UCLouvain

FACULTE DES SCIENCES

Faculté des sciences

Valuation of Financial Derivatives with Constrained Gaussian Process Regression

Author: **HARINDINTWALI CARL PRINCE**
Supervisor: **Prof. HAINAUT DONATIEN**
Reader: **Dr. BARBARIN JÉRÔME**
Academic year 2025–2026
Master [120] Actuarial Science

Abstract

This thesis studies Constrained Gaussian Process Regression (CGPR) for option pricing. The study puts the focus on a practical problem in quantitative finance where prices and Greeks often have to be computed many times for the same model. In practice, standard methods such as FFT or Monte Carlo are commonly used to obtain these prices. However, these approaches do not always provide a reusable pricing surface and greeks that can easily be evaluated again. The approach studied here is different because CGPR uses the Feynman–Kac PDE itself to approximate the pricing function, by imposing the PDE residual at interior points and the payoff or boundary conditions at boundary points.

The study begins with the Black–Scholes model, because it gives closed-form prices and Greeks and therefore provides a clear validation setting. Then the analysis moves to the Heston model, where the benchmark is computed with the Carr–Madan FFT. The framework is then extended to two additional applications. The first one is a two-asset Heston basket option, which makes the problem higher-dimensional and requires a Monte Carlo benchmark. The second one is a GMAB contract, where a mortality component is added to the financial pricing equation in order to move into the actuarial (insurance) setting.

Overall, the numerical results remain encouraging for the different applications considered in the study. In the Black–Scholes case, the CGPR approximation is very close to the exact benchmark. In the Heston model, the method remains effective, even if the errors increase when stochastic volatility is introduced. The two-asset extension is more difficult, but the CGPR prices still follow the Monte Carlo benchmark accuratly. The GMAB application also shows that the method can be extended to include a mortality component without modifying the PDE framework. The main difficulties is observed near the strike, at low variance levels and for short maturities. These are the regions where the pricing surface changes more sharply. Overall, CGPR does not replace FFT or Monte Carlo in every situation, although it remains a practical approach for constructing model-constrained pricing surfaces together with analytical Greeks. .

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my thesis supervisor, Prof. Donatien Hainaut. I am grateful for his support throughout this work, for his availability whenever I needed guidance, and for his patience and understanding during these past two years.

I would also like to thank my friends, who supported me during my studies and helped me keep going in difficult moments. I am very grateful to my brother and friend Régis, who has been by my side throughout my academic journey. His advice, encouragement and presence have been very valuable to me.

I also want to thank my family, especially my parents, for their constant support, trust and encouragement. Even from a distance, their support always felt close and comforting in times of need.

Above all, I thank God for His protection, His grace and the strength He gave me during this journey. "The Lord is my shepherd; I have all that I need." Psalms 23

Note: Generative AI was used for translation purposes.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Abbreviations

.

ATM : At The Money.

BS : Black–Scholes.

CGPR : Constrained Gaussian Process Regression.

CIR : Cox–Ingersoll–Ross.

CRN : Common Random Numbers.

FFT : Fast Fourier Transform.

GMAB : Guaranteed Minimum Accumulation Benefit.

GP : Gaussian Process.

GPR : Gaussian Process Regression.

ITM : In The Money.

MAE : Mean Absolute Error.

MC : Monte Carlo.

MRE : Mean Relative Error.

OTM : Out of The Money.

PDE : Partial Differential Equation.

CHAPTER 1

INTRODUCTION AND LITERATURE REVIEW

1.1 Introduction

Pricing a financial derivative sounds like a well-defined problem: given a model and a payoff, compute an expectation. In practice, however, this computation has to be repeated thousands of times. Models need to be recalibrated as market conditions change, risk indicators must be updated within the day, and hedge positions require continuous monitoring [17]. Each of these tasks demands a fresh evaluation of the pricing function, often under tight time constraints.

The Black–Scholes model of Black and Scholes [14] provides a natural first benchmark for this thesis. Its closed-form formula makes it possible to isolate the numerical error of the Constrained Gaussian Process Regression (CGPR) approximation from the error of the reference method itself. This is useful at the implementation stage, because any discrepancy between the CGPR output and the analytical price can be attributed to the construction of the kernel, the choice of training points, or the enforcement of the boundary conditions. At the same time, Black–Scholes remains too restrictive for a realistic pricing study: volatility is constant, the dependence between price and volatility is absent, and, as shown by Rubinstein [16], the model cannot reproduce the implied volatility patterns typically observed on equity markets.

This motivates the transition to the stochastic volatility model of Heston [6].. By introducing a stochastic variance process correlated with the stock price, the Heston model keeps a clear financial interpretation while producing a richer pricing surface. The numerical problem is also more demanding. Unlike in the Black–Scholes case,

prices are no longer obtained from an elementary closed-form formula, and they are usually evaluated through semi-analytical or numerical methods. For vanilla options, the Fourier-based approach of Carr and Madan [8] provides a natural benchmark, while Monte Carlo simulation and PDE-based methods can also be used in more general settings. In the context of this thesis, this distinction is important: Black–Scholes is used as a controlled validation environment, whereas Heston is the first setting in which the CGPR must approximate a higher-dimensional pricing function depending on time, spot and variance.

This is the starting point of this thesis. Rather than solving the pricing problem from scratch each time, the idea is to learn the pricing function once and then reuse it. Gaussian Process Regression (GPR) provides a natural framework for this by treating option pricing as a supervised learning task, fitting a smooth function to a set of pre-computed prices, and delivering instantaneous predictions at new parameter values. De Spiegeleer et al. [2] showed that this approach can match the accuracy of traditional methods while substantially reducing computation time.

However, this use of GPR does not fully solve the problem considered here. It makes the evaluation of new prices faster, but it still relies on prices that have already been computed elsewhere, for example by Monte Carlo or FFT. The Gaussian process then learns from these numerical outputs, not from the pricing equation itself. In that sense, the fitted surface may be accurate on the training domain while still having no explicit reason to satisfy the Feynman–Kac Partial Differential Equation (PDE), as discussed by Jeanblanc et al. [7].

The CGPR framework introduced by Hainaut and Vrins [1] addresses this directly. Instead of training on observed prices, it incorporates the Feynman-Kac PDE as a structural constraint within the regression procedure. The training data are not market prices or Monte Carlo outputs, they are evaluations of the PDE residual and the terminal payoff. The resulting estimator is model-driven by construction, and it comes with a built-in uncertainty measure through the posterior variance of the Gaussian process.

The thesis is organised progressively. The CGPR method is first tested in the Black–Scholes model, where closed-form prices and Greeks give a reliable benchmark. The analysis then moves to the Heston model, which is the main part of the numerical study. Finally, two extensions are considered: a two-asset Heston basket option and a Guaranteed Minimum Accumulation Benefit (GMAB) contract including mortality risk.

1.2 Literature Review

The application of machine learning techniques to derivative pricing has grown steadily over the past decade, driven by the computational demands of modern risk management. This section reviews the main contributions that form the foundation of this thesis.

1.2.1 Gaussian Processes for Machine Learning

The theoretical framework used throughout this thesis is developed in the reference textbook of Rasmussen and Williams [3]. A Gaussian process (GP) is a probabilistic model defined over functions: instead of estimating a fixed set of parameters, it places a prior distribution directly over the space of functions and updates it as observations arrive. The result is a posterior distribution that provides not only a point estimate of the unknown function, but also a measure of uncertainty at each prediction point.

What makes Gaussian processes particularly well suited to option pricing is this built-in uncertainty quantification. A classical numerical method gives a single number; a GP gives a number and a confidence level. In a context where model risk and approximation errors matter, this is a genuine advantage. The technical details of the GP framework, including the choice of kernel and the estimation of hyperparameters, are presented in Chapter 2.

1.2.2 Application of GPR to Derivative Pricing

De Spiegeleer et al. [2] show how GPR can be used to speed up derivative pricing. Their approach starts from a set of prices computed with a standard method, such as Monte Carlo simulation or FFT. These prices are then used to train the Gaussian process on different values of the model and contract parameters. Once this step is done, the GP can give prices at new points without running the full pricing method again.

The authors report substantial computational gains across several applications. For European call options under the Heston model, the average approximation error remains of order 10^{-4} while speed-ups of one order of magnitude are achieved compared to the FFT method. The most striking gains appear for exotic derivatives: barrier options priced by Monte Carlo yield speed-ups of several thousands, while

maintaining a comparable level of accuracy. This paper is the direct starting point of this thesis: it establishes that GPR is a viable alternative to classical pricing methods, and it motivates the question of whether the approximation can be made more model-consistent.

1.2.3 Constrained Gaussian Process Regression

The main methodological reference of this thesis is Hainaut and Vrins [1], who introduce the Constrained Gaussian Process Regression (CGPR) framework. Their starting observation is that standard GPR is purely data-driven: it learns from observed prices but has no knowledge of the financial model that generated them. In particular, nothing guarantees that the fitted surface satisfies the Feynman-Kac PDE associated with the underlying risk-neutral dynamics.

Hainaut and Vrins [1] take a different route. They do not train the Gaussian process on prices computed in advance. Instead, they use the pricing equation itself in the construction of the regression. In practice, the PDE residual is forced to be close to zero at interior points, while the payoff is imposed at the boundary. The information given to the model therefore comes from the financial dynamics and from the payoff condition, rather than from a separate pricing routine.

The result is a pricing function that is consistent with the Feynman–Kac structure imposed by the model, computationally efficient, and equipped with analytical Greeks. These three properties make CGPR particularly well suited to the repeated pricing problems that arise in risk management and model calibration. The full mathematical derivation of the estimator is presented in Chapter 2, and its numerical performance is assessed in Chapters 3 and 4.

Moreover, since the estimator is a differentiable function of the input variables, all sensitivities (Greeks) can be obtained by analytical differentiation, without resorting to finite differences. This property is derived in Hainaut and Vrins [1] and is discussed in detail in Chapter 2.

1.3 Framework and Assumptions

1.3.1 Market Setting

We consider a frictionless financial market defined on a complete filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{Q})$, where $\mathbb{Q}$ is the risk-neutral measure and $\mathbb{F} = (\mathcal{F}_t)_{t\geq 0}$ satisfies the usual conditions. The market consists of a risk-free asset growing at constant rate $r > 0$ and a risky asset whose dynamics are specified below. The risk-free rate is assumed constant throughout this thesis for simplicity; this assumption does not affect the general structure of the CGPR methodology..

The market is driven by $p-1$ state variables stored in a vector $y_t \in \mathbb{R}^{p-1}$, whose dynamics under $\mathbb{Q}$ are governed by the stochastic differential equation

$$dy_t = \mu_y(t, y_t)\, dt + \Sigma_y(t, y_t)\, dB_t, \tag{1.1}$$

where B_t is a $(p-1)$-vector of independent Brownian motions under $\mathbb{Q}$, μ_y is the drift vector and Σ_y is the diffusion matrix. Following Jeanblanc, Yor and Chesney [7], we assume the standard regularity conditions on μ_y and Σ_y required to ensure existence and uniqueness of the solution

We consider a European derivative with payoff $P(T, y_T)$ at maturity T. Under the absence of arbitrage, its price at time $t \leq T$ is the risk-neutral expectation of the discounted payoff:

$$f(t, y_t) = \mathbb{E}^{\mathbb{Q}}\left[e^{-r(T-t)} P(T, y_T) \;\middle|\; \mathcal{F}_t\right]. \tag{1.2}$$

The function $f : \mathcal{X} \to \mathbb{R}$, where $\mathcal{X} = [0, T] \times \mathbb{R}^{p-1}$, is called the *pricing function*. We write $x = (t, y_t)$ for the p-vector of time and state variables.

1.3.2 The Heston Stochastic Volatility Model

Throughout this thesis, he main underlying model is the stochastic volatility model introduced by Heston [6]. Under $\mathbb{Q}$, the state vector is $y_t = (S_t, V_t)^\top$, where S_t is the asset price and V_t its instantaneous variance. Their joint dynamics are

$$\begin{cases} dS_t = rS_t\, dt + \sqrt{V_t}\, S_t\, dW_t^{(1)}, \\ dV_t = \kappa(\gamma - V_t)\, dt + \sigma\sqrt{V_t}\, dW_t^{(2)}, \end{cases} \tag{1.3}$$

where $\kappa > 0$ is the mean-reversion speed, $\gamma > 0$ the long-term variance, $\sigma > 0$ the volatility of variance, and $d\langle W^{(1)}, W^{(2)}\rangle_t = \rho\, dt$ with $\rho \in [-1, 1]$. The variance process follows the square-root dynamics introduced by Cox, Ingersoll and Ross [5]. When the Feller condition $2\kappa\gamma \geq \sigma^2$ holds, the zero boundary is unattainable and the variance remains strictly positive. The numerical experiments below do not rely on imposing this condition; the variance domain is instead handled through the truncated computational domain.

In the notation of Equation (1.1):

$$\mu_y(t, y_t) = \begin{pmatrix} rS_t \\ \kappa(\gamma - V_t) \end{pmatrix}, \qquad \Sigma_y(t, y_t) = \begin{pmatrix} S_t\sqrt{V_t} & 0 \\ \rho\sigma\sqrt{V_t} & \sigma\sqrt{V_t(1-\rho^2)} \end{pmatrix},$$

so that $\Sigma_y\Sigma_y^\top = \begin{pmatrix} S_t^2 V_t & \rho\sigma S_t V_t \\ \rho\sigma S_t V_t & \sigma^2 V_t \end{pmatrix}$.

1.3.3 The Feynman–Kac Theorem and the Pricing PDE

The Feynman–Kac theorem establishes the link between the risk-neutral expectation in Equation (1.2) and a partial differential equation. It is the result that underpins the entire CGPR methodology.

Theorem 1.3.1 (Feynman–Kac [7])**.** *Under standard regularity conditions, the pricing function $f(t, y_t)$ defined in Equation* (1.2) *satisfies the PDE*

$$L_x f(x) = 0, \qquad x \in \mathcal{X}^\circ, \tag{1.4}$$

subject to the terminal condition $f(T, y_T) = P(T, y_T)$, where L_x is the linear differential operator

$$L_x f = \frac{\partial f}{\partial t} + \mu_y^\top \nabla_y f + \frac{1}{2}\mathrm{tr}\big(\Sigma_y\Sigma_y^\top H_y f\big) - r\, f, \tag{1.5}$$

with $\nabla_y f$ the gradient and $H_y f$ the Hessian of f with respect to y.

Proof. The proof relies on Itô's lemma applied to the discounted price process $e^{-rt}f(t, y_t)$. Under the risk-neutral measure, this process is a $(\mathbb{Q}, \mathbb{F})$-martingale, so its drift must vanish. Setting the drift to zero yields Equation (1.4). We refer to Jean Blanc [7] for the complete proof. □

This theorem explains why the PDE constraint is used in the CGPR method. If the pricing function satisfies the Feynman–Kac PDE (1.4) and the terminal condition, it gives the price implied by the model. In the numerical implementation,

the residual $L_x f$ is therefore required to be close to zero at interior training points, while the payoff is imposed on the boundary. In this way, the regression is linked to the pricing equation, instead of being trained only on prices computed in advance.

Applying Theorem 1.3.1 to the Heston model, with $x = (t, s, v)$ and substituting μ_y and $\Sigma_y \Sigma_y^\top$ into Equation (1.5):

$$\mu_y^\top \nabla_y f = rs\,\partial_s f + \kappa(\gamma - v)\,\partial_v f,$$
$$\frac{1}{2}\mathrm{tr}(\Sigma_y \Sigma_y^\top H_y f) = \frac{1}{2}s^2 v\,\partial_{ss} f + \rho\sigma s v\,\partial_{sv} f + \frac{1}{2}\sigma^2 v\,\partial_{vv} f.$$

The Feynman–Kac operator for the Heston model is therefore

$$L_x f = \frac{\partial f}{\partial t} + rs\,\frac{\partial f}{\partial s} + \kappa(\gamma - v)\,\frac{\partial f}{\partial v} + \frac{1}{2}vs^2\,\frac{\partial^2 f}{\partial s^2} + \rho\sigma v s\,\frac{\partial^2 f}{\partial s\,\partial v} + \frac{1}{2}\sigma^2 v\,\frac{\partial^2 f}{\partial v^2} - rf. \quad (1.6)$$

The pricing function of a European call with strike K satisfies $L_x f = 0$ in $\mathcal{X}^\circ$ with terminal condition $f(T, s, v) = (s - K)^+$. This is the PDE that the CGPR will enforce at interior training points in Chapter 2: the operator L_x defined here reappears in the pseudo-observations, in the covariance matrix C, and in the kernel derivatives of Chapter 2.

CHAPTER 2

MATHEMATICAL FRAMEWORK OF THE CGPR AND APPLICATION TO THE HESTON MODEL

2.1 Gaussian Process Regression

2.1.1 From Linear Regression to Gaussian Processes

The standard linear regression model assumes that the output $f(\mathbf{x})$ is a linear combination of the inputs $\mathbf{x} \in \mathbb{R}^d$:

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{w}, \qquad y = f(\mathbf{x}) + \varepsilon,$$

where $\mathbf{w} \in \mathbb{R}^d$ is a weight vector and $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$ is an i.i.d. noise term. This model is simple but restricted to linear relationships, which limits its ability to represent complex functions such as option pricing surfaces.

A natural way to make the model more flexible is to replace the original input x by a set of transformed variables $\phi : \mathbb{R}^d \to \mathbb{R}^N$. The regression model then becomes

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}.$$

The role of ϕ is to describe the same input in a richer space, so that the model can represent nonlinear relationships while remaining linear in the weights w.

Placing a Gaussian prior $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p)$ on the weights and marginalising over $\mathbf{w}$, the predictive distribution of f depends only on inner products of the form

$\phi(\mathbf{x})^\top \Sigma_p \, \phi(\mathbf{x}')$. We can therefore define a *kernel function*

$$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \Sigma_p \, \phi(\mathbf{x}'),$$

and compute predictions without evaluating ϕ explicitly, even when N is very large. This is the *kernel trick* [3].

Taking the limit $N \to \infty$, the parametric model disappears entirely and is replaced by a prior defined directly over functions. This leads to the Gaussian process framework.

2.1.2 Definition and Prior Distribution

Definition 1 (Gaussian Process [3])**.** *A Gaussian process (GP) is a collection of random variables* $\{g(\mathbf{x})\}_{\mathbf{x}\in\mathcal{X}}$ *such that, for any finite set of points* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$*, the vector* $(g(\mathbf{x}_1), \ldots, g(\mathbf{x}_n))^\top$ *follows a multivariate Gaussian distribution. A GP is completely characterised by its mean function* $m(\mathbf{x}) = \mathbb{E}[g(\mathbf{x})]$ *and its covariance function (or kernel)* $k(\mathbf{x}, \mathbf{x}') = \mathrm{Cov}(g(\mathbf{x}), g(\mathbf{x}'))$*.*

Throughout this thesis we set $m(\mathbf{x}) = 0$, which is standard practice and involves no loss of generality since a non-zero mean can always be absorbed into the kernel. The GP prior over functions is then written

$$g \sim \mathcal{GP}(0,\, k(\mathbf{x}, \mathbf{x}'))\,. \tag{2.1}$$

For any finite collection of input points $X = \{x_1, \ldots, x_n\} \subset \mathcal{X}$, the corresponding function values follow the joint distribution

$$\mathbf{g} = \big(g(x_1), \ldots, g(x_n)\big)^\top \sim \mathcal{N}(\mathbf{0},\, K(X, X))\,, \tag{2.2}$$

where $K(X, X)$ is the $n \times n$ Gram matrix with entries $[K(X, X)]_{ij} = k(x_i, x_j)$.

2.1.3 Posterior Distribution and Predictions

Suppose we observe noisy evaluations $y_i = g(x_i) + \varepsilon_i$, with $\varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_n^2)$, at the training points $X = \{x_1, \ldots, x_n\}$. We collect these in the vector $\mathbf{y} = (y_1, \ldots, y_n)^\top$.

