New High-order Methods using Gaussian Processes for Computational Fluid Dynamics Simulations

Dongwook Lee¹, Adam Reyes², Carlo Graziani³, Petros Tzeferacos³,⁴

¹ Applied Mathematics and Statistics, University of California, Santa Cruz, CA, U.S.A.
² Department of Physics, University of California, Santa Cruz, CA, U.S.A.
³ Flash Center for Computational Science, Department of Astronomy & Astrophysics, University of Chicago, IL, U.S.A.
⁴ Physics, Oxford University, U.K.

Abstract. We present an entirely new class of high-order numerical algorithms for computational fluid dynamics simulations. The new method is based on the Gaussian Processes (GP) modeling that generalizes the Gaussian probability distribution. Our approach is to adapt the idea of the GP prediction technique that utilizes the covariance kernel functions, and use the GP prediction to interpolate/reconstruct a high-order approximations for solving hyperbolic PDEs. We propose the GP high-order method as a new class of numerical high-order formulations, alternative to the conventional polynomial-based approaches.

1. Introduction

Cutting edge simulations of gas dynamics and magnetohydrodynamics (MHD) have been among the headliner applications of scientific high-performance computing (HPC) [1, 2, 3, 4]. One notable HPC development is that the newer architectures are expected to feature a dramatic change in the current balance between computation and memory resources, with memory per compute core declining drastically from current levels [2, 3, 5]. This trend tells us that there need to be algorithmic strategies to save memory at the cost of increased computation. Recently, this paradigm shift has become a high priority for HPC applications. In the context of numerical methods for computational fluid dynamics, an important approach to accomplish such strategies is the design of very high-order of accuracy methods [3]. As compared to low-order methods, high-order numerical methods achieve an increased target solution accuracy by computing increased higher-order floating-point approximations on a given grid resolution [6, 7, 8], thus embodying, in a concrete manner, the desired tradeoff between memory and computation by exercising more computation per memory (or equivalently, the equal amount of computation with less memory).

Within the broad framework of FD and FV discretization methods, discrete algorithms of data interpolation and reconstruction play a key role in numerical methods for PDE integration [6, 7, 9]. They are frequently a limiting factor in a numerical scheme’s overall convergence rate, efficiency, and algorithmic complexity. Interpolation and reconstruction are not only essential for estimating high-order accurate approximations of discrete solutions; but also of interface tracking; of prolonging states from coarse zones to corresponding refined zones in adaptive-mesh refinement (AMR) schemes; and of various other contexts associated with high-order solutions as well. In CFD simulations, these interpolation and reconstruction algorithms therefore must be carried out as accurately as possible, because, by and large, their accuracy is one of the key
main factors that determines the overall accuracy of each simulation.

Polynomial based approach has been most successful and popular among interpolation/reconstruction methods in this field. It is easy to understand why this should be the case. In the first place, it is easily relatable to Taylor expansion, the most familiar of function approximations. Secondly, it is well-studied that the nominal $N$-th order accuracy of polynomial interpolation/reconstruction is derived from using polynomials of degree $N - 1$, bearing a leading term of the error that scales with $O(\Delta^N)$ as the local grid spacing $\Delta$ approaches to zero [6, 7, 9]. However, the simplicity of polynomial interpolation/reconstruction has costs. In the first instance, the polynomial approach is notoriously prone to oscillations in data fitting, especially with discontinuous data [16]. Further, in many practical situations the high-order interpolation/reconstruction is carried out on a series of 1D stencils in each of the normal directions. In this sense, such 1D based polynomial interpolation/reconstruction works at the cost of overlooking available information from the neighborhood in transverse directions, which could otherwise be used to improve the accuracy and stability of the interpolation/reconstruction. Moreover, the algorithmic complexity of such schemes typically grows with order of accuracy [17]. Genuinely multidimensional polynomial based schemes can be constructed at the cost of greater computational effort and with more restrictions on mesh geometry [?].

We present a new high-order approach that readily overcomes the aforementioned issues in the polynomial approaches by developing the broader outlook afforded by the GP modeling perspective. Our new GP method is a class of high-order schemes designed for numerical evolution of hyperbolic PDEs, $u_t + \nabla \cdot F(u) = 0$. In this paper we describe new high-order GP approximation strategies in two steps focusing on one-dimensional formulation: firstly, Gaussian Process (GP) interpolation that works on pointwise values of $u(x_i)$, and secondly, GP reconstruction for finite volume methods (FVM) where volume-averaged values, $\bar{u}_i = \frac{1}{\Delta V_i} \int_{\Delta V_i} u(x) dV$, are considered. The first will provide a baseline formulation of using GP as a new high-order interpolator, while the latter will serve as a high-order reconstructor by applying the volume averaged integration to the mathematical components derived from the first. The second approach will provide a scheme that is better suited for compressible fluid flows including shocks and discontinuities.

