Gaussian Homotopy Optimization with Predictor–Corrector Path Tracking and Deterministic Sampling

Xiaozhu Luo, Markus Walker, and Uwe D. Hanebeck

***Abstract*— Non-convex optimization is a fundamental challenge across engineering and scientific applications, where gradient-based methods frequently converge to suboptimal local minima. Standard Gaussian Homotopy (GH) methods address this by progressively smoothing the objective landscape, yet rely on heuristic parameter schedules that do not exploit the local geometry of the homotopy path, limiting both convergence speed and robustness. To overcome these limitations, we propose a Predictor–Corrector Gaussian Homotopy (PCGH) method that formulates Gaussian homotopy optimization as a path tracking problem. By deriving the Davidenko continues Ordinary Differential Equation (ODE) for GH, stationary points are tracked through predictor–corrector continuation rather than iterative optimization. We further introduce geometry-aware arc-length parameterizations based on Fisher-information and Hessian-derived metrics, together with deterministic Fibonacci sampling for efficient approximation of the Gaussian convolution and its derivatives. Experiments on standard 2D benchmark functions demonstrate that PCGH method consistently reaches low-cost solutions with fewer function evaluations than conventional GH methods, exhibiting comparable convergence reliability.**

***Index Terms*— Unconstrained optimization, global optimization, Gaussian homotopy, deterministic sampling.**

I. Introduction

Many challenges across sensor fusion, state estimation, and robotic perception are formulated as non-convex optimization problems [1]–[3]. Gradient-based methods such as steepest descent and Newton are efficient in convex settings but inherently local, so in non-convex problems they can become trapped in poor local minima and are sensitive to initialization.

To circumvent the limitations of local optimization methods, a variety of global optimization techniques have been extensively developed. Stochastic and probabilistic optimization methods, including simulated annealing [4], evolutionary algorithm [5], and particle swarm optimization [6], have been widely employed to address non-convex optimization problems. Complementing these strategies, Homotopy optimization presents a systematic alternative by deforming the complex cost landscape into an easily optimizable surrogate, guiding the search toward the global optimum's basin of attraction.

Historically, homotopy methods were developed as numerical techniques to find roots of complex nonlinear equations by tracking the solution path from a simple, tractable system as it is continuously deformed into the target system [7]. This fundamental principle naturally extends to non-convex optimization [8], [9]. Instead of tracking roots, homotopy optimization constructs a continuous sequence of surrogate objective functions, smoothly bridging the complex original landscape with a tractable simplified function possessing a known, easily attainable global minimum, and tracks the minimizer across these functions (see Fig. 1). This deformation is parameterized by a homotopy parameter.

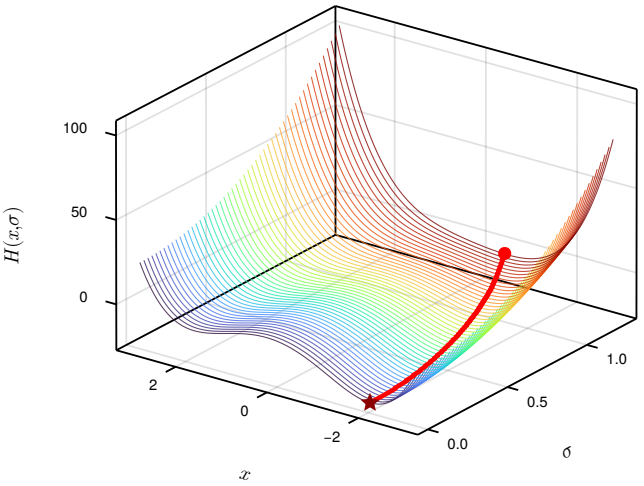


Fig. 1: Illustration of homotopy optimization. The homotopy $H(\underline{x}, \sigma)$ smoothly interpolates between a simplified surrogate and the original objective $f(\underline{x})$. The red curve tracks a stationary point of H as the homotopy parameter σ varies, connecting the surrogate's minimizer to a stationary solution of $f(\underline{x})$.

Among various choices of homotopy strategies, Gaussian Homotopy (GH) stands out as a particularly robust and reliable approach. GH continuously deforms the original cost function by convolving it with a Gaussian kernel. The homotopy path transitions from a highly smoothed (large standard deviation) landscape, where local non-convex structures are suppressed, to the original (small standard deviation) landscape with all its complexities. In addition, the relationship between GH and the convex envelope of the original function has been established in [10], providing theoretical insights into the advantages of GH in non-convex optimization.

Despite this theoretical elegance, classical GH suffers from severe computational overhead due to a cumbersome double-loop structure, where an outer loop progressively restores the original non-convex landscape while an inner loop employs a

This work is part of the German Research Foundation (DFG) AI Research Unit 5339 regarding the combination of physics-based simulation with AI-based methodologies for the fast maturation of manufacturing processes.

Xiaozhu Luo, Markus Walker, and Uwe D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology, Germany (e-mail: {firstname.lastname}@kit.edu).

local optimizer to track the evolving minimizer. To eliminate this bottleneck, recent frameworks such as Single Loop Gaussian Homotopy (SLGH) [11] optimize decision variables and the homotopy variable simultaneously in a single loop. However, SLGH optimizes the joint problem in Euclidean space, causing convergence difficulties due to the geometric mismatch between the decision and homotopy variables. The authors resort to clamping the homotopy update—a heuristic fix that can yield suboptimal performance.

