}^{km} - \delta_0 \mathrm{P}_{km,\varepsilon,\mathsf{C}}^{km}\right\|_V \leqslant B_c(\sqrt{\delta} + \mathtt{M}_R + \exp[-R]) \; . \tag{B.128}$$

The proof in the general case where $\delta \in (0, \bar{\delta}]$ is similar and we obtain that there exists $B_c \geqslant 0$ such that for any $\delta \in (0, \bar{\delta}]$

$$\|\pi_{\varepsilon,\delta} - \tilde{\pi}_\varepsilon\|_V \leqslant B_c(\sqrt{\delta} + \mathtt{M}_R + \exp[-R]) \; . \tag{B.129}$$

(b) For any C compact convex with $0 \in \mathsf{C}$ we define $\tilde{\pi}_\varepsilon$ and $\rho_{\varepsilon,\mathsf{C}}$ such that for any $x \in \mathbb{R}^d$

$$\rho_{\varepsilon,\mathsf{C}}(x) = \exp[-d^2(x,\mathsf{C})/(2\lambda)] p(y|x) p_\varepsilon^\alpha(x) \; , \qquad (\mathrm{d}\tilde{\pi}_\varepsilon/\mathrm{dLeb})(x) = \rho_{\varepsilon,\mathsf{C}}(x) \Big/ \int_{\mathbb{R}^d} \rho_{\varepsilon,\mathsf{C}}(\tilde{x}) \mathrm{d}\tilde{x} \; . \tag{B.130}$$

Similarly, define ρ_ε and π_ε such that for any $x \in \mathbb{R}^d$

$$\rho_\varepsilon(x) = p(y|x) p_\varepsilon^\alpha(x) \; , \qquad (\mathrm{d}\pi_\varepsilon/\mathrm{dLeb})(x) = \rho_\varepsilon(x) \Big/ \int_{\mathbb{R}^d} \rho_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x} \; . \tag{B.131}$$

Since for any $x \in \mathbb{R}^d$, $\rho_{\varepsilon,\mathsf{C}}(x) \leqslant \rho_\varepsilon(x)$ we get $\int_{\mathbb{R}^d} \rho_{\varepsilon,\mathsf{C}}(\tilde{x}) \mathrm{d}\tilde{x} \leqslant \int_{\mathbb{R}^d} \rho_\varepsilon(\tilde{x}) \mathrm{d}\tilde{x}$. Hence we obtain using the Cauchy-Schwarz inequality and the Markov inequality

$$\mathrm{KL}(\pi_\varepsilon | \pi_{\mathsf{C}}) \leqslant \int_{\mathbb{R}^d} \log(\rho_\varepsilon(\tilde{x})/\rho_{\varepsilon,\mathsf{C}}(\tilde{x})) \mathrm{d}\pi_\varepsilon(\tilde{x}) \tag{B.132}$$

$$\leqslant \int_{\mathsf{C}^c} \|\tilde{x}\|^2 \, \mathrm{d}\pi_\varepsilon(\tilde{x}) \leqslant \mathbb{P}^{1/2}(X \notin \mathsf{C}) \, \mathbb{E}^{1/2}[\|X\|^4] \leqslant \mathbb{E}[\|X\|^4] R_{\mathsf{C}}^{-2} \; . \tag{B.133}$$

with X a random variable with distribution π_ε. We conclude using the generalized Pinsker inequality (Durmus and Moulines, 2017, Lemma 24).

■

B.7 Proofs of Section 5.3.3

B.7.1 Proof of Proposition 5.3.5

Let $\alpha, \lambda, \varepsilon, \bar{\delta} > 0$, $\delta \in (0, \bar{\delta}]$ and $\mathsf{C} \subset \mathbb{R}^d$ convex and compact with $0 \in \mathsf{C}$. For any $x_1, x_2 \in \mathbb{R}^d$ we have

$$\|b_\varepsilon(x_1) - b_\varepsilon(x_2)\| \leqslant (\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon) \|x_1 - x_2\| \ . \tag{B.134}$$

Denote $(X_n, Y_n)_{n\in\mathbb{N}}$ the Markov chain obtained using the coupling described in (Bortoli and Durmus, 2020, Section 3) with initial condition $(x_1, x_2) \in \mathsf{C}$. Using (Bortoli and Durmus, 2020, Corollary 7-(b)) we get that for any $\ell \in \mathbb{N}$

$$\mathbb{E}\left[\mathbb{1}_{\Delta^c_{\mathbb{R}^d}}(X_{(\ell+1)\lceil 1/\delta\rceil}, Y_{(\ell+1)\lceil 1/\delta\rceil})\right] \leqslant (1-\beta)\mathbb{E}\left[\mathbb{1}_{\Delta^c_{\mathbb{R}^d}}(X_{\ell\lceil 1/\delta\rceil}, Y_{\ell\lceil 1/\delta\rceil})\right] \ , \tag{B.135}$$

where $\Delta_{\mathbb{R}^d} = \{(x, x) \ : \ x \in \mathbb{R}^d\}$ and $\beta \in (0, 1)$ with

$$\beta = 2\{-(1+\bar{\delta})(1 + \mathtt{L}_y + (\alpha\mathtt{L}/\varepsilon))\mathrm{diam}(\mathsf{C})\} \ , \tag{B.136}$$

where is the cumulative distribution function of the univariate Gaussian distribution with zero mean and unit variance. In addition, using that the coupling is absorbing, we have that for any $k \in \mathbb{N}$,

$$\mathbb{E}\left[\mathbb{1}_{\Delta^c_{\mathbb{R}^d}}(X_k, Y_k)\right] \leqslant \mathbb{E}\left[\mathbb{1}_{\Delta^c_{\mathbb{R}^d}}(X_{\lfloor k/\lceil 1/\delta\rceil\rfloor\lceil 1/\delta\rceil}, Y_{\lfloor k/\lceil 1/\delta\rceil\rfloor\lceil 1/\delta\rceil})\right] \ , \tag{B.137}$$

Combining this result and (B.135), we get that for any $k \in \mathbb{N}$

$$\left\|\delta_{x_1}\mathrm{Q}^k_{\varepsilon,\delta} - \delta_{x_2}\mathrm{Q}^k_{\varepsilon,\delta}\right\|_{\mathrm{TV}} \leqslant \mathbb{E}\left[\mathbb{1}_{\Delta^c_{\mathbb{R}^d}}(X_k, Y_k)\right] \leqslant (1-\beta)^{\lfloor k/\lceil 1/\delta\rceil\rfloor} \ . \tag{B.138}$$

Using that $\lfloor k/\lceil 1/\delta\rceil\rfloor \geqslant k\delta/(1+\delta) - 1$ concludes the proof upon letting $\tilde{\rho}_{\mathsf{C}} = (1-\beta)^{1/(1+\bar{\delta})}$ and $\tilde{A}_{\mathsf{C}} = (1-\beta)^{-1}$.