The theory of Gaussian Processes, and more generally of stochastic functions, goes back to the work of Wiener [18] and Kolmogorov [19]. Modern-day applications are numerous. Just in the physical sciences, GP prediction is in common use in meteorology, geology, and time-series analysis [20, 21], and in cosmology, where GP models furnish the standard description of the Cosmic Microwave Background [22]. Applications abound in many other fields, in particular wherever spatial or time-series data (or data otherwise scattered on some manifold) requires “nonparametric” modeling [21, 23]. Our goal in this study is the use of the ideas of predictive GP modeling that is processed by training observed data, e.g., cell-centered initial conditions of fluid variables. In what follows, we will stick to a relatively informal level of technical detail for the sake of brevity. Readers who wish to pursue the subject in greater detail are referred to [21, 23].

2. Gaussian Process Modelings

Gaussian Processes are a class of stochastic processes, that is, processes that yield sampling data (or functions) from an infinite dimensional function space. The function space is probabilistically constrained by specifying a prior probability to every function in the function space, where the functions are not exactly known. Random functions (or samples) drawn from of the prior are regarded as data, and the model is “trained or learned” on the data by means of Bayes’ theorem, producing an updated probabilistic constraint on the unknown functions. In this way GP produces a data-informed posterior model (an updated posterior mean function and a posterior covariance kernel function) from an “agnostic” prior model (a prior mean and a covariance kernel functions). The updated posterior mean function from the posterior is a data-
informed mean that is to be used for a data prediction in a GP sense. This GP prediction, by means of the updated mean function, is our target interpolation/reconstruction for FDM and FVM.

A GP is fully defined by two functions: (i) a mean function $\bar{f}(x) = \mathbb{E}[f(x)]$ over $\mathbb{R}^N$, and (ii) a covariance function which is a symmetric, positive-definite integral kernel $K(x,y)$ over $\mathbb{R}^N \times \mathbb{R}^N$. For instance, a function $f(x)$ represents a mathematical object describing a fluid density $\rho$. Such functions $f$ drawn randomly from this distribution are said to be sampled from a Gaussian Process with mean function $\bar{f}(x)$ and covariance function $K(x,y)$, and we write $f \sim \mathcal{GP}(\bar{f}, K)$. As with the case of finite-dimensional Gaussian distributions, the significance of the covariance is

$$K(x,y) = \mathbb{E}[(f(x) - \bar{f}(x))(f(y) - \bar{f}(y))],$$

where the averaging is over the GP distribution.

In standard statistical modeling practice, both $\bar{f}(x)$ and $K(x,y)$ are typically parametrized functions, with parameters controlling the character (e.g., scale lengths, differentiability, oscillation strength) of “likely” functions. Given a GP, and given $n$ “training” points $x_i$, $i = 1, \ldots, n$ at which the function values $f(x_i)$ are known, we may calculate the likelihood $L$ (the probability of $f$ given the GP model) of the data vector $f \equiv [f(x_1), \ldots, f(x_n)]^T$ (e.g., $n$ many point-wise values of density $\rho$ at $x_i$, $i = 1, \ldots, n$) by

$$L \equiv P(f) = (2\pi)^{-N/2} \det |K|^{-1/2} \exp \left[-\frac{1}{2} (f - \bar{f})^T K^{-1} (f - \bar{f}) \right],$$

where $K = [K_{ij}]_{i,j=1,\ldots,n}$ with $K_{ij} \equiv K(x_i,x_j)$.

Next, given the function samples $f \equiv [f(x_1), \ldots, f(x_n)]^T$ obtained at spatial points $x_i$, $i = 1, \ldots, n$, we wish to make probabilistic statements (or GP predictions) about the value $f_s \equiv f(x_s)$ of the unknown function $f \sim \mathcal{GP}(\bar{f}, K)$ at a new spatial point $x_s$. In other words, we are interested in making a new prediction of GP for $f$ at any randomly chosen point $x_s$. This is particularly of interest to us from the perspectives of FDM and FVM, because in this way, we can use GP to predict a function value (e.g., density flux) at cell interfaces $x_s \equiv x_{i+1/2}$, at which both FDM and FVM seek for the best possible high-order approximations of flux functions in designing conservative discrete hyperbolic PDEs. This can be done in GP as follow. We look at the augmented likelihood function $L_s$ by considering the joint distribution of the currently available training outputs, $f$, and the new test output $f_s$,