Another challenge in GH is that the Gaussian convolution and its derivatives are generally intractable for arbitrary objective functions and must be approximated numerically. Standard Monte Carlo approximations suffer from high variance at small sample sizes, degrading gradient and Hessian estimates when accuracy matters most. We instead use deterministic, low-discrepancy samples, which provide more uniform coverage and lower variance. Here we employ Fibonacci sampling [12]; other deterministic schemes, such as Localized Cumulative Distribution (LCD)-based sampling [13], are equally applicable.

Contributions: We propose a Predictor–Corrector Gaussian Homotopy (PCGH) method for solving multimodal non-convex optimization problem, drawing fundamental theoretical inspiration from numerical continuation [14]. Our specific contributions are

- deriving the Davidenko ODE for GH and using it for predictor-based path tracking,
- introducing arc-length parameterization with Fisher and Hessian based metrics for geometry-aware adaptive stepping, and
- employing deterministic Fibonacci sampling for lower-variance approximation of the Gaussian convolution and its derivatives.

Notation: Throughout this paper, scalars are denoted by lowercase letters, vectors by underlined lowercase letters (e.g., $\underline{x}$), matrices by boldface uppercase letters (e.g., $\mathbf{A}$), and sets by calligraphic letters (e.g., $\mathcal{C}$). We write $\mathbf{I}$ for the identity matrix, $\underline{0}$ for the zero vector, $(\cdot)^\top$ for transpose, and $\|\cdot\|$ for the Euclidean norm.

II. Problem Formulation

We consider unconstrained optimization problems of the form

$$\min_{\underline{x}} f(\underline{x})$$

where $\underline{x} \in \mathbb{R}^n$ is the n-dimensional decision variable and $f\colon \mathbb{R}^n \to \mathbb{R}$ is a continuous objective function. Iterative optimization algorithms, including gradient descent and Newton-type methods, seek stationary points satisfying $\nabla_{\underline{x}} f(\underline{x}) = \underline{0}$. While such methods are efficient for convex objectives, on non-convex landscapes they depend strongly on initialization and may converge to suboptimal stationary points because of local minima, saddle points, or flat regions.

Alternative global-search strategies can mitigate poor local minima, but they are often heuristic and require many function evaluations. In contrast to stochastic exploration, homotopy optimization presents a systematic alternative by constructing a continuous, mathematically well-defined deformation from a tractable auxiliary function to the target non-convex objective. Rather than optimizing the original objective function directly, the optimizer recursively tracks a continuous trajectory of stationary points induced by the deformation process. Since the neighboring optimization landscapes are similar, the minimizer obtained at the current homotopy iteration provides an effective warm start for the subsequent iteration.

Formally, a homotopy mapping is defined as a continuous function

$$H\colon \mathbb{R}^n \times [0, +\infty) \to \mathbb{R}$$

parameterized by a homotopy parameter σ[1], where larger values of σ produce a smoother and more tractable surrogate of the original objective, and

$$H(\underline{x}, \sigma = 0) = f(\underline{x}) \ .$$

Homotopy optimization starts from a sufficiently large σ_0, for which $H(\underline{x}, \sigma_0)$ is easy to optimize, and then tracks the stationary solutions as σ decreases toward zero.

We define the set of stationary solutions associated with the homotopy deformation as

$$\mathcal{C} = \left\{ (\underline{x}, \sigma) \in \mathbb{R}^n \times [0, +\infty) \mid \nabla_{\underline{x}} H(\underline{x}, \sigma) = \underline{0} \right\} \ .$$

Homotopy optimization tracks a path on $\mathcal{C}$ from the minimizer of the simplified function toward a stationary solution of the original objective $f(\underline{x})$, as illustrated in Fig. 1.

The effectiveness of homotopy optimization depends strongly on the homotopy construction, since different deformations induce different paths and numerical behavior. A homotopy that yields a smooth and stable solution path is therefore essential.

III. Gaussian Homotopy (GH)

We employ GH as the deformation strategy. Its connection to the heat diffusion equation [10], [15] provides a principled smoothing mechanism that suppresses local non-convex structures while preserving the large-scale geometry of the objective landscape.

A. Formal Definition

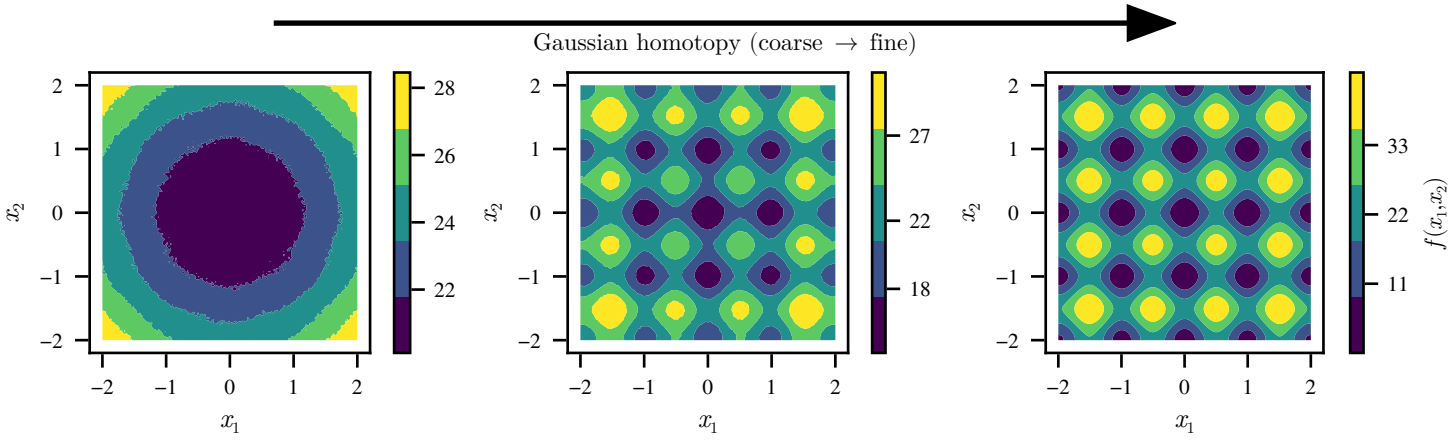
GH constructs a continuous deformation of the objective landscape through Gaussian smoothing. By convolving an isotropic Gaussian kernel with the objective function, local non-convex structures are progressively suppressed, producing a coarse-to-fine representation of the optimization landscape. The GH is formally defined as