For a new test point x_*, the joint prior distribution of $\mathbf{y}$ and $g(x_*)$ is Gaussian:

$$\begin{bmatrix} \mathbf{y} \\ g(x_*) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0},\ \begin{bmatrix} K(X,X) + \sigma_n^2 I_n & K(X,x_*) \\ K(x_*,X) & k(x_*,x_*) \end{bmatrix} \right), \tag{2.3}$$

where $K(X, x_*) = (k(x_i, x_*))_{i=1,\dots,n}$ is an n-vector. Conditioning on the observed data $\mathbf{y}$ yields the posterior distribution of $g(x_*)$, which is again Gaussian [3]:

$$\mathbb{E}[g(x_*) \mid \mathbf{y}] = K(x_*, X) \left[K(X,X) + \sigma_n^2 I_n\right]^{-1} \mathbf{y}, \tag{2.4}$$

$$\mathrm{Var}[g(x_*) \mid \mathbf{y}] = k(x_*, x_*) - K(x_*, X) \left[K(X,X) + \sigma_n^2 I_n\right]^{-1} K(X, x_*). \tag{2.5}$$

The posterior mean in Equation (2.4) is the GPR estimator of $g(x_*)$. The posterior variance in Equation (2.5) quantifies the uncertainty of this estimate and does not depend on the observed values $\mathbf{y}$, only on the locations of the training and test points.

2.1.4 Kernel Functions and Hyperparameters

The kernel $k(\mathbf{x}, \mathbf{x}')$ encodes prior assumptions about the smoothness and structure of the unknown function. A necessary and sufficient condition for k to be a valid covariance function is that the Gram matrix $K(X, X)$ is positive semi-definite for any finite collection of points [3].

In this thesis, we use the *anisotropic squared exponential* kernel (also called radial basis function or Gaussian kernel):

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\sum_{j=1}^{d} \frac{(x_j - x_j')^2}{2\eta_j^2} \right),$$

where $\eta_j > 0$ is the *length-scale* (or bandwidth) in dimension j. The anisotropic formulation allows each input dimension to contribute differently to the kernel, which is natural for pricing problems where time and state variables live on very different scales.

The parameters $\boldsymbol{\eta} = (\eta_1, \dots, \eta_d)$ are called *hyperparameters*. Their choice controls the smoothness of the estimated function: a small η_j makes the kernel more local in dimension j, while a large η_j enforces a smoother approximation. The squared exponential kernel produces functions that are infinitely differentiable, which is important for the analytical computation of Greeks in Section 2.2.

2.1.5 Hyperparameter Estimation

The hyperparameters $\boldsymbol{\eta}$ and the noise level σ_n^2 are estimated from the data by maximising the log-marginal likelihood of the observations. Integrating out the function values $\mathbf{g}$, the marginal distribution of $\mathbf{y}$ is

$$\mathbf{y} \sim \mathcal{N}\big(\mathbf{0},\ K(X,X) + \sigma_n^2 I_n\big)\,,$$

so that the log-marginal likelihood is

$$\log p(\mathbf{y} \mid X, \boldsymbol{\eta}) = -\frac{1}{2}\,\mathbf{y}^\top \big(K + \sigma_n^2 I\big)^{-1}\mathbf{y} - \frac{1}{2}\log\big|K + \sigma_n^2 I\big| - \frac{n}{2}\log 2\pi, \tag{2.6}$$

where $K = K(X,X)$ for brevity. The first term penalises poor data fit, while the second term penalises model complexity through the log-determinant of the covariance matrix. Maximising Equation (2.6) over $\boldsymbol{\eta}$ therefore balances fit and complexity automatically.

In the CGPR framework introduced in the next section, the hyperparameters are instead calibrated by minimising a validation criterion based on the PDE residual and boundary conditions. This alternative approach is described in detail in Section 2.2 and in Chapter 3.

2.1.6 Analytical Differentiation of the Posterior Mean

A key property of Gaussian processes is that any linear transformation of a GP remains Gaussian. In particular, if $g \sim \mathcal{GP}(0,k)$ and k is differentiable with respect to its arguments, then the derivative process $\partial g/\partial x_j$ is also a GP. Its covariance with g and its own covariance can be computed by differentiating k [3].

This property implies that the posterior mean of the derivative of g with respect to x_j at a test point x_* is obtained by differentiating the posterior mean in Equation (2.4):

$$\frac{\partial}{\partial x_{*,j}}\mathbb{E}[g(x_*) \mid \mathbf{y}] = \frac{\partial K(x_*, X)}{\partial x_{*,j}}\,\big[K(X,X) + \sigma_n^2 I_n\big]^{-1}\mathbf{y}. \tag{2.7}$$

No additional training is required: once the matrix $[K(X,X) + \sigma_n^2 I_n]^{-1}\mathbf{y}$ has been computed, all partial derivatives of the posterior mean are obtained by differentiating only the kernel. This observation extends naturally to linear differential operators, and it is the starting point of the CGPR construction in the next section.

2.2 Constrained Gaussian Process Regression

2.2.1 Motivation

The GPR framework of Section 2.1 treats option pricing as a standard regression: a training set of prices is used to fit a smooth function, and new prices are obtained by evaluating the posterior mean. The limitation is that the training set must be generated by an external method (Monte Carlo, finite differences or Fourier techniques) and nothing in the regression guarantees that the fitted surface satisfies the Feynman–Kac PDE associated with the underlying model. A purely data-driven GPR can therefore produce an accurate interpolation while remaining structurally inconsistent with the model.

The Constrained Gaussian Process Regression (CGPR) framework of Hainaut and Vrins [1] addresses this by incorporating the Feynman–Kac PDE directly into the regression. The training data are no longer simulated prices, instead they are evaluations of the PDE residual at interior points and the terminal payoff at the boundary. The fitted surface therefore satisfies the pricing equation by construction, and no external pricing method is needed.

2.2.2 Problem Formulation

Let $\mathcal{X} \subset \mathbb{R}^p$ be the pricing domain, with interior $\mathcal{X}^\circ$ and boundary $\partial\mathcal{X}$. The pricing function $f : \mathcal{X} \to \mathbb{R}$ satisfies

$$\begin{cases} L_x f(x) = 0, & x \in \mathcal{X}^\circ, \\ f(\bar{x}) = h(\bar{x}), & \bar{x} \in \partial\mathcal{X}, \end{cases} \tag{2.8}$$

where L_x is the Feynman–Kac differential operator and h is the boundary function. The unknown function f is modelled by a GP prior

$$g(x) \sim \mathcal{GP}(0,\, k(x, \tilde{x}))\,, \tag{2.9}$$

where k is a differentiable kernel.

2.2.3 Training Sets and Pseudo-Observations

Two training sets are introduced. The first is sampled in the interior:

$$X^\circ = \{x_i^\circ\}_{i=1}^m \subset \mathcal{X}^\circ.$$

At each interior point, the Feynman–Kac equation requires the PDE residual to vanish. This is incorporated through noisy pseudo-observations

$$z_i = L_{x_i^\circ} g(x_i^\circ) + \varepsilon_i^*, \qquad \varepsilon_i^* \sim \mathcal{N}(0, \sigma^{*2}), \tag{2.10}$$

with target values $z_i = 0$ for all $i = 1, \ldots, m$.

The second training set is sampled on the boundary:

$$\bar{X} = \{\bar{x}_i\}_{i=1}^d \subset \partial\mathcal{X}.$$

At boundary points, the pricing function is constrained to match h:

$$h_i = h(\bar{x}_i) + \bar{\varepsilon}_i, \qquad \bar{\varepsilon}_i \sim \mathcal{N}(0, \sigma^{*2}). \tag{2.11}$$

The parameter σ^* has both a statistical and a numerical role. From the GP point of view, it represents the noise added to the pseudo-observations. From the numerical point of view, it regularises the covariance matrix. Adding $\sigma^{*2} I$ on the diagonal makes the matrix inversion more stable, especially when the training points are close to each other. The choice of σ^* is discussed in Chapter 3.

A key difference with standard GPR is that no simulated or market prices appear in the training set. The vector $\mathbf{z} = (z_i)$ contains zeros (PDE residual targets) and $\mathbf{h} = (h_i)$ contains payoff evaluations. Both are collected in the observation vector

$$\mathbf{o} = \begin{pmatrix} \mathbf{z} \\ \mathbf{h} \end{pmatrix} \in \mathbb{R}^{m+d}. \tag{2.12}$$

2.2.4 Basis Function Representation

To derive the CGPR estimator rigorously, we work with a finite basis function representation of the GP. Without loss of generality, any kernel k can be approximated by a finite sum for a given accuracy, so there exist n basis functions

$\phi(x) = (\phi_j(x))_{j=1,\dots,n}$ such that

$$k(x, x') = \phi(x)^\top \phi(x'), \qquad \forall\, x, x' \in \mathcal{X}. \tag{2.13}$$

The GP prior $g \sim \mathcal{GP}(0, k)$ is then represented as

$$g(x) = \sum_{j=1}^{n} w_j \phi_j(x) = w^\top \phi(x), \tag{2.14}$$

where the weight vector $w \sim \mathcal{N}(\mathbf{0}, I_n)$.

Since L_x is a linear operator, applying it to g gives

$$L_x g(x) = \sum_{j=1}^{n} w_j \, L_x \phi_j(x) = w^\top \phi_L(x), \tag{2.15}$$

where $\phi_L(x) = (L_x \phi_j(x))_{j=1,\dots,n}$ is the vector of differentiated basis functions. This shows that $L_x g(x)$ is a linear combination of Gaussian weights w, and is therefore itself Gaussian. More generally, any collection of values drawn from $\{g(x), L_x g(x)\}$ follows a multivariate Gaussian distribution, fully determined by the kernel through the relations

$$\mathrm{Cov}(g(x), g(x')) = \phi(x)^\top \phi(x') = k(x, x'), \tag{2.16}$$

$$\mathrm{Cov}(L_x g(x), g(x')) = \phi_L(x)^\top \phi(x') = L_x k(x, x'), \tag{2.17}$$

$$\mathrm{Cov}(L_x g(x), L_{x'} g(x')) = \phi_L(x)^\top \phi_L(x') = L_x L_{x'} k(x, x'). \tag{2.18}$$

2.2.5 Conditional Distribution of the Weights

Define the $m \times n$ matrix $\mathring{\Phi}_L$ and the $d \times n$ matrix $\bar{\Phi}$ by

$$(\mathring{\Phi}_L)_{ij} = L_{x_i^\circ} \phi_j(x_i^\circ), \qquad (\bar{\Phi})_{ij} = \phi_j(\bar{x}_i).$$

Conditionally on w, the joint distribution of (Z, H) — the random vectors corresponding to the pseudo-observations in Equations (2.10)–(2.11) is

$$\begin{pmatrix} Z \\ H \end{pmatrix} \Bigg| w \sim \mathcal{N}\left(\begin{pmatrix} \mathring{\Phi}_L w \\ \bar{\Phi} w \end{pmatrix}, \, \sigma^{*2} I_{m+d} \right). \tag{2.19}$$

Proposition 2.2.1 (Conditional distribution of weights [1])**.** *The posterior distri-*

bution of w given the observations (z, h) is

$$w \mid z, h \;\sim\; \mathcal{N}(\mu_{z,h},\, \Sigma_{z,h})\,, \tag{2.20}$$

where

$$\Sigma_{z,h} = \Big(I_n + \sigma^{*-2}\mathring{\Phi}_L^\top \mathring{\Phi}_L + \sigma^{*-2}\bar{\Phi}^\top \bar{\Phi}\Big)^{-1}, \tag{2.21}$$

$$\mu_{z,h} = \sigma^{*-2}\,\Sigma_{z,h}\left(\mathring{\Phi}_L^\top z + \bar{\Phi}^\top h\right). \tag{2.22}$$

Proof. We apply Bayes' theorem to the prior $w \sim \mathcal{N}(0, I_n)$ and the likelihood in Equation (2.19). The posterior density satisfies

$$\log p(w \mid z, h) \propto -\frac{1}{2} w^\top w - \frac{1}{2\sigma^{*2}} \left\| z - \mathring{\Phi}_L w \right\|^2 - \frac{1}{2\sigma^{*2}} \left\| h - \bar{\Phi} w \right\|^2.$$

Expanding and collecting the quadratic and linear terms in w:

$$\begin{aligned} \text{Quadratic:} \quad & w^\top \Big(I_n + \sigma^{*-2}\mathring{\Phi}_L^\top \mathring{\Phi}_L + \sigma^{*-2}\bar{\Phi}^\top \bar{\Phi}\Big)\, w = w^\top \Sigma_{z,h}^{-1} w, \\ \text{Linear:} \quad & -2 w^\top \sigma^{*-2} \left(\mathring{\Phi}_L^\top z + \bar{\Phi}^\top h\right). \end{aligned}$$

Completing the square in w identifies the posterior as $\mathcal{N}(\mu_{z,h}, \Sigma_{z,h})$ with $\Sigma_{z,h}$ and $\mu_{z,h}$ as defined above. □

2.2.6 Covariance Matrix C

The matrix C is the covariance matrix of the observation vector $\mathbf{o}$. We compute $\mathrm{Cov}(\mathbf{o}, \mathbf{o})$ bloc by bloc using Equations (2.16)–(2.18).

Case 1: two interior residuals z_i and z_j.

$$\begin{aligned} \mathrm{Cov}(z_i, z_j) &= \mathrm{Cov}\Big(L_{x_i^\circ} g(x_i^\circ) + \varepsilon_i^*,\; L_{x_j^\circ} g(x_j^\circ) + \varepsilon_j^*\Big) \\ &= L_{x_i^\circ} L_{x_j^\circ}\, k(x_i^\circ, x_j^\circ) + \sigma^{*2}\, \mathbf{1}_{i=j}, \end{aligned}$$

where the noise terms ε_i^* are independent of g and of each other.

Case 2: an interior residual z_i and a boundary value h_j.

$$\begin{aligned} \mathrm{Cov}(z_i, h_j) &= \mathrm{Cov}\big(L_{x_i^\circ} g(x_i^\circ) + \varepsilon_i^*,\; g(\bar{x}_j) + \bar{\varepsilon}_j\big) \\ &= L_{x_i^\circ}\, k(x_i^\circ, \bar{x}_j), \end{aligned}$$

since ε_i^* and $\bar{\varepsilon}_j$ are independent of g and of each other.

Case 3: two boundary values h_i and h_j.

$$\begin{aligned}\mathrm{Cov}(h_i, h_j) &= \mathrm{Cov}(g(\bar{x}_i) + \bar{\varepsilon}_i,\ g(\bar{x}_j) + \bar{\varepsilon}_j) \\ &= k(\bar{x}_i, \bar{x}_j) + \sigma^{*2}\, \mathbf{1}_{i=j}.\end{aligned}$$

Assembling the three cases gives

$$C = \mathrm{Cov}(\mathbf{o}, \mathbf{o}) = \sigma^{*2} I_{m+d} + \begin{bmatrix} k_{L^2}(X^\circ, X^\circ) & k_L(X^\circ, \bar{X}) \\ k_L(X^\circ, \bar{X})^\top & k(\bar{X}, \bar{X}) \end{bmatrix}, \tag{2.23}$$

where $\sigma^{*2} I_{m+d}$ collects the diagonal noise contributions from Cases 1 and 3, and the three blocks are

$$\begin{aligned}\big[k_{L^2}(X^\circ, X^\circ)\big]_{ij} &= L_{x_i^\circ} L_{x_j^\circ}\, k(x_i^\circ, x_j^\circ), \\ \big[k_L(X^\circ, \bar{X})\big]_{ij} &= L_{x_i^\circ}\, k(x_i^\circ, \bar{x}_j), \\ \big[k(\bar{X}, \bar{X})\big]_{ij} &= k(\bar{x}_i, \bar{x}_j).\end{aligned}$$

2.2.7 CGPR Price Estimator

We now derive the main result. The CGPR estimator is defined as the conditional expectation of $g(x)$ given the observations (z, h). From the basis function representation in Equation (2.14):

$$\mathbb{E}[g(x) \mid z, h] = \phi(x)^\top\, \mathbb{E}[w \mid z, h] = \phi(x)^\top \mu_{z,h}. \tag{2.24}$$

Substituting Equation (2.22):

$$\mathbb{E}[g(x) \mid z, h] = \sigma^{*-2}\, \phi(x)^\top \Sigma_{z,h} \left(\mathring{\Phi}_L^\top z + \bar{\Phi}^\top h \right). \tag{2.25}$$

This expression involves the inverse of the $n \times n$ matrix $\Sigma_{z,h}^{-1}$. Since n can be very large (infinite in the kernel limit), we use the Woodbury matrix identity to rewrite it in terms of the $(m+d) \times (m+d)$ matrix C, which has fixed dimension. The Woodbury identity states that for matrices of compatible dimensions:

$$(I_n + UV)^{-1} U = U\, (I_{m+d} + VU)^{-1}, \tag{2.26}$$

where the proof is given in Appendix A.1. Applying this identity with

$$U = \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L^\top & \sigma^{*-1}\bar{\Phi}^\top \end{pmatrix}, \qquad V = \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L \\ \sigma^{*-1}\bar{\Phi} \end{pmatrix},$$

the expression in Equation (2.25) transforms into [1]:

$$\mathbb{E}[g(x) \mid z, h] = \begin{pmatrix} \phi(x)^\top \mathring{\Phi}_L^\top \\ \phi(x)^\top \bar{\Phi}^\top \end{pmatrix}^\top C^{-1} \begin{pmatrix} z \\ h \end{pmatrix}.$$

Recognising that $\phi(x)^\top \mathring{\Phi}_L^\top = (\mathring{\Phi}_L \phi(x))^\top$ with $(\mathring{\Phi}_L \phi(x))_i = L_{x_i^\circ} k(x_i^\circ, x)$, and similarly $(\bar{\Phi}\phi(x))_i = k(\bar{x}_i, x)$, gives the following result.

Proposition 2.2.2 (CGPR Price Estimator [1])**.** *Define the* $(m+d)$*-vector*

$$\boldsymbol{\beta}(x) = \begin{bmatrix} \big(L_{x_i^\circ} k(x_i^\circ, x)\big)_{i=1,\dots,m} \\ \big(k(\bar{x}_i, x)\big)_{i=1,\dots,d} \end{bmatrix}. \tag{2.27}$$

The CGPR estimator of the pricing function is

$$\hat{f}(x) = \mathbb{E}[g(x) \mid \mathbf{o}] = \boldsymbol{\beta}(x)^\top C^{-1} \mathbf{o}, \tag{2.28}$$

with posterior variance

$$\mathrm{Var}(g(x) \mid \mathbf{o}) = k(x,x) - \boldsymbol{\beta}(x)^\top C^{-1} \boldsymbol{\beta}(x). \tag{2.29}$$

The posterior variance is derived in Appendix A.2.

The vector $C^{-1}o$ is computed once for a fixed model configuration and does not depend on the prediction point x. This is the costly part of the method, since it requires solving a linear system of size $m+d$. Once this quantity is available, a new price only requires the computation of $\beta(x)$ and a dot product:

$$\widehat{f}(x) = \beta(x)^\top C^{-1} o.$$

Thus, for a fixed maturity and fixed model parameters, the expensive computation is done once, while prices at new points can be evaluated quickly.

Once the system has been solved, the vector $C^{-1}\mathbf{o}$ is fixed. This is the expensive part of the method, since it requires solving a linear system of size $(m+d)$. For a new point x, the estimator only needs the vector $\beta(x)$ and the price is obtained

from the product $\beta(x)^\top C^{-1}\mathbf{o}$. The cost of one new prediction is linear in $m+d$, while the costly matrix step is performed only once. This separation explains why CGPR becomes useful when many prices have to be evaluated.

2.2.8 Estimator of the PDE Residual

The same argument applies to the transformed process $L_x g(x)$. From Equation (2.15):

$$\mathbb{E}[L_x g(x) \mid z, h] = \phi_L(x)^\top \mu_{z,h}.$$

Applying the same Woodbury transformation as above, with $\phi_L(x)$ in place of $\phi(x)$, gives the following result.

Proposition 2.2.3 (PDE Residual Estimator [1]). *For $x \in \mathcal{X}^\circ$, define the $(m+d)$-vector*

$$\boldsymbol{\alpha}(x) = \begin{bmatrix} \big(L_{x_i^\circ} L_x\, k(x_i^\circ, x)\big)_{i=1,\dots,m} \\ \big(L_x\, k(\bar{x}_i, x)\big)_{i=1,\dots,d} \end{bmatrix}. \tag{2.30}$$

The posterior mean of the PDE residual is

$$\mathbb{E}[L_x g(x) \mid \mathbf{o}] = \boldsymbol{\alpha}(x)^\top C^{-1}\mathbf{o}, \tag{2.31}$$

with posterior variance

$$\mathrm{Var}(L_x g(x) \mid \mathbf{o}) = \Big[L_x L_{\tilde{x}}\, k(x, \tilde{x})\Big]_{\tilde{x}=x} - \boldsymbol{\alpha}(x)^\top C^{-1}\boldsymbol{\alpha}(x). \tag{2.32}$$

If the posterior mean of $L_x g(x)$ remains close to zero across the domain, the fitted surface is consistent with the Feynman–Kac equation. This provides a direct diagnostic criterion for the quality of the CGPR approximation, independently of any external benchmark.