$$L_s \equiv P(f,f_s) = (2\pi)^{-(n+1)/2} \det |M|^{-1/2} \exp \left[-\frac{1}{2} (g - \bar{g})^T M^{-1} (g - \bar{g}) \right],$$

where $g$ and $\bar{g}$ are the $(n+1)$-dimensional vectors whose components, in partitioned form, are

$$g^T \equiv [f_s, f], \quad \bar{g}^T \equiv [\bar{f}(x_s), \bar{f}],$$

and $M$ is the $(n+1) \times (n+1)$ augmented covariance matrix, given in partitioned form by

$$M = \begin{pmatrix} K_{ss} & k_s^T \\ k_s & K \end{pmatrix}.$$  

In Eq. (5), we’ve defined a scalar $k_{ss}$ and an $n$-dimensional vector $k_s = [k_{s,i}]_{i=1,\ldots,n}$ given by

$$k_{ss} \equiv K(x_s,x_s), \quad k_{s,i} \equiv K(x_s,x_i).$$

Using Bayes’ Theorem, the conditioning property applied to the joint Gaussian prior distribution on the observation $f$ yields the Gaussian posterior distribution of $f_s$ given $f$. As a result, the main outcome of the GP prediction [21] can be derived:
interpolation is then to operate on based polynomial interpolations, in that, the GP approach does not incur a great cost in multidimensional stencil is a qualitative advantage of the GP interpolation over the usual 1D-stencil configuration. This can be easily seen in the arguments of the covariance kernels. This point
approximation, say at etc. For an exposition purpose, let us use volume averaged data. We will see that an algorithmic design for GP interpolation of pointwise data based on the volume averaged data, we consider first an interpolation method based on the although in this paper we are more interested in developing a high-order reconstruction method — the weight vector may be computed and stored in advance, as an initialization, and can remain constant throughout the simulation. In case that an adaptive mesh refinement (AMR) configuration is under consideration, w may be computed at all possible grid refinement levels and stored a priori and used later. Another more efficient way is, a new set of weight vectors w can be newly computed only once and stored whenever there is a new change in grid resolution on any local AMR region during simulations. The GP interpolations then come at the cost of an inner product of a vector whose size is the number of points in the stencil of the interpolation \([x_{i-p}, x_{i+q}].\) Typically, the sizes of the stencil are three for 2nd order piecewise linear methods [24, 9]: five for both the 3rd order piecewise parabolic [25] and 5th order WENO methods [26]. This fact assures that the sizes of the linear system in Eq. (8) are not large at all, e.g., \(K\) is going to be a 5 \(\times\) 5 matrix when using the stencil of five.

An important feature of the GP interpolation is that it naturally supports multidimensional stencil configuration. This can be easily seen in the arguments of the covariance kernels. This multidimensional stencil is a qualitative advantage of the GP interpolation over the usual 1D-based polynomial interpolations, in that, the GP approach does not incur a great cost in

\[
P(f_*|\mathbf{f}) = (2\pi U^2)^{-1/2} \exp \left[ -\frac{1}{2} \frac{(f_* - \bar{f}_* )^2}{U^2} \right],
\]

where the newly updated posterior mean function

\[
f_* \equiv \bar{f}(\mathbf{x}_* + k_*^T \cdot \mathbf{K}^{-1} (\mathbf{f} - \bar{f})),
\]

and the newly updated posterior covariance kernel function

\[
U^2 \equiv k_{ss} - k_*^T \cdot \mathbf{K}^{-1} \cdot k_*.
\]

We emphasize here that, in Eq. (8), the GP on the unknown function \(f\) has resulted in a Gaussian probability distribution on the unknown function value \(f_*\), providing a probabilistic mean prediction of \(f_*\) at a new desired location \(\mathbf{x}_*\) as given in Eq. (8), whose uncertainty is given by the posterior covariance in Eq. (9).

2.1. GP Interpolation

Although in this paper we are more interested in developing a high-order reconstruction method based on the volume averaged data, we consider first an interpolation method based on the pointwise data. We will see that an algorithmic design for GP interpolation of pointwise data will provide a good mathematical foundation for FVM which reconstructs pointwise values from volume averaged data.

