A gradient based calibration of the Heston model with real data

Anna Clevenhaus, Claudia Totzeck, Matthias Ehrhardt

Bergische Universität Wuppertal,
{clevenhaus,totzeck,ehrhardt}@math.uni-wuppertal.de

MATHEMATICAL MODELLING, ANALYSIS AND COMPUTATIONAL MATHEMATICS

BERGISCHE UNIVERSITÄT WUPPERTAL

Outline

Heston Model

Gradient Descent Algorithm

Discretization

Numerical Results

Conclusion

European Plain Vanilla Put Option for the Heston Model

- exercise date T
- underlying price S, strike $K \in \mathbb{R}^+$
- payoff function $\phi(S) = \max(K - S, 0) =: (K - S)^+, \quad$ for $S \geq 0$
- Heston model [**Heston (1993)**]

$$\begin{cases} dS_t = (r-q)S_t\,dt + \sqrt{\nu_t}S_t\,dW_t^S, & S_0 > 0, \\ d\nu_t = \kappa(\mu - \nu_t)\,dt + \sigma\sqrt{\nu_t}\,dW_t^\nu, & \nu_0 > 0, \end{cases}$$

- riskless interest rate r, dividend rate q
- variance ν, mean reversion rate κ, long-term mean μ, volatility-of-variance σ
- dW_t^S, dW_t^ν Brownian motions, correlated by $\rho \in [-1, 1]$
- Feller condition: $\kappa\mu - \frac{1}{2}\sigma^2 \geq 0$

Heston PDE and its supplied boundary condition

Heston PDE for pricing a vanilla put option $\tilde{V}(S,\nu,t)$

$$\tilde{V}_t + \frac{1}{2}S^2\nu\tilde{V}_{SS} + \frac{1}{2}\sigma^2\nu\tilde{V}_{\nu\nu} + \rho\sigma\nu S\tilde{V}_{S\nu} + (r-q)S\tilde{V}_S + \kappa(\mu-\nu)\tilde{V}_\nu - r\tilde{V} = 0$$

with terminal condition (payoff of put option)

$$\phi(S) = \max(K - S, 0)$$

and boundary conditions proposed by Heston [**Heston (1993)**]

$$\begin{aligned} S = 0: &\quad \tilde{V} = K\exp\big(-r(T-t)\big) \\ S \to \infty: &\quad \tilde{V} = 0 \\ \nu = 0: &\quad \tilde{V}_t + rS\tilde{V}_S + \kappa\mu\tilde{V}_\nu - r\tilde{V} = 0 \\ \nu \to \infty: &\quad \tilde{V} = K\exp\big(-r(T-t)\big) \end{aligned}$$

log-transformed Heston PDE

Apply the following transformations

- $\log$-transformation asset $x = \log(S)$
- time reversal $\tau := T - t$ to obtain forward-in-time PDE
- $V(x, \nu, \tau) = \tilde{V}(S, \nu, t)$

and obtain $\log$-transformed Heston PDE

$$V_\tau = \frac{\nu}{2} V_{xx} + \frac{1}{2}\sigma^2 \nu V_{\nu\nu} + \left(r - q - \frac{\nu}{2}\right) V_x + \kappa(\mu - \nu) V_\nu + \sigma\nu\rho V_{x\nu} - rV$$

with initial condition

$$V(x, \nu, 0) = \tilde{\phi}(x) = \max(K - \exp(x), 0)$$

and boundary conditions

- for $x \to -\infty$: $V = K \exp(-r\tau)$
- for $x \to \infty$: $V = 0$
- for $\nu \to 0$: $rV = (r - q)V_x + \kappa\mu V_\nu - V_\tau$
- for $\nu \to \infty$: $V = K \exp(-r\tau)$