B.7.2 Proof of Proposition 5.3.6

Let $\alpha, \lambda > 0$, $\varepsilon \in (0, \varepsilon_0]$ such that $2\lambda(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon - \min(\mathtt{m}, 0)) \leqslant 1$ and $\bar{\delta}_1 = (1/3)(\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda)^{-1}$. Recall that for any $x_1, x_2 \in \mathbb{R}^d$

$$\|b_\varepsilon(x_1) - b_\varepsilon(x_2)\| \leqslant (\mathtt{L}_y + \alpha\mathtt{L}/\varepsilon + 1/\lambda) \|x_1 - x_2\| \ . \tag{B.139}$$

Using this result, the fact that for any $x \in \mathbb{R}^d$, $\langle b_\varepsilon(x), x\rangle \leqslant -\tilde{\mathtt{m}}\|x\|^2 + c$ and (Douc et al., 2019, Theorem 19.4.1) there exist $\bar{\delta}_2 > 0$, $\tilde{B} \geqslant 0$ and $\tilde{\rho} \in (0, 1]$ such that for any $\delta \in (0, \bar{\delta}_2]$, $x \in \mathbb{R}^d$ and $k \in \mathbb{N}$

$$\left\|\delta_x\mathrm{R}^k_{\varepsilon,\delta} - \pi_{\varepsilon,\delta}\right\|_V + \left\|\delta_x\mathrm{Q}^k_{\varepsilon,\delta} - \pi^{\mathsf{C}}_{\varepsilon,\delta}\right\|_V \leqslant \tilde{B}\tilde{\rho}^{k\delta}V(x) \ , \tag{B.140}$$

with $\tilde{B}$ and $\tilde{\rho}$ which do not depend on R. In addition, using Lemma B.5.1, for any $k \in \mathbb{N}$ and $\delta \in (0, \bar{\delta}_2]$ we have

$$\mathrm{R}^k_{\varepsilon,\delta} V(x) \leqslant \tilde{\lambda}^{k\delta} V(x) + \tilde{c}\delta \;, \tag{B.141}$$

with $\tilde{\lambda} \in [0,1)$ and $\tilde{c} > 0$ which do not depend on $R \geqslant 0$. For any $\delta \in (0, \bar{\delta}_2]$ we have

$$\lambda^\delta + c\delta \leqslant \lambda^\delta (1 + c\delta\lambda^{-\bar{\delta}_2}) \leqslant (\lambda \exp[c\lambda^{-\bar{\delta}_2}])^\delta \;. \tag{B.142}$$

Let $A = \lambda \exp[c\lambda^{-\bar{\delta}_2}]$, we have that for any $x \in \mathbb{R}^d$, $\mathrm{R}_{\varepsilon,\delta} V(x) \leqslant A^\delta V(x)$. Therefore we get that $(V(X_n)A^{-n})_{n\in\mathbb{N}}$ is a supermartingale. Hence using Doob maximal inequality and Markov inequality we get that

$$\mathbb{P}\left(\sup_{k\in\{0,\dots,n\}} \|X_k\| \geqslant R \right) \leqslant V(x) A^{n\delta} \exp[-R] \;. \tag{B.143}$$

Therefore, we get that for any $k \in \mathbb{N}$

$$\left\| \pi_{\varepsilon,\delta} - \pi^{\mathsf{C}}_{\varepsilon,\delta} \right\|_{\mathrm{TV}} \leqslant (V(0) + \tilde{c}\bar{\delta}_2) A^{k\delta} \exp[-R] + \tilde{B}\tilde{\rho}^{k\delta} V(0) \;. \tag{B.144}$$

We conclude upon letting $k = \lfloor r/(2\log(A)\delta) \rfloor$.

B.8 Proofs of Appendix B.4

B.8.1 Proof of Proposition B.4.1

The first part of the proposition is straightforward. Using Pinsker's inequality (Boucheron et al., 2013, Theorem 4.19) we have for any $x \in \mathbb{R}^d$

$$\|\mu - (\tau_x)_\# \mu\|^2_{\mathrm{TV}} \leqslant 2\mathrm{KL}((\tau_x)_\#\mu|\mu) \leqslant 2 \textstyle\int_{\mathbb{R}^d} \|U(\tilde{x}+x) - U(\tilde{x})\| \,\mathrm{d}\mu(\tilde{x}) \leqslant 2C_\gamma \|x\|^\gamma \;. \tag{B.145}$$

For the second part of the proof, since there exist $c_1, \varpi > 0$ and $c_2 \in \mathbb{R}$ such that for any $x \in \mathbb{R}^d$, $U(x) \geqslant c_1 \|x\|^\varpi + c_2$ then for any $k \in \mathbb{N}^*$ and $\alpha > 0$, $\int_{\mathbb{R}^d} (1+\|x\|)^k p(x) < +\infty$. Let $q(x) = (1+\|x\|)^{-(d+1)} / \int_{\mathbb{R}^d} (1+\|\tilde{x}\|)^{-(d+1)} \mathrm{d}\tilde{x}$. Then using that for any $t \geqslant 0$, $|\mathrm{e}^t - 1| \leqslant |t| \, \mathrm{e}^{|t|}$ we get that for any $x \in \mathbb{R}^d$

$$\begin{aligned}
&\int_{\mathbb{R}^d} |p(\tilde{x}) - p(x-\tilde{x})| \, q^{1-1/\alpha}(\tilde{x}) \mathrm{d}\tilde{x} \\
&\leqslant C_\gamma \|x\|^\gamma \exp[C_\gamma \|x\|^\gamma] \int_{\mathbb{R}^d} (1+\|\tilde{x}\|)^{(d+1)(1/\alpha-1)} p(\tilde{x}) \mathrm{d}\tilde{x} \left(\int_{\mathbb{R}^d} (1+\|\tilde{x}\|)^{-(d+1)} \mathrm{d}\tilde{x} \right)^{1-1/\alpha} ,
\end{aligned} \tag{B.146}$$

which concludes the proof.

B.8.2 Proof of Proposition B.4.2

First we show the following technical lemma.

Lemma B.8.1 *For any $x, y \geqslant 0$ and $\beta > 0$, $(x+y)^\beta - x^\beta \leqslant 2^\beta (y^\beta + x^{(\beta-1)\wedge 0} y)$.*

Proof: The result is straightforward if $\beta \in (0,1]$, since in this case $(x+y)^\beta \leqslant x^\beta + y^\beta$. Assume that $\beta > 1$. If $x = 0$ the result holds. Now assume that $x > 0$. If $y \geqslant x$ then $(x+y)^\beta - x^\beta \leqslant 2^\beta y^\beta$. Assume that $y \leqslant x$. Since $\mathrm{f}: t \mapsto (1+t)^\beta - 1$ is convex we obtain that for any $t \in [0,1]$, $\mathrm{f}(t) \leqslant 2^\beta t$. Using this result we have

$$(x+y)^\beta - x^\beta \leqslant x^\beta f(y/x) \leqslant 2^\beta x^{\beta-1} y \;, \tag{B.147}$$

which concludes the proof. ■

Before proving Proposition B.4.2 we state the following lemma.

