Gaussian processes for high order finite volume methods

November 22, 2019

Ian May, Dongwook Lee

Department of Applied Mathematics
University of California Santa Cruz
Santa Cruz, CA
Introduction

Goal
Solve the compressible Euler equations (2D)

\[
\frac{\partial U}{\partial t} + \frac{\partial}{\partial x} F(U) + \frac{\partial}{\partial y} G(U) = 0
\]

\[U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \quad F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \end{pmatrix}, \quad G(U) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E + p) \end{pmatrix}\]

accurately and robustly

Finite volume considerations

- Handles shocks naturally
- Discretely conservative
- Agreeable with AMR
- Non-trivial to take beyond 2\text{nd} order accuracy
Finite volume formulation

Integrate over \( D_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}] \) and normalize

\[
\int_{D_{i,j}} \frac{\partial U}{\partial t} dV = - \int_{D_{i,j}} \nabla \cdot \mathbf{F} dV
\]

\[
\frac{\partial \langle U \rangle_{i,j}}{\partial t} = - \frac{1}{\Delta x \Delta y} \int_{\partial D_{i,j}} \mathbf{F} \cdot \mathbf{n} dS
\]

\[
\frac{\partial \langle U \rangle_{i,j}}{\partial t} = \frac{1}{\Delta x} \left( \langle F \rangle_{i-1/2,j} - \langle F \rangle_{i+1/2,j} \right) + \frac{1}{\Delta y} \left( \langle G \rangle_{i,j-1/2} - \langle G \rangle_{i,j+1/2} \right)
\]

where

\[
\langle h \rangle_{i,j} = \frac{1}{\Delta x \Delta y} \int_{D_{i,j}} h dV \quad \langle h \rangle_{i \pm 1/2,j} = \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} h(x_{i \pm 1/2}, y) dy
\]
Accuracy requirement

Numerical flux

\[ \langle F \rangle_{i \pm 1/2, j} = \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} F(U(x_{i \pm 1/2}, y)) \, dy \]

\[ \approx \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} \hat{F} \left( U^-_{i \pm 1/2}(y), U^+_{i \pm 1/2}(y) \right) \, dy \]

Two barriers to high order in multiple dimensions

- Integral must be done accurately
- Numerical flux is defined pointwise, thus need accurate pointwise values of \( U^\pm_{i \pm 1/2} \)
Polynomial reconstruction

Given the stencil \( \{ \langle U \rangle_{i-r,j}, \cdots, \langle U \rangle_{i,j}, \cdots, \langle U \rangle_{i+r,j} \} \), there is a unique polynomial \( Q(x) \) of degree \( p = 2r \) satisfying:

\[
\frac{1}{\Delta x} \int_{x_{i+s-1/2}}^{x_{i+s+1/2}} Q(x) \, dx = \langle U \rangle_{i+s,j}, \quad s = -r, \ldots, r
\]

which yields the approximation

\[
\langle U \rangle_{i \pm 1/2,j} = Q(x_{i \pm 1/2}) + \mathcal{O}(\Delta x^{p+1})
\]

Just plug it in, what could go wrong?

First note

\[
U(x_{i \pm 1/2}, y_j) = \langle U \rangle_{i \pm 1/2,j} + \mathcal{O}(\Delta x^{p+1}) + \mathcal{O}(\Delta y^2),
\]

and
Special cases

If $F(U) = AU$ then,

$$\langle F \rangle_{i \pm 1/2, j} = \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} AU(x_{i \pm 1/2}, y) \, dy$$

$$= A \langle U \rangle_{i \pm 1/2, j}$$

$$= AQ(x_{i \pm 1/2}) + O(\Delta x^p + 1).$$

Linear benchmarks

Consider the Euler equations on $[0, 1]^2$ with periodic boundaries and the initial condition:

$$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} 1 + e^{-50(x+y-2)^2} + e^{-50(x+y-1)^2} + e^{-50(x+y)^2} \\ 1 \\ 1 \\ 1/\gamma \end{pmatrix}.$$

run to a final time of $T = 1/2$. 
Figure: Modifications to retain non-linear accuracy don’t matter for this benchmark.
### Special cases