2.2.9 Greeks via Analytical Differentiation

The estimator $\hat{f}(x) = \boldsymbol{\beta}(x)^\top C^{-1}\mathbf{o}$ is linear in $\boldsymbol{\beta}(x)$, and $C^{-1}\mathbf{o}$ is fixed after training. The Greeks are therefore obtained by differentiating $\boldsymbol{\beta}(x)$ with respect to the relevant state variable, with no additional computation required. For the Heston

model, where $x = (t, s, v)$:

$$\widehat{\Delta}(x) = \frac{\partial \hat{f}}{\partial s}(x) = \frac{\partial \boldsymbol{\beta}(x)^\top}{\partial s} C^{-1}\mathbf{o}, \tag{2.33}$$

$$\widehat{\mathcal{V}}(x) = \frac{\partial \hat{f}}{\partial v}(x) = \frac{\partial \boldsymbol{\beta}(x)^\top}{\partial v} C^{-1}\mathbf{o}, \tag{2.34}$$

$$\widehat{\Theta}(x) = \frac{\partial \hat{f}}{\partial t}(x) = \frac{\partial \boldsymbol{\beta}(x)^\top}{\partial t} C^{-1}\mathbf{o}. \tag{2.35}$$

Since the squared exponential kernel is infinitely differentiable, all partial derivatives of $\boldsymbol{\beta}(x)$ are available in closed form. The explicit expressions are derived in Section 2.4 and listed in Appendix A.3.

2.3 CGPR Algorithm

The previous sections derived all the mathematical ingredients of the CGPR estimator. We now summarise the full procedure. The kernel bandwidths $\boldsymbol{\eta} = (\eta_t, \eta_s, \eta_v)$ are calibrated by minimising the internal validation criterion

$$J(\boldsymbol{\eta}) = \frac{1}{m'} \sum_{x_i' \in X^{\circ\prime}} \left(L_{x_i'} \hat{f}(x_i') \right)^2 + \frac{1}{d'} \sum_{\bar{x}_j' \in \bar{X}'} \left(\hat{f}(\bar{x}_j') - h(\bar{x}_j') \right)^2, \tag{2.36}$$

where $X^{\circ\prime}$ and $\bar{X}'$ are validation sets independent of X° and $\bar{X}$.

Algorithm 1. CGPR approximation of $\hat{f}(x)$ [1].

Offline step — performed once per maturity.

1. Sample m interior points $\{x_i^\circ\}_{i=1}^m \subset \mathcal{X}^\circ$ uniformly and set $z_i = 0$. Sample d boundary points $\{\bar{x}_j\}_{j=1}^d \subset \partial\mathcal{X}$, compute $h_j = h(\bar{x}_j)$, and form $\mathbf{o} = (z_1, \ldots, z_m, h_1, \ldots, h_d)^\top$. Sample two additional validation sets $X^{\circ\prime}$ and $\bar{X}'$.

2. Calibrate $\boldsymbol{\eta}^*$ by minimising $J(\boldsymbol{\eta})$ in Equation (2.36) via Nelder–Mead over $\log(\boldsymbol{\eta})$.

3. Assemble C in Equation (2.23) using $\boldsymbol{\eta}^*$, with blocks computed via Equations (2.41), (2.39) and (2.37). Factorise C and compute $\boldsymbol{\gamma} = C^{-1}\mathbf{o}$ at cost $\mathcal{O}((m+d)^3)$.

Online phase — repeated for each prediction point x.

4. Compute $\boldsymbol{\beta}(x)$ via Equation (2.27) using Equation (2.40), and evaluate

$$\hat{f}(x) = \boldsymbol{\beta}(x)^\top \boldsymbol{\gamma}, \qquad \mathrm{Var}(g(x) \mid \mathbf{o}) = k(x,x) - \boldsymbol{\beta}(x)^\top C^{-1} \boldsymbol{\beta}(x),$$

at cost $\mathcal{O}(m+d)$. If Greeks are required:

$$\widehat{\Delta}(x) = \partial_s \boldsymbol{\beta}(x)^\top \boldsymbol{\gamma}, \qquad \widehat{\mathcal{V}}(x) = \partial_v \boldsymbol{\beta}(x)^\top \boldsymbol{\gamma}, \qquad \widehat{\Theta}(x) = \partial_t \boldsymbol{\beta}(x)^\top \boldsymbol{\gamma},$$

using derivatives from Appendix A.4.

The dominant cost is Step 3 at $\mathcal{O}((m+d)^3)$, performed once per maturity. Each subsequent price evaluation costs only $\mathcal{O}(m+d)$. The sensitivity of the results to m, d and σ^* is analysed in Chapter 3.

2.4 Application to the Heston Model

We now instantiate the CGPR framework to the pricing of European call options under the Heston model introduced in Section 1.3. The state variable is $x = (t, s, v) \in \mathcal{X} = [0, T] \times \mathbb{R}_+^2$, where s denotes the asset price and v the instantaneous variance. The objective of this section is to derive the two quantities $L_{\tilde{x}} k(x, \tilde{x})$ and $L_x L_{\tilde{x}} k(x, \tilde{x})$ that enter the covariance matrix C in Equation (2.23), and to give the explicit expressions for the Greeks.

2.4.1 The Feynman–Kac Operator and Boundary Conditions

The Feynman–Kac operator for the Heston model is L_x defined in Equation (1.6), which we recall here:

$$L_x f = \frac{\partial f}{\partial t} + rs\,\frac{\partial f}{\partial s} + \kappa(\gamma - v)\,\frac{\partial f}{\partial v} + \frac{1}{2}vs^2\,\frac{\partial^2 f}{\partial s^2} + \rho\sigma vs\,\frac{\partial^2 f}{\partial s\,\partial v} + \frac{1}{2}\sigma^2 v\,\frac{\partial^2 f}{\partial v^2} - rf.$$

The pricing domain is truncated to $\mathcal{X} = [0, T] \times (0, s_{\max}] \times \mathbb{R}_+$, where $s_{\max}$ is chosen large enough to contain the full validation range. The boundary $\partial\mathcal{X}$ has three components, at each of which the boundary function h is defined as follows:

$$h(\bar{x}) = \begin{cases} (s-K)^+ & \bar{x} = (T, s, v), \quad \text{(terminal payoff)}, \\ 0 & \bar{x} = (t, 0, v), \quad \text{(option worthless at } s = 0), \\ s - Ke^{-r(T-t)} & \bar{x} = (t, s_{\max}, v), \quad \text{(deep in-the-money approximation)}. \end{cases}$$

These boundary conditions have a simple interpretation. At maturity, the option value is equal to its payoff. When $s = 0$, the stock price stays at zero, so the call is worthless. At the upper boundary $s = s_{\max}$, the call is treated as a deep in-the-money option and its value is approximated by

$$s_{\max} - Ke^{-r(T-t)}.$$

This approximation becomes exact when $s \to \infty$.

2.4.2 Anisotropic Gaussian Kernel

For the three-dimensional state vector (t, s, v), we use the anisotropic squared exponential kernel

$$k(x, \tilde{x}) = \exp\left(-\frac{(t-\tilde{t})^2}{2\eta_t^2} - \frac{(s-\tilde{s})^2}{2\eta_s^2} - \frac{(v-\tilde{v})^2}{2\eta_v^2}\right), \tag{2.37}$$

with bandwidths $\eta_t, \eta_s, \eta_v > 0$. To lighten notation, define the normalised differences

$$\delta_t = \frac{t-\tilde{t}}{\eta_t^2}, \qquad \delta_s = \frac{s-\tilde{s}}{\eta_s^2}, \qquad \delta_v = \frac{v-\tilde{v}}{\eta_v^2}. \tag{2.38}$$

The partial derivatives of k with respect to (t, s, v) are:

$$\partial_t k = -\delta_t k, \quad \partial_s k = -\delta_s k, \quad \partial_v k = -\delta_v k,$$

$$\partial_{ss}k = \left(\delta_s^2 - \frac{1}{\eta_s^2}\right)k, \quad \partial_{vv}k = \left(\delta_v^2 - \frac{1}{\eta_v^2}\right)k, \quad \partial_{sv}k = \delta_s\delta_v k.$$

Derivatives with respect to $(\tilde{t}, \tilde{s}, \tilde{v})$ introduce opposite signs in the first-order terms since, for example, $\partial_{\tilde{s}}k = +\delta_s k$.

2.4.3 First Operator: $L_{\tilde{x}}\, k(x, \tilde{x})$

Applying $L_{\tilde{x}}$ to the kernel (that is, applying the operator in the tilde variable) gives a product of the form

$$L_{\tilde{x}}\, k(x, \tilde{x}) = \ell_{\tilde{x}}(x, \tilde{x}) \cdot k(x, \tilde{x}), \tag{2.39}$$

where $\ell_{\tilde{x}}$ is a scalar coefficient obtained by collecting all derivative terms.

Proposition 2.4.1 (First operator [1])**.**

$$\ell_{\tilde{x}} = \delta_t + r\tilde{s}\,\delta_s + \kappa(\gamma - \tilde{v})\,\delta_v + \frac{1}{2}\tilde{s}^2\tilde{v}\left(\delta_s^2 - \frac{1}{\eta_s^2}\right) + \rho\sigma\tilde{s}\tilde{v}\,\delta_s\delta_v + \frac{1}{2}\sigma^2\tilde{v}\left(\delta_v^2 - \frac{1}{\eta_v^2}\right) - r. \tag{2.40}$$

Proof. Each term of $L_{\tilde{x}}$ is applied to k using the tilde derivative sign convention. Since $\partial_{\tilde{t}}k = +\delta_t k$ and similarly for $\tilde{s}$ and $\tilde{v}$, the first-order terms give $\delta_t + r\tilde{s}\delta_s + \kappa(\gamma - \tilde{v})\delta_v$. The second-order terms give $\frac{1}{2}\tilde{s}^2\tilde{v}(\delta_s^2 - \eta_s^{-2}) + \rho\sigma\tilde{s}\tilde{v}\delta_s\delta_v + \frac{1}{2}\sigma^2\tilde{v}(\delta_v^2 - \eta_v^{-2})$. Summing all terms and factoring out k gives Equation (2.40). □

2.4.4 Double Operator: $L_x L_{\tilde{x}}\, k(x, \tilde{x})$

The interior–interior block of C requires applying L_x to the product $\ell_{\tilde{x}} \cdot k$ from Equation (2.39). Using the Leibniz rule on each term of L_x:

$$\begin{aligned} L_x L_{\tilde{x}} k = {} & \ell_{\tilde{x}}(\partial_t k - rk) + k\,\partial_t \ell_{\tilde{x}} \\ & + rs(k\,\partial_s \ell_{\tilde{x}} + \ell_{\tilde{x}}\,\partial_s k) + \kappa(\gamma - v)(k\,\partial_v \ell_{\tilde{x}} + \ell_{\tilde{x}}\,\partial_v k) \\ & + \tfrac{1}{2}s^2 v(k\,\partial_{ss}\ell_{\tilde{x}} + 2\,\partial_s \ell_{\tilde{x}}\,\partial_s k + \ell_{\tilde{x}}\,\partial_{ss}k) \\ & + \tfrac{1}{2}\sigma^2 v(k\,\partial_{vv}\ell_{\tilde{x}} + 2\,\partial_v \ell_{\tilde{x}}\,\partial_v k + \ell_{\tilde{x}}\,\partial_{vv}k) \\ & + \rho\sigma s v(k\,\partial_{sv}\ell_{\tilde{x}} + \partial_s \ell_{\tilde{x}}\,\partial_v k + \partial_v \ell_{\tilde{x}}\,\partial_s k + \ell_{\tilde{x}}\,\partial_{sv}k)\,. \end{aligned} \tag{2.41}$$

All partial derivatives of $\ell_{\tilde{x}}$ with respect to (t, s, v) are available in closed form and listed in Appendix A.3. With Equations (2.39) and (2.41), the three blocks of C in Equation (2.23) are fully computable: the interior–interior block uses Equa-

tion (2.41), the interior–boundary block uses Equation (2.39), and the boundary–boundary block uses Equation (2.37) directly.

Once C is assembled and factorised at offline cost $\mathcal{O}((m+d)^3)$, the vector $C^{-1}\mathbf{o}$ is computed once. Any new price is then evaluated at cost $\mathcal{O}(m+d)$ via Equation (2.28).

2.4.5 Expressions for the Greeks

Since $\hat{f}(x) = \boldsymbol{\beta}(x)^\top C^{-1}\mathbf{o}$ and $C^{-1}\mathbf{o}$ is fixed after training, each Greek reduces to differentiating $\boldsymbol{\beta}(x)$. The vector $\boldsymbol{\beta}(x)$ has two parts: the first m entries are $\beta_i(x) = \ell_{x_i^\circ}(x_i^\circ, x) \cdot k(x_i^\circ, x)$ and the last d entries are $\beta_{m+j}(x) = k(\bar{x}_j, x)$.

We develop the Delta $\widehat{\Delta} = \partial_s \hat{f}$ explicitly as an illustration. For the last d entries:

$$\partial_s k(\bar{x}_j, x) = -\delta_s \, k(\bar{x}_j, x).$$

For the first m entries, the product rule gives

$$\partial_s \beta_i(x) = \Big(\partial_s \ell_{x_i^\circ}\Big)\, k(x_i^\circ, x) - \delta_s \, \ell_{x_i^\circ}\, k(x_i^\circ, x), \tag{2.42}$$

where $\delta_s = (s - s_i^\circ)/\eta_s^2$ and $\partial_s \ell_{x_i^\circ}$ is the derivative of $\ell_{\tilde{x}}$ with respect to s (the non-tilde variable), evaluated at $(x, \tilde{x}) = (x, x_i^\circ)$. From Appendix A.3:

$$\partial_s \ell_{\tilde{x}} = \frac{1}{\eta_s^2}\Big(r\tilde{s} + \tilde{s}^2\tilde{v}\,\delta_s + \rho\sigma\tilde{s}\tilde{v}\,\delta_v\Big). \tag{2.43}$$

Assembling Equation (A.8) for all i and j gives the full vector $\partial_s \boldsymbol{\beta}(x)$, and the Delta is

$$\widehat{\Delta}(x) = \frac{\partial \boldsymbol{\beta}(x)^\top}{\partial s} C^{-1}\mathbf{o}.$$

The Vega $\widehat{\mathcal{V}} = \partial_v \hat{f}$ and Theta $\widehat{\Theta} = \partial_t \hat{f}$ follow the same structure, replacing δ_s by δ_v or δ_t and using the corresponding derivatives of $\ell_{\tilde{x}}$ from Appendix A.3.

CHAPTER 3

EXPERIMENTAL FRAMEWORK

3.1 Validation in the Black–Scholes Framework

The first experiment validates the CGPR implementation in the Black–Scholes framework. Since exact prices and Greeks are available through the closed-form formula, any discrepancy between the CGPR approximation and the analytical solution reflects the numerical approximation error of the method itself. This controlled setting also allows us to study the effect of the main implementation choices (training size, regularisation parameter and kernel bandwidths) before moving to the more demanding Heston experiment.

In the Black–Scholes model, the stock price follows

$$dS_t = rS_t\,dt + \sigma S_t\,dW_t,$$

and the pricing function $f(t,s)$ of a European call with strike K and maturity T satisfies $L_x f = 0$ with the two-dimensional Feynman–Kac operator

$$L_x f = \frac{\partial f}{\partial t} + rs\,\frac{\partial f}{\partial s} + \frac{1}{2}\sigma^2 s^2\,\frac{\partial^2 f}{\partial s^2} - rf.$$

This is a one-dimensional special case of the general operator derived in Section 1.3, with v fixed and $\rho = 0$. The analytical benchmark is the Black–Scholes formula:

$$C_{\mathrm{BS}}(t,s) = s\,\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2),$$

with

$$d_1 = \frac{\ln(s/K) + (r + \frac{1}{2}\sigma^2)(T-t)}{\sigma\sqrt{T-t}}, \qquad d_2 = d_1 - \sigma\sqrt{T-t}.$$

This formula is used only as a benchmark. The CGPR estimator is trained from the Feynman–Kac residual and the boundary conditions; no Black–Scholes prices are used in the training set.

3.1.1 Implementation Choices

Since the stock price is unbounded, the computation is done on a finite domain,

$$[0, T) \times (0, s_{\max}], \qquad s_{\max} = 200,$$

This domain covers the validation range $S_0 \in [50, 150]$. The boundary sample $\bar{X}$ is split into three parts:

$$\bar{X}_0 = \{(t, 0)\}, \qquad \bar{X}_{\max} = \{(t, s_{\max})\}, \qquad \bar{X}_T = \{(T, s)\},$$

with corresponding boundary values

$$f(t, 0) = 0, \qquad f(t, s_{\max}) \simeq s_{\max} - Ke^{-r(T-t)}, \qquad f(T, s) = (s - K)^+.$$

The boundary points are placed approximately equally on these three parts. This allows the CGPR surface to be guided not only by the terminal payoff, but also by the behaviour at the lower and upper stock-price boundaries.

The regularisation parameter is fixed at $\sigma^* = 0.01$ throughout the Black–Scholes experiment. As discussed in Section 2.3, a smaller value enforces the PDE and boundary constraints more tightly but risks an ill-conditioned system, while a larger value improves stability at the cost of a bias in the fitted surface. The value $\sigma^* = 0.01$ provides a good balance in this two-dimensional setting. The same baseline value is retained in the Heston experiment, where its effect is analysed in Section 3.2.2.

The kernel bandwidths (η_t, η_1) are calibrated by minimising the criterion $J(\boldsymbol{\eta})$ defined in Equation (2.36) via Nelder–Mead over $\log(\boldsymbol{\eta})$. The optimisation is performed separately for each maturity. The starting point is $(\eta_t^{(0)}, \eta_1^{(0)}) = (0.75, 14)$, based on the values reported in [1] for the Black–Scholes case.

The financial parameters used throughout this section are $r = 5\%$, $\sigma = 25\%$ and $K = 100$. The validation grid consists of 51 initial stock prices $S_0 \in [50, 150]$ for each of the six maturities $T \in \{0.5, 1, 1.5, 2, 2.5, 3\}$, giving 306 validation options.

3.1.2 Convergence with Respect to the Training Size

We first study the effect of the number of training points on pricing accuracy, fixing the maturity at $T = 1$ and varying $m = d$ from 200 to 600. The objective is to assess whether imposing the Feynman–Kac constraint and the boundary conditions on a larger set of points improves the quality of the CGPR approximation.

Table 3.1 – Convergence of the CGPR approximation in the Black–Scholes case ($T = 1$, $\sigma^* = 0.01$, $s_{\max} = 200$).

$m = d$	MAE	MRE ITM	Mean Std.	η_t	η_s
200	0.0941	0.34%	0.0025	0.8434	23.0119
400	0.0435	0.22%	0.0052	0.3435	12.0764
600	0.0062	0.01%	0.0017	0.2123	20.0860

Both the MAE and the MRE on in-the-money options decrease when m increases. Between $m = 200$ and $m = 600$, the MAE falls from 0.0941 to 0.0062, while the MRE decreases from 0.34% to 0.01%. This shows that adding more training constraints improves the approximation of the Black–Scholes pricing surface.

For an at-the-money call option with $S_0 = K = 100$ and $T = 1$, the Black–Scholes price is of the order of ten euros under the parameters of Table 3.1. The relative errors in Table 3.1 therefore correspond to small pricing errors in absolute terms.

The posterior standard deviation does not decrease monotonically with m. As explained in Section 2.2, it measures the internal uncertainty of the Gaussian process posterior. It depends on the location of the training points and on the calibrated bandwidths. It should therefore not be read as a direct measure of the error against the Black–Scholes benchmark.

The bandwidth η_1 is also non-monotone. It decreases from 23.01 at $m = 200$ to 12.08 at $m = 400$, before increasing again to 20.09 at $m = 600$. This is not surprising, since the bandwidths are recalibrated for each training set and the criterion $J(\boldsymbol{\eta})$ is not convex. The values of $\boldsymbol{\eta}$ should therefore be interpreted together with the training configuration used to obtain them.

Although $m = d = 600$ gives the lowest error in Table 3.1, we retain $m = d = 400$ in the following experiments as a compromise between accuracy and computational cost.