$$\begin{aligned} H_{\mathrm{G}}(\underline{x}, \sigma) &= f(\underline{x}) * k_\sigma(\underline{x}) \\ &= \int_{\mathbb{R}^n} f(\underline{\eta}) k_\sigma(\underline{x} - \underline{\eta}) \, \mathrm{d}\underline{\eta} \ , \end{aligned} \tag{1}$$

where $*$ denotes the convolution operator and k_σ is the isotropic Gaussian kernel

$$k_\sigma(\underline{x}) = \frac{1}{(2\pi)^{\frac{n}{2}} \sigma^n} \exp\left(-\frac{\|\underline{x}\|^2}{2\sigma^2} \right) \ , \tag{2}$$

[1]The homotopy parameter is often defined in $[0, 1]$ in other publications, with 0 corresponding to the simplified function and 1 to the original objective. A mapping from $[0, 1]$ to $[0, +\infty)$ can be used to transform the parameter.



(a) Smoothed ($\sigma = 0.5$) (b) Smoothed ($\sigma = 0.25$) (c) Original

Fig. 2: Gaussian smoothing of Rastrigin function for different values of the smoothing parameter σ.

with $\|\cdot\|^2$ denoting the squared Euclidean norm and σ the standard deviation.

The smoothing parameter σ controls the scale of the deformation. Large values yield strongly smoothed landscapes, while smaller values gradually recover the original objective. This coarse-to-fine transition is shown in Fig. 2.

By interpreting the Gaussian kernel (2) as the probability density function, the GH in (1) can equivalently be expressed as the expectation

$$H_{\mathrm{G}}(\underline{x}, \sigma) = \mathrm{E}_{p(\underline{\eta};\underline{x},\sigma^2\mathbf{I})}\{f(\underline{\eta})\} \ . \tag{3}$$

This probabilistic interpretation enables approximation of the GH through Monte Carlo integration, avoiding direct calculation of the high-dimensional convolution integral, which in general has no analytic solution.

B. Differential Properties

The differentiability of the Gaussian kernel allows the gradient, Hessian, and mixed derivatives of the GH to be derived explicitly without requiring derivatives of the objective. These identities play a central role in the path-tracking and Newton-based solvers.

1) Gradient: The gradient of the GH w.r.t. $\underline{x}$ can be derived by interchanging differentiation and integration, which is justified under standard regularity assumptions on f, yielding

$$\begin{aligned} \nabla_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma) &= \int_{\mathbb{R}^n} f(\underline{\eta})\, \nabla_{\underline{x}} k_\sigma(\underline{x} - \underline{\eta})\ \mathrm{d}\underline{\eta} \\ &= \frac{1}{\sigma^2} \int_{\mathbb{R}^n} f(\underline{\eta}) \big[\underline{\eta} - \underline{x}\big] k_\sigma(\underline{x} - \underline{\eta})\, \mathrm{d}\underline{\eta} \ . \end{aligned} \tag{4}$$

2) Hessian: Similar to the gradient, the Hessian of the GH w.r.t. $\underline{x}$ is given by

$$\begin{aligned} \nabla^2_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma) &= \int_{\mathbb{R}^n} f(\underline{\eta})\, \nabla^2_{\underline{x}} k_\sigma(\underline{x} - \underline{\eta})\ \mathrm{d}\underline{\eta} \\ &= \int_{\mathbb{R}^n} f(\underline{\eta}) \left[\frac{\big[\underline{\eta} - \underline{x}\big]\big[\underline{\eta} - \underline{x}\big]^\top}{\sigma^4} - \frac{\mathbf{I}}{\sigma^2} \right] k_\sigma(\underline{x} - \underline{\eta})\, \mathrm{d}\underline{\eta} \ . \end{aligned} \tag{5}$$