Lemma B.8.2 *Let π_1, π_2 two probability measures and $q_1, q_2 : \mathbb{R}^d \to [0, +\infty)$ two measurables functions such that for any $x \in \mathbb{R}^d$, $(\mathrm{d}\pi_i/\mathrm{dLeb})(x) = q_i(x)/c_i$ with $c_i = \int_{\mathbb{R}^d} q_i(\tilde{x})\mathrm{d}\tilde{x}$. Denote $\mathrm{D} = \int_{\mathbb{R}^d} |q_1(x) - q_2(x)|$. We have*

$$\|\pi_1 - \pi_2\|_{\mathrm{TV}} \leqslant 2c_1^{-1}\mathrm{D} \;. \tag{B.148}$$

Proof: We have

$$\|\pi_1 - \pi_2\|_{\mathrm{TV}} = \int_{\mathbb{R}^d} \left| \frac{q_1(x)}{c_1} - \frac{q_2(x)}{c_2} \right| \mathrm{d}x \leqslant c_1^{-1}(\mathrm{D} + |c_2 - c_1|) \;, \tag{B.149}$$

which concludes the proof using that $|c_2 - c_1| \leqslant \mathrm{D}$. ■

We now give the proof of Proposition B.4.2.

Proof: Let $\alpha > 0$. For any $\varepsilon > 0$ and $x \in \mathbb{R}^d$ denote $\bar{p}(x) = p(y|x)p^\alpha(x)$ and $\bar{p}_\varepsilon(x) = (py|x)p_\varepsilon^\alpha(x)$, where we recall that for any $x \in \mathbb{R}^d$

$$p_\varepsilon(x) = (2\pi\varepsilon)^{-d/2} \int_{\mathbb{R}^d} p(\tilde{x}) \exp[-\|x - \tilde{x}\|^2 / (2\varepsilon)] \mathrm{d}\tilde{x} \;. \tag{B.150}$$

For any $\varepsilon > 0$ we have

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant \|p(y|\cdot)\|_\infty \int_{\mathbb{R}^d} |p^\alpha(x) - p_\varepsilon^\alpha(x)| \, \mathrm{d}x \;. \tag{B.151}$$

Using Lemma B.8.1 and that $\|p_\varepsilon\|_\infty \leqslant \|p\|_\infty < +\infty$, we have for any $\varepsilon > 0$ and $x \in \mathbb{R}^d$

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant 2^\alpha \|p(y|\cdot)\|_\infty (1 + \|p\|_\infty^{(\alpha-1)\wedge 0}) \tag{B.152}$$

$$\times \left\{ \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, \mathrm{d}x + \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)|^\alpha \, \mathrm{d}x \right\} \;. \tag{B.153}$$

Using Jensen's inequality, for any $q : \mathbb{R}^d \to (0, +\infty)$ with $\int_{\mathbb{R}^d} q(\tilde{x})\mathrm{d}\tilde{x} = 1$ we have

$$\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)|^\alpha \, \mathrm{d}x \leqslant \left(\int_{\mathbb{R}^d} \left| p(x) - p_\varepsilon(x) q^{1-1/\alpha}(x) \right| \mathrm{d}x \right)^\alpha \;. \tag{B.154}$$

Combining this result with (B.152) we get that

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant 2^\alpha \|p(y|\cdot)\|_\infty (1 + \|p\|_\infty^{(\alpha-1)\wedge 0})$$

$$\times \left\{ \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, \mathrm{d}x + \left(\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{1-1/\alpha}(x) \mathrm{d}x \right)^\alpha \right\} \;. \tag{B.155}$$

If $\alpha \geqslant 1$, choosing q such that $\|q\|_\infty \leqslant 1$ we get

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant 2^\alpha \, \|p(y|\cdot)\|_\infty \, (1 + \|p\|_\infty^{(\alpha-1)\wedge 0}) \tag{B.156}$$

$$\times \left\{ \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, \mathrm{d}x + \left(\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, (x) \mathrm{d}x \right)^\alpha \right\} . \tag{B.157}$$

Hence since $p \in \mathrm{L}^1(\mathbb{R}^d)$ and $\{\tilde{x} \mapsto (2\pi\varepsilon)^{-d/2} \exp[-\|\tilde{x}\|^2/(2\varepsilon)] \ : \ \varepsilon > 0\}$ is a family of mollifiers, we have $\lim_{\varepsilon \to 0} \int_{\mathbb{R}^d} |p(x) - p_\varepsilon| \, \mathrm{d}x = 0$. Combining this result, (B.157) and Lemma B.8.2 concludes the first part of the proof.

Now let $\alpha > 0$ and assume H8(α). If $\alpha \geqslant 1$ then using (B.152) we have

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant 2^\alpha (1 + 2^{\alpha-1}) \, \|p(y|\cdot)\|_\infty \, (1 + \|p\|_\infty^{(\alpha-1)\wedge 0}) \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, \mathrm{d}x \, . \tag{B.158}$$

If $\alpha < 1$ then using that $\|q\|_\infty < +\infty$, we get that

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant 2^\alpha \, \|p(y|\cdot)\|_\infty \, (1 + \|q\|_\infty^{1/\alpha - 1})(1 + \|p\|_\infty^{(\alpha-1)\wedge 0})$$

$$\times \left\{ \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{1-1/\alpha}(x) \mathrm{d}x + \left(\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{1-1/\alpha}(x) \mathrm{d}x \right)^\alpha \right\} . \tag{B.159}$$

Hence, in any case, there exists $\tilde{C}_0 \geqslant 0$ such that

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)|$$

$$\leqslant \tilde{C}_0 \left\{ \int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{\min(1-1/\alpha, 0)}(x) \mathrm{d}x + \left(\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{\min(1-1/\alpha, 0)}(x) \mathrm{d}x \right)^\alpha \right\} . \tag{B.160}$$

Using Jensen's inequality and the change of variable $\tilde{x} \mapsto \varepsilon^{1/2}\tilde{x}$, we have for any $\varepsilon \in (0, (4\kappa)^{-1}]$

$$\int_{\mathbb{R}^d} |p(x) - p_\varepsilon(x)| \, q^{\min(1-1/\alpha, 0)}(x) \mathrm{d}x \tag{B.161}$$

$$\leqslant \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} |p(x) - p(x - \tilde{x})| \, q^{\min(1-1/\alpha, 0)}(x) (2\pi\varepsilon)^{-d/2} \exp[-\|\tilde{x}\|^2/(2\varepsilon)] \mathrm{d}x \mathrm{d}\tilde{x} \tag{B.162}$$

$$\leqslant \int_{\mathbb{R}^d} \exp[\kappa \|\tilde{x}\|^2] \, \|\tilde{x}\|^\beta \, (2\pi\varepsilon)^{-d/2} \exp[-\|\tilde{x}\|^2/(2\varepsilon)] \mathrm{d}\tilde{x} \tag{B.163}$$