3.1.3 Pricing Results

We now fix $m = d = 400$ and evaluate the CGPR approximation over the full validation grid of 300 options, covering six maturities and 50 initial stock prices $S_0 \in [50, 150]$.

Table 3.2 – CGPR pricing errors against Black–Scholes closed-form prices ($m = d = 400$, $\sigma^* = 0.01$, $s_{\max} = 200$).

T	MAE	MRE ITM	Std.	$\lvert\Delta_{\text{err}}\rvert$	$\lvert\Gamma_{\text{err}}\rvert$	$\lvert\Theta_{\text{err}}\rvert$	η_t	η_1
0.5	0.0163	0.05%	0.0037	0.0020	0.000404	0.7365	0.1702	11.9892
1.0	0.0435	0.22%	0.0052	0.0027	0.000512	0.7043	0.3435	12.0764
1.5	0.0621	0.28%	0.0032	0.0024	0.000380	0.3043	0.7591	14.2717
2.0	0.1066	0.41%	0.0131	0.0042	0.000656	1.0981	0.4032	15.0357
2.5	0.1242	0.42%	0.0127	0.0034	0.000520	0.8788	0.4983	16.1106
3.0	0.1132	0.38%	0.0120	0.0021	0.000280	0.6289	0.5734	17.6291

The MRE on in-the-money options stays below 0.42% across all maturities. To put this in perspective, a MRE of 0.42% on an option worth 10€ represents an absolute error of about 4 cents; a level of precision that is acceptable for most practical pricing applications, given that the dominant source of error in a real setting is model misspecification rather than numerical approximation.

The MAE increases overall with maturity, from 0.0163 at $T = 0.5$ to 0.1242 at $T = 2.5$, before decreasing slightly at $T = 3$. This is mainly due to the fact that longer-maturity options have larger prices in absolute terms. A similar relative error can therefore lead to a larger absolute error. The MRE on in-the-money options remains low across all maturities and never exceeds 0.42%.

The bandwidth η_1 also increases with maturity, from 11.99 at $T = 0.5$ to 17.63 at $T = 3$. This means that the CGPR fit becomes smoother in the stock-price direction for longer maturities. This is consistent with the behaviour of call prices, with a price curve that becomes smoother around the strike when maturity increases. The time bandwidth η_t is less regular, but it remains of the same order of magnitude across maturities.

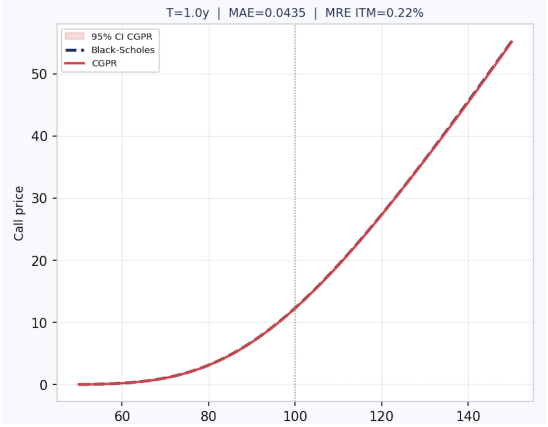

Figure 3.1 – CGPR and Black–Scholes call prices for $T = 1$ ($K = 100$, $r = 5\%$, $\sigma = 25\%$, $m = d = 400$, $\sigma^* = 0.01$, $s_{\max} = 200$).

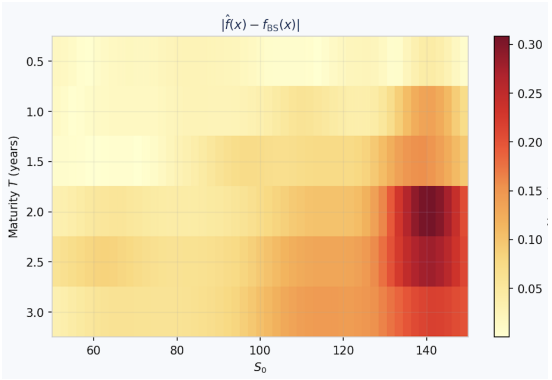

Figure 3.2 – Absolute pricing errors of CGPR against Black–Scholes across maturities and spot values.

Figure 3.1 compares the CGPR prices with the Black–Scholes benchmark for $T = 1$. The two curves are almost identical over the full range of spot values. The vertical line marks the strike $K = 100$, and the title reports a MAE of 0.0435 and a MRE of 0.22% on in-the-money options. The posterior confidence interval is also very narrow, which shows that the fitted surface is stable in this Black–Scholes validation case.

Figure 3.2 gives a more detailed view of the absolute errors over the validation grid. The largest errors appear for high values of S_0 and for longer maturities. These points are closer to the upper boundary $s_{\max} = 200$, where the call value is approximated by

$$f(t, s_{\max}) \simeq s_{\max} - Ke^{-r(T-t)}.$$

This suggests that the fitted surface is more affected by the upper boundary approximation in this region.

3.1.4 Accuracy of the Greeks

We finally assess the quality of the Greeks produced by the CGPR estimator. As explained in Chapter 2, the sensitivities are obtained by analytical differentiation of the kernel-based estimator. This avoids finite-difference approximations and provides Greeks that are directly consistent with the estimated pricing surface.

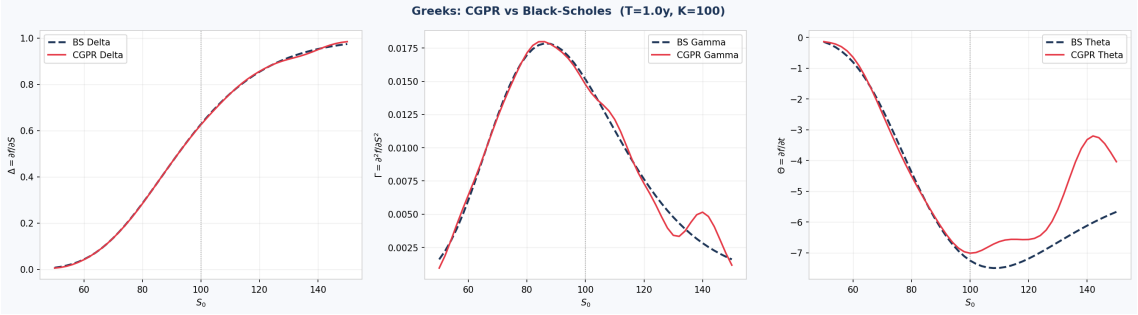

Figure 3.3 – Comparison of CGPR and Black–Scholes Greeks for $T = 1$ and $K = 100$.

Figure 3.3 compares the CGPR Greeks with the Black–Scholes Greeks for $T = 1$. Delta is very close to the benchmark over the full range of spot values. Gamma is also well reproduced around the strike, where it reaches its highest value. For larger values of S_0, small differences become more visible.

Theta is less precise than the two others. However, the CGPR curve follows the Black–Scholes curve up to the at-the-money region, but the difference increases for high spot prices. This shows that the time derivative is more difficult to approximate in this experiment, especially near the upper part of the stock-price range.

3.2 Validation in the Heston Model Framework

After validating the CGPR in the Black–Scholes framework, we now apply the method to the Heston stochastic volatility model. The state vector is now $x = (t, s, v)$, which makes the problem more demanding in two ways. First, the covariance matrix C is built from a three-dimensional kernel with three bandwidths (η_t, η_s, η_v) instead of two. Second, the Heston price surface depends on the variance level V_0, which adds a full dimension to the validation grid. The theoretical framework — the Feynman–Kac operator, the anisotropic kernel and the analytical expressions for the Greeks — was derived in Section 2.4.

3.2.1 Implementation Choices

The financial parameters and numerical setup are summarised in Table 3.3. The Heston parameters are representative of equity markets: a negative correlation $\rho = -30\%$ captures the leverage effect, the long-term variance $\gamma = 0.0225$ corresponds to a long-run volatility of 15%, and the volatility of variance $\sigma = 30\%$ produces a pronounced implied volatility smile.

Table 3.3 – Numerical setup for the Heston CGPR experiment.

Category	Quantity	Value
Financial parameters	Strike	$K = 100$
	Risk-free rate	$r = 5\%$
	Mean reversion speed	$\kappa = 0.95$
	Long-term variance	$\gamma = 0.0225$
	Volatility of variance	$\sigma = 30\%$
	Correlation	$\rho = -30\%$
Training domain	Stock-price interval	$s \in [10, 300]$
	Variance interval	$v \in [0, 8\gamma]$
	Time interval	$t \in [0, T]$
Training design	Interior sample size	$m = 400$
	Boundary sample size	$d = 240 = 3m/5$
	Boundary condition	$f(T, s, v) = (s - K)^+$
	Regularisation parameter	$\sigma^* = 0.01$
Kernel calibration	Kernel	Anisotropic Gaussian
	Bandwidths	η_t, η_s, η_v
	Optimisation	Nelder–Mead on $\log(\boldsymbol{\eta})$
	Criterion	$J(\boldsymbol{\eta})$ in Equation (2.36)
Validation grid	Spot values	20 values in $[50, 150]$
	Variance values	20 values in $[\gamma/5, 5\gamma]$
	Maturities	$T \in \{0.5, 1, 1.5, 2, 2.5, 3\}$
	Total options	2400
Benchmark	FFT [8]	$N = 4096$, $\alpha = 1.5$, spacing $= 0.25$

FFT benchmark. Since the Heston model does not provide an elementary closed-form call price, the benchmark is computed via the Carr–Madan Fast Fourier Transform algorithm [8].. The key ingredient is the Heston characteristic function

$$\phi_T(u) = \mathbb{E}^{\mathbb{Q}}\big[e^{iu \log S_T} \mid S_0, V_0\big] ,$$

which is available in closed form [6]. The idea of the Carr–Madan approach is to introduce a damping factor $e^{\alpha k}$ (with $k = \log K$ and $\alpha > 0$) to ensure integrability, and then to recover the call price via an inverse Fourier transform of a transformed quantity $\psi(u)$ involving ϕ_T. This integral is then approximated numerically using a standard FFT routine. The full mathematical derivation and the implementation parameters ($N = 4096$, $\alpha = 1.5$, frequency spacing $= 0.25$) are given in Appendix A.6.

Since the FFT introduces a small discretization error, the reported MAE reflects the combined discrepancy between the CGPR and the FFT rather than the exact pricing error of the CGPR alone.

Boundary conditions and computational structure. Unlike in the Black–Scholes experiment, the boundary sample is placed only at the terminal date T, with no spatial boundary condition at $s = 0$ or $s = s_{\max}$. The baseline choice $m = 400$, $d = 240 = 3m/5$, gives more weight to the interior PDE constraints while keeping the covariance matrix manageable. This is a compromise: larger training sets improve accuracy but increase the optimisation time, as shown in Table 3.4. Once the system is solved, the validation grid is evaluated in less than one second. The FFT remains faster for this vanilla benchmark, but CGPR provides a reusable pricing surface with analytical Greeks and posterior uncertainty. Further sensitivity results are reported in Appendix A.1.

Table 3.4 – Accuracy and computation times for the Heston CGPR experiment ($T = 1$, $d = 3m/5$, $\sigma^* = 0.01$).

m	d	MAE	MRE ITM	Mean Std.	Opt. time	CGPR time	FFT time
400	240	0.1805	1.47%	0.0064	48.12s	0.60s	0.16s
600	360	0.1118	0.64%	0.0050	192.83s	0.61s	0.14s
800	480	0.0952	0.32%	0.0042	385.10s	0.83s	0.15s

3.2.2 Sensitivity to the Regularisation Parameter

Before analysing the pricing results, we study the effect of the regularisation parameter σ^* on the CGPR approximation. As discussed in Section 2.2, this parameter plays two roles: it is the noise level attached to the pseudo-observations in the Bayesian formulation, and it acts as a Tikhonov regularisation term that improves the conditioning of the covariance matrix C.

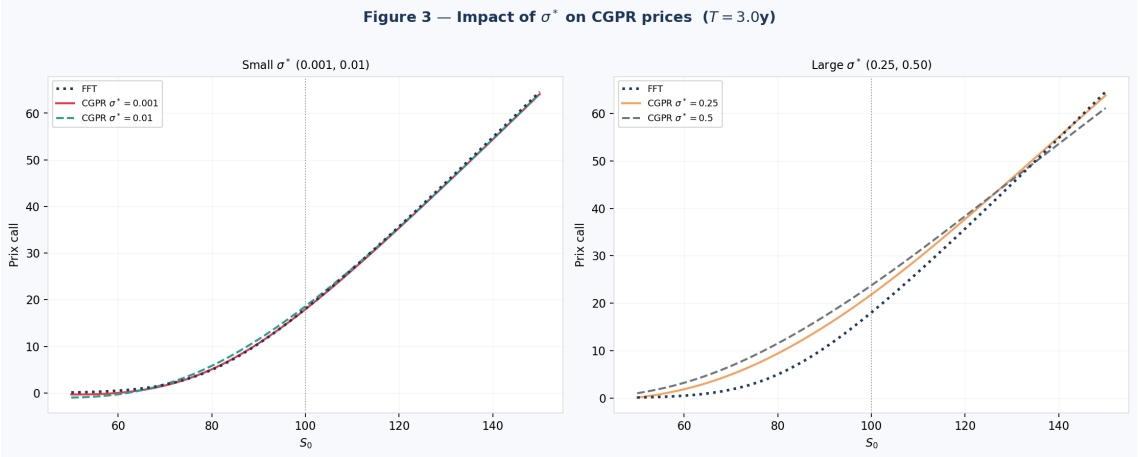

Figure 3.4 – Impact of σ^* on Heston CGPR prices for $T = 3$ ($m = 400$, $d = 240$). Left panel: small values ($\sigma^* \in \{0.001, 0.01\}$). Right panel: large values ($\sigma^* \in \{0.25, 0.50\}$).

In Figure 3.4, the bandwidths are kept fixed in order to isolate the visual effect of changing σ^*. In Table 3.5, they are re-optimised for each value of σ^*.

Table 3.5 – Sensitivity to σ^* in the Heston CGPR experiment ($T = 3$, $m = 400$, $d = 240$).

σ^*	MAE	MRE ITM	Mean Std.	η_t	η_s	η_v
0.001	0.3213	0.67%	0.0011	1.126	31.10	3.171
0.01	0.3534	0.75%	0.0068	1.538	25.78	1.674
0.1	0.7903	2.70%	0.0487	1.909	21.44	2.120
0.25	2.9361	10.08%	0.0871	2.435	28.55	2.846
0.5	1.9580	5.45%	0.1566	6.163	22.77	1.704

The value $\sigma^* = 0.001$ gives the lowest MRE in Table 3.5. However, it also gives a very small posterior standard deviation, which suggests an overconfident fit. More importantly, such a small value adds very little diagonal regularisation to the covariance matrix C, making the inversion more sensitive. For this reason, $\sigma^* = 0.01$ is kept as the baseline value.

The comparison between $\sigma^* = 0.25$ and $\sigma^* = 0.5$ should also be read with care. Although $\sigma^* = 0.5$ gives a lower MRE than $\sigma^* = 0.25$, both values produce much larger errors than the baseline case. This happens because the bandwidths $\boldsymbol{\eta}$ are re-optimised for each value of σ^*. Changing σ^* therefore changes both the strength of the PDE constraint and the bandwidths selected by the optimiser.

For the rest of the Heston experiment, $\sigma^* = 0.01$ is kept as the baseline value. It does not give the lowest MRE in this table, but it gives a good compromise between pricing accuracy, posterior uncertainty and numerical stability.

3.2.3 Pricing Results

Table 3.6 reports the pricing errors and optimized bandwidths for the six maturities.

Table 3.6 – Heston CGPR pricing errors and optimized bandwidths by maturity ($m = 400$, $d = 240$, $\sigma^* = 0.01$).

T	MAE	MRE ITM	Mean Std.	η_t	η_s	η_v
0.5y	0.1785	1.86%	0.0055	1.070	20.14	0.205
1.0y	0.1805	1.49%	0.0064	0.555	24.09	0.361
1.5y	0.2121	1.57%	0.0076	1.083	26.20	0.256
2.0y	0.3230	0.94%	0.0059	1.090	24.37	1.167
2.5y	0.3459	0.79%	0.0064	1.321	25.10	1.389
3.0y	0.3535	0.73%	0.0067	1.537	25.79	1.675

The MRE on in-the-money options stays below 2% across all maturities. Compared to the Black–Scholes experiment where the MRE reached 0.42%, the Heston errors are larger — which is expected since the pricing surface now depends on an additional state variable V_0 that the kernel must approximate simultaneously. To put the numbers in perspective, a MRE of 1.86% at $T = 0.5$ means that on an at-the-money option worth approximately 5€, the CGPR price deviates by less than 10 cents from the FFT benchmark.

The MRE decreases with maturity, from 1.86% at $T = 0.5$ to 0.73% at $T = 3$. Over the same range, the MAE increases from about 0.18 to 0.35. This difference mainly comes from the scale of the option prices: longer-maturity options are more expensive, so a similar relative error can lead to a larger absolute error. For this reason, both the MAE and the MRE are useful to read Table 3.6.

The variance bandwidth η_v increases with maturity, from 0.205 at $T = 0.5$ to 1.675 at $T = 3$ and it can be explained by the variance process. For longer maturities, the variance has more time to move back toward its long-term level γ. As a result, the option price becomes less sensitive to the initial variance V_0, and a larger bandwidth in the variance direction is selected. On the other hand, the stock-price bandwidth η_s remains more stable, between 20.14 and 26.20.

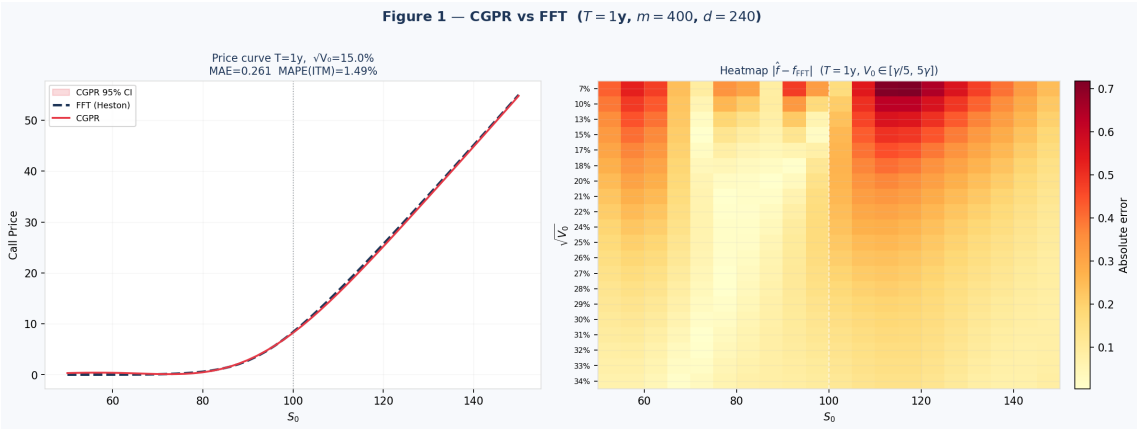

Figure 3.5 – Left: CGPR and FFT call prices for $T = 1$, $\sqrt{V_0} = 15\%$ ($m = 400$, $d = 240$). The shaded area is the 95% posterior confidence interval. Right: absolute error over the 20×20 validation grid in (S_0, V_0).

Figure 3.5 shows the result for $T = 1$. On the left panel, the CGPR price curve is close to the FFT benchmark for $\sqrt{V_0} = 15\%$, and the posterior band remains narrow. This band mainly reflects the internal uncertainty of the CGPR estimator, not a direct error bound against the FFT price. The heatmap on the right shows that the largest errors occur for low initial volatility levels, especially for low stock

prices and near the strike. These regions are more difficult because the pricing surface changes more rapidly there.