The mean \(f_*\) of the distribution given in Eq. (8) is our interpolation of the function \(f\) at the point \(\mathbf{x}_*\), where \(f\) is any given fluid variable such as density, pressure, velocity, flux functions, etc. For an exposition purpose, let us use \(u\) to denote a fluid variable (e.g., density). The GP interpolation is then to operate on \(f = u\), predicting an output value \(\bar{u}\) with a highly accurate approximation, say at \(x_{i+\frac{1}{2}}\), \(\{\bar{u}_{i+\frac{1}{2}}\} = \mathcal{I}_{GP}(u_{i-p}, \ldots, u_{i+q}),\) where \(\mathcal{I}_{GP}(\cdot)\) is the GP interpolation given in Eq. (8). As shown in Eq. (8), the interpolant \(f_*\) is a simple linear combination of the observed data \(\mathbf{f}\) and the covariance kernels \(k_* , K\), anchored by one of its arguments to one of the data points, \(\mathbf{x}_*, \mathbf{x}_1, \ldots, \mathbf{x}_n\). The term added to the mean function \(\bar{f}(\mathbf{x}_*)\) in Eq. (8) may also be cast as an inner product between a vector of weights \(\mathbf{w} \equiv \mathbf{K}^{-1} \cdot k_*\) and a vector of data residuals \((\mathbf{f} - \bar{f})\). The noteworthy fact about this is that the weights \(\mathbf{w}\) are absolutely independent of the data values \(\mathbf{f}\); and they depend only on the \(n\) locations of the data points \(\mathbf{x}_i\) and on the desired interpolation point \(\mathbf{x}_*\). This means that, for an interpolation scheme in which the training point locations (viz. the stencil) and the interpolation point (viz. a cell interface) are known in advance — which is often the case as initial grid configurations are set in static grid setups — the weight vector \(\mathbf{w}\) may be computed and stored in advance, as an initialization, and can remain constant throughout the simulation. In case that an adaptive mesh refinement (AMR) configuration is under consideration, \(\mathbf{w}\) may be computed at all possible grid refinement levels and stored a priori and used later. Another more efficient way is, a new set of weight vectors \(\mathbf{w}\) can be newly computed only once and stored whenever there is a new change in grid resolution on any local AMR region during simulations. The GP interpolations then come at the cost of an inner product of a vector whose size is the number of points in the stencil of the interpolation \([x_{i-p}, x_{i+q}].\) Typically, the sizes of the stencil are three for 2nd order piecewise linear methods [24, 9]: five for both the 3rd order piecewise parabolic [25] and 5th order WENO methods [26]. This fact assures that the sizes of the linear system in Eq. (8) are not large at all, e.g., \(K\) is going to be a 5 \(\times\) 5 matrix when using the stencil of five.
algorithmic design due to the inherent complexity that arises in multidimensional polynomial
approaches especially when designing high-order algorithms [27, 13, 15]. The inversion of \( K - \) whose size is not big – is also computationally efficient with Cholesky decomposition because the
matrix is symmetric positive definite, about a factor two faster than the usual LU decomposition
for inversion.

There is an additional piece of information beyond the point estimate \( f_s - \) we also have
an uncertainty in the estimate, given by \( U \) in Eq. (9). This posterior uncertainty is of crucial
importance in many GP modeling applications, but it is of limited interest for the purpose of this
paper. We will overlook posterior uncertainty in the current study, and focus on the posterior
mean formula given in Eq. (8).

### 2.2. GP Reconstruction

In FVM the fluid variables to be evolved are not point-wise values. Rather, they are volume-
averaged quantities, \( \bar{u}_i = \frac{1}{\Delta V} \int_{\Delta V} u(x) dV \). The GP interpolation strategy discussed so far
should therefore be modified in order to fit well with the data type in FVM. Note that the
integral averages over cell (and face) constitute linear operations on a function \( f(x) \). As
with ordinary finite-dimensional multivariate Gaussian distributions, where linear operations on
Gaussian random variables result in new Gaussian random variables with linearly transformed
means and covariances, a set of \( m \) linear functionals operating on a GP-distributed function \( f \)
has an \( m \)-dimensional Gaussian distribution with mean and covariance that are linear functionals
of the GP mean function and covariance function.