3) Mixed Derivative: In predictor-corrector path tracking, we need to quantify how the gradient $\nabla_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma)$ changes as the homotopy parameter σ varies. This dependence is captured by the mixed derivative $\nabla^2_{\underline{x}\sigma} H_{\mathrm{G}}(\underline{x}, \sigma)$, which is required in the predictor step. It is given by

$$\begin{aligned} \nabla^2_{\underline{x}\sigma} H_{\mathrm{G}}(\underline{x}, \sigma) &= \int_{\mathbb{R}^n} f(\underline{\eta})\, \nabla^2_{\underline{x}\sigma} k_\sigma(\underline{x} - \underline{\eta})\ \mathrm{d}\underline{\eta} \\ &= \int_{\mathbb{R}^n} f(\underline{\eta}) \big[\underline{\eta} - \underline{x}\big] \left[\frac{\|\underline{\eta} - \underline{x}\|^2}{\sigma^5} - \frac{n+2}{\sigma^3} \right] k_\sigma(\underline{x} - \underline{\eta})\, \mathrm{d}\underline{\eta} \ . \end{aligned} \tag{6}$$

C. Standard Gaussian Homotopy Method

Standard GH solves a sequence of smoothed optimization problems with decreasing values of the homotopy parameter σ. The process begins with a sufficiently large σ, for which the objective landscape is strongly smoothed, making it easier to optimize. The minimizer obtained at one smoothing level is then used as the initialization for a local solver at the next level. The procedure is summarized in Alg. 1.

Remark 1: A key practical advantage of GH is its reduced sensitivity to initialization. When σ_0 is sufficiently large, the resulting smoothed function has a simpler landscape, and different choices of $\underline{x}_0$ often converge to the same or nearby minimizers. In practice, selecting a sufficiently large initial smoothing level is therefore more important than carefully tuning the initial point.

Although effective, this sequential re-optimization strategy does not explicitly exploit the local geometry of the homotopy path. As a result, performance depends strongly on the chosen schedule, i.e., $\sigma_0 > \sigma_1 > \cdots > \sigma_{m-1} > 0$, where m is the number of iterations. Schedules with small decreases in σ increase computational cost by requiring many closely spaced optimization steps. However, overly aggressive σ schedules (those that decrease quickly) can cause the solver to lose the path and converge to an incorrect stationary point.

Algorithm 1: Standard GH Method

Input: Cost function $f\colon \mathbb{R}^n \to \mathbb{R}$, sigma schedule $\sigma_0 > \sigma_1 > \cdots > \sigma_{m-1} > 0$, Initial solution $\underline{x}_0$.
for $k = 0$ **to** $m-1$ **do**
  $\underline{x}_k =$ minimizer of $H_{\mathrm{G}}(\underline{x}, \sigma_k)$, initialized at $\underline{x}_{k-1}$.
Output: $\underline{x}_m$

IV. Predictor–Corrector Gaussian Homotopy Method (PCGH)

To address the limitations of standard GH, we propose a PCGH method that explicitly leverages the local geometry of the homotopy path to accelerate convergence and improve stability. The algorithm is summarized in Alg. 2. The key idea is to use the predictor–corrector method widely used in numerical continuation for tracking solution paths of parameterized equations [14], [16], [17]. The predictor step estimates the trajectory of the minimizer as the homotopy parameter σ changes, while the corrector step refines this estimate to ensure it remains on the homotopy path.

A. Predictor Step

To know how the minimizer changes w.r.t. σ, we start from a stationary point $(\underline{x}_0, \sigma_0)$ of the homotopy function, which satisfies

$$\nabla_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma) = \underline{0} \ .$$

Applying implicit differentiation w.r.t. the homotopy parameter σ yields the Davidenko Ordinary Differential Equation (ODE) [18]

$$\nabla^2_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma) \frac{\mathrm{d}\underline{x}}{\mathrm{d}\sigma} + \nabla^2_{\underline{x}\sigma} H_{\mathrm{G}}(\underline{x}, \sigma) = \underline{0} \ , \tag{7}$$

which describes how the roots of an equation evolve w.r.t. changes in σ. Rewriting (7) explicitly yields

$$\frac{\mathrm{d}\underline{x}}{\mathrm{d}\sigma} = -\left[\nabla^2_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma)\right]^{-1} \nabla^2_{\underline{x}\sigma} H_{\mathrm{G}}(\underline{x}, \sigma) = \underline{v}(\underline{x}, \sigma) \ .$$

This ODE, together with the initial condition $\underline{x}(\sigma_0) = \underline{x}_0$, predicts the minimizer at subsequent values of σ along the homotopy path. When the path is parameterized directly by σ, the trajectory can be highly non-uniform, which may lead to overshooting or undershooting in the predictor step. To mitigate this issue, we use arc-length parameterization so that the predictor evolves more uniformly along the path.

B. Arc-Length Parameterization of Homotopy Path

Arc-length parameterization is a common technique in numerical continuation for ensuring uniform progression along a path [14]. Parameterizing the homotopy by arc-length s allows the predictor to evolve at constant speed along the path, improving stability.

We first specify the distance metric used to measure the arc-length. The infinitesimal arc-length is written as

$$(\mathrm{d}s)^2 = [\mathrm{d}\underline{x}^\top, \mathrm{d}\sigma]\, \mathbf{M}(\underline{x}, \sigma)\, [\mathrm{d}\underline{x}^\top, \mathrm{d}\sigma]^\top \ ,$$

where $\mathbf{M}(\underline{x}, \sigma)$ is a positive definite metric matrix that balances the contributions of $\underline{x}$ and σ.