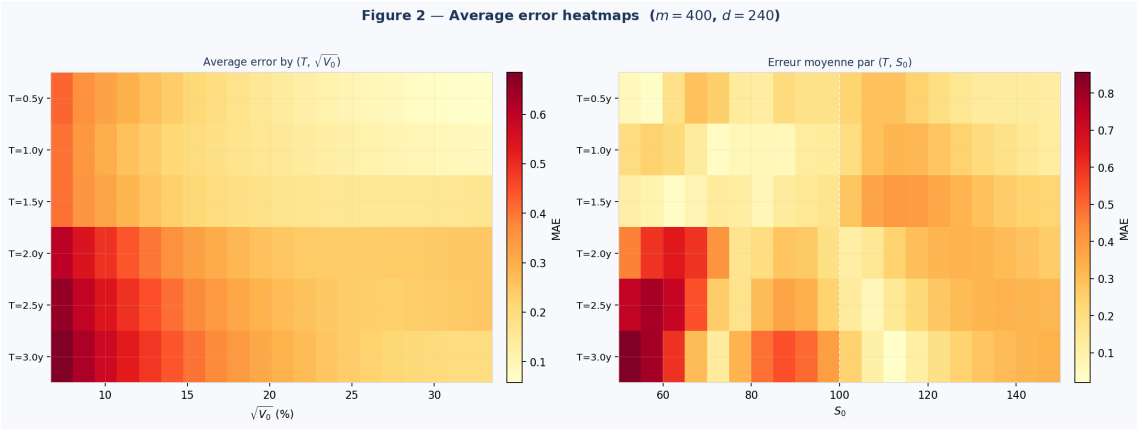

Figure 3.6 – Average absolute errors by maturity and initial variance (left) and by maturity and initial stock price (right) ($m = 400$, $d = 240$).

Figure 3.6 shows that this pattern is also present across maturities. Errors are higher when the initial volatility is low, especially for longer maturities. The errors are also more visible for low stock prices and around the strike region. This suggests that the approximation is more difficult in regions where the price surface changes more rapidly.

3.2.4 Accuracy of the Greeks

The CGPR estimator produces analytical sensitivities by differentiating $\boldsymbol{\beta}(x)$, as derived in Section 2.4.

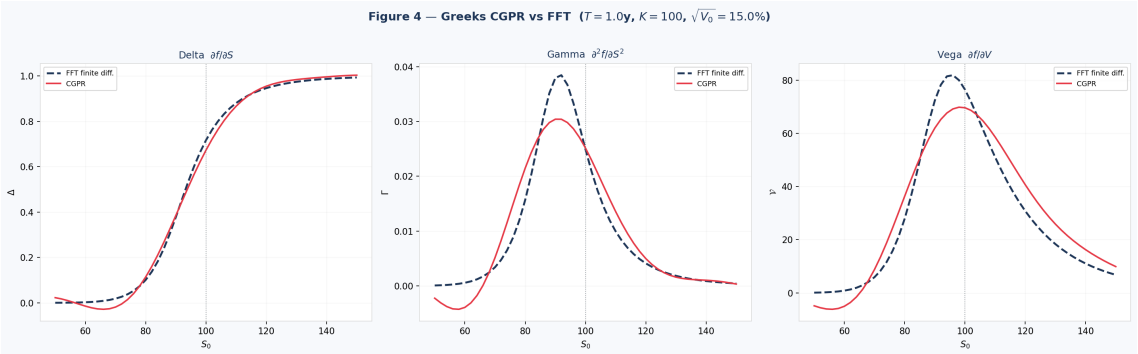

Figure 3.7 – CGPR and FFT finite-difference Greeks for $T = 1$, $K = 100$ and $\sqrt{V_0} = 15\%$ ($m = 400$, $d = 240$).

Figure 3.7 illustrates the behaviour of the Greeks for $T = 1$, while Table 3.7

reports the MAE across all maturities. Delta is the most accurate sensitivity, with an error below 0.035 in all cases. Gamma also keeps the expected shape: it increases near the strike, reaches a peak, and then decreases for larger stock prices. However, the CGPR curve is not perfectly aligned with the FFT benchmark.

Vega shows a similar situation. Its overall shape is well reproduced, with a clear increase around the strike followed by a decrease for larger stock prices. The main difference with the FFT benchmark is again in the level of the curve, which explains the larger MAE values reported in Table 3.7. .

Table 3.7 – Greek errors: CGPR vs FFT finite differences ($m = 400$, $d = 240$, $\sigma^* = 0.01$).

T	MAE Δ	MAE Γ	MAE Vega
0.5y	0.02103	0.003736	6.8360
1.0y	0.01770	0.002577	6.5314
1.5y	0.01595	0.002208	6.1175
2.0y	0.02876	0.003087	8.1773
2.5y	0.03067	0.003101	8.1406
3.0y	0.03405	0.002964	8.0508

These limitations are partly attributable to the benchmark itself: the FFT Greeks are computed by finite differences, which introduces additional numerical noise. The reported Greek errors should therefore be interpreted with care, since they include both the CGPR approximation error and the finite-difference error of the FFT benchmark.

CHAPTER 4

EXTENSION: EXTENSIONS TO MULTI-ASSET AND INSURANCE PRODUCTS

4.1 Extension to a Two-Asset Heston Model

The previous sections validated the CGPR methodology in the Black–Scholes and one-asset Heston frameworks. We now extend the method to a basket option on two assets, each driven by its own stochastic variance process. The state vector becomes $x = (t, S_1, S_2, V_1, V_2) \in \mathbb{R}^5$, which increases the dimension of the pricing problem from three to five. This extension tests whether the CGPR construction remains accurate when several sources of dependence are introduced simultaneously.

4.1.1 Model, Payoff and Feynman–Kac Operator

Under the risk-neutral measure, the two assets follow Heston-type dynamics:

$$\begin{aligned} dS_{i,t} &= rS_{i,t}\,dt + \sqrt{V_{i,t}}\,S_{i,t}\,dW_t^{S_i}, \quad i = 1, 2, \\ dV_{i,t} &= \kappa_i(\gamma_i - V_{i,t})\,dt + \sigma_i\sqrt{V_{i,t}}\,dW_t^{V_i}, \quad i = 1, 2. \end{aligned} \tag{4.1}$$

The four Brownian motions are correlated through the matrix

$$R = \begin{pmatrix} 1 & \rho_1 & \rho_{12} & 0 \\ \rho_1 & 1 & 0 & \rho_V \\ \rho_{12} & 0 & 1 & \rho_2 \\ 0 & \rho_V & \rho_2 & 1 \end{pmatrix}, \tag{4.2}$$

with ordering $(W^{S_1}, W^{V_1}, W^{S_2}, W^{V_2})$. The parameter ρ_i is the leverage effect for asset i, ρ_{12} is the cross-asset correlation, and ρ_V captures a direct dependence between the two variance shocks — for instance when both assets are exposed to common market volatility. The positive definiteness of R is verified before running the Monte Carlo simulation.

The contract is a European basket call option with payoff

$$h(S_1, S_2) = (\alpha S_1 + (1-\alpha) S_2 - K)^+ , \tag{4.3}$$

with $\alpha = 0.5$ in the numerical experiment, corresponding to an equally weighted basket.

Applying Theorem 1.3.1 to the two-asset dynamics, the pricing function $f(t, S_1, S_2, V_1, V_2)$ satisfies $Lf = 0$ in the interior of the pricing domain, where

$$\begin{aligned} Lf = {} & \frac{\partial f}{\partial t} + rS_1 \frac{\partial f}{\partial S_1} + rS_2 \frac{\partial f}{\partial S_2} + \kappa_1(\gamma_1 - V_1)\frac{\partial f}{\partial V_1} + \kappa_2(\gamma_2 - V_2)\frac{\partial f}{\partial V_2} - rf \\ & + \frac{1}{2} S_1^2 V_1 \frac{\partial^2 f}{\partial S_1^2} + \frac{1}{2} S_2^2 V_2 \frac{\partial^2 f}{\partial S_2^2} + \frac{1}{2}\sigma_1^2 V_1 \frac{\partial^2 f}{\partial V_1^2} + \frac{1}{2}\sigma_2^2 V_2 \frac{\partial^2 f}{\partial V_2^2} \\ & + \rho_{12} S_1 S_2 \sqrt{V_1 V_2} \frac{\partial^2 f}{\partial S_1 \partial S_2} + \rho_1 \sigma_1 S_1 V_1 \frac{\partial^2 f}{\partial S_1 \partial V_1} + \rho_2 \sigma_2 S_2 V_2 \frac{\partial^2 f}{\partial S_2 \partial V_2} \\ & + \rho_V \sigma_1 \sigma_2 \sqrt{V_1 V_2} \frac{\partial^2 f}{\partial V_1 \partial V_2}. \end{aligned} \tag{4.4}$$

This operator is the direct application of Equation (1.5) to the two-asset Heston dynamics. The last term, proportional to ρ_V, arises from the direct correlation between the two variance processes and introduces a mixed derivative $\partial^2 f / \partial V_1 \partial V_2$ that does not appear in the one-asset case. The instantaneous covariance matrix of the state variables (S_1, S_2, V_1, V_2) that generates the second-order terms is given in Appendix A.7.

4.1.2 CGPR Construction and Kernel

The CGPR estimator follows the same construction as in Chapter 2:

$$\hat{f}(x) = \boldsymbol{\beta}(x)^\top C^{-1} \mathbf{o}.$$

The dimension of the input vector increases from three to five, and the operator L in Equation (4.4) replaces the one-asset Heston operator (**??**). The anisotropic

Gaussian kernel extends naturally to the five-dimensional state vector:

$$k_\eta(x, x') = \exp\left(-\frac{1}{2} \sum_{\ell=1}^{5} \frac{(x_\ell - x'_\ell)^2}{\eta_\ell^2} \right), \tag{4.5}$$

with bandwidths $\boldsymbol{\eta} = (\eta_t, \eta_{S_1}, \eta_{S_2}, \eta_{V_1}, \eta_{V_2})$.

The boundary sample is more complex than in the one-asset case. Five components are used:

— terminal payoff at $t = T$: $h = (\alpha S_1 + (1-\alpha)S_2 - K)^+$,
— lower boundary $S_1 = 0$: approximated by a one-asset Black–Scholes call on $(1-\alpha)S_2$,
— lower boundary $S_2 = 0$: approximated by a one-asset Black–Scholes call on αS_1,
— upper boundary $S_1 = S_{\max}$: discounted forward approximation of the basket,
— upper boundary $S_2 = S_{\max}$: symmetric approximation.

Half of the boundary points are placed at maturity, and the other half are split across the four spatial boundaries. These boundary values are not external prices. They are used to guide the CGPR surface near the edges of the numerical domain, where the interior PDE points alone may not be sufficient.

The kernel derivatives required to build C follow the same structure as in Section 2.4, with the operator L replaced by Equation (4.4). The explicit expressions are given in Appendix A.8.

4.1.3 Implementation Choices

The numerical setup is summarised in Table 4.1. The two assets are assigned different Heston parameters to avoid a degenerate symmetric configuration. Asset 1 is calibrated close to the one-asset experiment of Section 3.2, with a long-term variance of $\gamma_1 = 0.15^2$ and volatility of variance $\sigma_1 = 0.30$. Asset 2 has a higher long-term variance $\gamma_2 = 0.20^2$ and a lower volatility of variance $\sigma_2 = 0.25$. The cross-asset correlation $\rho_{12} = 0.30$ and variance correlation $\rho_V = 0.30$ introduce two distinct dependence channels, as described in Section 4.1.1.

Table 4.1 – Numerical setup for the two-asset Heston experiment.

Category	Parameter	Value
Basket option	Strike	$K = 100$
	Weight	$\alpha = 0.5$
	Risk-free rate	$r = 5\%$
Asset 1	Mean reversion	$\kappa_1 = 0.95$
	Long-term variance	$\gamma_1 = 0.15^2$
	Vol. of variance	$\sigma_1 = 0.30$
	Leverage	$\rho_1 = -0.30$
Asset 2	Mean reversion	$\kappa_2 = 0.80$
	Long-term variance	$\gamma_2 = 0.20^2$
	Vol. of variance	$\sigma_2 = 0.25$
	Leverage	$\rho_2 = -0.20$
Dependence	Asset correlation	$\rho_{12} = 0.30$
	Variance correlation	$\rho_V = 0.30$
Training	Interior points	$m = 600$
	Boundary points	$d = 600$
	Regularisation	$\sigma^* = 0.01$
	Interior sampling	Sobol sequence
Validation	Slice	$S_1 = S_2 = S_0$, $V_{2,0} = \gamma_2$
	S_0 grid	20 points in $[60, 150]$
	$V_{1,0}$ grid	10 points in $[\gamma_1/5,\ 5\gamma_1]$
	Maturities	$T \in \{0.5, 1, 1.5, 2, 2.5, 3\}$
Benchmark	Method	Monte Carlo
	Paths	$n_{\text{sim}} = 8{,}000$
	Variance reduction	Antithetic variates
	Steps per year	100

Three implementation choices differ from the one-asset experiment and deserve comment.

Sobol sampling. Interior points are generated from randomized Sobol sequence[20] rather than a uniform distribution. In five dimensions, a Sobol sequence provides better coverage of the state space than a standard random sample of moderate m, which reduces the risk of leaving large regions of the domain without PDE constraints.

Choice of $m = d = 600$. Unlike the one-asset Heston case where $d = 3m/5$, we use $d = m$ here because the boundary sample covers five components including four spatial boundaries. Table 4.2 reports the sensitivity to the training size for $T = 1.0$.

Table 4.2 – Accuracy and computation time for different training sizes ($T = 1.0$, $d = m$, $\sigma^* = 0.01$).

$m = d$	MAE	MRE ITM	Mean Std	Opt. time(s)	CGPR time(s)	MC time(s)
400	0.4450	1.06%	0.00564	157.9	0.00	0.25
600	0.3113	0.77%	0.00600	239.2	0.00	0.24
800	0.3808	1.12%	0.00423	530.3	0.00	0.24
1000	0.1906	0.50%	0.00524	483.0	0.01	0.25

The accuracy is not monotone in m because going from $m = 400$ to $m = 600$ improves both the MAE and MRE, but $m = 800$ gives worse results than $m = 600$. This reflects the non-convex optimization landscape of $J(\boldsymbol{\eta})$ where a larger training set can lead the optimizer to a different local minimum. The best absolute result is at $m = 1000$ with a MRE of 0.50%, but the optimization time of 483 seconds per maturity makes it prohibitive for the full six-maturity experiment. We retain $m = d = 600$ as the best compromise between accuracy and computation time.

It is easy to show that Monte Carlo is relatively cheap for one benchmark run, but it remains simulation-based and must be repeated when new prices, Greeks or higher precision are required. By contrast, CGPR has a costly offline calibration step, but once the system has been solved, the fitted pricing surface is evaluated almost instantly. This is the main computational advantage of CGPR in this higher-dimensional setting.

Monte Carlo benchmark. Since no simple Fourier or closed-form benchmark is available for this basket option, the benchmark is computed with Monte Carlo simulation. The correlation between the Brownian motions is introduced through the Cholesky decomposition of the matrix R. The variance processes are simulated with an Euler scheme, while keeping the variance non-negative.

Since the benchmark is itself numerical, the MAE does not measure the exact error of the CGPR method. It measures the difference between the CGPR price and the Monte Carlo price. For this reason, the Monte Carlo standard error is also reported.

Bandwidth calibration. The five bandwidths $\boldsymbol{\eta}$ are calibrated by minimising the same criterion $J(\boldsymbol{\eta})$ defined in Equation (2.36). The optimisation is performed in log-space with bounds on each bandwidth to prevent the optimizer from removing a dimension from the kernel. The bounds are

$$\eta_t \in [0.05, 2T], \quad \eta_{S_i} \in [5, S_{\max}], \quad \eta_{V_i} \in [0.02, 5],$$

and the minimisation uses the Powell [15] method with multiple starting points.

4.1.4 Sensitivity to the Regularisation Parameter

Table 4.3 reports the effect of σ^* for $T = 2.0$ with $m = d = 600$.

Table 4.3 – Sensitivity to σ^* ($T = 2.0$, $m = d = 600$).

σ^*	MAE	MRE ITM	Mean Std	cond(C)	η_{V_1}	η_{V_2}
0.001	0.7975	2.05%	0.00129	4.9×10^8	3.7742	0.6141
0.005	0.4541	0.74%	0.00287	1.0×10^7	4.9997	2.4720
0.010	0.4933	0.90%	0.00572	2.7×10^6	2.1847	2.6609
0.050	0.3827	0.76%	0.03574	8.7×10^4	0.2892	1.2099
0.100	0.5987	1.56%	0.04919	1.9×10^4	0.6902	1.2783
0.250	0.6490	1.48%	0.09705	3.7×10^3	0.7457	1.1965

The pattern is consistent with the one-asset Heston experiment. At $\sigma^* = 0.001$, the condition number reaches 4.9×10^8, which makes the bandwidth optimisation numerically unstable. It then explains why the MAE of 0.797 is the worst in the table despite the tightest constraint enforcement. At $\sigma^* = 0.25$, the larger posterior standard deviation (0.097) suggests that the PDE and boundary constraints are imposed less tightly.

The link between σ^* and accuracy is not monotone. The lowest MRE, 0.74%, is obtained for $\sigma^* = 0.005$, but this value also gives a high condition number, equal to 10^7. This suggests that the system may be less stable numerically. The lowest MAE, 0.383, is obtained for $\sigma^* = 0.05$, but the Mean Std is much larger than for $\sigma^* = 0.01$. This means that the PDE constraint is imposed less tightly. Overall, $\sigma^* = 0.01$ remains a safer choice for the baseline experiment.

The table also confirms the coupling between σ^* and the variance bandwidths: η_{V_1} ranges from 0.289 to 4.999 across the values tested. This reinforces the conclusion from Section 3.2.2 that σ^* and $\boldsymbol{\eta}$ must be understood as jointly determined hyperparameters.

We retain $\sigma^* = 0.01$ as the baseline, for the same reasons as in the one-asset case: it provides a good balance between accuracy, posterior uncertainty and numerical stability, with a condition number of 2.7×10^6 that keeps the optimisation well-behaved.

4.1.5 Pricing Results

Table 4.4 reports the pricing errors across the six maturities. The benchmark is Monte Carlo with $n_{\text{sim}} = 8{,}000$ paths; the MC standard error is reported alongside the MAE to quantify the benchmark uncertainty.

Table 4.4 – Two-asset Heston CGPR pricing errors and optimized bandwidths ($m = d = 600$, $\sigma^* = 0.01$).

T	MAE	MRE ITM	MC SE	Mean Std	η_t	η_{S_1},η_{S_2}	η_{V_1},η_{V_2}
0.5	0.5431	2.42%	0.0285	0.0040	0.253	77.1, 65.8	1.423, 3.051
1.0	0.3556	1.06%	0.0494	0.0056	0.992	50.9, 79.3	0.571, 1.907
1.5	0.3581	0.67%	0.0659	0.0064	1.269	50.7, 88.5	0.611, 1.776
2.0	0.4915	0.85%	0.0823	0.0057	1.050	40.8, 126.6	2.185, 2.661
2.5	0.4378	0.92%	0.0961	0.0067	1.172	47.5, 128.4	1.923, 1.705
3.0	0.4742	0.79%	0.1095	0.0089	4.831	47.7, 54.2	0.477, 4.721

Overall, the results remain stable across maturities. The MAE ranges from 0.3556 to 0.5431, while the MRE on in-the-money options ranges from 0.67% to 2.42%. The average MRE is about 1.12%, which remains reasonable given that the problem is now five-dimensional.

The largest relative error is observed for the shortest maturity, with 2.42% at $T = 0.5$. After that, the MRE decreases clearly and remains below 1.1% for all maturities with $T \geq 1.0$. This indicates that the short-maturity case is the most difficult for pricing. In this region, the basket payoff still has a strong kink near the strike, while for longer maturities the pricing surface becomes smoother.

The MC standard error gives an indication of the precision of the Monte Carlo benchmark. It increases with maturity, from 0.0285 at $T = 0.5$ to 0.1095 at $T = 3.0$ meaning that part of the reported MAE, especially for longer maturities, may come from the benchmark itself and not only from the CGPR approximation. The mean posterior standard deviation of the CGPR remains small, between 0.0040 and 0.0089, which suggests a stable fitted price surface.