Suppose in a FVM sense, for example, that we have \( m \) measures \( dg_i(x), i = 1, \ldots, m \), defining
\( m \) linear functionals,

\[
G_i = \int dg_i(x) f(x), \quad i = 1, \ldots, m. \tag{10}
\]

Then, the vector \( G = [G_1, \ldots, G_m]^T \) is normally distributed with mean \( \bar{G} = [\bar{G}_1, \ldots, \bar{G}_m]^T \) and
covariance matrix \( C = [C_{i,j}]_{i,j=1,\ldots,m} \), where

\[
\bar{G}_i = \mathbb{E}[G_i] = \int dg_i(x) \mathbb{E}[f(x)] = \int dg_i(x) \bar{f}(x), \quad i = 1, \ldots, m.
\tag{11}
\]

and

\[
C_{i,j} = \mathbb{E}[(G_i - \bar{G}_i)(G_j - \bar{G}_j)] = \int dg_i(x) dg_j(y) \mathbb{E}[(f(x) - \bar{f}(x))(f(y) - \bar{f}(y))] = \int dg_i(x) dg_j(y) K(x,y).
\tag{12}
\]

Thus, we see that the GP distribution on the function \( f \) leads to a multivariate Gaussian
distribution on any \( m \)-dimensional vector \( G \) of linear functionals of \( f \). This Gaussian distribution
can be used for likelihood maximization in a manner completely analogous to the more standard
case where training is performed using point-wise data in the case of FDM, rather than averages
in FVM.

For reconstruction, it is also not hard to generalize Eq. (8) – define the \( m \)-dimensional
**prediction vector** \( T_s = [T_{s,i}]_{i=1,\ldots,m} \) at \( x_s \) by

\[
T_{s,i} = \int dg_i(x) K(x, x_s). \tag{13}
\]

Then the point-wise function value \( \tilde{f}_s \) at the point \( x_s \), reconstructed from the volume-averaged
data \( \bar{G} \), is given by

\[
\tilde{f}_s = \bar{f}(x_s) + T_{s}^T C^{-1}(\bar{G} - \bar{G}). \tag{14}
\]
a straightforward generalization of Eq. (8).

For the sake of a reconstruction scheme on a mesh of control volumes, it is clear that the measures \( d_g(x) \) should be chosen as the cell volume-average measures,

\[
d_g(x) = \begin{cases} 
  d^N x / \Delta^N, & [x \in I_i], \\
  0, & [x \notin I_i], 
\end{cases} 
\]

(15)

where \( \Delta \) is the grid spacing, and \( I_i \) is the \( i \)-th cell, \( I_i = \prod_{d=1}^N I_i^{(d)} \) with 1D cell interval \( I_i^{(d)} = [x_i^{(d)} - \Delta^{(d)}/2, x_i^{(d)} + \Delta^{(d)}/2] \) for each \( d \) direction (note \( N = 3 \) for 3D). For the purpose of the current discussion, we will assume a locally-uniform rectilinear grid of cubical cells \( I_i \) of uniform size \( \Delta \) = \( \Delta^{(1)} = \Delta^{(2)} = \Delta^{(3)} \).

2.3. GP Implementation with Squared Exponential Kernel

To illustrate a single interpolation/reconstruction procedure we consider a stencil \( S_R \) of radius \( R \) points centered on the cell \( I_i \) containing \( 2R + 1 \) points:

\[
S_R = \{ I_{i-R}, \ldots, I_i, \ldots, I_{i+R} \} 
\]

(16)

The fluid variables on this stencil then act as the input data \( (G) \) to the GP model in Eq. (14). In the current study, we adopt a GP model consisting of an isotropic “Squared Exponential (SE)” covariance [21],

\[
K(x, y) = \Sigma^2 \exp\left[ -\frac{(x - y)^2}{2\ell^2} \right], 
\]

(17)

and a constant mean function, \( \bar{f}(x) = f_0 u \), where \( u = [1, \ldots, 1]^T \). The model features three free parameters, \( f_0 \), \( \Sigma^2 \), and \( \ell \). The later two are called “hyperparameters” which are the parameters that are built into the kernel. The parameter \( f_0 \) is determined from the data as part of the interpolation process, by maximizing the likelihood function, Eq. (2). This can easily be done analytically, and it turns out that \( f_0 \) is a weighted mean of the function data values,

\[
f_0 = \begin{cases} 
  u^T C^{-1} f, & \text{for interpolation,} \\
  u^T C^{-1} G, & \text{for reconstruction.} 
\end{cases} 
\]

(18)

The derivation is omitted here for the sake of brevity. The hyperparameter \( \Sigma^2 \) has no effect on the posterior mean function, hence we set \( \Sigma^2 = 1 \) for simplicity. The hyperparameter \( \ell \) is the correlation scale length of the model. It determines the length scale of variation preferred by the GP model. Our GP predictions of interpolation/reconstruction, which necessarily agree with the observed values of the function at the training points \( x_i \), may wiggle on this scale between training points. In this sense, \( \ell \) is a “rigidity”, controlling the curvature scales of the prediction. Since we want function interpolations/reconstructions that are smooth on the scale of the grid, we certainly want \( \ell > \Delta \), and would also prefer \( \ell \geq R \). The optimal values of \( \ell \) should be determined by experience with the algorithm, and can further be re-estimated by considering relevant physical length scales of the data [28].