Convergence study

#### Experimental convergence rates

<table>
<thead>
<tr>
<th>$r$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.29</td>
<td>2.32</td>
<td>2.29</td>
<td>2.32</td>
</tr>
<tr>
<td>2</td>
<td>3.65</td>
<td>3.75</td>
<td>3.65</td>
<td>3.75</td>
</tr>
<tr>
<td>3</td>
<td>4.94</td>
<td>5.08</td>
<td>4.94</td>
<td>5.08</td>
</tr>
</tbody>
</table>
Expensive, but intuitive method

- Use multidimensional reconstruction to point values on faces directly
- Approximate flux integral with a Gauss rule
- \( \Rightarrow \) Need multiple point values on each face, multiple calls to Riemann solver
Expensive, but intuitive method

- Use multidimensional reconstruction to point values on faces directly
- Approximate flux integral with a Gauss rule
- ⇒ Need multiple point values on each face, multiple calls to Riemann solver

Modified dimension-by-dimension

- Use 1D stencils to get accurate face-averaged states
- Reconstruct along faces to get accurate face-centered states
- Call Riemann solver once per interface
- Reconstruct face-average fluxes from face-centered fluxes
Modified dimension-by-dimension
Diagrammatically
Modified dimension-by-dimension
Diagrammatically
Modified dimension-by-dimension
Diagramatically
Modified dimension-by-dimension
Diagrammatically
Modified dimension-by-dimension
Diagrammatically
Modified dimension-by-dimension
Diagramatically
Why Gaussian processes?

They generalize very well

- Dimension agnostic
- Order agnostic
- (Un)Structured grid agnostic
- Flexible stencil choices
- Directly incorporate problem physics
Why Gaussian processes?

They generalize very well

- Dimension agnostic
- Order agnostic
- (Un)Structured grid agnostic
- Flexible stencil choices
- Directly incorporate problem physics

Downsides of Gaussian processes (subjective)

- Ill-conditioning problems
- Less intuitive
- Almost too flexible, lot’s of choices to investigate
- The buzzword factor is high
Gaussian processes

Definition
For a domain $D$, a Gaussian process is given by a distribution over a function space:

$$f(x) \sim \mathcal{N}(0, K(x, x'; \ell)),$$

such that for $y \in D$, $f(y)$ belongs to a multivariate normal distribution:

$$f(y) \sim \mathcal{N}(0, K)$$

$$K_{ij} = K(y_i, y_j; \ell),$$

for some correlation (kernel) function $K$. Defined here as:

$$K(x, y; \ell) = e^{-\frac{||x-y||^2}{2\ell^2}}$$
Gaussian process interpolation

Posterior distribution
The Gaussian process conditioned on some given data, \( f(y_k) = q \) at some locations \( y_k \in D \), goes as:

\[
(f(x) | f(y) = q) \sim \mathcal{N}(\mu_y, K_y)
\]

\[
\mu_y = K(x, y; \ell)K^{-1}q
\]

\[
K_y = K(x, x'; \ell) - K(x, y; \ell)K^{-1}K(y, x'; \ell).
\]

All functions described by this process interpolate the data.
Gaussian process interpolation

**Posterior distribution**
The Gaussian process conditioned on some given data, \( f(y_k) = q \) at some locations \( y_k \in D \), goes as:

\[
(f(x) \mid f(y) = q) \sim \mathcal{N}(\mu_y, K_y)
\]

\[
\mu_y = K(x, y; \ell) K^{-1} q
\]

\[
K_y = K(x, x'; \ell) - K(x, y; \ell) K^{-1} K(y, x'; \ell).
\]

*All functions described by this process interpolate the data.*

**Mean is the most likely interpolant**
To predict \( f(x^*) \) for some \( x^* \in D \) evaluate the mean: \( f(x^*) \approx \mu_y(x^*) \).

