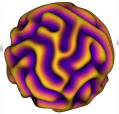
Gaussian processes for high order finite volume methods

November 22, 2019

Ian May, Dongwook Lee

Department of Applied Mathematics
University of California Santa Cruz

Santa Cruz, CA



Goal

Solve the compressible Euler equations (2D)

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

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix} \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(E+p) \end{pmatrix} \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}$$

accurately and robustly

Finite volume considerations

- Handles shocks naturally
- Discretely conservative
- Agreeable with AMR
- Non-trivial to take beyond 2^{nd} order accuracy

Finite volume formulation

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

$$\int\limits_{D_{i,j}} \frac{\partial \mathbf{U}}{\partial t} dV = -\int\limits_{D_{i,j}} \nabla \cdot \underline{\underline{\mathbf{F}}} dV$$

$$\frac{\partial \langle \mathbf{U} \rangle_{i,j}}{\partial t} = -\frac{1}{\Delta x \Delta y} \int\limits_{\partial D_{i,j}} \underline{\underline{\mathbf{F}}} \cdot \hat{\mathbf{n}} dS$$

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

where

$$\langle h \rangle_{i,j} = \frac{1}{\Delta x \Delta y} \int\limits_{D_{i,j}} h dV \qquad \langle h \rangle_{i\pm 1/2,j} = \frac{1}{\Delta y} \int\limits_{y_{j-1/2}}^{y_{j+1/2}} h(x_{i\pm 1/2}, y) dy$$

Accuracy requirement

Numerical flux

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

$$\approx \frac{1}{\Delta y} \int\limits_{y_{j-1/2}}^{y_{j+1/2}} \hat{\mathbf{F}}\left(\mathbf{U}^{-}_{i\pm 1/2}(y), \mathbf{U}^{+}_{i\pm 1/2}(y)\right) dy$$

Two barriers to high order in multiple dimensions

- Integral must be done accurately
- Numerical flux is defined *pointwise*, thus need accurate *pointwise* values of $\mathbf{U}^{\pm}_{i\pm 1/2}$

Polynomial reconstruction

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

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

which yields the approximation

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

Just plug it in, what could go wrong?

First note

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

and

Special cases

If $\mathbf{F}(\mathbf{U}) = \mathbf{A}\mathbf{U}$ then,

$$\begin{aligned}
\langle \mathbf{F} \rangle_{i\pm 1/2,j} &= \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{A}\mathbf{U}\left(x_{i\pm 1/2}, y\right) dy \\
&= \mathbf{A}\langle \mathbf{U} \rangle_{i\pm 1/2,j} \\
&= \mathbf{A}\mathbf{Q}(x_{i\pm 1/2}) + \mathcal{O}(\Delta x^{p+1}).
\end{aligned}$$

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 $T = 1/2$.

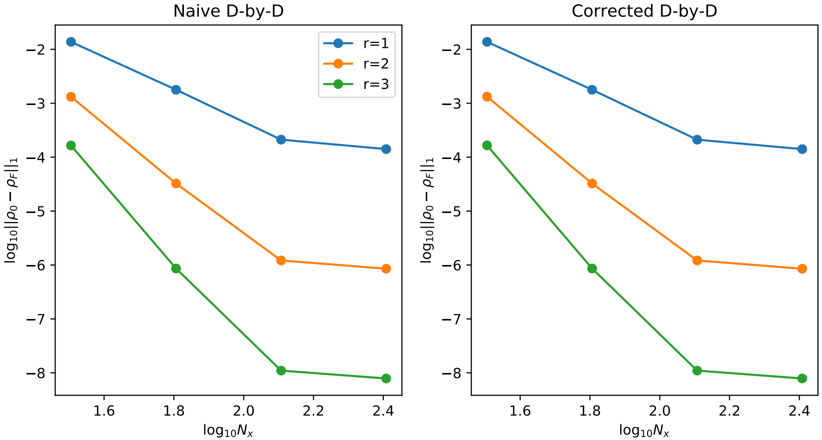


Figure: Modifications to retain non-linear accuracy don't matter for this benchmark.