This covariance has the property of having a native space of \( C^\infty \) functions, and one of the practical advantages of the SE covariance model stems from its dimensional factorization. Therefore, the volume averages in Eqs. (11) and (12) simplify to iterated integrals, and in fact they can be expressed analytically in terms of a pre-computed list of error functions of arguments proportional to one-dimensional cell center differences. Additionally the SE kernel only depends on the spatial distance between data points, belonging to the class of isotropic kernels. This
means that from the perspective of the underlying interpolating/reconstructing functions there is no additional complexity or restrictions to using genuinely multidimensional stencils. Extension of the full GP method for finite volume schemes remains a topic of future work.

Other choices of kernel functions are also available, such as the Matérn, rational quadratic and Wendland classes of covariances [23, 29]. Such functions contain additional smoothness hyperparameters that fix the smoothness of the assumed GP model. We observe that use of suitably unsmooth kernel functions can negatively impact the error convergence and overall accuracy of the GP method. In addition such choices of kernels would require numerical approximations for Eqs. (12) and (13), rather than an analytical form that is afforded in the SE case, which can further negatively affect the accuracy. Further, in the limiting case of large smoothness these functions approach the SE kernel function.

3. Numerical Results

3.1. 1D Smooth Advection

We present numerical results using GP reconstruction applied to the 1D compressible Euler equations, using a fourth-order Runge-Kutta (RK4) method for temporal advancement. The test considered here involves the passive advection of a Gaussian density profile. We initialize a computational box on [0,1] with periodic boundary conditions. The initial density profile is defined by \( \rho(x) = 1 + e^{-100(x-x_0)^2} \), with \( x_0 = 0.5 \), with constant velocity, \( u = 1 \), and pressure, \( P = 1/\gamma \). The specific heat ratio is chosen to be \( \gamma = 5/3 \). The resulting profile is propagated for one period through the boundaries. At \( t = 1 \), the profile returns to its initial position at \( x = x_0 \).

Any deformation of the initial profile is due to either phase errors or numerical diffusion. We perform this test using a length hyperparameter of \( \ell = 0.1 \) for stencil radii \( R = 1, 2, 3, 4 \) and 5, with a fixed Courant number, \( C_{\text{crit}} = 0.8 \) and vary the resolution of computational box, with \( N = 32, 64, 128, 256 \) and 512.

The results of this study are shown in Fig. 1(a). From these numerical experiments, GP reconstruction shows convergence that goes as the size of the stencil, \( 2R + 1 \). The error plateaus at an \( L_1 \) error \( \sim 10^{-12} \), a few orders of magnitude greater than double-precision. This happens because at high resolution the length hyperparameter, \( \ell \), becomes very large relative to the grid spacing, \( \Delta \). The covariance matrix, \( \mathbf{C} \) given in Eq. (12) becomes nearly singular in the regime \( \ell/\Delta \gg 1 \), yielding very large condition numbers for \( \mathbf{C} \). We find the plateau in the \( L_1 \) error occurs for condition numbers, \( \kappa \sim 10^{18} \), corresponding to the point where the errors in inverting \( \mathbf{C} \) in Eq. (14) begin to dominate.

The choice of floating point precision then has an immense impact on the possible \( \ell/\Delta \) so that the condition number errors do not dominate. To produce the results in Fig. 1 we use quadruple-precision for the calculation of \( \mathbf{z}^T = \mathbf{T}_f^T \mathbf{C}^{-1} \) in Eq. (13). Since \( \mathbf{z}^T \) needs to be calculated only once, before starting the simulation, it can then be truncated to double-precision for use in the actual reconstruction procedure. It should be noted that this is necessary for the purposes of a grid convergence study, which is a very unusual situation from the statistical point of view. In a practical simulation it would be an overkill to do so, because in this case the GP algorithm tries to over resolve a feature with the characteristic length \( \ell \) on a relatively very small grid spacing \( \Delta \), yielding \( \ell/\Delta \gg 1 \).