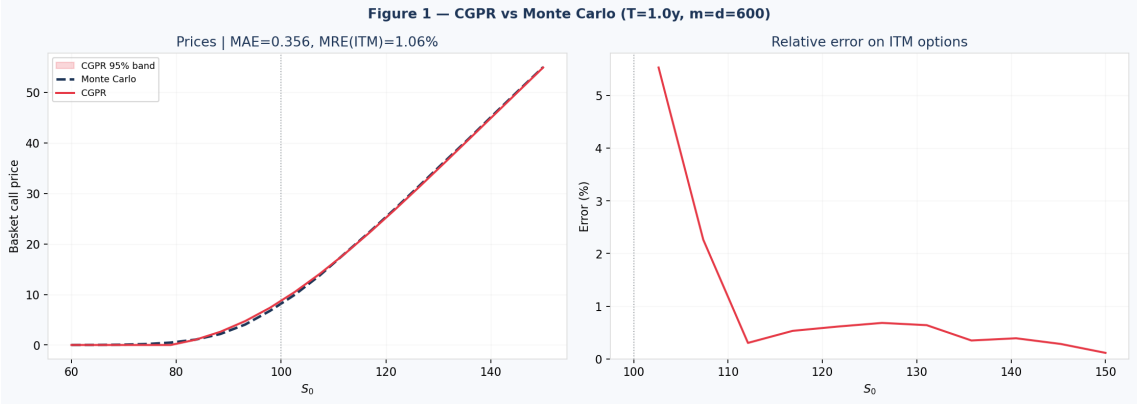

Figure 4.1 – CGPR and Monte Carlo prices for the two-asset Heston basket call ($T = 1.0$, $m = d = 600$, $\sigma^* = 0.01$). Left: price curve on the slice $S_1 = S_2 = S_0$, $V_{2,0} = \gamma_2$. Right: relative error on in-the-money options.

Figure 4.1 shows the result for $T = 1.0$ on the slice $S_1 = S_2 = S_0$ and $V_{2,0} = \gamma_2$. The left panel shows that the CGPR price curve is close to the Monte Carlo benchmark in absolute terms. The largest visible difference is around the transition region, close to the strike.

The right panel gives the relative error on in-the-money options. The error is high just above the strike, reaching about 7% around $S_0 \approx 102$. This does not necessarily mean that the absolute error is large: in this region, the option price is still small, so a small difference in price can give a large relative error. For larger values of S_0, the relative error decreases quickly and stays around or below 1%.

4.1.6 Structure of the Approximation Error

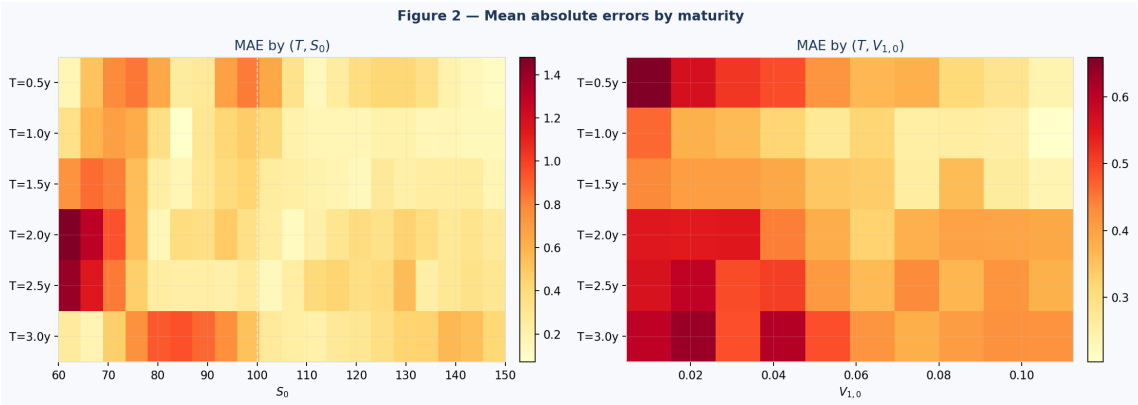

Figure 4.2 – Mean absolute errors by maturity and S_0 (left) and by maturity and $V_{1,0}$ (right) ($m = d = 600$, $\sigma^* = 0.01$).

Figure 4.2 breaks down the pricing errors by maturity and state variable. Two patterns emerge.

The left panel shows that the largest errors are concentrated for $S_0 \in [60, 80]$ and long maturities. This is the out-of-the-money region where the basket price is small but not zero. The difficulty comes from the transition between near-zero values at low $S_0 = 60$ and the increasing payoff around the strike.

The right panel shows that errors are largest at low variance levels, almost indepedetly of maturity. When $V_{1,0} \approx \gamma_1/5 = 0.0045$, the option price becomes sensitive to small variance shocks. A fixed variance bandwidth calibrated on the full interval $[\gamma_1/5,\, 5\gamma_1]$ may be too wide near the lower end of the variance range.

These two patterns together suggest that the main limitations of the current configuration are the boundary anchoring at low spot prices and the variance resolution at low variance levels, rather than the overall dimensionality of the problem. Both could be addressed by adaptive sampling; placing more training points in the regions where the surface changes most rapidly.

4.1.7 Accuracy of the Greeks

The CGPR estimator produces analytical sensitivities by differentiating $\beta(x)$ with respect to each state variable, using the kernel derivatives derived in Appendix A.8. Six Greeks are computed: $\Delta_i = \partial \hat{f}/\partial S_i$, $\Gamma_i = \partial^2 \hat{f}/\partial S_i^2$, and $\mathcal{V}_i = (\partial \hat{f}/\partial V_i) \cdot 2\sqrt{V_{i,0}}$ for $i = 1, 2$. The Monte Carlo benchmarks use central finite differences with common random numbers (CRN). Figure 4.3 shows the results for $T = 1.0$y.

Delta. Both Δ_1 and Δ_2 are reproduced quite well on the full range. For large values of S_0, both curves converge to the basket weights, which are both equal to 0.5 in this experiment. The fit is slightly less precise below the strike, where the CGPR curve is a bit higher than the Monte Carlo finite-difference estimate. A similar effect was already visible in the one-asset Heston case.

Gamma. The CGPR captures the general bell shape of Gamma, but the CGPR curve underestimates the peak around the strike. The maximum s close to 0.006 for the CGPR curve, while the Monte Carlo benchmark reaches about 0.009. The difference becomes smaller for larger values of S_0. This is consistent with the smoothing effect of the Gaussian kernel near the at-the-money region.

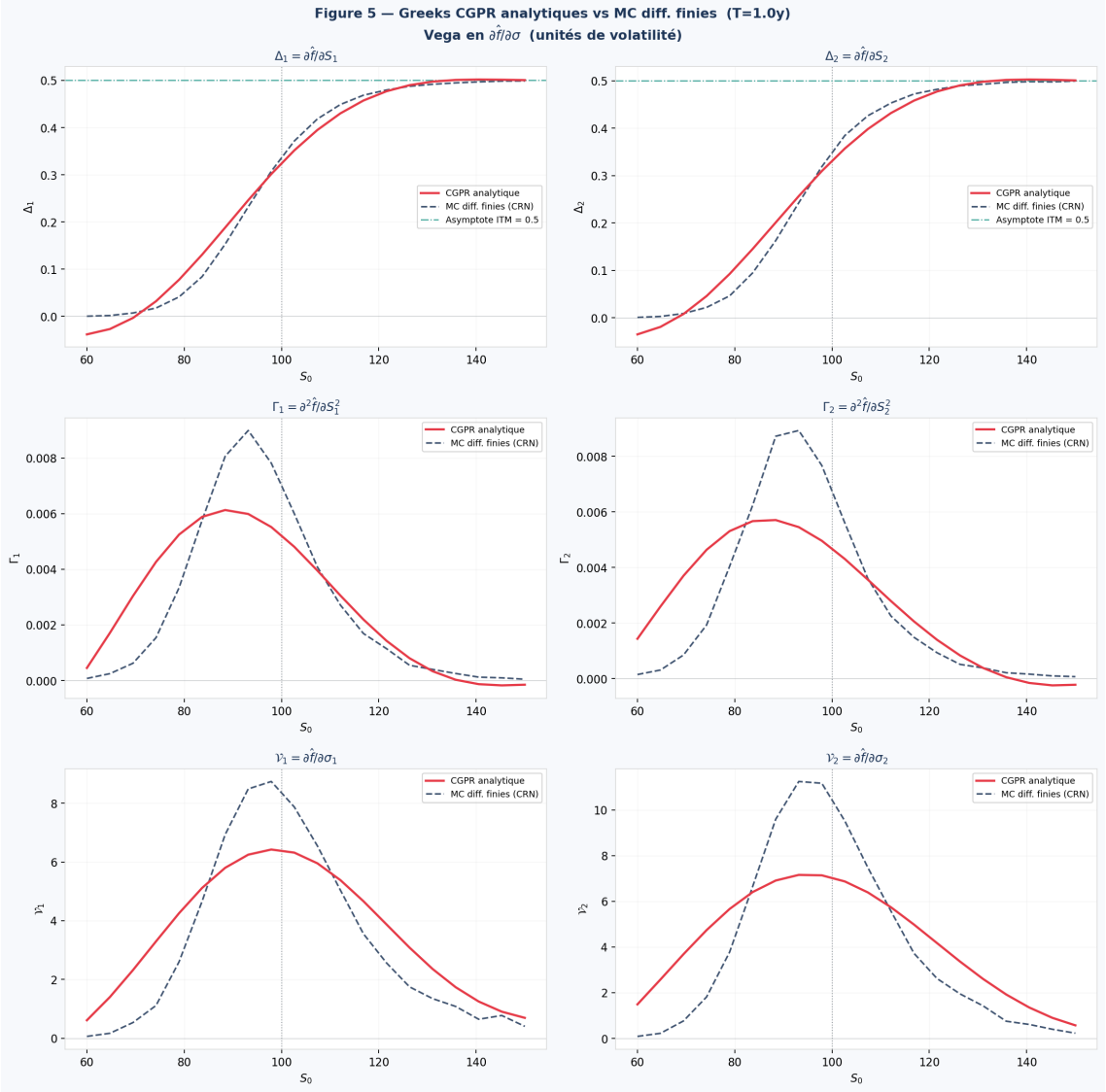

Figure 4.3 – Two-asset Heston basket call Greeks: CGPR analytical versus Monte Carlo finite differences (CRN), $T = 1.0$y, $m = d = 600$, $\sigma^* = 0.01$, slice $S_1 = S_2 = S_0$, $V_{1,0} = \gamma_1$, $V_{2,0} = \gamma_2$. Vegas in volatility units. Vertical dotted line: strike $K = 100$.

Vega. The Vega curves have the right general shape, but the level difference is more visible than for Delta. The CGPR reaches a peak of about 6.3 for $\mathcal{V}_1$ and 7.5 for $\mathcal{V}_2$, while the Monte Carlo benchmarks reaches about 8.5 and 11.5, respectively. The method still captures the relative ranking between the two assets, since the second asset has the larger Vega, which is coherent with its higher long-term variance.

Overall, Delta is the most accurate sensitivity in this experiment. Gamma and Vega reproduce the main shape of the benchmark curves, but their peaks are smoother near the at-the-money region.

4.2 Application to GMAB Insurance Contracts

The previous sections applied the CGPR method to purely financial payoffs. We now show that the same framework extends to life insurance contracts, where a mortality component modifies the pricing operator. The product considered is a Guaranteed Minimum Accumulation Benefit (GMAB), a unit-linked contract widely distributed in the Belgian and European markets (Hainaut (2024)[11]).

4.2.1 Product and Analytical Price

A GMAB (Hainaut (2024) [11]) is a single-premium contract in which the policyholder invests a lump sum (the single-premium) in a fund F_t. At maturity T, the surviving policyholder receives $\max(F_T, G)$, where G is a guaranteed floor. If the policyholder dies before T, the beneficiaries receive the current fund value F_τ. The contract therefore combines a financial option (the guaranteed minimum) with a life insurance component driven by the force of mortality $\mu(t)$.

Under Heston dynamics [6] with a management fee rate δ deducted continuously from the fund, the fair value of the contract decomposes as

$$f_{\text{ins}}(F_0, V_0; T) = {}_Tp_x \cdot f_{\text{pur}}(F_0, G; T) + F_0 \int_0^T e^{-\delta s}\, \mu(s)\, {}_sp_x\, ds, \tag{4.6}$$

where ${}_tp_x = \exp(-\int_0^t \mu(s)\, ds)$ is the survival probability and

$$f_{\text{pur}} = Ge^{-rT} + C_{\text{Heston}}(F_0, G;\, r - \delta,\, T)$$

is the pure financial GMAB price. The decomposition in Equation (4.6) is itself analytical: it follows directly from the independence between the financial mar-

ket and the deterministic force of mortality. However, the Heston call component C_{Heston} does not admit a closed-form expression and is evaluated numerically via the Carr–Madan FFT algorithm [8] with drift $r - \delta$, exactly as in Section 3.2.1. The benchmark used throughout this section is therefore semi-analytical: the mortality decomposition is exact, while the Heston call is an FFT-based numerical evaluation.

Mortality model. Rather than using a standard parametric model such as Gompertz or Makeham [12], we use the official Belgian life table published by Statbel [13]. It provides annual death probabilities q_x for integer ages $x \in \{0, \ldots, 110\}$, separately for men, women, and the total population. The continuous force of mortality is recovered as $\mu_x = -\ln(1 - q_x)$ and linearly interpolated between integer ages. For a policyholder of age x_0 at inception, the hazard rate at time t after issue is $\mu(t) = \mu_{x_0+t}$.

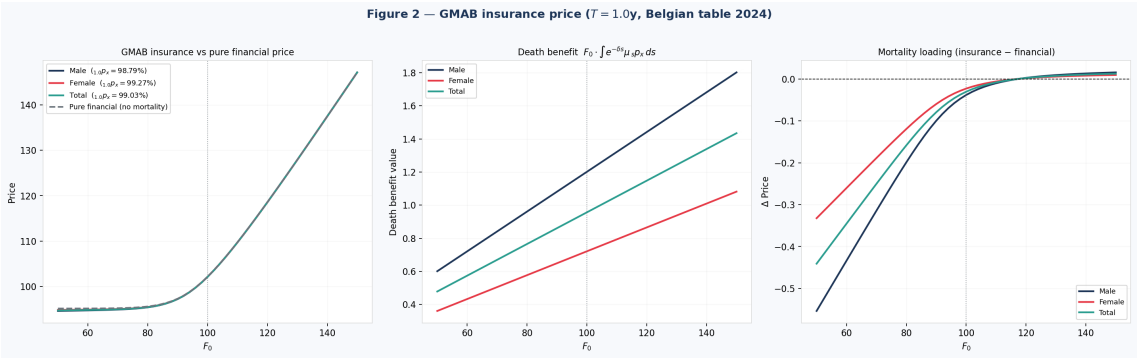

Figure 4.4 – Decomposition of the GMAB insurance price into survival and death benefit components ($T = 1.0$y, $G = 100$, $\delta = 2\%$, $x_0 = 65$, Statbel 2024 table). Left: total insurance price for male, female and total population versus pure financial price (no mortality). Center: death benefit component $F_0 \cdot \int_0^T e^{-\delta s} \mu(s)\, {}_s p_x\, ds$. Right: mortality loading (insurance minus financial price).

Figure 4.4 illustrates the price decomposition for the three demographic groups of the Statbel table. As expected, the insurance price always lies below the pure financial price: mortality reduces the probability of reaching maturity, so the insurer pays less than the financial value of the guarantee alone. Perhaps more surprisingly, despite their different hazard rates, all three populations yield prices that are visually indistinguishable on the left panel. At age 65 over a one-year horizon, mortality remains low enough (${}_1p_{65} = 98.79\%$ for males and 99.27% for females) that the gender distinction has virtually no impact on the contract value. This point is confirmed numerically in Section 4.2.3.

These observations motivate the modified Feynman–Kac operator incorporating

both the mortality discount and the instantaneous death benefit into the pricing PDE.

4.2.2 Modified Feynman–Kac Operator

The pricing function $f(t, F, V)$ satisfies the following PDE:

$$\mathcal{A}^{\mu} f(t, F, V) = -\mu(t)F, \tag{4.7}$$

where the linear operator $\mathcal{A}^{\mu}$ is defined by

$$\begin{aligned}\mathcal{A}^{\mu} f = \frac{\partial f}{\partial t} + (r-\delta)F\frac{\partial f}{\partial F} + \kappa(\gamma - V)\frac{\partial f}{\partial V} + \tfrac{1}{2}F^2 V\frac{\partial^2 f}{\partial F^2} + \tfrac{1}{2}\sigma^2 V\frac{\partial^2 f}{\partial V^2} + \rho\sigma FV\frac{\partial^2 f}{\partial F \partial V} \\ - (r + \mu(t))f. \end{aligned} \tag{4.8}$$

Compared with the standard Heston operator (1.6), two modifications appear. First, the drift of the fund changes from rF to $(r-\delta)F$ to account for the management fee. Second, the discount rate is augmented from r to $r + \mu(t)$, reflecting the possibility of death before maturity. The right-hand side $-\mu(t)F$ represents the instantaneous death benefit paid to beneficiaries when the policyholder dies.

The terminal condition is

$$f(T, F, V) = \max(F, G),$$

and the spatial boundary conditions are obtained from Equation (4.6), evaluated at $F = 0$ and $F = F_{\max}$.

Since $\mu(t)$ is deterministic, it does not introduce a new state variable. The CGPR construction of Chapter 2 therefore applies without structural change. The same Gaussian kernel in (t, F, V) is used, and the covariance matrix is made with the modified operator $\mathcal{A}^{\mu}$. The only difference is in the interior pseudo-observations, which are set to

$$z_i = -\mu(t_i)F_i,$$

instead of zero as in the purely financial case.

4.2.3 Pricing Results

The Heston parameters are taken from Table 3.3. The fee rate is $\delta = 2\%$, the guaranteed floor $G = 100$, and the policyholder is aged $x_0 = 65$ at inception with mortality from the Statbel 2024 total population table. The CGPR uses $m = 400$ interior points, $d = 300$ boundary points and $\sigma^* = 0.01$, with bandwidths calibrated by Nelder–Mead separately for each maturity.

Figure 4.5 shows the CGPR approximation against the semi-analytical benchmark for $T = 1.0$y at $\sqrt{V_0} = 15\%$. The two curves are visually coincident; the MRE on in-the-money contracts is 0.17% and the MAE is 0.1964. The heatmap reveals that the largest errors concentrate around the at-the-money region at low variance levels ($\sqrt{V_0} \leq 10\%$), where the kink in $\max(F_T, G)$ makes the surface hardest to approximate; this is a pattern already observed in the one-asset Heston experiment of Section 3.2.

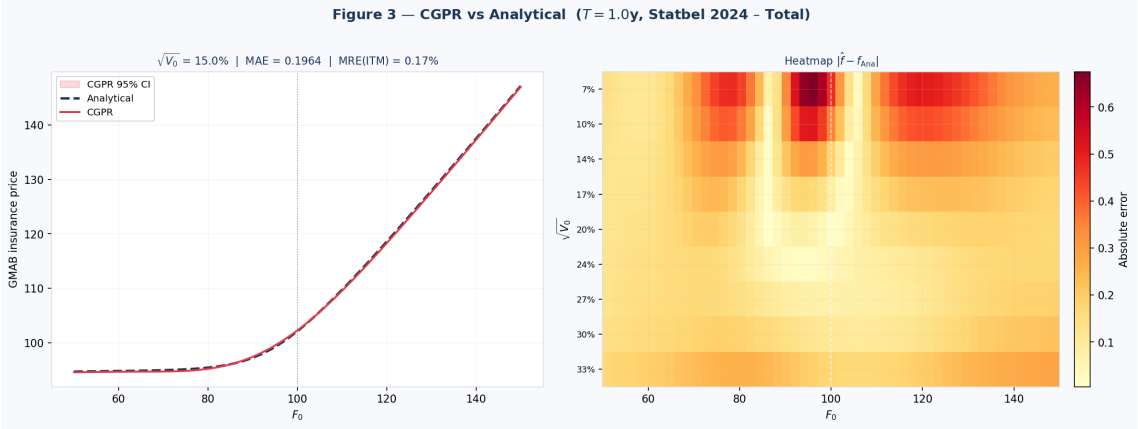

Figure 4.5 – CGPR versus analytical GMAB price ($T = 1.0$y, Statbel 2024 — Total, $\sqrt{V_0} = 15\%$). Left: price curve with 95% CGPR posterior interval. Right: absolute error heatmap over the $(F_0, \sqrt{V_0})$ grid.

Table 4.5 reports the results to six maturities. The MRE stays below 0.55% throughout, growing from 0.14% at $T = 0.5$y to 0.54% at $T = 3.0$y. The MAE grows with maturity, driven by the increasing weight of the death-benefit component and the larger price scale at longer horizons. The calibrated bandwidths remain stable overall. In particular η_F stays close to 22, while η_t increases from 0.234 to 1.297 with the maturity. This is consistent with the observation in the Heston experiment.

Table 4.5 – CGPR GMAB pricing errors by maturity ($G = 100$, $x_0 = 65$, $m = 400$, $\sigma^* = 0.01$, Statbel 2024 — Total).