$$\leqslant \varepsilon^{\beta/2} (2\pi)^{-d/2} \int_{\mathbb{R}^d} \exp[\kappa\varepsilon \|\tilde{x}\|^2] \, \|\tilde{x}\|^\beta \exp[-\|\tilde{x}\|^2/2] \mathrm{d}\tilde{x} \tag{B.164}$$

$$\leqslant \varepsilon^{\beta/2} (2\pi)^{-d/2} \int_{\mathbb{R}^d} \|\tilde{x}\|^\beta \exp[-\|\tilde{x}\|^2/4] \mathrm{d}\tilde{x} \leqslant C_0 \varepsilon^{\beta/2} \, , \tag{B.165}$$

with $C_0 = (2\pi)^{-d/2} \int_{\mathbb{R}^d} \|\tilde{x}\|^\beta \exp[-\|\tilde{x}\|^2/4] \mathrm{d}\tilde{x}$. Hence, we have

$$\int_{\mathbb{R}^d} |\bar{p}(x) - \bar{p}_\varepsilon(x)| \, \mathrm{d}x \leqslant C_1 (\varepsilon^{\beta/2} + \varepsilon^{\beta\alpha/2}) \, , \tag{B.166}$$

with $C_1 = \tilde{C}_0(C_0 + C_0^\alpha)$. Let $\varepsilon_1 = \min((cC_1)^{-2/\beta}/2, (cC_1)^{-2/(\beta\alpha)}/2, (4\kappa)^{-1})$ and $c = \int_{\mathbb{R}^d} \bar{p}(x)\mathrm{d}x$. Combining (B.166) with Lemma B.8.2, we get that for any $\varepsilon \in (0, \varepsilon_1]$

$$\|\pi - \pi_\varepsilon\|_{\mathrm{TV}} \leqslant 2c^{-1}C_1(\varepsilon^{\beta/2} + \varepsilon^{\beta\alpha/2}) \,, \tag{B.167}$$

which concludes the proof upon letting $A_0 = 2c^{-1}C_1$. ■

Bibliography

Introduction to markov random fields. In *Markov Random Fields for Vision and Image Processing*. The MIT Press, 2011. doi: 10.7551/mitpress/8579.003.0001. URL https://doi.org/10.7551/mitpress/8579.003.0001. [Cited on page 28.]

A. Abdulle, I. Almuslimani, and G. Vilmart. Optimal explicit stabilized integrator of weak order 1 for stiff and ergodic stochastic differential equations. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):937–964, 2018. doi: 10.1137/17M1145859. URL https://doi.org/10.1137/17M1145859. [Cited on page 87.]

J. Adler and O. Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 37(6):1322–1332, 2018a. doi: 10.1109/TMI.2018.2799231. [Cited on pages 6 and 15.]

J. Adler and O. Öktem. Deep bayesian inversion, 2018b. URL https://arxiv.org/abs/1811.05910. [Cited on page 33.]

M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*, 19(9):2345–2356, 2010. doi: 10.1109/TIP.2010.2047910. [Cited on pages 6 and 15.]

C. Aguerrebere. *On the Generation of High Dynamic Range Images: Theory and Practice from a Statistical Perspective*. Theses, Télécom ParisTech ; Universidad de la República, Uruguay, May 2014. URL https://tel.archives-ouvertes.fr/tel-01136641. [Cited on pages 22 and 115.]

C. Aguerrebere, A. Almansa, J. Delon, Y. Gousseau, and P. Muse. A Bayesian Hyperprior Approach for Joint Image Denoising and Interpolation, With an Application to HDR Imaging. *IEEE Transactions on Computational Imaging*, 3(4):633–646, dec 2017. ISSN 2333-9403. doi: 10.1109/TCI.2017.2704439. [Cited on page 28.]

R. Ahmad, C. A. Bouman, G. T. Buzzard, S. Chan, S. Liu, E. T. Reehorst, and P. Schniter. Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery. *IEEE signal processing magazine*, 37(1):105–116, 2020. [Cited on pages 35 and 36.]

S. Aja-Fernandez, T. Pie, G. Vegas-Sánchez-Ferrero, et al. Spatially variant noise estimation in mri: A homomorphic approach. *Medical image analysis*, 20(1):184–197, 2015. [Cited on page 115.]

G. Alain and Y. Bengio. What Regularized Auto-Encoders Learn from the Data-Generating Distribution. *Journal of Machine Learning Research*, 15:3743–3773, 2014. ISSN 1532-4435. URL http://jmlr.org/papers/v15/alain14a.html. [Cited on page 39.]

Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, and S. McLaughlin. Lidar waveform-based analysis of depth images constructed using sparse single-photon data. *IEEE Transactions on Image Processing*, 25(5):1935–1946, 2016. [Cited on page 115.]

Y. Altmann, R. Aspden, M. Padgett, and S. McLaughlin. A bayesian approach to denoising of single-photon binary images. *IEEE Transactions on Computational Imaging*, 3(3): 460–471, 2017a. [Cited on page 115.]

Y. Altmann, S. McLaughlin, and M. Padgett. Unsupervised restoration of subsampled images constructed from geometric and binomial data. In *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 1–5. IEEE, 2017b. [Cited on page 115.]

A. Andrle, N. Farchmin, P. Hagemann, S. Heidenreich, V. Soltwisch, and G. Steidl. Invertible neural networks versus mcmc for posterior reconstruction in grazing incidence x-ray fluorescence. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 528–539. Springer, 2021. [Cited on page 115.]

V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen. On instabilities of deep learning in image reconstruction and the potential costs of ai. *Proceedings of the National Academy of Sciences*, 117(48):30088–30095, 2020. [Cited on page 33.]

V. Antun, M. J. Colbrook, and A. C. Hansen. Can stable and accurate neural networks be computed?–on the barriers of deep learning and smale's 18th problem. *arXiv preprint arXiv:2101.08286*, 2021. [Cited on page 33.]

S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, 2019. doi: 10.1017/S0962492919000059. [Cited on pages 4, 14, and 33.]

F. Bach. Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681, 2017. [Cited on pages 45 and 69.]

J. M. Bardsley. Mcmc-based image reconstruction with uncertainty quantification. *SIAM J. Sci. Comput.*, 34, 2012. [Cited on pages 3 and 13.]

R. Bassett and J. Deride. Maximum a posteriori estimators as a limit of bayes estimators. *Mathematical Programming*, 174(1):129–144, 2019. [Cited on page 60.]

H. H. Bauschke, P. L. Combettes, et al. *Convex analysis and monotone operator theory in Hilbert spaces*, volume 408. Springer, 2011. [Cited on pages 35 and 70.]

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. doi: 10.1137/080716542. URL https://doi.org/10.1137/080716542. [Cited on page 26.]

Y. Bengio, L. Yao, G. Alain, and P. Vincent. Generalized denoising auto-encoders as generative models. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/559cb990c9dffd8675f6bc2186971dc2-Paper.pdf. [Cited on pages 4 and 14.]

J. Bernardo and A. Smith. *Bayesian Theory*, volume 15. 01 2000. ISBN 0 471 49464 X. doi: 10.2307/2983298. [Cited on pages 42, 62, and 63.]