Fichera Theory

- Parabolic PDE has a singularity at $\nu = 0$

$$V_\tau = \frac{\nu}{2} V_{xx} + \frac{1}{2}\sigma^2 \nu V_{\nu\nu} + (r - q - \frac{\nu}{2}) V_x + \kappa(\mu - \nu) V_\nu + \sigma \nu \rho V_{x\nu} - rV$$

- It reduces to hyperbolic PDE

$$V_\tau = (r - q) V_x + \kappa \mu V_\nu + -rV$$

- For hyperbolic PDEs the boundary flow can be computed explicitly
- We use Fichera theory to determine the need of boundary conditions [**Fichera (1963)**]

$\rightsquigarrow$ Heston PDE is rewritten in divergent form

$$V_\tau - \nabla \cdot A \nabla V + \vec{b} \nabla V - rV = 0$$

with

$$A = \frac{1}{2}\nu \begin{pmatrix} \sigma^2 & \sigma\rho \\ \sigma\rho & 1 \end{pmatrix}, \qquad \vec{b} = -\begin{pmatrix} \kappa(\mu - \nu) - \frac{1}{2}\sigma^2 \\ r - q - \frac{\nu}{2} - \frac{1}{2}\sigma\rho \end{pmatrix}$$

Fichera Theory w.r.t. the variance

For the variance $\nu = 0$

$$\vec{b} = -\begin{pmatrix} \kappa(\mu - \nu) - \frac{1}{2}\sigma^2 \\ r - q - \frac{\nu}{2} - \frac{1}{2}\sigma\rho \end{pmatrix}, \quad \vec{n} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

the Fichera condition is given by

$$b(\nu) = \lim_{\nu \to 0^+} \vec{b} \cdot \vec{n} = \lim_{\nu \to 0^+} \left(\kappa(\mu - \nu) - \frac{1}{2}\sigma^2 - \frac{1}{2}\rho\sigma\nu \right) = \kappa\mu - \frac{1}{2}\sigma^2$$

It follows

- if $b(\nu) \geq 0$ (outflow boundary) we must not supply any BCs at $\nu = 0$
- if $b(\nu) < 0$ (inflow boundary) we have to supply BCs at $\nu = 0$

- Outflow boundary if and only if the Feller condition is fulfilled (which is assumed in the sequel)
- Note: we have to supply a numerical closure boundary condition later

Parameter calibration

- Goal: minimize the cost functional

$$J(V,\xi) = \frac{1}{2}\int_0^T \|V - V_{\mathsf{data}}\|^2 \, dt$$

V is computed using parameter set $\xi = (\kappa, \mu, \sigma, \rho)$

- Lagrange multipliers $\psi = (\varphi, \varphi_a, \varphi_b, \varphi_c, \varphi_d)$
- Domain $\Omega = \Omega_{(x,\nu)} = (-\infty, \infty)_x \times (0, \infty)_\nu$
- Partition boundary $\Gamma = \partial\Omega$ into

$$\Gamma_a = \partial\Omega \cap \{x = -\infty\}, \qquad \Gamma_b = \partial\Omega \cap \{x = \infty\}$$
$$\Gamma_c = \partial\Omega \cap \{\nu = 0\}, \qquad \Gamma_d = \partial\Omega \cap \{\nu = \infty\}$$

- regarding Heston model in divergent form

$$\frac{\partial V}{\partial \tau} - \nabla \cdot A\nabla V + \vec{b} \cdot \nabla V + rV = 0$$

Constraint operator e for the Lagrangian

$$\langle e(V,\xi),\psi\rangle = \underbrace{\int_0^T \int_{\tilde{\Omega}} \Big[\frac{\partial V}{\partial \tau} - \nabla \cdot A\nabla V + \vec{b}\cdot\nabla V + rV\Big]\varphi \, dz d\tau}_{=T}$$

$$+ \underbrace{\int_0^T \int_{\Gamma_a} \big[V - K\exp(-r\tau)\big]\varphi_a \, ds d\tau}_{B_a}$$

$$+ \underbrace{\int_0^T \int_{\Gamma_b} \big[V - K\exp(-r\tau)\big]\varphi_a \, ds d\tau}_{B_b}$$

$$+ \underbrace{\int_0^T \int_{\Gamma_d} \big[V - K\exp(-r\tau)\big]\varphi_d \, ds d\tau}_{B_d}$$