The length hyperparameter, \( \ell \), provides an additional avenue to tune solution accuracy that is not present in polynomial based methods. Fig. 1(b) shows how the errors in the smooth-advection problem changes with the choice of \( \ell \), compared with the error from a fifth-order WENO5 + RK4 solution (denoted in dotted lines) [10]. At large \( \ell \) the errors become roughly the same as in WENO5, and at small values becomes worse. The error finds a minimum at a value of \( \ell \) near the half-width of the Gaussian density profile. This would suggest that the optimal choice of \( \ell \) should match the physical length scale of the feature being resolved. It should be noted that, for a given value of \( R \), varying the value of \( \ell \) does not affect the computational
efficiency of the scheme.

The result in Fig. 1(b) may give a false impression that a larger value of $\ell$ would just work appropriately for a wide range of general problems, thus removing the need to tune this parameter from the model consideration. However, such a case is to be considered as an oversimplified case (e.g., Gaussian advection) and only valid if a simulation under study has one dominant physical length scale that is almost invariant during the simulation. This is not true in practice because there could be a range of dynamically important length scales that can vary over time. In this case a method that allows adaptively changing $\ell$ locally will be ideal. We currently investigate on such an algorithm, "Adaptive $\ell$ Re-estimate (ALR)" method, to tune the GP model sensitivity on $\ell$, the approach of which is logically similar to the standard Adaptive Mesh Refinement (AMR) algorithms.

![Figure 1](image_url)

**Figure 1.** (a) Convergence for smooth Gaussian density advection using different GP stencil radii, all using the length scale $\ell = 0.1$. Black dotted lines show $2R + 1$ convergence rates, ranging from 3rd-order ($R = 1$) to 11th-order ($R = 5$). Red dotted line represents a plateau where a large condition number of the covariance matrix is obtained and no further accuracy is achievable. (b) $L_1$ errors as a function of the hyperparameter $\ell$, using $R = 2$. Dotted lines are the error for fifth-order WENO5 + RK4 method on the same stencil.

Shown in Scheme A of Table 1 is a performance comparison of the GP method with $R = 2$ relative to three most popular piecewise polynomial methods, including first-order Godunov (FOG) [30], PPM [25] with 2nd-order characteristic tracing, and WENO5 + RK4 [10]. As demonstrated, the performance of GP + RK4 is about 40% more expensive than the usual performance of PPM with characteristic tracing, while it is more than twice faster than WENO5 + RK4. Notice that we temporally integrated GP with $R = 2$ using RK4 in order to provide a fair comparison with WENO5+RK4 in terms of delivering the equivalent order of numerical solution accuracies in both space and time. The temporal schemes used in the other two, FOG and PPM, are much cheaper and lighter, delivering the overall solution accuracies first-order and second-order respectively at most.

In Scheme B, we now compare performances of the four spatial methods, all combined with RK4. This reveals us the computational expenses only due to the four different spatial solvers. As seen, GP is only twice expensive than FOG which does not even involve computing any high-order interpolation/reconstruction at all. The relative gain of GP compared to PPM and WENO5 is very significant, in that it only requires less than half of the computing times than those of the two methods.
| Scheme A          | Speedup | Scheme B          | Speedup |
|------------------|---------|------------------|---------|
| GP + RK4         | 1.0     | GP + RK4         | 1.0     |
| FOG              | 0.17    | FOG + RK4        | 0.5     |
| PPM + charTr     | 0.83    | PPM + RK4        | 2.8     |
| WENO5 + RK4      | 2.4     | WENO5 + RK4      | 2.4     |

Table 1. (Scheme A) Relative speedup of the GP + RK4 compared with piecewise polynomial schemes of first-order Godunov (FOG), PPM, and WENO5 + RK4 for the 1D Gaussian advection problem. (Scheme B) The same relative speedup except that all spatial schemes are now integrated with RK4. The comparisons have been obtained from a serial calculation on a single CPU.

3.2. 1D Shu-Osher Shock Tube Problem

The second test is the Shu-Osher problem [26] to test a shock-capturing scheme’s ability to also resolve small-scale flow features in addition to the shock. The test gives a good indication of the method’s numerical diffusivity, and is a popular benchmark to demonstrate numerical errors of a given method. In this problem, a (nominally) Mach 3 shock wave propagates into a constant density field with sinusoidal perturbations. As the shock advances, two sets of density features appear behind the shock. One set has the same spatial frequency as the unshocked perturbations, while in the second set the frequency is doubled and follows more closely behind the shock. The test of the numerical method then is to accurately resolve the dynamics and strengths of the oscillations behind the shock.