T	${}_Tp_x$	Ana (ITM)	MAE	MRE (ITM)	Mean σ	η_t	η_F	η_V
0.5y	99.530%	124.3735	0.1580	0.14%	0.0060	0.234	27.13	0.213
1.0y	99.034%	123.6552	0.1964	0.17%	0.0093	0.424	22.02	0.241
1.5y	98.511%	122.8497	0.2942	0.31%	0.0104	0.657	21.46	0.267
2.0y	97.963%	121.9778	0.3547	0.42%	0.0116	0.877	21.37	0.280
2.5y	97.395%	121.0604	0.3940	0.50%	0.0130	1.089	21.32	0.286
3.0y	96.817%	120.1117	0.4032	0.54%	0.0141	1.297	21.44	0.295

As an additional check, the CGPR is compared against a Monte Carlo simulation (30,000 Euler–Maruyama paths, 52 steps per year), yielding MAE(MC) = 0.2031, in line with MAE(Ana) = 0.1964. Both benchmarks give consistent results, which supports the validity of the CGPR approximation (see Appendix A.9, Figure A.1).

Greeks. The Greeks are reported in Figure 4.6.The Figure shows the Delta, Gamma and Vega of the CGPR estimator against finite-difference approximations of the semi-analytical benchmark for $T = 1.0$y. Delta is well reproduced across the full range of fund values. Gamma and Vega reproduce the main shape of the benchmark curves. Their peaks are slightly underestimated near the at-the-money region, which is also where the payoff changes most sharply. This is the same pattern as in Section 3.2.4: a smooth Gaussian kernel naturally struggles to capture the high curvature generated by the kink in $\max(F_T, G)$. Overall, the Greeks remain sufficiently accurate for hedging purposes.

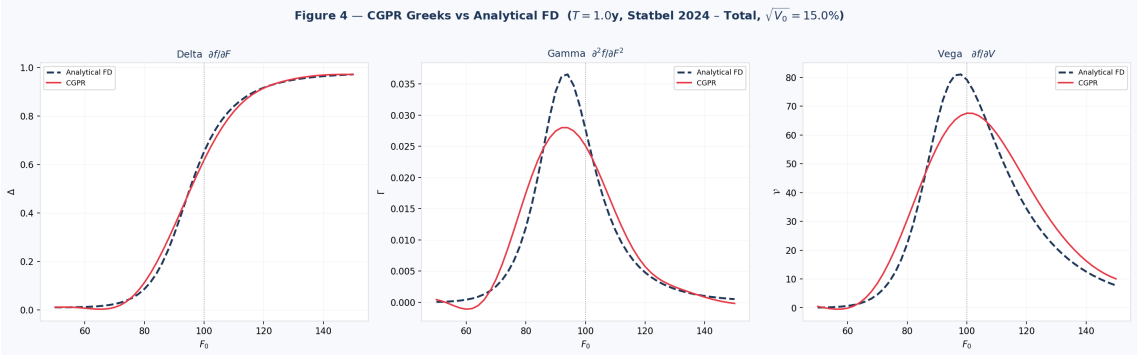

Figure 4.6 – CGPR Greeks versus finite-difference approximations of the semi-analytical benchmark ($T = 1.0$y, Statbel 2024 — Total, $\sqrt{V_0} = 15\%$). Left: Delta $\partial f/\partial F$. Centre: Gamma $\partial^2 f/\partial F^2$. Right: Vega $\partial f/\partial V$. The vertical dotted line marks the guaranteed floor $G = 100$.

CHAPTER 5

CONCLUSION

This thesis studied Constrained Gaussian Process Regression (CGPR) for derivative pricing under the Feynman–Kac framework. Following Hainaut and Vrins [1], the pricing function is approximated by using the pricing PDE directly with the residual imposed at interior points, while payoff and boundary conditions are imposed at selected boundary points. The aim was therefore not to learn from a large database of precomputed prices, but to build a pricing surface that is constrained by the model itself. This also gives access to analytical sensitivities and to a posterior uncertainty measure.

The numerical study started with the Black–Scholes model, mainly as a validation benchmark. Even if this model is too simple for realistic volatility dynamics, it is useful because exact prices and Greeks are available. The observations were that the CGPR reproduced the pricing surface well, producing relative error on in-the-money options below (0.42%) for the baseline configuration. The convergence test also confirmed that increasing the training size improves the approximation. For the Greeks, Delta and Gamma were well recovered, while Theta remained less stable near the upper boundary of the stock-price domain.

The main numerical experiment was the Heston stochastic volatility model. This was the first real stress test of the method, because the pricing function now depends on time, the stock price and the variance level. Against the Carr–Madan FFT benchmark [8], the CGPR kept relative errors below (2%) across maturities in the baseline configuration. The timing results also showed that the costly part is the construction and calibration of the CGPR system, while the fitted surface can then be evaluated quickly. For this vanilla Heston benchmark, FFT remains faster, but

CGPR gives a reusable model-constrained pricing surface with analytical Greeks and a posterior uncertainty measure. The main errors appeared at low variance levels and near the at-the-money region, where the pricing surface has stronger curvature. This was also visible in the Greeks where Delta was the most stable sensitivity, while Gamma and Vega were more difficult to reproduce near the strike.

The two-asset Heston basket option was the most demanding financial extension, which moved the problem from a three-dimensional to a five-dimensional state space. This case brought another challenge because classical benchmark methods become less straightforward in higher dimension. The Carr–Madan FFT used in the one-asset Heston case is not directly reusable in the same form in a multidimensional stochastic-volatility framework, while Monte Carlo remains flexible but is affected by simulation and discretisation error. The largest error (2.42%) was obtained for the shortest maturity, while the other maturities remained close to or below 1%. The online evaluation time was essentially negligible once the CGPR system had been solved, confirming the interest of the offline–online structure in higher dimension. The largest errors were concentrated in low spot and low variance regions, where the pricing surface changes more rapidly. This suggests that future improvements should focus not only on increasing the number of training points, but also on placing them more effectively in the most sensitive parts of the domain.

The last extension applied the same methodology to a Guaranteed Minimum Accumulation Benefit contract. This part brought the financial pricing framework into an actuarial application, since a GMAB combines a financial guarantee with a mortality component. To do so the mortality was introduced through a deterministic force of mortality in the Feynman–Kac operator, without changing the general CGPR construction. Only the operator and the interior pseudo-observations were modified by this added component. The numerical results remained accurate across maturities, with relative errors below 0.55%. This confirms that the PDE-based logic can be extended beyond purely financial payoffs on condition that the insurance component can be written inside the pricing equation.

Overall, the results support the relevance of CGPR as a model-constrained pricing method for both derivatives and simple actuarial products. In line with Hainaut and Vrins [1], its main strengths are its direct use of the Feynman–Kac structure, its analytical Greeks, its posterior uncertainty measure and its offline–online decomposition. The main takeaway is not that CGPR uniformly dominates FFT or Monte Carlo. In simple one-dimensional or semi-analytical settings, classical methods can remain faster or easier to use. The interest of CGPR is different as it builds

a reusable pricing surface directly constrained by the model dynamics, while also providing sensitivities and uncertainty estimates. This makes the method especially interesting when many prices and Greeks must be computed for the same model configuration, or when the benchmark method becomes harder to implement in higher dimension.

The choice of the kernel bandwidths is important and the its optimisation can be very sensitive to local minima. Moreover, The covariance matrix becomes expensive to invert when the number of training points or the dimension increases. Finally, general observations show that the approximation is less precise in regions of high curvature, especially near the strike, at short maturities and for low variance levels. These limitations do not invalidate the method, but they indicate where future improvements should focus.

Future research could follow three directions. First, the method could be tested on more complex products, such as exotic or American options. This would show whether CGPR can handle payoffs with path-dependence or early exercise features, as in the penalised approach of Hainaut [18]. Second, the computational cost could be reduced by using sparse or low-rank approximations of the covariance matrix, since the matrix inversion becomes expensive in higher dimension [3]. Third, other kernels could be considered. The Gaussian kernel works well in this thesis, but Matérn or rational quadratic kernels may be better in regions where the price surface changes more sharply, such as near the strike [19].

APPENDIX A

MATHEMATICAL COMPLEMENTS TO CHAPTER 2

A.1 Woodbury Matrix Identity

The Woodbury matrix identity is used in the derivation of the CGPR estimator in Section 2.2 to convert an expression involving the inversion of an $n \times n$ matrix into one involving the inversion of the $(m+d) \times (m+d)$ matrix C. We state and prove the version used in this thesis.

Lemma A.1.1 (Woodbury identity)**.** *Let U be an $n \times (m+d)$ matrix and V a $(m+d) \times n$ matrix such that $I_n + UV$ is invertible. Then*

$$(I_n + UV)^{-1} U = U (I_{m+d} + VU)^{-1}. \tag{A.1}$$

Proof. Post-multiplying both sides by $(I_{m+d} + VU)$:

$$\begin{aligned}(I_n + UV)^{-1} U (I_{m+d} + VU) &= (I_n + UV)^{-1} (U + UVU) \\ &= (I_n + UV)^{-1} (I_n + UV) U \\ &= U.\end{aligned}$$

Since $(I_{m+d} + VU)$ is invertible (this follows from the invertibility of $I_n + UV$ by a standard determinant argument), dividing both sides on the right by $(I_{m+d} + VU)^{-1}$ gives Equation (A.1). □

In the CGPR derivation, the identity is applied with

$$U = \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L^\top & \sigma^{*-1}\bar{\Phi}^\top \end{pmatrix}, \qquad V = \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L \\ \sigma^{*-1}\bar{\Phi} \end{pmatrix},$$

so that $I_{m+d} + VU = \sigma^{*-2}C$ up to a constant, which transforms the $n \times n$ inversion into the $(m+d) \times (m+d)$ inversion of C.

A.2 Posterior Variance of the CGPR Estimator

We derive the posterior variance formula in Equation (2.29). From the basis function representation $g(x) = w^\top \phi(x)$, the conditional variance of $g(x)$ given (z, h) is

$$\mathrm{Var}(g(x) \mid z, h) = \phi(x)^\top \Sigma_{z,h}\, \phi(x),$$

where $\Sigma_{z,h}$ is the posterior covariance of w defined in Equation (2.21). We now express this in terms of C.

From the Woodbury identity applied to $\Sigma_{z,h}$:

$$\Sigma_{z,h} = I_n - \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L^\top & \sigma^{*-1}\bar{\Phi}^\top \end{pmatrix} C^{-1} \begin{pmatrix} \sigma^{*-1}\mathring{\Phi}_L \\ \sigma^{*-1}\bar{\Phi} \end{pmatrix}.$$

Substituting into $\phi(x)^\top \Sigma_{z,h} \phi(x)$:

$$\mathrm{Var}(g(x) \mid z, h) = \phi(x)^\top \phi(x) - \begin{pmatrix} \mathring{\Phi}_L \phi(x) \\ \bar{\Phi}\phi(x) \end{pmatrix}^\top C^{-1} \begin{pmatrix} \mathring{\Phi}_L \phi(x) \\ \bar{\Phi}\phi(x) \end{pmatrix}.$$

Recognising that $\phi(x)^\top \phi(x) = k(x, x)$ from Equation (2.13), and that

$$\begin{pmatrix} \mathring{\Phi}_L \phi(x) \\ \bar{\Phi}\phi(x) \end{pmatrix} = \boldsymbol{\beta}(x),$$

gives directly

$$\mathrm{Var}(g(x) \mid z, h) = k(x, x) - \boldsymbol{\beta}(x)^\top C^{-1} \boldsymbol{\beta}(x),$$

which is Equation (2.29).

A.3 Kernel Derivatives for the Heston Model

This appendix collects all partial derivatives of the coefficient $\ell_{\tilde{x}}(x, \tilde{x})$ defined in Equation (2.40), needed to evaluate the double operator $L_x L_{\tilde{x}} k(x, \tilde{x})$ in Equation (2.41). We recall the normalised differences from Equation (2.38):

$$\delta_t = \frac{t - \tilde{t}}{\eta_t^2}, \qquad \delta_s = \frac{s - \tilde{s}}{\eta_s^2}, \qquad \delta_v = \frac{v - \tilde{v}}{\eta_v^2}.$$

All derivatives below are with respect to the non-tilde variables (t, s, v).

First-order derivatives of $\ell_{\tilde{x}}$

$$\partial_t \ell_{\tilde{x}} = \frac{1}{\eta_t^2}, \tag{A.2}$$

$$\partial_s \ell_{\tilde{x}} = \frac{1}{\eta_s^2}\Big(r\tilde{s} + \tilde{s}^2\tilde{v}\,\delta_s + \rho\sigma\tilde{s}\tilde{v}\,\delta_v\Big), \tag{A.3}$$

$$\partial_v \ell_{\tilde{x}} = \frac{1}{\eta_v^2}\Big(\kappa(\theta - \tilde{v}) + \rho\sigma\tilde{s}\tilde{v}\,\delta_s + \sigma^2\tilde{v}\,\delta_v\Big). \tag{A.4}$$

Second-order derivatives of $\ell_{\tilde{x}}$

$$\partial_{ss} \ell_{\tilde{x}} = \frac{\tilde{s}^2\tilde{v}}{\eta_s^4}, \tag{A.5}$$

$$\partial_{vv} \ell_{\tilde{x}} = \frac{\sigma^2\tilde{v}}{\eta_v^4}, \tag{A.6}$$

$$\partial_{sv} \ell_{\tilde{x}} = \frac{\rho\sigma\tilde{s}\tilde{v}}{\eta_s^2\,\eta_v^2}. \tag{A.7}$$

A.4 Derivatives of beta for the Greeks

The Greeks in Equations (2.33)–(2.35) require the partial derivatives of $\boldsymbol{\beta}(x)$ with respect to s, v and t. Recall from Equation (2.27) that $\boldsymbol{\beta}(x)$ has two parts:

— the first m entries: $\beta_i(x) = L_{x_i^\circ} k(x_i^\circ, x) = \ell_{x_i^\circ}(x_i^\circ, x) \cdot k(x_i^\circ, x)$,

— the last d entries: $\beta_{m+j}(x) = k(\bar{x}_j, x)$.

Derivatives of the last d entries

These follow directly from the kernel derivatives:

$$\frac{\partial}{\partial s}k(\bar{x}_j, x) = -\delta_s\, k(\bar{x}_j, x),$$
$$\frac{\partial}{\partial v}k(\bar{x}_j, x) = -\delta_v\, k(\bar{x}_j, x),$$
$$\frac{\partial}{\partial t}k(\bar{x}_j, x) = -\delta_t\, k(\bar{x}_j, x),$$

where $\delta_s, \delta_v, \delta_t$ are evaluated at $(x, \tilde{x}) = (x, \bar{x}_j)$.

Derivatives of the first m entries

For the first m entries, using the product rule on $\ell_{x_i^\circ} \cdot k$:

$$\frac{\partial \beta_i}{\partial s} = \frac{\partial \ell_{x_i^\circ}}{\partial s} \cdot k(x_i^\circ, x) + \ell_{x_i^\circ} \cdot \partial_s k(x_i^\circ, x), \tag{A.8}$$
$$\frac{\partial \beta_i}{\partial v} = \frac{\partial \ell_{x_i^\circ}}{\partial v} \cdot k(x_i^\circ, x) + \ell_{x_i^\circ} \cdot \partial_v k(x_i^\circ, x), \tag{A.9}$$
$$\frac{\partial \beta_i}{\partial t} = \frac{\partial \ell_{x_i^\circ}}{\partial t} \cdot k(x_i^\circ, x) + \ell_{x_i^\circ} \cdot \partial_t k(x_i^\circ, x), \tag{A.10}$$

where $\partial_s k$, $\partial_v k$ and $\partial_t k$ are given in Section 2.4, and the derivatives of $\ell_{x_i^\circ}$ with respect to (s, v, t) are obtained from Equations (A.3)–(A.4) after swapping the roles of x and $\tilde{x}$ (i.e. replacing $\tilde{x}$ by x_i° and x by the prediction point).

A.5 Sensitivity to Training Size in the Heston Experiment

This appendix reports the sensitivity of the Heston CGPR results to the number of interior points m and boundary points d, for a fixed maturity $T = 1$ and fixed bandwidths equal to the average of the maturity-specific optimal values.

Sensitivity to m and d

Table A.1 – Sensitivity to m and d in the Heston CGPR experiment ($T = 1$, $\sigma^* = 0.01$, bandwidths fixed).

m	$d = 3m/5$	MAE	MRE ITM	$d = 3m/4$	MAE	MRE ITM	$d = m$	MAE	MRE ITM
400	240	0.1805	1.47%	300	0.2315	2.01%	400	0.1141	1.08%
600	360	0.1001	0.59%	450	0.0637	0.33%	600	0.0637	0.33%
800	480	0.1068	0.40%	600	0.0830	0.28%	800	0.0805	0.26%

Table A.2 – Heston CGPR with fixed bandwidths ($m = 400$, $d = 240$, $\sigma^* = 0.01$, η = average of maturity-specific optima).

T	MAE	MRE ITM	Mean Std.	η_t	η_s	η_v	CGPR time	FFT time
0.5y	0.2168	1.20%	0.0037	1.110	24.28	0.842	0.60s	0.16s
1.0y	0.2349	1.06%	0.0043	1.110	24.28	0.842	0.57s	0.16s
1.5y	0.2561	0.97%	0.0053	1.110	24.28	0.842	0.59s	0.16s
2.0y	0.2753	0.95%	0.0065	1.110	24.28	0.842	0.59s	0.17s
2.5y	0.2962	0.95%	0.0078	1.110	24.28	0.842	0.69s	0.18s
3.0y	0.3259	1.03%	0.0096	1.110	24.28	0.842	0.59s	0.16s

Fixed vs optimised bandwidths

Table A.3 compares the accuracy obtained with maturity-specific optimal bandwidths (Table 7 in the main text) against a configuration where a single fixed bandwidth triplet — equal to the average of the optimal values across maturities — is used for all maturities.

Table A.3 – Heston CGPR with fixed bandwidths ($m = 400$, $d = 240$, $\sigma^* = 0.01$, $\boldsymbol{\eta}$ = average of maturity-specific optima).

T	MAE	MRE ITM	Mean Std.	η_t	η_s	η_v	CGPR time	FFT time
0.5y	0.2168	1.20%	0.0037	1.110	24.28	0.842	0.53s	0.15s
1.0y	0.2349	1.06%	0.0043	1.110	24.28	0.842	0.52s	0.14s
1.5y	0.2561	0.97%	0.0053	1.110	24.28	0.842	0.51s	0.14s
2.0y	0.2753	0.95%	0.0065	1.110	24.28	0.842	0.61s	0.18s
2.5y	0.2962	0.95%	0.0078	1.110	24.28	0.842	0.68s	0.34s
3.0y	0.3259	1.03%	0.0096	1.110	24.28	0.842	0.54s	0.15s

Using fixed bandwidths eliminates the optimisation cost entirely — the CGPR system is assembled and solved in under one second per maturity. The accuracy loss is moderate: the MRE on in-the-money options increases from below 1.86% (optimised) to below 1.20% (fixed) across maturities, with the fixed-bandwidth configuration actually performing slightly better at short maturities. This suggests

that the average bandwidth triplet is a reasonable default when computation time is a binding constraint, for example in a calibration loop requiring repeated CGPR evaluations.

A.6 Carr–Madan FFT Pricing Algorithm

The Carr–Madan algorithm [8] recovers European call prices from the characteristic function of the log-price via a Fast Fourier Transform. We summarise the key steps used in the numerical implementation of Chapter 3.