Experimental convergence rates

	L_1	L_∞	L_1	L_∞
$r = 1$	2.29	2.32	2.29	2.32
$r = 2$	3.65	3.75	3.65	3.75
$r = 3$	4.94	5.08	4,94	5.08

Routes to high order

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

Routes to high order

Expensive, but intuitive method

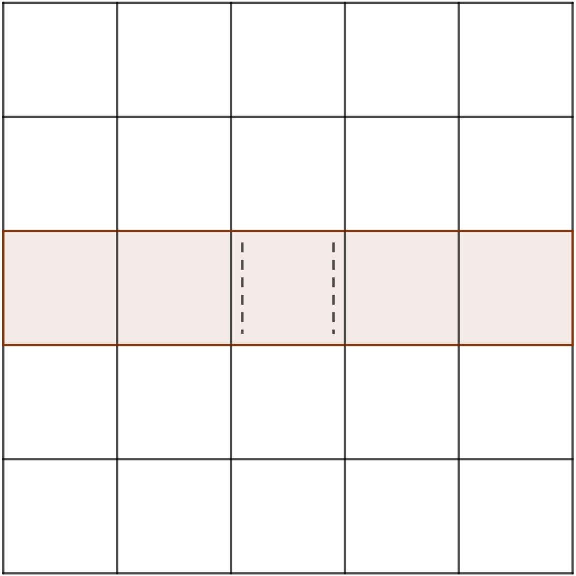
- Use multidimensional reconstruction to point values on faces directly
- Approximate flux integral with a Gauss rule
- $\Rightarrow$ Need multiple point values on each face, multiple calls to Riemann solver