- Γ_c: Due to an pure outflow boundary $\tilde{\Omega} = \Omega \cup \Gamma_c$

Lagrangian for the parameter calibration

- Lagrangian

$$L(V, \xi, \psi) = J(V, \xi) - \langle e(V, \xi), \psi \rangle$$

- first-order optimality conditions [**Hinze (2009)**]

$$0 = d_V L(V, \xi, \psi)[h] = d_V J(V, \xi)[h] - d_V \langle e(V, \xi), \psi \rangle [h]$$

- Computation of $d_V J(V, \xi)$

$$J(V, \xi) = \frac{1}{2} \int_0^T \int_\Omega (V - V_{\mathsf{data}})^2 \, dz d\tau$$

$$d_V J(V, \xi)[h] = \int_0^T \int_\Omega (V - V_{\mathsf{data}}) \, dz d\tau$$

Computation of $d_V \langle e(V,\xi), \psi\rangle[h]$

Using Green's first identity, we rewrite T

$$T = \int_0^T \int_{\tilde\Omega} \Big[\frac{\partial V}{\partial \tau} - \nabla \cdot A\nabla V + \vec{b}\cdot\nabla V + rV\Big]\varphi\, dz d\tau$$

$$= \underbrace{\Big[\int_{\tilde\Omega} \varphi h\, dz\Big]_{\tau=0}^{\tau=T}}_{=0}$$

$$+ \int_0^T \int_{\tilde\Omega} V\Big[-\frac{\partial \varphi}{\partial \tau} \underbrace{-\nabla\cdot A^\top \nabla\varphi - \vec{b}\cdot\nabla\varphi + (r - \nabla\cdot\vec{b})\varphi}_{=\mathcal{L}\varphi}\Big]\, dz d\tau$$

$$+ \int_0^T \int_{\Gamma_a\cup\Gamma_b\cup\Gamma_d} \Big(\big[(A^\top\nabla\varphi)\cdot\vec{n} + (\vec{b}\cdot\vec{n})\varphi\big]V - (A\nabla V)\cdot\vec{n}\varphi\Big)\, ds d\tau$$

Overall

Including the boundary condition leads to

$$d_V \langle e(V,u), \psi\rangle[h] = \int_0^T \int_{\tilde{\Omega}} h\Big(-\frac{\partial \varphi}{\partial \tau} + \mathcal{L}\varphi\Big)\, dz$$
$$+ \underbrace{\int_0^T \Big(\int_{\Gamma_a} h\varphi_a\, ds + \int_{\Gamma_b} h\varphi_b\, ds + \int_{\Gamma_d} h\varphi_d\, ds\Big) d\tau}_{B_1}$$
$$+ \underbrace{\int_0^T \int_{\Gamma_a \cup \Gamma_b \cup \Gamma_d} [(A^\top \nabla\varphi)\cdot \vec{n} + (\vec{b}\cdot\vec{n})\varphi] h\, ds - (A\nabla h)\cdot \vec{n}\varphi\, ds d\tau}_{B_2}$$

- $B_1 + B_2 = 0$ **[Clevenhaus (2024)]**

Derivation of the Adjoint

$$
\begin{aligned}
0 &= d_V J(V,u)[h] - d_V \langle e(V,u), \psi \rangle [h] \\
&= \int_0^T \int_\Omega (V - V_{\text{data}}) - \left(-\frac{\partial \varphi}{\partial \tau} + \mathcal{L}\varphi \right) dz d\tau
\end{aligned}
$$

Using the Variational Lemma gives the (formal) Adjoint equation

$$
\frac{\partial \varphi}{\partial \tau} + \nabla \cdot A^\top \nabla \varphi + \vec{b} \cdot \nabla \varphi - (r - \nabla \cdot b)\varphi = -(V - V_{\text{data}}) \quad \text{on } \tilde{\Omega}
$$

(Analytic) Boundary Conditions:

- Γ_a: $\varphi = 0$
- Γ_b: $\varphi = 0$
- Γ_c: No analytic boundary conditions needed (pure outflow boundary according to Fichera Theory)
- Γ_d: $\varphi = 0$

Computation of the Gradient

- Recall: κ, μ, σ and ρ are parameters (and not variables!)