Characteristic function

Under the Heston model, the risk-neutral characteristic function of $\log S_T$ given (S_0, V_0) is available in closed form [6]:

$$\phi_T(u) = \mathbb{E}^{\mathbb{Q}}\big[e^{iu \log S_T} \mid S_0, V_0\big] = \exp(C(\tau, u) + D(\tau, u)V_0 + iu \log S_0)\,, \tag{A.11}$$

where $\tau = T - t$ and

$$\begin{aligned}
\xi &= \kappa - \rho\sigma i u, \\
d &= \sqrt{\xi^2 + \sigma^2 u(u+i)}, \\
g &= \frac{\xi - d}{\xi + d}, \\
C(\tau, u) &= r\, iu\, \tau + \frac{\kappa\theta}{\sigma^2}\left[(\xi - d)\tau - 2\log\left(\frac{1 - g e^{-d\tau}}{1 - g}\right)\right], \\
D(\tau, u) &= \frac{\xi - d}{\sigma^2} \cdot \frac{1 - e^{-d\tau}}{1 - g e^{-d\tau}}.
\end{aligned}$$

Damped call price

To ensure integrability of the inverse Fourier transform, Carr and Madan introduce a damping factor $e^{\alpha k}$ where $k = \log K$ and $\alpha > 0$. The modified quantity

$$\psi(u) = \frac{e^{-rT}\phi_T\big(u - i(\alpha + 1)\big)}{\alpha^2 + \alpha - u^2 + i(2\alpha + 1)u} \tag{A.12}$$

is integrable over $\mathbb{R}_+$, and the call price at log-strike k is recovered via

$$C(K) = \frac{e^{-\alpha k}}{\pi} \int_0^{+\infty} \mathrm{Re}\big(e^{-iuk}\psi(u)\big)\, du. \tag{A.13}$$

Numerical discretization

The integral in Equation (A.13) is approximated by a discrete sum on a uniform frequency grid $u_j = j\,\eta_{\text{fft}}$ for $j = 0, \ldots, N-1$, where η_{fft} is the frequency spacing. The corresponding log-strike grid is $k_j = -b + \lambda j$ with $\lambda = 2\pi/(N\eta_{\text{fft}})$ and $b = N\lambda/2$. The discrete sum is then evaluated via a standard FFT routine. The implementation uses:

$$N = 4096, \qquad \alpha = 1.5, \qquad \eta_{\text{fft}} = 0.25.$$

The resulting price curve is interpolated at the log-strike of the target option using linear interpolation. These parameter choices are standard in the literature and produce FFT prices with a discretization error that is negligible relative to the CGPR approximation errors reported in Section 3.2.

A.7 Two-Asset Heston: Covariance Matrix and Kernel Derivatives

Instantaneous covariance matrix

The second-order terms of the Feynman–Kac operator in Equation (4.4) are generated by the instantaneous covariance matrix of the state variables (S_1, S_2, V_1, V_2):

$$A(S_1, S_2, V_1, V_2) = \begin{pmatrix} S_1^2 V_1 & \rho_{12} S_1 S_2 \sqrt{V_1 V_2} & \rho_1 \sigma_1 S_1 V_1 & 0 \\ \rho_{12} S_1 S_2 \sqrt{V_1 V_2} & S_2^2 V_2 & 0 & \rho_2 \sigma_2 S_2 V_2 \\ \rho_1 \sigma_1 S_1 V_1 & 0 & \sigma_1^2 V_1 & \rho_V \sigma_1 \sigma_2 \sqrt{V_1 V_2} \\ 0 & \rho_2 \sigma_2 S_2 V_2 & \rho_V \sigma_1 \sigma_2 \sqrt{V_1 V_2} & \sigma_2^2 V_2 \end{pmatrix}. \tag{A.14}$$

The diagonal entries generate the pure second derivatives in Lf. The off-diagonal entries generate the mixed derivative terms: A_{12} gives the $\partial^2 f/\partial S_1 \partial S_2$ term, A_{13} gives $\partial^2 f/\partial S_1 \partial V_1$, A_{24} gives $\partial^2 f/\partial S_2 \partial V_2$, and A_{34} gives $\partial^2 f/\partial V_1 \partial V_2$.

A.8 Two-Asset Heston: Kernel Derivatives

The construction of the covariance matrix C requires $L_{\tilde{x}}k(x,\tilde{x})$ and $L_x L_{\tilde{x}}k(x,\tilde{x})$, where L is the operator in Equation (4.4) and k is the five-dimensional kernel in Equation (4.5).

First operator: $L_{\tilde{x}}k(x,\tilde{x})$

Following the same derivation as in Section 2.4, the first operator factorises as

$$L_{\tilde{x}}k(x,\tilde{x}) = F(x,\tilde{x}) \cdot k(x,\tilde{x}), \tag{A.15}$$

where the scalar coefficient F is obtained by applying $L_{\tilde{x}}$ term by term to k. Defining the normalised differences

$$\delta_\ell = \frac{x_\ell - \tilde{x}_\ell}{\eta_\ell^2}, \qquad \ell = 0, \ldots, 4,$$

with the ordering $x = (t, S_1, S_2, V_1, V_2)$, and using the sign convention $\partial_{\tilde{x}_\ell} k = +\delta_\ell k$, the coefficient is

$$\begin{aligned} F(x,\tilde{x}) = {} & \delta_t - r + r\tilde{S}_1\delta_{S_1} + r\tilde{S}_2\delta_{S_2} + \kappa_1(\gamma_1 - \tilde{V}_1)\delta_{V_1} + \kappa_2(\gamma_2 - \tilde{V}_2)\delta_{V_2} \\ & + \frac{1}{2}A_{11}(\tilde{x})\left(\delta_{S_1}^2 - \frac{1}{\eta_{S_1}^2}\right) + \frac{1}{2}A_{22}(\tilde{x})\left(\delta_{S_2}^2 - \frac{1}{\eta_{S_2}^2}\right) \\ & + \frac{1}{2}A_{33}(\tilde{x})\left(\delta_{V_1}^2 - \frac{1}{\eta_{V_1}^2}\right) + \frac{1}{2}A_{44}(\tilde{x})\left(\delta_{V_2}^2 - \frac{1}{\eta_{V_2}^2}\right) \\ & + A_{12}(\tilde{x})\,\delta_{S_1}\delta_{S_2} + A_{13}(\tilde{x})\,\delta_{S_1}\delta_{V_1} + A_{24}(\tilde{x})\,\delta_{S_2}\delta_{V_2} + A_{34}(\tilde{x})\,\delta_{V_1}\delta_{V_2}, \end{aligned} \tag{A.16}$$

where $A_{ij}(\tilde{x})$ denotes the (i,j) entry of the matrix A in Equation (A.14) evaluated at $\tilde{x}$.

Double operator: $L_x L_{\tilde{x}} k(x, \tilde{x})$

Applying L_x to the product $F \cdot k$ via the Leibniz rule gives

$$\begin{aligned} L_x L_{\tilde{x}} k = &\frac{\partial F}{\partial t} k + F \frac{\partial k}{\partial t} - rFk \\ &+ rS_1 \left(\frac{\partial F}{\partial S_1} k + F \frac{\partial k}{\partial S_1} \right) + rS_2 \left(\frac{\partial F}{\partial S_2} k + F \frac{\partial k}{\partial S_2} \right) \\ &+ \kappa_1(\gamma_1 - V_1) \left(\frac{\partial F}{\partial V_1} k + F \frac{\partial k}{\partial V_1} \right) + \kappa_2(\gamma_2 - V_2) \left(\frac{\partial F}{\partial V_2} k + F \frac{\partial k}{\partial V_2} \right) \\ &+ \frac{1}{2} A_{11}(x) \left(\frac{\partial^2 F}{\partial S_1^2} k + 2 \frac{\partial F}{\partial S_1} \frac{\partial k}{\partial S_1} + F \frac{\partial^2 k}{\partial S_1^2} \right) + \dots \end{aligned} \tag{A.17}$$

where the ellipsis covers the remaining second-order and mixed-derivative terms, which follow the same Leibniz structure. The partial derivatives of F with respect to (t, S_1, S_2, V_1, V_2) are obtained by direct differentiation of Equation (A.16):

$$\begin{aligned} \frac{\partial F}{\partial t} &= \frac{1}{\eta_t^2}, \\ \frac{\partial F}{\partial S_1} &= \frac{1}{\eta_{S_1}^2} \left(rS_1 \delta_{S_1}^* + A_{11}(x) \delta_{S_1}^* + A_{12}(x) \delta_{S_2} + A_{13}(x) \delta_{V_1} \right), \\ \frac{\partial F}{\partial S_2} &= \frac{1}{\eta_{S_2}^2} \left(rS_2 \delta_{S_2}^* + A_{22}(x) \delta_{S_2}^* + A_{12}(x) \delta_{S_1} + A_{24}(x) \delta_{V_2} \right), \\ \frac{\partial F}{\partial V_1} &= \frac{1}{\eta_{V_1}^2} \left(\kappa_1(\gamma_1 - V_1) \delta_{V_1}^* + A_{33}(x) \delta_{V_1}^* + A_{13}(x) \delta_{S_1} + A_{34}(x) \delta_{V_2} \right), \\ \frac{\partial F}{\partial V_2} &= \frac{1}{\eta_{V_2}^2} \left(\kappa_2(\gamma_2 - V_2) \delta_{V_2}^* + A_{44}(x) \delta_{V_2}^* + A_{24}(x) \delta_{S_2} + A_{34}(x) \delta_{V_1} \right), \end{aligned}$$

where $\delta_\ell^* = \delta_\ell \cdot \eta_\ell^2 / \eta_\ell^2 = (x_\ell - \tilde{x}_\ell)/\eta_\ell^4$ denotes the second-order normalised difference. The second-order derivatives follow the same pattern:

$$\begin{aligned} \frac{\partial^2 F}{\partial S_1^2} &= \frac{A_{11}(x)}{\eta_{S_1}^4}, & \frac{\partial^2 F}{\partial S_2^2} &= \frac{A_{22}(x)}{\eta_{S_2}^4}, \\ \frac{\partial^2 F}{\partial V_1^2} &= \frac{A_{33}(x)}{\eta_{V_1}^4}, & \frac{\partial^2 F}{\partial V_2^2} &= \frac{A_{44}(x)}{\eta_{V_2}^4}, \\ \frac{\partial^2 F}{\partial S_1 \partial S_2} &= \frac{A_{12}(x)}{\eta_{S_1}^2 \eta_{S_2}^2}, & \frac{\partial^2 F}{\partial S_1 \partial V_1} &= \frac{A_{13}(x)}{\eta_{S_1}^2 \eta_{V_1}^2}, \\ \frac{\partial^2 F}{\partial S_2 \partial V_2} &= \frac{A_{24}(x)}{\eta_{S_2}^2 \eta_{V_2}^2}, & \frac{\partial^2 F}{\partial V_1 \partial V_2} &= \frac{A_{34}(x)}{\eta_{V_1}^2 \eta_{V_2}^2}. \end{aligned}$$

These expressions are consistent with the implementation in the `VecKernel` class of the numerical code, specifically the methods `_dF_dS1`, `_dF_dS2`, `_dF_dV1`, `_dF_dV2` and `_d2F`.

A.9 GMAB Supplementary Figures and Tables

Monte Carlo Validation

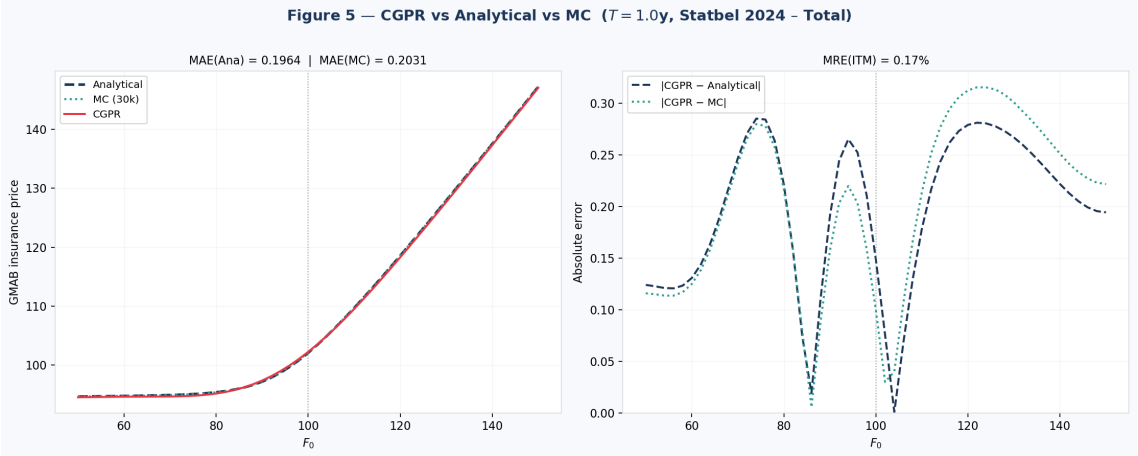

Figure A.1 – CGPR versus semi-analytical benchmark and Monte Carlo ($T = 1.0$y, Statbel 2024 — Total, $\sqrt{V_0} = 15\%$, 30,000 paths). Left: price curves. Right: absolute errors against both benchmarks.

Multi-Maturity Results

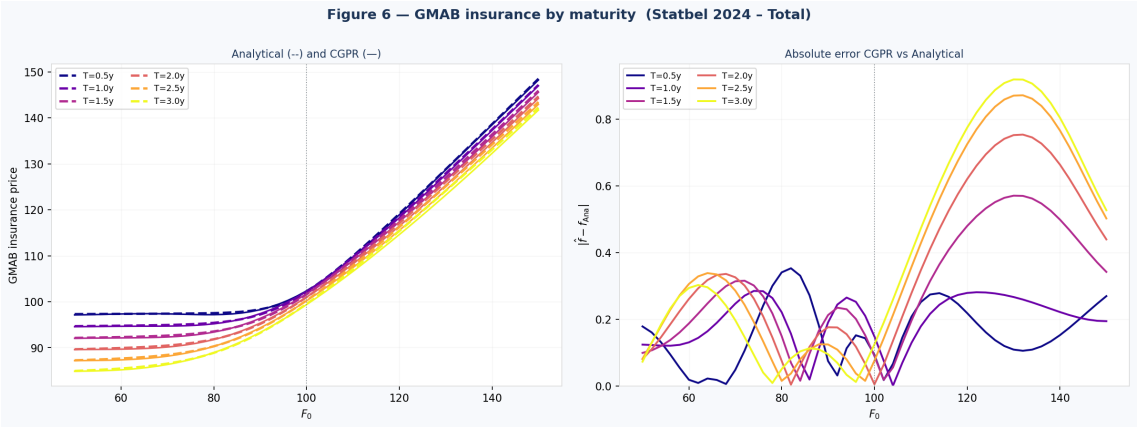

Figure A.2 – GMAB insurance price by maturity (Statbel 2024 — Total, $x_0 = 65$). Left: semi-analytical benchmark (dashed) and CGPR (solid) for $T \in \{0.5, 1.0, 1.5, 2.0, 2.5, 3.0\}$y. Right: absolute error $|\hat{f} - f_{\text{Ana}}|$.

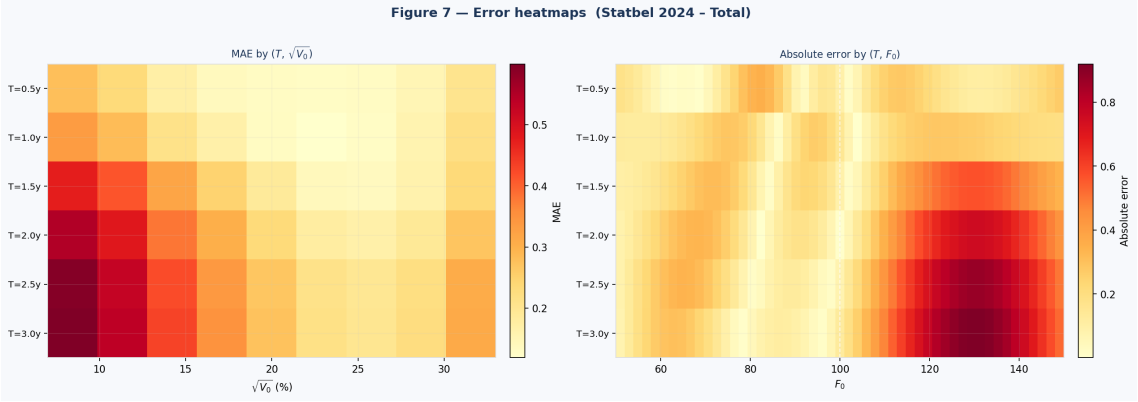

Figure A.3 – Error heatmaps (Statbel 2024 — Total, $x_0 = 65$). Left: MAE by $(T, \sqrt{V_0})$. Right: absolute error by (T, F_0). The white dashed line marks the guaranteed floor $G = 100$.

Gender Sensitivity

Gender sensitivity. Running the pricer independently for men, women, and the total-population table (see Appendix A.9, Table A.A.4) shows that the analytical ITM price differs by less than 0.001 across genders at $x_0 = 65$, $T = 1$y: at this age and horizon, mortality is low enough that the gender distinction is financially negligible. The CGPR achieves MRE = 0.17% uniformly for all three tables with fixed bandwidths, confirming that accuracy is insensitive to the mortality specification once the kernel geometry is fixed.

Table A.4 – Gender sensitivity ($T = 1.0$y, $G = 100$, $x_0 = 65$, $m = 400$, $\sigma^* = 0.01$). Fixed bandwidths calibrated on the total-population table.

Gender	${}_Tp_x$	Ana Price (ITM)	MAE	MRE (ITM)	Mean σ
Male	98.786%	123.6553	0.1959	0.17%	0.0093
Female	99.272%	123.6550	0.1968	0.17%	0.0093
Total	99.034%	123.6552	0.1964	0.17%	0.0093

BIBLIOGRAPHY

[1] D. Hainaut and F. Vrins. "European option pricing with model constrained Gaussian process regressions". *LIDAM Discussion Paper ISBA 2024/21*, Universit'e catholique de Louvain, 2024. Available at: `https://dial.uclouvain.be`.

[2] J. De Spiegeleer, D. B. Madan, S. Reyners and W. Schoutens. "Machine learning for quantitative finance: fast derivative pricing, hedging and fitting". *Quantitative Finance*, 18(10), 1637–1654, 2018. doi: 10.1080/14697688.2018.1495335.

[3] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006. ISBN: 026218253X.

[4] H. Herfurth. "Gaussian Process Regression in Computational Finance". Master thesis, Uppsala University, January 2020. Supervisor: Maciej Klimek.

[5] J. C. Cox, J. E. Ingersoll and S. A. Ross. "A theory of the term structure of interest rates". *Econometrica*, 53(2), 385–407, 1985.

[6] S. L. Heston. "A closed-form solution for options with stochastic volatility with applications to bond and currency options". *The Review of Financial Studies*, 6(2), 327–343, 1993.

[7] M. Jeanblanc, M. Yor and M. Chesney. *Mathematical Methods for Financial Markets*. Springer-Verlag London, 2009. ISBN: 978-1-85233-376-8.

[8] P. Carr and D. Madan. "Option valuation using the fast Fourier transform". *Journal of Computational Finance*, 2(4), 61–73, 1999.

[9] R. Lord, R. Koekkoek and D. Van Dijk. "A comparison of biased simulation schemes for stochastic volatility models". *Quantitative Finance*, 10(2), 177–194, 2010. doi: 10.1080/14697680802392496.

[10] M. E. da Silva and T. Barbe. "Quasi-Monte Carlo in finance: extending for problems of high effective dimension". *Working Paper*, BNP Paribas, 2005.

[11] D. Hainaut. "Valuation of guaranteed minimum accumulation benefits (GMABs) with physics-inspired neural networks". *Annals of Actuarial Science*, 18(2), 442–473, 2024. doi: 10.1017/S1748499524000095.

[12] D. C. M. Dickson, M. R. Hardy and H. R. Waters. *Actuarial Mathematics for Life Contingent Risks*. 2nd ed. Cambridge University Press, 2013. Available at: `https://www.cambridge.org/9781107044074`.

[13] Statbel. "Life expectancy and life tables". Belgian statistical office, 2024 life tables. Available at: `https://statbel.fgov.be/en/themes/population/mortality-life-expectancy-and-causes-death/life-expectancy-and-life-tables`.

[14] F. Black and M. Scholes. "The pricing of options and corporate liabilities". *Journal of Political Economy*, 81(3), 637–654, 1973.

[15] M.J.D. Powell. "An efficient method for finding the minimum of a function of several variables without calculating derivatives". In: *The Computer Journal* 7.2 (1964), pp. 155–162.

[16] M. Rubinstein. "Implied binomial trees". *Journal of Finance*, 49(3), 771–818, 1994.

[17] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, New York, 2004.

[18] D. Hainaut. "American option pricing with model constrained Gaussian process regressions". UCLouvain LIDAM Working Paper, 2024. Available at: `https://dial.uclouvain.be/pr/boreal/object/boreal:292665`.

[19] T. Beckers. "An introduction to Gaussian process models". arXiv preprint arXiv:2102.05497, 2021.

[20] A.B. Owen. "Scrambling Sobol' and Niederreiter–Xing points". In: *Journal of Complexity* 14.4 (1998), pp. 466–489.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Faculté des sciences
Place des Sciences, 2 bte L6.06.01, 1348 Louvain-la-Neuve, Belgique | www.uclouvain.be/sc