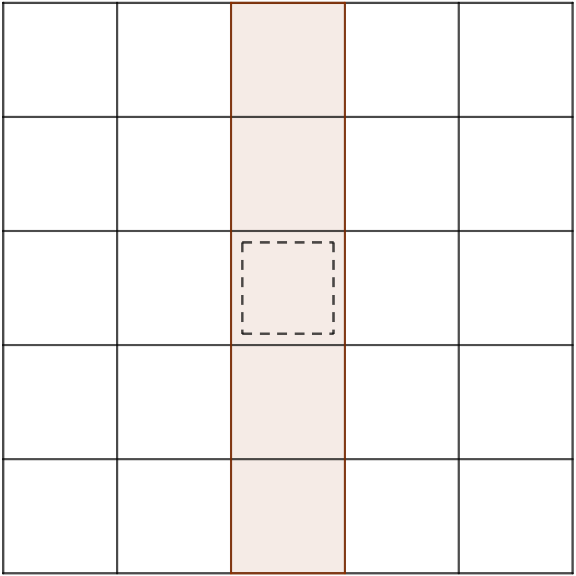
Modified dimension-by-dimension

- 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

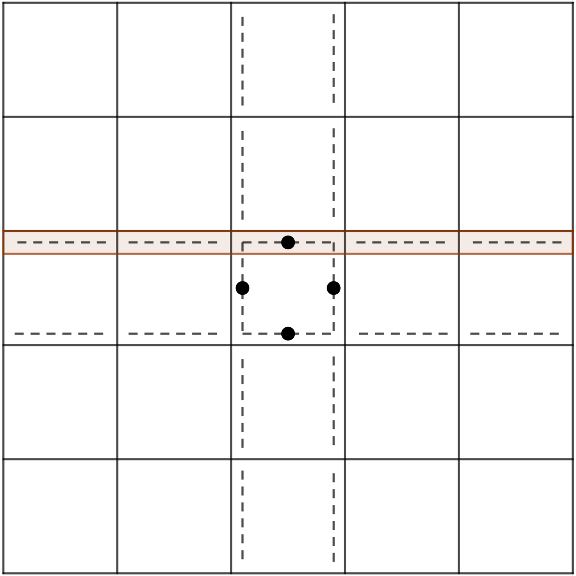
Modified dimension-by-dimension

Diagramatically



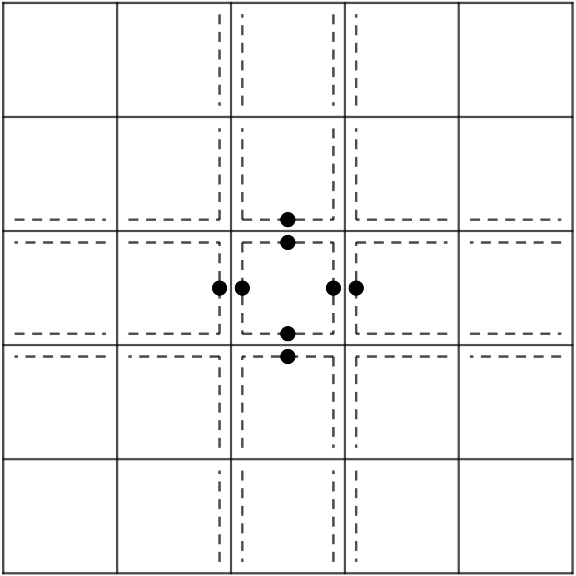


Modified dimension-by-dimension
Diagramatically



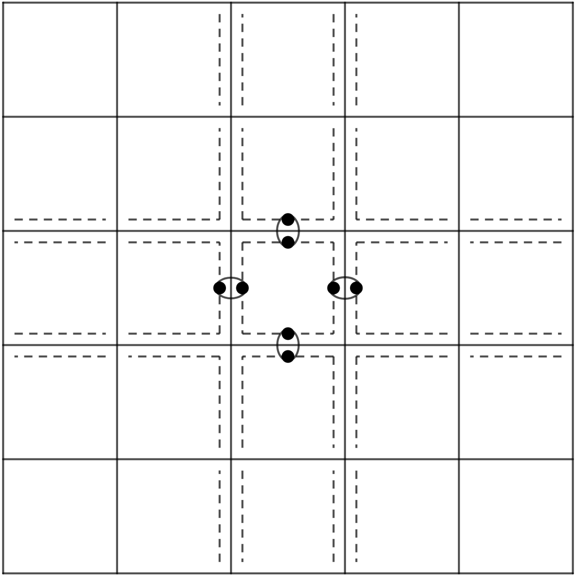
Modified dimension-by-dimension

Diagramatically



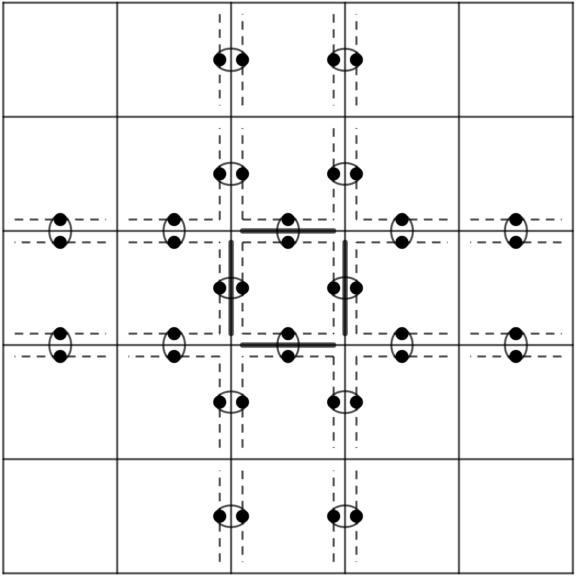
Modified dimension-by-dimension

Diagramatically



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

Definition

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

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

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

$$f(\mathbf{y}) \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}\right)$$
$$\mathbf{K}_{ij} = K(y_i, y_j; \ell),$$

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

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

Gaussian process interpolation

Posterior distribution

The Gaussian process conditioned on some given data, $f(\mathbf{y}_k) = \mathbf{q}$ at some locations $\mathbf{y}_k \in D$, goes as:

$$\left(f(\mathbf{x})\,|\,f(\mathbf{y}) = \mathbf{q}\right) \sim \mathcal{N}\left(\mu_{\mathbf{y}}, K_{\mathbf{y}}\right)$$
$$\mu_{\mathbf{y}} = K\left(\mathbf{x}, \mathbf{y}; \ell\right) \mathbf{K}^{-1}\mathbf{q}$$
$$K_y = K(\mathbf{x}, \mathbf{x}'; \ell) - K(\mathbf{x}, \mathbf{y}; \ell)\mathbf{K}^{-1}K(\mathbf{y}, \mathbf{x}'; \ell).$$

All functions described by this process interpolate the data.