$\rightsquigarrow$ gradient for V, V_{data} and φ w.r.t. those is zero and we can focus on

$$\int_0^T \int_{\hat{\Omega}} -V\Big[\frac{1}{2}\nu\sigma^2\varphi_{\nu\nu} + \nu\sigma\rho\varphi_{x\nu} + \frac{1}{2}\nu\varphi_{xx} + (\sigma^2 - \kappa(\mu-\nu))\varphi_\nu + (q - r + \frac{\nu}{2} + \sigma\rho)\varphi_x + (\kappa - r)\varphi\Big]\, dz d\tau$$

to derive the gradient, by computing the derivative to

- σ: $\int_0^T \int_{\hat{\Omega}} -V\left[\nu\sigma\varphi_{\nu\nu} + \nu\rho\varphi_{x\nu} + 2\sigma\varphi_\nu + \rho\varphi_x\right] dz d\tau$
- ρ: $\int_0^T \int_{\hat{\Omega}} -V\left[\nu\sigma\varphi_{x\nu} - \sigma\varphi_x\right] dz d\tau$
- κ: $\int_0^T \int_{\hat{\Omega}} V\left[(\mu-\nu)\varphi_\nu - \varphi\right] dz d\tau$
- μ: $\int_0^T \int_{\hat{\Omega}} V\left[\kappa\varphi_\nu\right] dz d\tau$

- Note: A time dependent gradient can be derived accordingly

Amijo Rule

Since the parameter domain for κ, μ, σ, and ρ is restricted, and the Feller condition must be satisfied, we use the *projected Armijo rule* [**Troeltzsch (2009)**]

- **projected Armijo rule:**
  we choose the maximum $\sigma_k \in \{1, 1/2, 1/4, \ldots\}$, for which

$$f\big(\mathcal{P}(\xi_k - \sigma_k \nabla f(\xi_k))\big) - f(\xi_k) \leq -\frac{\gamma}{\sigma_k} \|\mathcal{P}(\xi_k - \sigma_k \nabla f(\xi_k)) - \xi_k\|_2^2$$

- $\gamma \in (0, 1)$ is a numerical constant that depends on the problem and is typically chosen to be $\gamma = 10^{-4}$ (used later in numerical results)

Gradient Descent Algorithm

Algorithm 1: Gradient descent method for Heston parameter calibration

Result: calibrated parameters for Heston model

- initialize parameters u_0

while $\|\text{gradient}\| > \epsilon$ **do**

- solve the Heston PDE
- solve the formal Adjoint for the Heston PDE
- compute the gradient
- line search for step size
- update the parameter set

end

Discretization

Spatial discretization

- $x_i = x_{\min} + i\Delta_x$ for $i = 0, \ldots, N_x$ with $\Delta_x = (x_{\max} - x_{\min})/N_x$
- $\nu_j = \nu_{\min} + j\Delta_\nu$ for $j = 0, \ldots, N$ with $\Delta_\nu = \nu_{\max}/N_\nu$
- (standard) second order FD stencils

Time discretization

- $\tau_k = k \cdot \Delta_\tau$ for $k = 0, \ldots, N_\tau$ with $\Delta_\tau = \frac{T}{N_\tau}$
- Hundsdorfer-Verwer (HV) with $\theta = 0.75$ [**Hundsdorfer (2002)**]
  - Alternating Direction Implicit (ADI) scheme
  - scheme of order two for any θ