The derivatives of $\underline{x}$ and σ w.r.t. s must then satisfy

$$\left[\frac{\mathrm{d}\underline{x}}{\mathrm{d}s}^\top, \frac{\mathrm{d}\sigma}{\mathrm{d}s}\right] \mathbf{M}(\underline{x}, \sigma) \left[\frac{\mathrm{d}\underline{x}}{\mathrm{d}s}^\top, \frac{\mathrm{d}\sigma}{\mathrm{d}s}\right]^\top = 1 \tag{8}$$

By substituting the explicit Davidenko path trajectory tracking condition $\left(\frac{\mathrm{d}\underline{x}}{\mathrm{d}\sigma} = \underline{v}(\underline{x}, \sigma)\right)$ into (8), we obtain the coupled system of tracking ODEs:

$$\frac{\mathrm{d}\underline{x}}{\mathrm{d}s} = -\frac{\underline{v}(\underline{x}, \sigma)}{\alpha(\underline{x}, \sigma)} \ , \tag{9}$$

$$\frac{\mathrm{d}\sigma}{\mathrm{d}s} = -\frac{1}{\alpha(\underline{x}, \sigma)} \ , \tag{10}$$

Here, the negative sign is explicitly chosen to ensure that the homotopy parameter σ decreases monotonically toward zero along the homotopy path. The trajectory speed is scaled by the path-dependent normalization factor

$$\alpha(\underline{x}, \sigma) = \sqrt{[\underline{v}(\underline{x}, \sigma)^\top, 1]\, \mathbf{M}(\underline{x}, \sigma)\, [\underline{v}(\underline{x}, \sigma)^\top, 1]^\top} \ . \tag{11}$$

Because this factor is entirely determined by the underlying metric tensor $\mathbf{M}(\underline{x}, \sigma)$, we propose two geometry-aware metric formulations that exploit the structural topology of the objective function landscape, rather than relying on a naive Euclidean distance.

1) Fisher Metric: Since this homotopy is constructed from Gaussian smoothing, each point on the path can be associated with a Gaussian distribution. This motivates an information-theoretic distance based on the divergence between these distributions, rather than the Euclidean distance in parameter space.

For this case, the matrix $\mathbf{M}(\underline{x}, \sigma)$ can be defined based on the Fisher information matrix of the isotropic Gaussian distribution as

$$\mathbf{M}(\underline{x}, \sigma) = \begin{bmatrix} \frac{1}{\sigma^2}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \frac{2n}{\sigma^2} \end{bmatrix} \tag{12}$$

Therefore, the specific normalization factor can be derived by substituting (12) into (11), which yields

$$\alpha_{\text{Fisher}}(\underline{x}, \sigma) = \frac{1}{\sigma} \sqrt{\|\underline{v}(\underline{x}, \sigma)\|^2 + 2n} \ .$$

2) Hessian-squared Metric: Alternatively, the matrix $\mathbf{M}(\underline{x}, \sigma)$ can be defined based on the local curvature of the GH, which can be captured by the Hessian matrix of the GH. To ensure positive definiteness, we can use the squared Hessian as the metric in $\underline{x}$ space and Fisher information metric in σ space, i.e.

$$\mathbf{M}(\underline{x}, \sigma) = \begin{bmatrix} \nabla^2_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma)^\top \nabla^2_{\underline{x}} H_{\mathrm{G}}(\underline{x}, \sigma) & \mathbf{0} \\ \mathbf{0} & \frac{2n}{\sigma^2} \end{bmatrix} \tag{13}$$

In comparison with (12), the Hessian-squared metric incorporates local curvature information. High curvature reduces the step size to prevent overshooting, whereas low curvature allows larger steps.

Algorithm 2: Predictor-Corrector Gaussian Homotopy (PCGH)

Input: Cost function f, Initial values $(\underline{x}_{\text{init}}, \sigma_0)$, step size Δs

$\underline{x}_0 = \text{NewtonSolver}(H_{\text{G}}(\cdot, \sigma_0), \underline{x}_{\text{init}})$

for $k = 1$ **to** m **do**

  // Prediction step

  $\underline{x}_k^p, \sigma_k = \text{Solve ODE}(\underline{x}_{k-1}, \sigma_{k-1}, \Delta s)$

  // Correction step

  $\underline{x}_k = \text{NewtonSolver}(H_{\text{G}}(\cdot, \sigma_k), \underline{x}_k^p)$

  // Termination condition

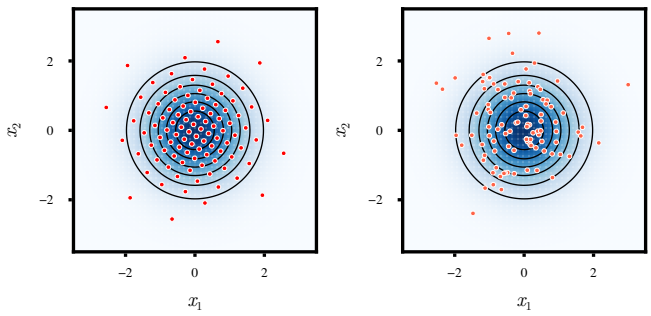
  if $\sigma_k < \sigma_{ter}$ **then**

    $\underline{x}_{\text{res}} = \underline{x}_k$

    break

$\underline{x}_{\text{res}} = \underline{x}_m$

Output: $\underline{x}_{\text{res}}$



(a) Fibonacci samples (b) Pseudo-random samples

Fig. 3: 100 samples from a standard 2D Gaussian: deterministic Fibonacci (left) vs. pseudo-random Monte Carlo (right). Deterministic samples achieve uniform coverage without sample clumps or empty spatial voids.