Gaussian process interpolation

Posterior distribution

The Gaussian process conditioned on some given data, $f(\mathbf{y}_k) = \mathbf{q}$ at some locations $\mathbf{y}_k \in D$, goes as:

$$\begin{gathered}(f(\mathbf{x})\,|\,f(\mathbf{y}) = \mathbf{q}) \sim \mathcal{N}\left(\mu_{\mathbf{y}}, K_{\mathbf{y}}\right) \\ \mu_{\mathbf{y}} = K\left(\mathbf{x}, \mathbf{y}; \ell\right)\mathbf{K}^{-1}\mathbf{q} \\ K_y = K(\mathbf{x}, \mathbf{x}'; \ell) - K(\mathbf{x}, \mathbf{y}; \ell)\mathbf{K}^{-1}K(\mathbf{y}, \mathbf{x}'; \ell).\end{gathered}$$

All functions described by this process interpolate the data.

Mean is the most likely interpolant

To predict $f(\mathbf{x}^*)$ for some $\mathbf{x}^* \in D$ evaluate the mean: $f(\mathbf{x}^*) \approx \mu_{\mathbf{y}}(\mathbf{x}^*)$. Compactly:

$$\begin{aligned} f(\mathbf{x}^*) &\approx K\left(\mathbf{x}^*, \mathbf{y}; \ell\right)\mathbf{K}^{-1}\mathbf{q} \\ &\approx \mathbf{w}_*^{\mathrm{T}}\mathbf{q}\end{aligned}$$

Dealing with cell/face averaged values

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

$$\mathbf{C}_{ij} = \frac{1}{||D_i||\,||D_j||} \int_{D_i} \int_{D_j} K(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}$$

and sample respecting the correlation between data types,

$$\mathbf{T}_i = \frac{1}{||D_i||} \int_{D_i} K(\mathbf{x}^*, \mathbf{x}) d\mathbf{x}$$

to find

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

for known cell/face averages, $\mathbf{g}$

Converting point values back to average values

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

$$\mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$$

$$\mathbf{T}_i = \frac{1}{||D_i||} \int\limits_{D_*} K(\mathbf{x}_i, \mathbf{x}) d\mathbf{x}$$

we find

$$\langle f(\mathbf{x}) \rangle_i \approx \mathbf{T}^{\mathrm{T}} \mathbf{K}^{-1} \mathbf{q}$$

from known point values $\mathbf{q}$

The isentropic vortex problem

A truly nonlinear benchmark problem

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

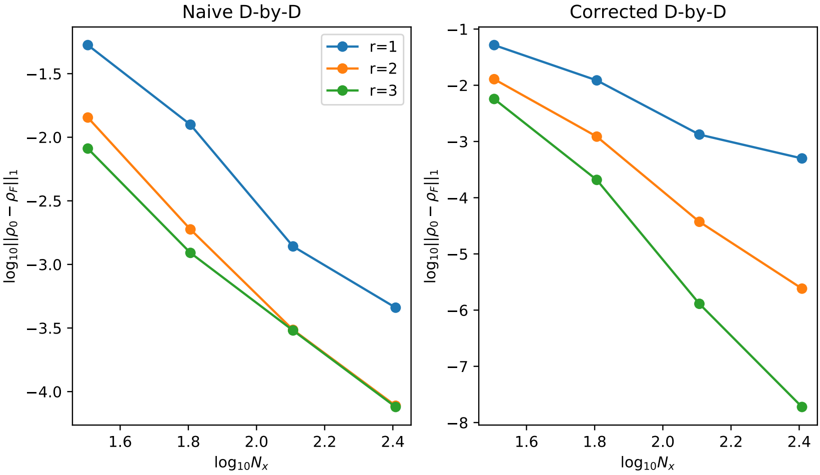
$$\begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix} = \begin{pmatrix} T^{1/(\gamma-1)} \\ 1 - y\omega \\ 1 + x\omega \\ T^{\gamma/(\gamma-1)} \end{pmatrix}$$

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

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

recover the intial condition at time $T_f = 2L$

The isentropic vortex problem



Experimental convergence rates

	L_1	L_∞	L_1	L_∞
$r=1$	2.38	2.22	2.33	2.21
$r=2$	2.53	2.95	4.22	4.08
$r=3$	2.23	2.37	6.19	6.16

Dealing with shocks – GP-WENO

Nonlinear GP reconstruction

The reconstruction presented is linear, i.e.

$$\langle \mathbf{U} \rangle_{i\pm 1/2,j} = \sum_{s=-r}^{r} w_k^{\pm} \langle \mathbf{U} \rangle_{i+s,j}$$