Compactly:

\[
f(x^*) \approx K(x^*, y; \ell) K^{-1} q
\]

\[
\approx w_\ast^T q
\]
Dealing with cell/face averaged values

We want to use GP to convert between data types. Define correlation matrix to match input data,

\[ \mathbf{C}_{ij} = \frac{1}{||D_i|| \cdot ||D_j||} \int_{D_i} \int_{D_j} K(x, y) \, dx \, dy \]

and sample respecting the correlation between data types,

\[ \mathbf{T}_i = \frac{1}{||D_i||} \int_{D_i} K(x^*, x) \, dx \]

to find

\[ f(x^*) \approx \mathbf{T}^T \mathbf{C}^{-1} \mathbf{g} \]

for known cell/face averages, \( \mathbf{g} \)
Gaussian process reconstruction – 1D
Point values $\rightarrow$ average values

Converting point values back to average values

Very similar to interpolation, but with appropriate sample vector. Defining:

\[
K_{ij} = K(x_i, x_j)
\]

\[
T_i = \frac{1}{||D_i||} \int_{D^*} K(x_i, x) dx
\]

we find

\[
\langle f(x) \rangle_i \approx T^T K^{-1} q
\]

from known point values $q$
The isentropic vortex problem

A truly nonlinear benchmark problem

The Euler equations on \([-L, L]^2\) with periodic boundaries and initial condition

\[
\begin{pmatrix}
\rho \\
u \\
v \\
p
\end{pmatrix} = \begin{pmatrix}
T^{1/(\gamma-1)} \\
1 - y\omega \\
1 + x\omega \\
T^{\gamma/(\gamma-1)}
\end{pmatrix}
\]

\[
T = 1 - \frac{\gamma - 1}{8\gamma\pi^2} e^{1-x^2-y^2}
\]

\[
\omega = \frac{1}{2\pi} e^{(1-x^2-y^2)/2}
\]

recover the intial condition at time \(T_f = 2L\)
The isentropic vortex problem

Ian May
GP-FVM
UC Santa Cruz
### Experimental convergence rates

<table>
<thead>
<tr>
<th>$r$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.38</td>
<td>2.22</td>
<td>2.33</td>
<td>2.21</td>
</tr>
<tr>
<td>2</td>
<td>2.53</td>
<td>2.95</td>
<td>4.22</td>
<td>4.08</td>
</tr>
<tr>
<td>3</td>
<td>2.23</td>
<td>2.37</td>
<td>6.19</td>
<td>6.16</td>
</tr>
</tbody>
</table>
Dealing with shocks – GP-WENO

Nonlinear GP reconstruction
The reconstruction presented is linear, i.e.

\[
\langle U \rangle_{i \pm 1/2, j} = \sum_{s=-r}^{r} w_{k}^{\pm} \langle U \rangle_{i+s, j}
\]

which is hopeless near discontinuities (Godunov)

WENO (weighted essentially non-oscillatory) methods
Idea: Break full stencil into substencils, reconstruct on each separately, use a weighted combination of these reconstructions

\[
\langle U \rangle_{i \pm 1/2, j; k} = \sum_{s=k-r-1}^{k-1} \langle U \rangle_{i+s, j}
\]

\[
\langle U \rangle_{i \pm 1/2, j; k} = \sum \omega_{k}^{\pm} \langle U \rangle_{i \pm 1/2, j; k}
\]
For *smooth* data, $\omega_k^\pm$ should reduce to some optimal weights such that

$$\sum_{s=-r}^{r} w_s^\pm \langle U \rangle_{i+s,j} \approx \sum \gamma_k^\pm \langle U \rangle_{i\pm 1/2,j;k}$$

which can be found by solving the least squares problem (e.g. $r = 2$)

$$\begin{pmatrix}
  w_{1,1} & 0 & 0 \\
  w_{2,1} & w_{1,2} & 0 \\
  w_{3,1} & w_{2,2} & w_{1,3} \\
  0 & w_{3,2} & w_{2,3} \\
  0 & 0 & w_{3,3}
\end{pmatrix} \begin{pmatrix}
  \gamma_1 \\
  \gamma_2 \\
  \gamma_3
\end{pmatrix} = \begin{pmatrix}
  w_1 \\
  w_2 \\
  w_3 \\
  w_4 \\
  w_5
\end{pmatrix}$$
Smoothness indicators

Following Jiang and Shu, we can define

\[
\omega_k = \frac{\tilde{\omega}_k}{\sum \tilde{\omega}_s} \quad \tilde{\omega}_k = \frac{\gamma_k}{(\epsilon + \beta_k)^p}
\]

where \(\beta_k\) measures the smoothness of the data on the \(k^{th}\) sub-stencil.