The normalization factor for the Hessian-squared metric can be derived by substituting (13) into (11), which yields

$$\alpha_{\text{Hessian}}(\underline{x}, \sigma) = \sqrt{\left\| \nabla^2_{\underline{x}\sigma} H_{\text{G}}(\underline{x}, \sigma) \right\|^2 + \frac{2n}{\sigma^2}} \ .$$

In practice, the system of ODEs (9) and (10) can be integrated using any standard ODE solver (e.g., Runge–Kutta methods) with step size Δs and initial conditions to obtain the predicted minimizer at the next value of σ.

C. Corrector Step

The purpose of the corrector step is to refine this predicted minimizer back onto the exact homotopy solution path for the updated homotopy parameter. Provided the step size is chosen appropriately, the predictor equations (9) and (10) yield an initial guess that falls reliably within the local basin of attraction, ensuring rapid convergence. Consequently, we can employ a second-order solver, such as Newton method, which exhibits rapid quadratic convergence in this localized region. Starting from the predicted solution, Newton method iterates

$$\underline{x}_{k+1} = \underline{x}_k - \left[\nabla^2_{\underline{x}} H_{\text{G}}(\underline{x}_k, \sigma)\right]^{-1} \nabla_{\underline{x}} H_{\text{G}}(\underline{x}_k, \sigma) \ ,$$

where the gradient $\nabla_{\underline{x}} H_{\text{G}}$ and Hessian $\nabla^2_{\underline{x}} H_{\text{G}}$ are given by (4) and (5), respectively.

Newton method excels in low dimensions, but computing and inverting the Hessian scales poorly to high-dimensional problems. To ensure scalability, alternative local optimization techniques, such as standard gradient descent or quasi-Newton methods such as BFGS / L-BFGS [19], can be employed instead.

V. Sample-Based Approximation

In this work, the integrals involved in the GH and its derivatives are approximated via sampling-based methods. We explore both (A.) standard Monte Carlo sampling and (B.) deterministic sampling techniques to evaluate these integrals efficiently and accurately.

A. Monte Carlo Approximation

By rewriting (3) to (6) in terms of a standard Gaussian perturbation, the GH and its derivatives can be expressed as expectations over $\hat{\underline{\epsilon}} \sim \mathcal{N}(\underline{0}, \mathbf{I})$. With $\underline{\eta} = \underline{x} + \sigma\hat{\underline{\epsilon}}$, this yields the following single-sample unbiased estimators:

$$\begin{aligned}
\widehat{H}_{\text{G}}(\underline{x}, \sigma) &= f(\underline{x} + \sigma\hat{\underline{\epsilon}}) \ , \\
\widehat{\nabla}_{\underline{x}} H_{\text{G}}(\underline{x}, \sigma) &= \frac{1}{\sigma}\hat{\underline{\epsilon}} f(\underline{x} + \sigma\hat{\underline{\epsilon}}) \ , \\
\widehat{\nabla}^2_{\underline{x}} H_{\text{G}}(\underline{x}, \sigma) &= \frac{1}{\sigma^2}\left(\hat{\underline{\epsilon}}\hat{\underline{\epsilon}}^\top - \mathbf{I}\right) f(\underline{x} + \sigma\hat{\underline{\epsilon}}) \ , \\
\widehat{\nabla}^2_{\underline{x}\sigma} H_{\text{G}}(\underline{x}, \sigma) &= \frac{1}{\sigma^2}\hat{\underline{\epsilon}}\left(\|\hat{\underline{\epsilon}}\|^2 - (n+2)\right) f(\underline{x} + \sigma\hat{\underline{\epsilon}}) \ .
\end{aligned}$$

The corresponding multi-sample estimators are obtained by averaging these expressions over N independent and identically distributed (i.i.d.) samples $\hat{\underline{\epsilon}}$.

A key challenge is that the factors $1/\sigma$ and $1/\sigma^2$ amplify estimator variance as $\sigma \to 0$. In particular, the variance scales as $\mathcal{O}(1/\sigma^2 N)$ for the gradient and $\mathcal{O}(1/\sigma^4 N)$ for both the Hessian and mixed derivative. To mitigate this effect, we apply a simple control-variate technique by subtracting the baseline value $f(\underline{x})$ inside the derivative estimators. This preserves unbiasedness because the added terms have zero expectation, while significantly reducing variance when σ is small. For a more comprehensive discussion on control variates and alternative variance reduction techniques, we refer the reader to [20].

B. Deterministic Sampling

The standard Monte Carlo method for approximating the expectations converges at a rate of $\mathcal{O}(1/\sqrt{N})$ that is independent of the dimensionality of the problem [21]. However, i.i.d. random sampling naturally leads to samples clumping together in some regions of the space while leaving other regions sparsely covered, which can result in high variance and poor estimation quality. To achieve lower variance at a fixed sample budget, pseudo-random Monte Carlo can be replaced by deterministic low-discrepancy sampling. Specifically, we