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

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

$$\langle \mathbf{U} \rangle_{i\pm 1/2,j;k} = \sum \omega_k^{\pm} \langle \mathbf{U} \rangle_{i\pm 1/2,j;k}$$

Optimal weights

For *smooth* data, $\omega_k^\pm$ should reduce to some optimal weights such that

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

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

$$\omega_k = \frac{\widetilde{\omega}_k}{\sum \widetilde{\omega}_s} \qquad \widetilde{\omega}_k = \frac{\gamma_k}{(\epsilon + \beta_k)^p}$$

where β_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

$$\log L_k = -\frac{1}{2}\left(\log|\mathbf{K}_k| + \mathbf{q}^{\mathrm{T}}\mathbf{K}_k^{-1}\mathbf{q} + 2\log(2\pi)\right)$$

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

$$\beta_k = \mathbf{q}^{\mathrm{T}}\mathbf{K}_k^{-1}\mathbf{q}$$

works well for equispaced grids

Notes on (GP) WENO

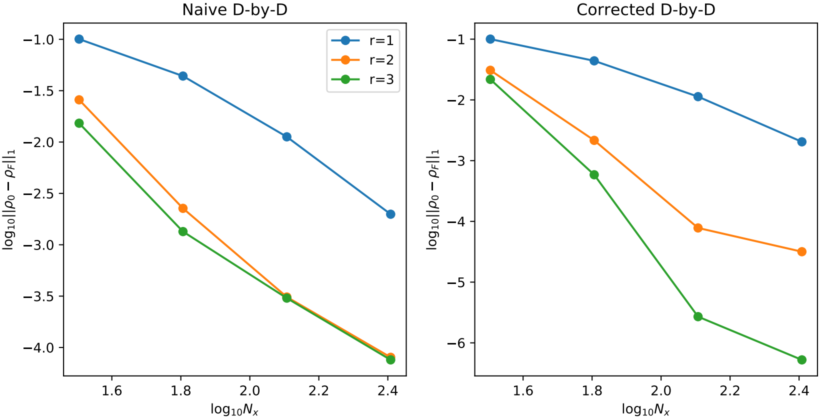
General WENO

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

GP-WENO

- Note $\mathbf{K}_k$ instead of $\mathbf{C}_k$ in formula for β_k
- Could use completely different types of stencils
- No need to derive new expressions for each order

The isentropic vortex – WENO active



Experimental convergence rates

	L_1	L_∞	L_1	L_∞
$r=1$	1.90	1.65	1.88	1.66
$r=2$	2.79	3.12	3.46	3.98
$r=3$	2.51	3.29	5.38	5.80

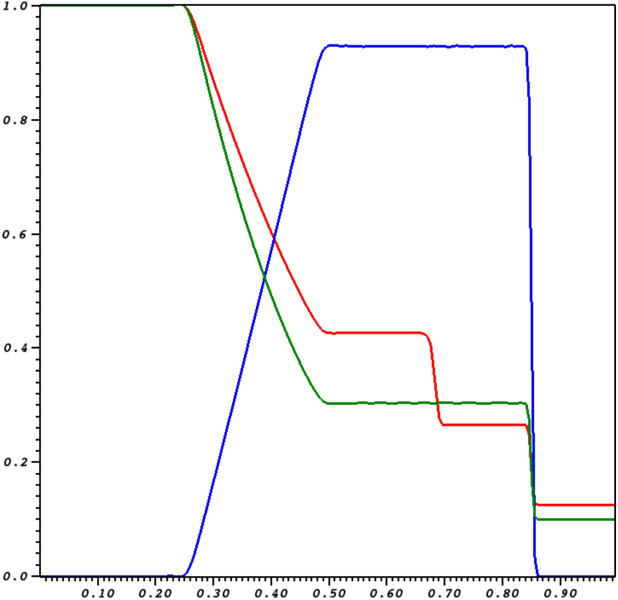
Sod shock tube

The standard Riemann problem

Euler equations on $[0,1]$ with inflow/outflow boundaries and initial condition

$$\begin{pmatrix}\rho\\ u\\ p\end{pmatrix} = \begin{cases}\begin{pmatrix}1 & 0 & 1\end{pmatrix}^{\mathrm{T}}, & x < 0.5\\ \begin{pmatrix}0.125 & 0 & 0.1\end{pmatrix}^{\mathrm{T}}, & x > 0.5\end{cases}$$