- Note: For simplicity we use standard schemes, as the main focus is on the gradient and later on the space mapping

Boundary treatment for the variance

Based on the Fichera Theory, we need to supply a closure boundary condition for $\nu \to 0$. As it is a pure outflow boundary, we use

- Extrapolation via Ghost layer
- at $\nu = \nu_{\min} - \Delta_\nu$

$$V(x_i, \nu_{\min} - \Delta_\nu, \tau_k) = V(x_i, \nu_{\min}, \tau_k)$$

As we truncate the domain at $\nu_{\max} = 1$ and the proposed boundary condition is given for $\nu \to \infty$, we also propose extrapolation

- Extrapolation via Ghost layer
- at $\nu = \nu_{\max} + \Delta_\nu$

$$V(x, \nu_{\max}, \tau) = V(x, \nu_{\max} + \Delta_\nu, \tau)$$

Following the approach of [**Kutik (2015)**], as it is the best closure condition for the gradient method [**Clevenhaus (2023)**]

Numerical adjustment of the cost functional

In the theoretical approach, the cost functional is given by

$$J(V,\xi) = \frac{1}{2}\int_0^T \|V - V_{\mathsf{data}}\|^2 \, dt$$

suggesting that the option price is given for all grid points in x and ν direction and at all time instances t

- However, when using real data, we usually get only one option price at a particular $\hat{x}$ and maturity T, where K, r, and q are given

$$\tilde{V}_{\mathsf{data}}(\hat{x}) = P\big(\log(\hat{S}), K, r, q, T\big)$$

- Thus we search for the optimal parameters

$$\hat{\xi} = (\hat{\nu}, \kappa, \mu, \sigma, \rho)$$

- desired option price is a discrete value within the Heston PDE solution
- Therefore, we adjust the cost functional by a minimal expansion of the given value to an additional grid point by constant extrapolation, s.t. the computation of the gradient is possible

Concrete adjustment strategy

- Adjustment in space:
  - Since there are no constraints on $\hat{\nu}$, we place $\hat{\nu}$ on a grid point to avoid extrapolation w.r.t. ν
  - We extrapolate the constant values over the grid points used within the finite difference stencils in the gradient computation
- Adjustment w.r.t. time:
  Since $\varphi(x, \nu, T) = 0$, we use the values at τ_{k-2} and τ_{k-1} for the gradient computation

Resulting V_{data}:
Let $x_k = \hat{x}$ and $\nu_l = \hat{\nu}$, then set $V_{\text{data}}(x_{k\pm5}, \nu_{l\pm5}, T)$ to $\tilde{V}_{\text{data}}(\hat{x})$ else $V_{\text{data}}(x_i, \nu_j, \tau_k) = 0$
Overall the cost function is given by

$$\hat{J}(V, \hat{\xi}) = \frac{1}{2} \int_{\tau_{k-2}}^{\tau_{k-1}} \|V - V_{\text{data}}\|^2 \, dt$$

Heston Model

Gradient Descent Algorithm

Discretization

Numerical Results

Conclusion

Numerical Setup

To obtain $\hat{x}$ on a grid point, we set

- $x_{\min} = dx$, $x_{\max} = 1.2 \log(\hat{S})$, $N_x = 120$
- $\nu_{\min} = 0.01$, $\nu_{\max} = 1$, $N_\nu = 100$
- $\tau_{\min} = 0$, $\tau_{\max} = 1$, $\Delta_\tau = 0.45\Delta_x^2$

From the Nikkei Stock Index 300 on December 31, 2012, we obtain K, r as well as q for T

	κ^{guess}	μ^{guess}	σ^{guess}	ρ^{guess}
Test 1	5.0	0.2	0.6	-0.3
Test 2	3.0	0.1	0.3	-0.2
Test 3	4.0	0.15	2	-0.4