The results of this test are shown for various reconstruction methods in Fig. 2. The solutions are calculated at $t = 1.8$ using a resolution of $N = 200$ and are compared to a reference solution resolved on $N = 1600$. The GP solution uses $\ell/\Delta = 6$ and the shock detector on (see [31]). It is evident that the GP solution exhibits the least diffusive solution of the shown methods, producing a very-high order accurate solution that is more quickly approaching the high resolution reference solution.
4. Conclusion
We have introduced a new Gaussian Process approach in designing a family of high-order numerical algorithms to numerically solve the hyperbolic system of conservation laws. The GP methods presented here show an extremely fast rate of solution accuracy in smooth advection problems by controlling one single parameter $R$, strongly suggesting the new promising method is more effective than the conventional polynomial based high-order methods.

References
[1] Dongarra J J, Meuer H W, Simon H D and Strohmaier E 2010 The Birth of Numerical Analysis 93
[2] Dongarra J 2012 On the Future of High Performance Computing: How to Think for Peta and Exascale Computing (Hong Kong Science and Technology)
[3] Subcommittee A 2014 US Department Of Energy Report
[4] Keyes D E, McInnes L C, Woodward C, Gropp W, Myra E, Pernice M, Bell J, Brown J, Clo A, Connors J et al. 2013 International Journal of High Performance Computing Applications 27 4–83
[5] Attig N, Gibbon P and Lipper T 2011 Computer Physics Communications 182 2041–2046
[6] LeVeque R J 2002 Finite volume methods for hyperbolic problems vol 31 (Cambridge university press)
[7] LeVeque R J 2007 Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems vol 98 (Siam)
[8] Hesthaven J S, Gottlieb S and Gottlieb D 2007 Spectral methods for time-dependent problems vol 21 (Cambridge University Press)
[9] Toro E 2009 Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction (Springer) ISBN 9783540420293 URL http://books.google.com/books?id=5qEjK0um8oC
[10] Jiang G S and Shu C W 1996 Journal of Computational Physics 126 202 – 228 ISSN 0021-9991 URL http://www.sciencedirect.com/science/article/pii/S0021999196901308
[11] Mignone A, Tzeferacos P and Bodo G 2010 Journal of Computational Physics 229 5896–5920
[12] Lee D 2013 Journal of Computational Physics 243 269–292
[13] McCorquodale P and Colella P 2011 Communications in Applied Mathematics and Computational Science 6 1–25
[14] Shu C W 2009 SIAM review 51 82–126
[15] Buchmiller P and Helzel C 2014 Journal of Scientific Computing 61 343–368
[16] Gottlieb D and Shu C W 1999 Journal of Scientific Computing 41 644–668
[17] Gerolymos G, Senéchal D and Vallet I 2009 Journal of Computational Physics 228 8481 – 8524 ISSN 0021-9991 URL http://www.sciencedirect.com/science/article/pii/S0021999109003908
[18] Wiener N 1949 Extrapolation, interpolation, and smoothing of stationary time series, with engineering applications (Cambridge: Technology Press of the Massachusetts Institute of Technology) "First published during the war as a classified report to Section D 2, National Defense Research Committee."; Stationary time series
[19] Kolmogorov A 1941 Izv. Akad. Nauk. SSSR 5 3–14
[20] Wahba G, Johnson D, Gao F and Gong J 1995 Monthly Weather Review 123 3358–3369
[21] Rasmussen C and Williams C 2005 Gaussian Processes for Machine Learning Adaptive Computation And Machine Learning (MIT Press) ISBN 9780262182539 URL http://books.google.com/books?id=VtVQgAAACAAJ
[22] Bond J, Crittenden R, Jaffe A and Knox L 1999 Computing in science & engineering 1 21–35
[23] Stein M 1999 Interpolation of Spatial Data: Some Theory for Kriging Springer Series in Statistics New York) ISBN 9780387986296 URL http://books.google.com/books?id=5nXuL2w1EC
[24] Van Leer B 1979 Journal of Computational Physics 32 101–136
[25] Colella P and Woodward P R 1984 Journal of Computational Physics 54 174–201
[26] Shu C W and Osher S 1989 Journal of Computational Physics 83 32–78
[27] Zhang R, Zhang M and Shu C W 2011 Communications in Computational Physics 9 807–827
[28] Ambikasaran S, Foreman-Mackey D, Greengard L, Hogg D W and O’Neil M 2016 IEEE transactions on pattern analysis and machine intelligence 38 252–265
[29] Cressie N 2015 Statistics for spatial data (John Wiley &amp; Sons)
[30] Godunov S K 1959 Matematicheskii Sbornik 89 271–306
[31] Balsara D and Spicer D 1999 Journal of Computational Physics 149 270–292