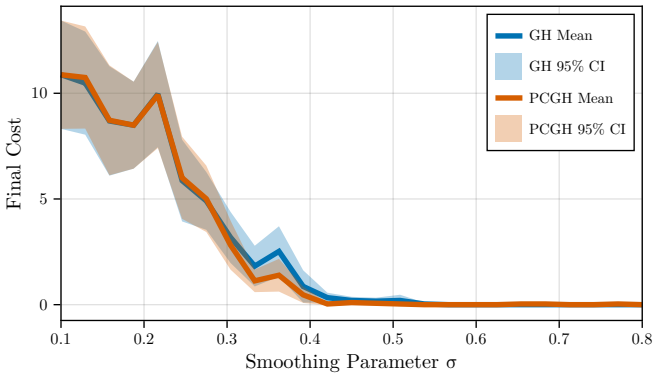


Fig. 4: Initial sigma sweep on the 2D Rastrigin function. The curves show the mean final cost and 95% confidence interval over 20 trials for standard GH and PCGH. Both methods are robust once σ_0 is sufficiently large, while too-small values of σ_0 fail to smooth out the local non-convex structure.

employ generalized Fibonacci grids [12], which offer better space-filling properties and nearly uniform neighbor spacing. By mapping this deterministic grid to the target Gaussian density, we obtain significantly more stable approximations of the Gaussian convolution and its derivatives compared to standard pseudo-random sampling.

Fig. 3 shows a visual comparison of 100 samples drawn from a standard 2D Gaussian distribution using deterministic Fibonacci sampling and pseudo-random Monte Carlo sampling. The deterministic Fibonacci samples exhibit a more uniform coverage of the space, avoiding the clumping and voids that are characteristic of random sampling.

VI. Experiment

We evaluate the proposed PCGH method in two experiments. The first studies sensitivity to the initial smoothing parameter σ_0, and the second compares convergence against standard Gaussian continuation on a suite of benchmark functions.

A. Initial Sigma Sweep

We first study how the initial smoothing parameter σ_0 affects performance on the 2D *Rastrigin* function. This experiment compares standard GH and PCGH using the Fisher metric. For each value of σ_0, we run 20 trials with random initial points and report the mean final cost with a $95\,\%$ confidence interval. Both solvers use random samples to approximate the Gaussian homotopy and its derivatives.

The results are shown in Fig. 4. Both methods exhibit stable behavior once σ_0 is sufficiently large, while performance deteriorates rapidly when σ_0 becomes too small. On this benchmark, the transition occurs at around $\sigma_0 \approx 0.5$, indicating that a sufficiently large initial smoothing level suppresses local non-convex structures and makes the final solutions of both methods less sensitive to initialization. However, this empirical stability does not guarantee global optimality: the stationary path induced by Gaussian homotopy does not necessarily connect the initial smoothed solution to a global minimizer of the original objective.

B. Comparison of Solver Convergence

We next compare convergence behavior on four standard 2D benchmark functions: *Rastrigin*, *Rosenbrock*, *Ackley*, and *Beale*. The evaluated solvers are:

- **GC**: standard Gaussian homotopy,
- **PCGH (Fisher)**: predictor–corrector Gaussian homotopy with the Fisher metric, and
- **PCGH (Hessian)**: predictor–corrector Gaussian homotopy with the Hessian-squared metric.

For each solver, we report one deterministic run using Fibonacci sampling and 20 stochastic runs using random sampling. The stochastic curves show the mean and $95\,\%$ confidence interval, and all plots report the objective value in log scale against the number of function evaluations.

The convergence results are summarized in Fig. 5. Across the four benchmarks, both PCGH variants generally reach low-cost regions with fewer function evaluations than standard Gaussian continuation, while the deterministic and stochastic traces show the same overall trend. This improvement is especially visible on strongly non-convex landscapes, where predictor–corrector path tracking makes more efficient progress along the homotopy path than the fixed schedule used by the standard method.

C. Discussion

The experiments highlight three main observations. First, the initial sigma sweep confirms that homotopy-based optimization is robust to initialization once the initial smoothing parameter is sufficiently large. On the Rastrigin benchmark, both standard GH and PCGH remain stable above a clear threshold in σ_0, whereas smaller values fail to suppress the local non-convex structure. Second, the convergence comparison on four benchmark functions shows that the predictor–corrector variants generally reach low-cost regions with fewer function evaluations than standard Gaussian homotopy, which is advantageous for problems in which function evaluations are expensive, such as model predictive control with complex dynamics. Third, as evidenced by the simulations, deterministic Fibonacci sampling enhances the performance of both solvers by providing a more stable and accurate approximation of the derivatives.

VII. Conclusion

This paper presents a predictor–corrector Gaussian homotopy framework for multimodal non-convex optimization. By deriving the Davidenko ODE for GH and coupling it with arc-length parameterization, the proposed PCGH method explicitly exploits the local geometry of the homotopy path instead of relying on a fixed smoothing schedule. In addition, deterministic Fibonacci sampling provides a practical way to reduce approximation variance in the Gaussian convolution and its derivatives.

While this study focuses on low-dimensional benchmarks, future work will first evaluate PCGH in high-dimensional spaces, directly addressing the critical cost of Hessian-based correctors. We will subsequently explore anisotropic Gaussian kernels for landscape-aware smoothing, as well as alternative