Sod shock tube



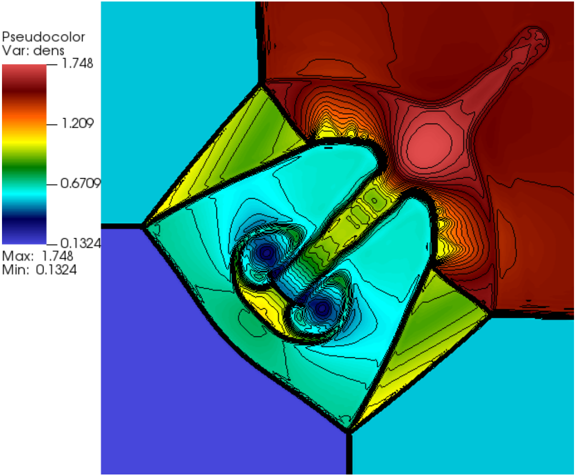
2D Riemann problem configuration 3

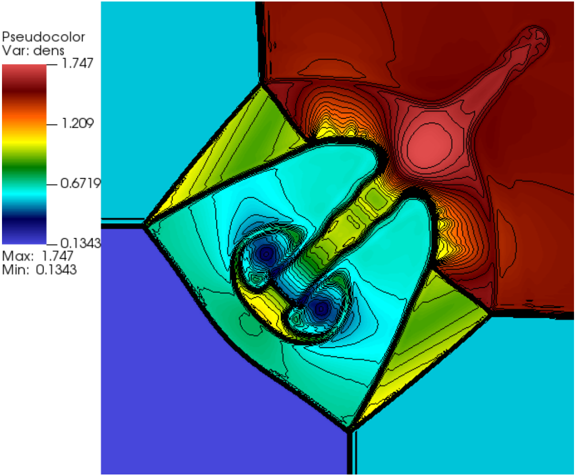
Euler equations on $[0,1]^2$ with outflow boundaries and initial condition

$$\begin{pmatrix}\rho_1\\u_1\\v_1\\p_1\end{pmatrix}=\begin{pmatrix}0.5323\\1.206\\0\\0.3\end{pmatrix}\qquad\begin{pmatrix}\rho_2\\u_2\\v_2\\p_2\end{pmatrix}=\begin{pmatrix}1.5\\0\\0\\1.5\end{pmatrix}$$

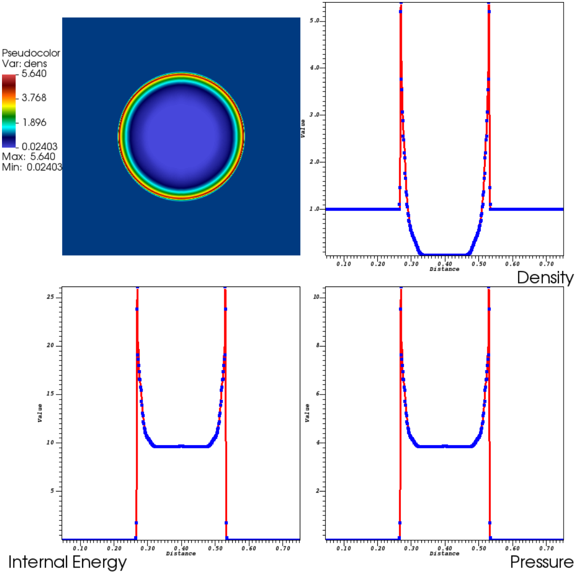
$$\begin{pmatrix}\rho_3\\u_3\\v_3\\p_3\end{pmatrix}=\begin{pmatrix}0.138\\1.206\\1.206\\0.029\end{pmatrix}\qquad\begin{pmatrix}\rho_4\\u_4\\v_4\\p_4\end{pmatrix}=\begin{pmatrix}0.5323\\0\\1.206\\0.3\end{pmatrix}$$

Without flux correction




Pseudocolor
Var: dens
1.747
1.209
0.6719
0.1343
Max: 1.747
Min: 0.1343

Sedov blast problem



Final thoughts

Conclusion

- Naive use of 1D stencils in 2D yields 2^{nd} 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_∞