Table: Test Cases for the calibration to market data

Calibrated values

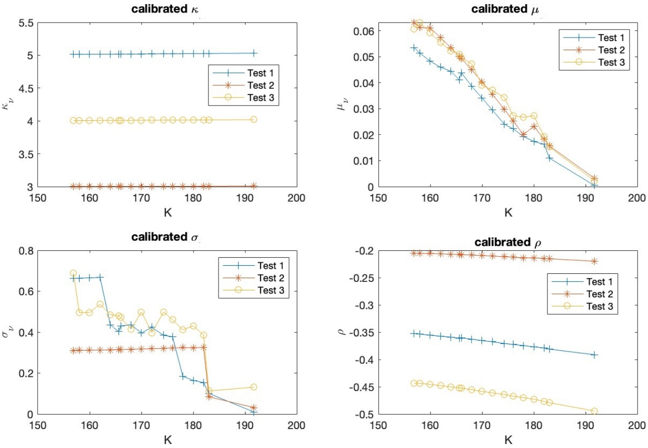

Figure: Calibrated constant values for the different test cases

Relative cost functional improvement

$$\delta = 100 \frac{J(V, \hat{\xi}^{\text{guess}}) - J(V, \hat{\xi}^{\text{calibrated}})}{J(V, \hat{\xi}^{\text{guess}})}$$

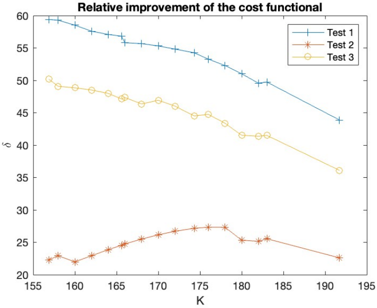

Figure: Relative cost functional improvement in percent for the different test cases in a constant setting

Mean Square Error

$$\text{MSE} = \frac{1}{N}\sum_{i=0}^{N}\left(V_{\text{data}}{}^{i} - V^{i}(\hat{\xi}_i^{\text{calibrated}})\right)^2$$

	Test 1	Test 2	Test 3
MSE	$5.2 \cdot 10^{-3}$	$7.9 \cdot 10^{-3}$	$3.5 \cdot 10^{-3}$

Table: MSE for different strike values for the different test cases from the Nikkei Stock Index 300 on December 31, 2012

Conclusion

- The gradient algorithm works well with real market data
- The time dependent parameters need more real market data, application to American option pricing or path-dependent options
- Next step: Apply the gradient method as a coarse solver within a space mapping approach

References I

[1] A. Clevenhaus, C. Totzeck, M. Ehrhardt, A gradient based calibration method for the Heston model, accepted: International Journal of Computer Mathematics, 2024.

[2] A. Clevenhaus, C. Totzeck, M. Ehrhardt, A numerical study of the impact of variance boundary conditions for the Heston model. In: K. Burnecki, J. Szwabiński, and M. Teuerle (eds.) Progress in Industrial Mathematics at ECMI 2023, The European Consortium for Mathematics in Industry, Springer, 2024.

[3] G. Fichera, "On a unified theory of boundary value problems for elliptic-parabolic equations of second order", Matematika, 7:6 (1963), 99–122; Boundary problems in differential equations, Univ. of Wisconsin Press, Madison, 1960, 97–120

[4] S.L. Heston. A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options, Rev. Fin. Stud. 6(2) (1993), 327-343

[5] Hinze, M. and Pinnau, R. and Ulbrich, M. and Ulbrich, S., Optimization with PDE Constraints, Springer (2009)

[6] W. Hundsdorfer. Accuracy and Stability of Splitting with Stabilizing Corrections, Appl. Numer. Math., 42(1-3) (2002), 213-233

[7] P. Kútik, K. Mikula. Diamond-cell finite volume scheme for the Heston model, DCDS-S, 8(5) (2015), 913-931.