Likelihood measures smoothness

A GP with a SE kernel is good at representing smooth functions, thus the log-likelihood

\[
\log L_k = -\frac{1}{2} \left( \log |K_k| + q^T K_k^{-1} q + 2 \log(2\pi) \right)
\]

indicates how smooth the \(k^{th}\) sub-stencil is. Choosing

\[
\beta_k = q^T K_k^{-1} q
\]

works well for equispaced grids.
Notes on (GP) WENO

General WENO

- Transform to characteristic variables first
- Componentwise limiting works well enough, lower dissipation
- Lots of non-linear weights other than WENO-JS

GP-WENO

- Note $K_k$ instead of $C_k$ in formula for $\beta_k$
- Could use completely different types of stencils
- No need to derive new expressions for each order
The isentropic vortex – WENO active

Naive D-by-D

Corrected D-by-D

Ian May
GP-FVM
UC Santa Cruz
## Experimental convergence rates

<table>
<thead>
<tr>
<th>$r$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
<th>$L_1$</th>
<th>$L_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.90</td>
<td>1.65</td>
<td>1.88</td>
<td>1.66</td>
</tr>
<tr>
<td>2</td>
<td>2.79</td>
<td>3.12</td>
<td>3.46</td>
<td>3.98</td>
</tr>
<tr>
<td>3</td>
<td>2.51</td>
<td>3.29</td>
<td>5.38</td>
<td>5.80</td>
</tr>
</tbody>
</table>
The standard Riemann problem
Euler equations on $[0, 1]$ with inflow/outflow boundaries and initial condition

$$
\begin{pmatrix}
\rho \\
u \\
p
\end{pmatrix}
= \begin{cases}
\begin{pmatrix}
1 & 0 & 1
\end{pmatrix}^T, & x < 0.5 \\
\begin{pmatrix}
0.125 & 0 & 0.1
\end{pmatrix}^T, & x > 0.5
\end{cases}
$$
Sod shock tube
Euler equations on $[0, 1]^2$ with outflow boundaries and initial condition

\[
\begin{align*}
\begin{pmatrix}
\rho_1 \\
u_1 \\
v_1 \\
p_1 \\
\end{pmatrix} &= 
\begin{pmatrix}
0.5323 \\
1.206 \\
0 \\
0.3 \\
\end{pmatrix}, \\
\begin{pmatrix}
\rho_2 \\
u_2 \\
v_2 \\
p_2 \\
\end{pmatrix} &= 
\begin{pmatrix}
1.5 \\
0 \\
0 \\
1.5 \\
\end{pmatrix}, \\
\begin{pmatrix}
\rho_3 \\
u_3 \\
v_3 \\
p_3 \\
\end{pmatrix} &= 
\begin{pmatrix}
0.138 \\
1.206 \\
1.206 \\
0.029 \\
\end{pmatrix}, \\
\begin{pmatrix}
\rho_4 \\
u_4 \\
v_4 \\
p_4 \\
\end{pmatrix} &= 
\begin{pmatrix}
0.5323 \\
0 \\
1.206 \\
0.3 \\
\end{pmatrix}.
\end{align*}
\]
2D Riemann problem configuration 3
With flux correction
Sedov blast problem
Final thoughts

Conclusion

- Naive use of 1D stencils in 2D yields 2nd nonlinear accuracy
- A cheap modification to the reconstruction recovers accuracy (Buchmuller and Helzel)
- Gaussian process reconstruction is super flexible, same formulas for many orders
- GP yields simple, effective, smoothness indicators for WENO

Next steps

- Appropriate limiting for flux reconstruction
- Time stepping without RK
- Ideal MHD – Divergence free GP methods
- 3D problems
The isentropic vortex problem

\[ L_\infty \]

\begin{align*}
\text{Naive D-by-D} & \\
\text{Corrected D-by-D} & \\
\end{align*}

\begin{align*}
\text{log}_{10} N_x & \\
\text{log}_{10} |\rho_0 - \rho_i|_\infty & \\
\end{align*}

\begin{align*}
r = 1 & \\
r = 2 & \\
r = 3 & \\
\end{align*}

Ian May
GP-FVM
UC Santa Cruz