M. Bertero, P. Boccacci, and V. Ruggiero. Inverse imaging with poisson data. *IOP Publish*, 2018. [Cited on page 115.]

S. A. Bigdeli and M. Zwicker. Image Restoration using Autoencoding Priors. Technical report, 2017. [Cited on pages 39 and 94.]

S. A. Bigdeli, M. Jin, P. Favaro, and M. Zwicker. Deep Mean-Shift Priors for Image Restoration. In *(NIPS) Advances in Neural Information Processing Systems 30*, pages 763–772, sep 2017. URL http://papers.nips.cc/paper/6678-deep-mean-shift-priors-for-image-restoration. [Cited on page 39.]

V. I. Bogachev. *Measure Theory*, volume Volume 1. Springer, 1 edition, 2007. ISBN 9783540345138,3-540-34513-2. URL http://gen.lib.rus.ec/book/index.php?md5=ffbdc7e3d8e571c6cd5f9e8633cdfdc2. [Cited on page 132.]

A. Bora, A. Jalal, E. Price, and A. G. Dimakis. Compressed sensing using generative models. In *(ICML) International Conference on Machine Learning*, volume 2, pages 537–546. JMLR. org, 2017. ISBN 9781510855144. [Cited on page 92.]

V. D. Bortoli and A. Durmus. Convergence of diffusions and their discretizations: from continuous to discrete processes and back, 2020. [Cited on pages 130, 132, 133, 135, and 137.]

V. D. Bortoli, A. Durmus, A. F. Vidal, and M. Pereyra. Maximum likelihood estimation of regularisation parameters in high-dimensional inverse problems: an empirical bayesian approach. part ii: Theoretical analysis, 2020. [Cited on page 127.]

S. Boucheron, G. Lugosi, and P. Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013. [Cited on pages 129, 136, and 138.]

S. Boyd, N. Parikh, and E. Chu. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011. [Cited on pages 6, 15, 27, and 34.]

K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010. doi: 10.1137/090769521. URL https://doi.org/10.1137/090769521. [Cited on page 26.]

A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005a. [Cited on pages 35 and 107.]

A. Buades, B. Coll, and J.-M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005b. [Cited on page 38.]

G. T. Buzzard, S. H. Chan, S. Sreehari, and C. A. Bouman. Plug-and-play unplugged: Optimization-free reconstruction using consensus equilibrium. *SIAM Journal on Imaging Sciences*, 11(3):2001–2020, 2018. [Cited on page 35.]

X. Cai, M. Pereyra, and J. D. McEwen. Uncertainty quantification for radio interferometric imaging – i. proximal MCMC methods. *Monthly Notices of the Royal Astronomical Society*, 480(3):4154–4169, July 2018. doi: 10.1093/mnras/sty2004. URL https://doi.org/10.1093/mnras/sty2004. [Cited on page 84.]

A. P. Calderon. Uniqueness in the cauchy problem for partial differential equations. *American Journal of Mathematics*, 80(1):16–36, 1958. ISSN 00029327, 10806377. URL http://www.jstor.org/stable/2372819. [Cited on page 25.]

A. P. Calderon and A. Zygmund. *On the Existence of Certain Singular Integrals*, pages 19–73. Springer Netherlands, Dordrecht, 1989. ISBN 978-94-009-1045-4. doi: 10.1007/978-94-009-1045-4_3. URL https://doi.org/10.1007/978-94-009-1045-4_3. [Cited on page 25.]

A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004. doi: 10.1023/B:JMIV.0000011325.36760.1e. [Cited on page 28.]

A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76:167–188, 1997. [Cited on page 26.]

A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40(1):120–145, 2011. [Cited on pages 27 and 51.]

T. Chen, E. Fox, and C. Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1683–1691. PMLR, 22–24 Jun 2014. URL https://proceedings.mlr.press/v32/cheni14.html. [Cited on page 28.]

Y. Chen and T. Pock. Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1256–1272, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2596743. [Cited on pages 6, 15, 31, and 107.]

R. Cohen, M. Elad, and P. Milanfar. Regularization by denoising via fixed-point projection (red-pro), 2020. [Cited on pages 8, 17, and 37.]

P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer, 2011. [Cited on page 34.]

A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018. doi: 10.1109/MSP.2017.2765202. [Cited on page 32.]

G. V. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314, 1989. [Cited on page 28.]

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising with block-matching and 3d filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, page 606414. International Society for Optics and Photonics, 2006. [Cited on pages 35 and 107.]

A. S. Dalalyan. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 79(3):651–676, 2017. ISSN 1369-7412. doi: 10.1111/rssb.12183. URL https://doi.org/10.1111/rssb.12183. [Cited on page 38.]

A. Daniely. Depth separation for neural networks. In *COLT*, 2017. [Cited on page 29.]

V. Debarnot and P. Weiss. Deep-blur: Blind identification and deblurring with convolutional neural networks. 2022. [Cited on page 114.]

B. Delyon. General results on the convergence of stochastic algorithms. *IEEE Transactions on Automatic Control*, 41(9):1245–1255, 1996. [Cited on page 45.]

B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the em algorithm. *Annals of statistics*, pages 94–128, 1999. [Cited on page 45.]

S. Diamond, V. Sitzmann, F. Heide, and G. Wetzstein. Unrolled optimization with deep priors. 2017. [Cited on pages 6, 15, and 31.]

C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014. [Cited on page 31.]

D. L. Donoho and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *biometrika*, 81(3):425–455, 1994. [Cited on pages 2, 12, and 26.]

R. Douc, E. Moulines, P. Priouret, and P. Soulier. *Markov Chains*. Springer Series in Operations Research and Financial Engineering. Springer International Publishing, 2019. ISBN 9783319977034. URL https://books.google.fr/books?id=eTYnuQEACAAJ. [Cited on page 137.]

M. M. Dunlop. Multiplicative noise in bayesian inverse problems: Well-posedness and consistency of map estimators, 2019. URL https://arxiv.org/abs/1910.14632. [Cited on page 22.]

A. Durmus and E. Moulines. Nonasymptotic convergence analysis for the unadjusted Langevin algorithm. *Ann. Appl. Probab.*, 27(3):1551–1587, 2017. ISSN 1050-5164. doi: 10.1214/16-AAP1238. URL https://doi.org/10.1214/16-AAP1238. [Cited on pages 38, 133, and 137.]

A. Durmus, E. Moulines, and M. Pereyra. Efficient bayesian computation by proximal markov chain monte carlo: when langevin meets moreau. *SIAM Journal on Imaging Sciences*, 11(1):473–506, 2018. [Cited on pages 3, 13, 28, 38, and 66.]

B. Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. [Cited on pages 33, 36, 42, and 64.]

M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006. doi: 10.1109/TIP.2006.881969. [Cited on page 26.]