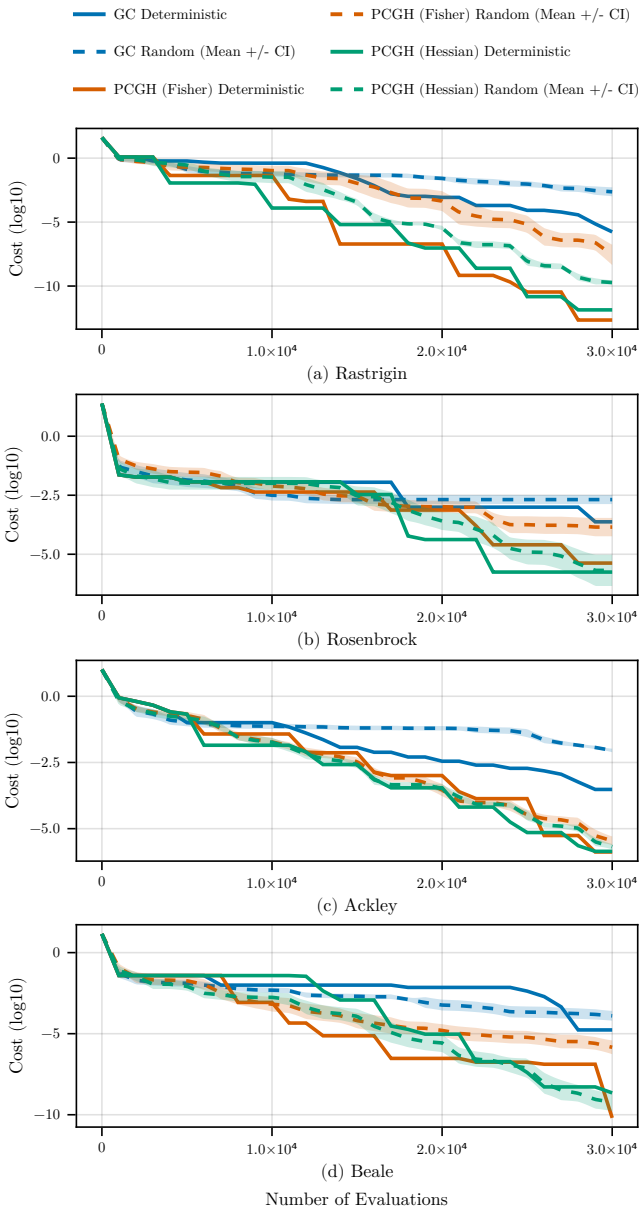


Fig. 5: Solver convergence comparison on four 2D benchmark functions. Solid lines denote deterministic runs with Fibonacci sampling, while dashed lines denote the mean stochastic performance with 95 % confidence intervals over 20 trials. The PCGH variants generally converge quicker than standard Gaussian continuation.

geometric path metrics. Overall, the results suggest that predictor–corrector path tracking is a promising direction for making Gaussian homotopy optimization more efficient and more practically reliable.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert, "Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 4597–4604.

[3] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard, "SE-sync: A certifiably correct algorithm for synchronization over the special Euclidean group," *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 95–125, 2019.

[4] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[5] T. Bäck and H.-P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evolutionary Computation*, vol. 1, no. 1, pp. 1–23, 1993.

[6] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, vol. 4, 1995, pp. 1942–1948.

[7] C. G. Broyden, "A new method of solving nonlinear simultaneous equations," *The Computer Journal*, vol. 12, no. 1, pp. 94–99, Feb. 1969.

[8] L. T. Watson and R. T. Haftka, "Modern homotopy methods in optimization," *Computer Methods in Applied Mechanics and Engineering*, vol. 74, no. 3, pp. 289–305, Sep. 1989.

[9] D. M. Dunlavy and D. P. O'Leary, "Homotopy optimization methods for global optimization." Sandia National Laboratories, Tech. Rep., 12 2005. [Online]. Available: https://www.osti.gov/biblio/876373

[10] H. Mobahi and J. W. Fisher, "On the link between Gaussian homotopy continuation and convex envelopes," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Cham: Springer International Publishing, 2015, vol. 8932, pp. 43–56.

[11] H. Iwakiri, Y. Wang, S. Ito, and A. Takeda, "Single loop Gaussian homotopy method for non-convex optimization," Nov. 2022, arXiv:2203.05717.

[12] D. Frisch and U. D. Hanebeck, "Deterministic Gaussian sampling with generalized Fibonacci grids," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, 2021, pp. 1–8.

[13] U. D. Hanebeck, M. F. Huber, and V. Klumpp, "Dirac mixture approximation of multivariate Gaussian densities," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 2009, pp. 3851–3858.

[14] E. L. Allgower and K. Georg, *Numerical Continuation Methods*, ser. Springer Series in Computational Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg, 1990, vol. 13.

[15] H. Mobahi and J. Fisher III, "A theoretical analysis of optimization by Gaussian continuation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, Feb. 2015.

[16] C. B. Garcia and W. I. Zangwill, *Pathways to Solutions, Fixed Points, and Equilibria*, ser. Prentice-Hall Series in Computational Mathematics. Prentice-Hall, 1981.

[17] H. B. Keller, "Numerical solution of bifurcation and nonlinear eigenvalue problems," in *Applications of Bifurcation Theory*, P. H. Rabinowitz, Ed. Academic Press, 1977, pp. 359–384.

[18] D. F. Davidenko, "On a new method of numerical solution of systems of nonlinear equations," *Doklady Akademii Nauk SSSR*, vol. 88, no. 4, pp. 601–602, 1953, in Russian.

[19] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York: Springer, 2006.

[20] S. Mohamed, M. Rosca, M. Figurnov, and A. Mnih, "Monte Carlo Gradient Estimation in Machine Learning," Sep. 2020, arXiv:1906.10652.

[21] R. E. Caflisch, "Monte Carlo and quasi-Monte Carlo methods," *Acta Numerica*, vol. 7, p. 1–49, 1998.