[8] Tröltzsch, F., Optimale Steuerung partieller Differentialgleichungen, Springer (2009)

Conclusion

- The gradient algorithm works well with real market data
- The time dependent parameters need more real market data, application to American option pricing or path-dependent options
- Next step: Apply the gradient method as a coarse solver within a **space mapping approach**

Thank you for your attention!

Basic Ideas of the Space Mapping Technique

This approach bridges the gap between accuracy and efficiency, making it valuable in scenarios where computational costs matter

Problem Statement

we want to optimize some system for which we have a **model hierarchy** consisting of a **fine model (f)** and a **coarse model (c)**

- **fine model:** accurate, but expensive to evaluate
- **coarse model:** less accurate, but cheap to evaluate

Assumption

Optimization of the coarse model is possible and comparatively cheap, whereas the optimization of the fine model is very expensive or even impossible due to the high cost of the fine model simulations

- **fine model response** $f\colon X \to Y$, where X, Y Banach spaces (X is called 'control space', Y is called 'codomain')
  - **fine model optimization problem**

$$\min_{x \in X} J(f(x)) \tag{F}$$

i.e., we minimize a cost functional $J\colon Y \to \mathbb{R}$ which depends on $f(x)$

(F) is, in general, too complex and too expensive to be solved directly

- **coarse model response** $c\colon X \to Y$. For simplicity, we assume that the control spaces as well as the codomains of the model responses coincide

We approximate (F) by the **coarse model optimization problem**

$$\min_{x \in X} J(c(x)) \tag{C}$$

To obtain well-posed problems, we assume that both (F) and (C) have a unique minimizer. Consequently, we introduce

$$x_{\mathrm{f}}^* := \operatorname*{argmin}_{x \in X} J(f(x)) \quad \text{and} \quad x_{\mathrm{c}}^* := \operatorname*{argmin}_{x \in X} J(c(x)) \tag{M}$$

A naive approach for approximating x_f^* is to solve (C) and use its minimizer x_c^* as approximation of x_f^*.

The space mapping technique extends and generalizes this approach.

Space Mapping Function

We introduce the space mapping function $s\colon X \to X$, $x_\mathrm{f} \mapsto s(x_\mathrm{f})$, where

$$s(x_\mathrm{f}) := \operatorname*{argmin}_{x_\mathrm{c} \in X} r\big(c(x_\mathrm{c}), f(x_\mathrm{f})\big) \tag{S}$$

with some **misalignment function** $r\colon Y \times Y \to \mathbb{R}$, which is used to measure the discrepancy between the fine and coarse model responses

To get a well-defined space mapping function, we assume that problem (S) is well-posed, i.e., that it has a unique minimizer for all $x_\mathrm{f} \in X$.

Assumption: the misalignment function r is **exact** in the sense that

$$\operatorname*{argmin}_{x_\mathrm{c} \in X} r\big(c(x_\mathrm{c}), c(z)\big) = z \quad \text{for all} \quad z \in X$$

For a given fine model control x_f, the space mapping function s computes the best coarse model control x_c such that the discrepancy between the fine and coarse model responses $f(x_\mathrm{f})$ and $c(x_\mathrm{c})$ is minimized.

Assumption: coarse model is a suitable approximation of the fine model, at least in the vicinity of their respective minimizers, i.e., $c(x_\mathrm{c}^*) \approx f(x_\mathrm{f}^*)$

Under this assumption, we expect that

$$s(x_\mathrm{f}^*) = \operatorname*{argmin}_{x_\mathrm{c} \in X} r\big(c(x_\mathrm{c}), f(x_\mathrm{f}^*)\big) \approx \operatorname*{argmin}_{x_\mathrm{c} \in X} r\big(c(x_\mathrm{c}), c(x_\mathrm{c}^*)\big) = x_\mathrm{c}^*$$

due to the exactness of the misalignment function.

$\rightsquigarrow$ **fundamental idea** of space mapping technique is to solve the equation

$$s(x_\mathrm{f}^*) - x_\mathrm{c}^* = 0$$