R. Eldan and O. Shamir. The power of depth for feedforward neural networks. In *COLT*, 2016. [Cited on page 29.]

C. Fefferman, S. Mitter, and H. Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016. ISSN 08940347, 10886834. URL https://www.jstor.org/stable/jamermathsoci.29.4.983. [Cited on pages 2 and 12.]

Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html. [Cited on page 31.]

H. Gao, X. Tao, X. Shen, and J. Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3848–3856, 2019. [Cited on page 31.]

M. Gharbi, G. Chaurasia, S. Paris, and F. Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)*, 35(6):191, 2016. [Cited on page 31.]

D. Gilton, G. Ongie, and R. Willett. Neumann networks for inverse problems in imaging. 2019. [Cited on pages 6, 15, and 31.]

M. Girolami and B. Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73 (2):123–214, 2011. [Cited on page 28.]

R. Glowinski and A. Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *Revue franccaise d'automatique, informatique, recherche opérationnelle. Analyse numérique*, 9 (R2):41–76, 1975. [Cited on pages 6, 15, 27, and 34.]

H. Goh, S. Sheriffdeen, J. Wittmer, and T. Bui-Thanh. Solving bayesian inverse problems via variational autoencoders. In J. Bruna, J. Hesthaven, and L. Zdeborova, editors, *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, volume 145 of *Proceedings of Machine Learning Research*, pages 386–425. PMLR, 16–19 Aug 2022. URL https://proceedings.mlr.press/v145/goh22a.html. [Cited on page 33.]

M. González, A. Almansa, and P. Tan. Solving Inverse Problems by Joint Posterior Maximization with Autoencoding Prior. mar 2021. [Cited on pages 92 and 114.]

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversaria l Networks. *Advances in Neural Information Processing Systems 27*, pages 2672–2680, jun 2014. ISSN 10495258. [Cited on page 32.]

I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. http://www.deeplearningbook.org. [Cited on page 30.]

K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 399–406. Omnipress, 2010. [Cited on pages 6, 15, and 31.]

R. Gribonval. Should penalized least squares regression be interpreted as maximum a posteriori estimation? *IEEE Transactions on Signal Processing*, 59(5):2405–2410, 2011. [Cited on pages 33, 37, and 118.]

A. A. Gritsenko, J. Snoek, and T. Salimans. On the relationship between normalising flows and variational- and denoising autoencoders, 2019. URL https://openreview.net/forum?id=HklKEUUY_E. [Cited on page 114.]

B. Guo, Y. Han, and J. Wen. Agem: Solving linear inverse problems via deep priors and sampling. In *Advances in Neural Information Processing Systems*, pages 547–558, 2019. [Cited on pages 4, 14, 39, and 62.]

J. Hadamard. Sur les problèmes aux dérivées partielles leur signification physique. *Princeton University Bulletin*, pages 49–52. [Cited on page 25.]

J. Hadamard. Lectures on cauchy's problem in linear partial differential equations, 1923. [Cited on page 25.]

P. Hagemann, J. Hertrich, and G. Steidl. Stochastic normalizing flows for inverse problems: a markov chains viewpoint, 2022. [Cited on page 33.]

A. Halimi, Y. Altmann, A. McCarthy, X. Ren, R. Tobin, G. S. Buller, and S. McLaughlin. Restoration of intensity and depth images constructed using sparse single-photon data. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 86–90. IEEE, 2016. [Cited on page 115.]

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90. [Cited on page 30.]

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf. [Cited on pages 5 and 14.]

T. Hohage and F. Werner. Inverse problems with poisson data: statistical regularization theory, applications and algorithms. *Inverse Problems*, 32(9):093001, 2016. [Cited on page 115.]

M. Holden, M. Pereyra, and K. C. Zygalakis. Bayesian imaging with data-driven priors encoded by neural networks. *SIAM Journal on Imaging Sciences*, 15(2):892–924, 2022. doi: 10.1137/21M1406313. URL https://doi.org/10.1137/21M1406313. [Cited on page 109.]

K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4:251–257, 1991. [Cited on page 28.]

A. Houdard, C. Bouveyron, and J. Delon. High-dimensional mixture models for unsupervised image denoising (hdmi). *SIAM Journal on Imaging Sciences*, 11(4):2815–2846, 2018. [Cited on page 28.]

S. Hurault, A. Leclaire, and N. Papadakis. Gradient Step Denoiser for convergent Plug-and-Play. In *(ICLR) International Conference on Learning Representations*, 2022a. URL http://arxiv.org/abs/2110.03220. [Cited on pages xviii, 4, 14, 37, 92, 95, and 113.]

S. Hurault, A. Leclaire, and N. Papadakis. Proximal denoiser for convergent plug-and-play optimization with nonconvex regularization. (2), jan 2022b. URL http://arxiv.org/abs/2201.13256. [Cited on pages 4, 14, 37, 92, 94, 95, 99, and 113.]

A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html. [Cited on pages 4 and 14.]

Z. Kadkhodaie and E. P. Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. *arXiv preprint arXiv:2007.13640*, 2020. [Cited on pages 4, 14, and 39.]

I. Karatzas and S. E. Shreve. *Brownian motion and stochastic calculus*, volume 113 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 1991. ISBN 0-387-97655-8. doi: 10.1007/978-1-4612-0949-2. URL https://doi.org/10.1007/978-1-4612-0949-2. [Cited on page 135.]

B. Kawar, G. Vaksman, and M. Elad. Snips: Solving noisy inverse problems stochastically. *arXiv preprint arXiv:2105.14951*, 2021a. [Cited on pages 4, 14, and 39.]

B. Kawar, G. Vaksman, and M. Elad. Stochastic image denoising by sampling from the posterior distribution, 2021b. [Cited on pages 4, 14, and 39.]

B. Kawar, M. Elad, S. Ermon, and J. Song. Denoising diffusion restoration models. *arXiv preprint arXiv:2201.11793*, 2022. [Cited on pages xiv, 5, 14, 15, and 114.]

D. Kingma and M. Welling. *An Introduction to Variational Autoencoders*. Foundations and trends in machine learning. Now Publishers, 2019. ISBN 9781680836226. URL https://books.google.fr/books?id=pLX0ywEACAAJ. [Cited on pages 31 and 32.]

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. [Cited on page 30.]

E. Kobler, A. Effland, K. Kunisch, and T. Pock. Total deep variation for linear inverse problems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. [Cited on page 92.]

I. Kobyzev, S. J. Prince, and M. A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, 2021. doi: 10.1109/TPAMI.2020.2992934. [Cited on page 32.]

S. Kullback. *Information theory and statistics*. Courier Corporation, 1997. [Cited on page 133.]

C. Laroche, A. Almansa, and M. Tassano. Deep model-based super-resolution with non-uniform blur. *arXiv preprint arXiv:2204.10109*, 2022. [Cited on pages xiv, 6, 15, and 16.]

J. Latz. On the well-posedness of bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 8(1):451–482, 2020. doi: 10.1137/19M1247176. URL https://doi.org/10.1137/19M1247176. [Cited on page 65.]

R. Laumont, V. de Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. On maximum-a-posteriori estimation with plug & play priors and stochastic gradient descent. 2021. URL https://hal.archives-ouvertes.fr/hal-03348735/document. [Cited on page 41.]

R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. Bayesian imaging using plug & play priors: When langevin meets tweedie. *SIAM Journal on Imaging Sciences*, 15(2):701–737, 2022. doi: 10.1137/21M1406349. URL https://doi.org/10.1137/21M1406349. [Cited on page 62.]

M. Lebrun, A. Buades, and J.-M. Morel. A nonlocal bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013. [Cited on page 35.]

J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018. [Cited on pages 45 and 69.]

R. S. Liptser and A. N. Shiryaev. *Statistics of random processes. I*, volume 5 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, expanded edition, 2001. ISBN 3-540-63929-2. General theory, Translated from the 1974 Russian original by A. B. Aries, Stochastic Modelling and Applied Probability. [Cited on page 133.]

Y. Liu, Z. Qin, S. Anwar, S. Caldwell, and T. Gedeon. Are deep neural architectures losing information? invertibility is indispensable. In *International Conference on Neural Information Processing*, pages 172–184. Springer, 2020. [Cited on page 114.]

C. Louchet and L. Moisan. Posterior expectation of the total variation model: Properties and experiments. *SIAM Journal on Imaging Sciences*, 6(4):2640–2684, dec 2013. ISSN 19364954. doi: 10.1137/120902276. [Cited on pages 3, 12, 13, and 28.]

A. Lugmayr, M. Danelljan, L. Van Gool, and R. Timofte. Srflow: Learning the super-resolution space with normalizing flow. In A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, editors, *Computer Vision – ECCV 2020*, pages 715–732, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58558-7. [Cited on pages 33 and 114.]

T. Meinhardt, M. Moller, C. Hazirbas, and D. Cremers. Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *(ICCV) International Conference on Computer Vision*, pages 1781–1790, 2017. doi: 10.1109/ICCV.2017.198. [Cited on page 35.]

M. Metivier and P. Priouret. Applications of a kushner and clark lemma to general classes of stochastic algorithms. *IEEE Transactions on Information Theory*, 30(2):140–151, 1984. [Cited on page 45.]

S. P. Meyn and R. L. Tweedie. Stability of Markovian processes. III. Foster-Lyapunov criteria for continuous-time processes. *Adv. in Appl. Probab.*, 25(3):518–548, 1993. ISSN 0001-8678. doi: 10.2307/1427522. URL https://doi.org/10.2307/1427522. [Cited on page 126.]

P. Milanfar. Symmetrizing Smoothing Filters. *SIAM Journal on Imaging Sciences*, 6(1): 263–284, jan 2013. ISSN 1936-4954. doi: 10.1137/120875843. [Cited on page 37.]

M. Mirza and S. Osindero. Conditional generative adversarial nets. *ArXiv*, abs/1411.1784, 2014. [Cited on page 32.]

K. Miyasawa et al. An empirical bayes estimator of the mean of a normal population. *Bull. Inst. Internat. Statist*, 38(181-188), 1961. [Cited on page 33.]

T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. [Cited on pages 45 and 69.]

S. Mukherjee, S. Dittmer, Z. Shumaylov, S. Lunz, O. Öktem, and C.-B. Schönlieb. Learned convex regularizers for inverse problems, 2021. [Cited on page 92.]

J. R. Munkres. Topology, 2000. [Cited on page 43.]

P. Nair, R. G. Gavaskar, and K. N. Chaudhury. Fixed-point and objective convergence of plug-and-play algorithms. *IEEE Transactions on Computational Imaging*, 7:337–348, 2021. [Cited on page 35.]

F. Natterer. Regularisierung schlecht gestellter probleme durch projektionsverfahren. *Numerische Mathematik*, 28:329–341, 1977. [Cited on page 25.]

F. Natterer. *The Mathematics of Computerized Tomography*. Society for Industrial and Applied Mathematics, 2001. doi: 10.1137/1.9780898719284. URL https://epubs.siam.org/doi/abs/10.1137/1.9780898719284. [Cited on page 25.]

F. Natterer and F. Wübbeling. Mathematical methods in image reconstruction. In *SIAM monographs on mathematical modeling and computation*, 2001. [Cited on page 25.]

F. Panloup et al. Unajusted langevin algorithm with multiplicative noise: Total variation and wasserstein bounds. *arXiv preprint arXiv:2012.14310*, 2020. [Cited on page 114.]

G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. 2019. [Cited on page 32.]

M. Pereyra. Proximal Markov chain Monte Carlo algorithms. *Statistics and Computing*, 26(4):745–760, jul 2016. ISSN 0960-3174. doi: 10.1007/s11222-015-9567-4. [Cited on page 28.]

M. Pereyra. Revisiting Maximum-a-Posteriori estimation in log-concave models. *SIAM Journal on Imaging Sciences*, 12(1):650–670, 2019. [Cited on pages 28 and 60.]

M. Pereyra, P. Schniter, E. Chouzenoux, J.-C. Pesquet, J.-Y. Tourneret, A. O. Hero, and S. McLaughlin. A survey of stochastic simulation and optimization methods in signal processing. *IEEE Journal of Selected Topics in Signal Processing*, 10(2):224–241, 2015. [Cited on page 38.]

M. Pereyra, L. Vargas Mieles, and K. C. Zygalakis. Accelerating proximal Markov chain Monte Carlo by using an explicit stabilized method. *SIAM J. Imaging Sci.*, 13(2):905–935, 2020. doi: 10.1137/19M1283719. URL https://doi.org/10.1137/19M1283719. [Cited on pages 67, 87, and 91.]

J.-C. Pesquet, A. Repetti, M. Terris, and Y. Wiaux. Learning maximally monotone operators for image recovery, 2020. [Cited on pages xviii, 37, 92, 95, 96, and 114.]

J. Rapp and V. K. Goyal. A few photons among many: Unmixing signal and noise for photon-efficient active imaging. *IEEE Transactions on Computational Imaging*, 3(3): 445–459, 2017. [Cited on page 115.]

D. Ray, H. Ramaswamy, D. V. Patel, and A. A. Oberai. The efficacy and generalizability of conditional gans for posterior inference in physics-based inverse problems. *ArXiv*, abs/2202.07773, 2022. [Cited on page 33.]

E. T. Reehorst and P. Schniter. Regularization by denoising: Clarifications and new interpretations. *IEEE Transactions on Computational Imaging*, 5(1):52–67, 2018. doi: 10.1109/TCI.2018.2880326. [Cited on pages 35, 37, and 107.]

A. Repetti, M. Pereyra, and Y. Wiaux. Scalable bayesian uncertainty quantification in imaging inverse problems via convex optimization. *SIAM Journal on Imaging Sciences*, 12(1):87–118, 2019. ISSN 1936-4954. doi: 10.1137/18M1173629. [Cited on pages xiv, 3, 13, and 28.]

H. Robbins. An empirical bayes approach to statistics. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, volume 3, pages 157–164. University of California Press, 1956. [Cited on page 33.]

C. Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer Texts in Statistics. Springer New York, 2007. ISBN 9780387715988. URL https://books.google.fr/books?id=6oQ4s8Pq9pYC. [Cited on pages 31, 64, 93, and 94.]

G. O. Roberts, R. L. Tweedie, et al. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996. [Cited on pages 38 and 135.]

Y. Romano, M. Elad, and P. Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017. [Cited on pages 37, 94, 104, and 107.]

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *ArXiv*, abs/1505.04597, 2015. [Cited on page 30.]

L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992. ISSN 01672789. doi: 10.1016/0167-2789(92)90242-F. [Cited on pages 2, 12, 26, 28, and 51.]

E. K. Ryu, J. Liu, S. Wang, X. Chen, Z. Wang, and W. Yin. Plug-and-play methods provably converge with properly trained denoisers. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 5546–5557, 2019. URL http://proceedings.mlr.press/v97/ryu19a.html. [Cited on pages xviii, 4, 7, 8, 14, 17, 35, 36, 38, 45, 50, 51, 52, 55, 69, 73, 75, 94, 113, and 114.]

C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi. Image super-resolution via iterative refinement. *arXiv preprint arXiv:2104.07636*, 2021. [Cited on pages 5 and 15.]

E. Schwartz, R. Giryes, and A. M. Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing*, 28(2):912–923, 2018. [Cited on page 31.]

L. Schwartz. Désintégration d'une mesure. *Séminaire Équations aux dérivées partielles (Polytechnique)*, pages 1–10. [Cited on page 63.]

D. Shin, A. Kirmani, V. K. Goyal, and J. H. Shapiro. Photon-efficient computational 3-d and reflectivity imaging with single-photon detectors. *IEEE Transactions on Computational Imaging*, 1(2):112–125, 2015. [Cited on page 115.]

V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein. Implicit neural representations with periodic activation functions. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546. [Cited on page 22.]

Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/file/3001ef257407d5a371a96dcd947c7d93-Paper.pdf. [Cited on pages 4, 14, 39, and 114.]

S. Sreehari, S. V. Venkatakrishnan, B. Wohlberg, G. T. Buzzard, L. F. Drummy, J. P. Simmons, and C. A. Bouman. Plug-and-Play Priors for Bright Field Electron Tomography and Sparse Interpolation. *IEEE Transactions on Computational Imaging*, 2(4):1–1, 2016. ISSN 2333-9403. doi: 10.1109/TCI.2016.2599778. [Cited on page 35.]

J.-L. Starck and F. Murtagh. Astronomical image and data analysis. 2007. [Cited on page 115.]

A. M. Stuart. Inverse problems: A bayesian perspective. *Acta Numerica*, 19:451–559, 2010. doi: 10.1017/S0962492910000061. [Cited on page 64.]

Y. Sun, B. Wohlberg, and U. S. Kamilov. An online plug-and-play algorithm for regularized image reconstruction. *IEEE Transactions on Computational Imaging*, 2019. [Cited on pages 4, 8, 14, 17, 35, 113, and 114.]

Y. Sun, Z. Wu, B. Wohlberg, and U. S. Kamilov. Scalable plug-and-play admm with convergence guarantees. *arXiv preprint arXiv:2006.03224*, 2020. [Cited on pages 4, 8, 14, 17, 35, and 113.]

V. B. Tadić, A. Doucet, et al. Asymptotic bias of stochastic gradient search. *The Annals of Applied Probability*, 27(6):3255–3304, 2017. [Cited on pages 45, 46, 118, and 119.]

A. M. Teodoro, J. M. Bioucas-Dias, and M. A. Figueiredo. A convergent image fusion algorithm using scene-adapted gaussian-mixture-based denoising. *IEEE Transactions on Image Processing*, 28(1):451–463, 2018a. [Cited on page 34.]

A. M. Teodoro, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Scene-Adapted Plug-and-Play Algorithm with Guaranteed Convergence: Applications to Data Fusion in Imaging, jan 2018b. [Cited on page 28.]

A. N. Tikhonov. On the stability of inverse problems. *Proceedings of the USSR Academy of Sciences*, 39:195–198, 1943. [Cited on page 25.]

S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE, 2013. [Cited on pages iii, iv, 4, 14, and 35.]

A. F. Vidal, V. De Bortoli, M. Pereyra, and A. Durmus. Maximum likelihood estimation of regularization parameters in high-dimensional inverse problems: An empirical bayesian approach part i: Methodology and experiments. *SIAM Journal on Imaging Sciences*, 13 (4):1945–1989, 2020. [Cited on pages 58, 95, 111, and 114.]

C. Villani. *Optimal transport*, volume 338 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 2009. ISBN 978-3-540-71049-3. doi: 10.1007/978-3-540-71050-9. URL https://doi.org/10.1007/978-3-540-71050-9. Old and new. [Cited on page 71.]

M. Vono, N. Dobigeon, and P. Chainais. Asymptotically exact data augmentation: models, properties and algorithms. *arXiv preprint arXiv:1902.05754*, 2019. [Cited on page 125.]

K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang. Generative adversarial networks: introduction and outlook. *IEEE/CAA Journal of Automatica Sinica*, 4(4): 588–598, 2017. doi: 10.1109/JAS.2017.7510583. [Cited on page 32.]

Z. Wang and A. C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, 2009. doi: 10.1109/MSP.2008.930649. [Cited on page 83.]

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612, 2004. [Cited on page 83.]

H. Wu, J. Köhler, and F. Noe. Stochastic normalizing flows. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5933–5944. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/41d80bfc327ef980528426fc810a6d7a-Paper.pdf. [Cited on page 32.]

X. Xu, Y. Sun, J. Liu, B. Wohlberg, and U. S. Kamilov. Provable Convergence of Plug-and-Play Priors with MMSE denoisers. (4):1–10, 2020. URL http://arxiv.org/abs/2005.07685. [Cited on pages 4, 14, 33, 35, 36, 37, 50, 51, 52, 113, and 114.]

G. Yu, G. Sapiro, and S. Mallat. Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2011. [Cited on page 28.]

K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7): 3142–3155, 2017. [Cited on pages 31, 35, 38, 95, and 107.]

K. Zhang, W. Zuo, and L. Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. [Cited on pages 31 and 114.]

K. Zhang, L. Van Gool, and R. Timofte. Deep unfolding network for image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3217–3226, 2020. [Cited on pages 6 and 15.]

K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. doi: 10.1109/TPAMI.2021.3088914. [Cited on pages 4, 8, 14, 17, 95, and 114.]

D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, nov 2011. ISBN 978-1-4577-1102-2. doi: 10.1109/ICCV.2011.6126278. URL http://people.csail.mit.edu/danielzoran/EPLLICCVCameraReady.pdf. [Cited on pages 27, 28, and 34.]

A. Zygmund. *On singular integrals*, pages 68–105. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-10918-8. doi: 10.1007/978-3-642-10918-8_3. URL https://doi.org/10.1007/978-3-642-10918-8_3. [Cited on page 25.]