Shock Capturing via Limiting for High-Order Methods Including Discontinuous Galerkin and Flux Reconstruction

ICOSAHOM

McGill University, July 13-18, 2025

H. T. Huynh

NASA Glenn Research Center

Supported by

NASA Transformational Tools and Technologies Project

Motivation

- High-order methods are becoming popular in CFD
- For flows with shocks, despite recent successes, challenges remain: the solutions can oscillate, lose accuracy, or worse, we get NaN.

Approaches to Capture Shocks

- Approaches to capture shocks: Limiting (the focus here), filtering, artificial viscosity, and method modification

- A few important works on limiting: Cockburn and Shu (1990…), Krivodonova and coauthors (2004, 2007), Wang and coauthors (2009…), (Kim, Park, You, and Kim 2010 …)

Limiting

- Existing approach for high-order methods:
  - Obtain a solution using the base scheme.
  - For each cell, detect if the solution is "smooth" or "not smooth", i.e., the cell is "good" or "troubled"
  - If cell is "good", move on to next cell, if cell is "troubled" apply limiting.

  Detection can fail

- Current work: a limiter where detection is not critical to preserve accuracy near extrema

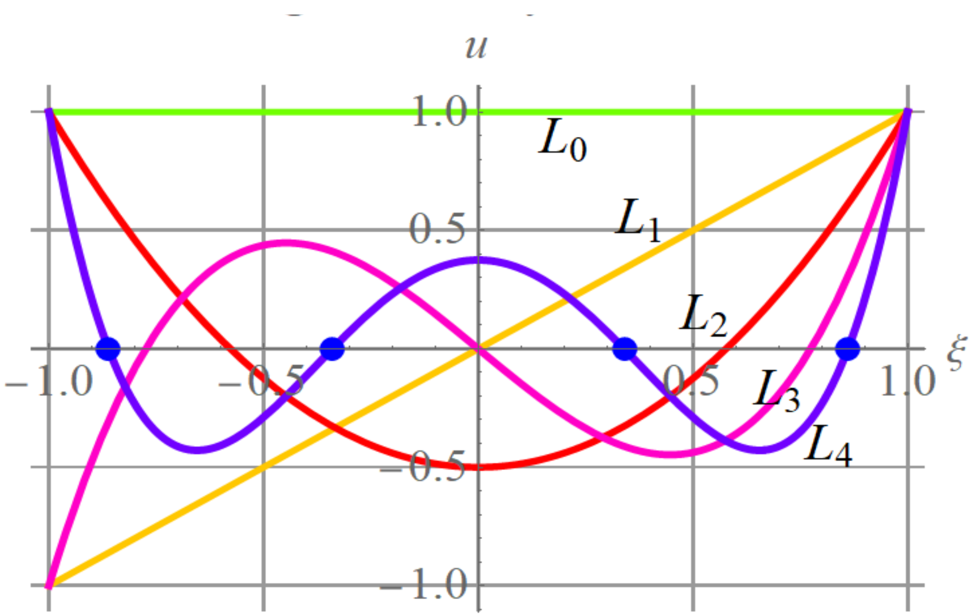
Outline

- Discuss oscillations caused by a discontinuity
- Present a new limiting
- Detect smooth regions to save computing time
- Some numerical examples
- Conclusions and discussion

Legendre Polynomials

On $[-1,1]$, the **Legendre** polynomial L_k of degree k is defined by:

- $L_k(1) = 1$
- L_k is orthogonal to all polynomials of degree less than k: for $m = 0, 1, \dots, k-1$,

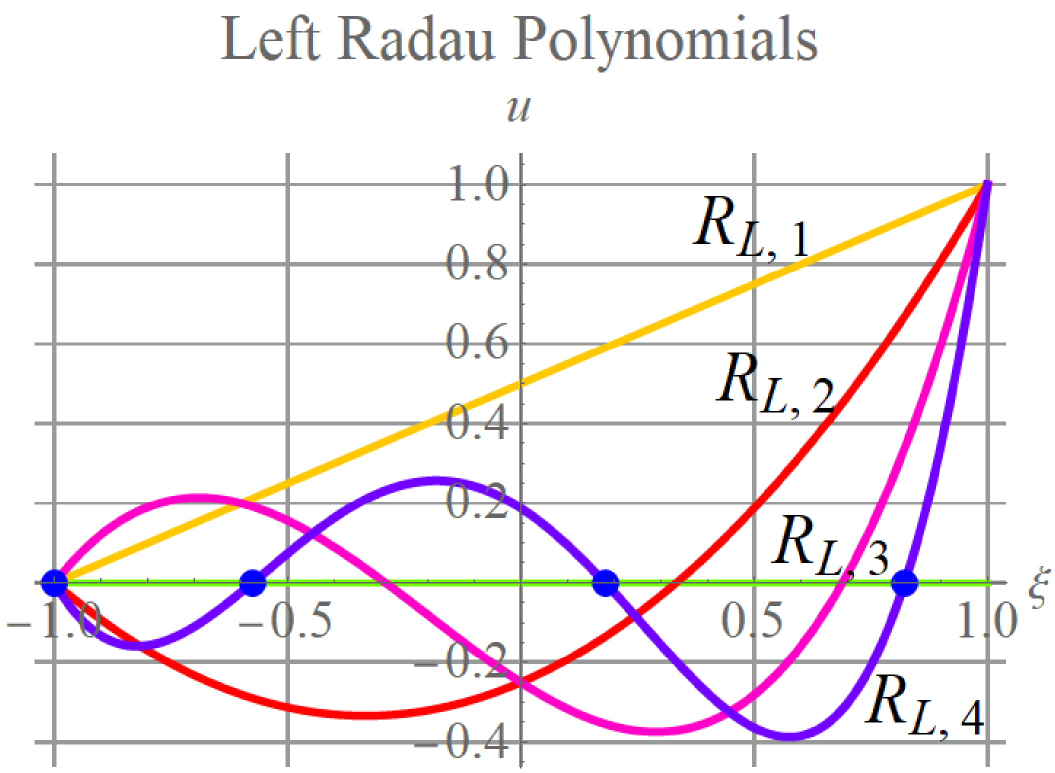


$$\int_{-1}^{1} L_k(\xi)\xi^m d\xi = 0.$$

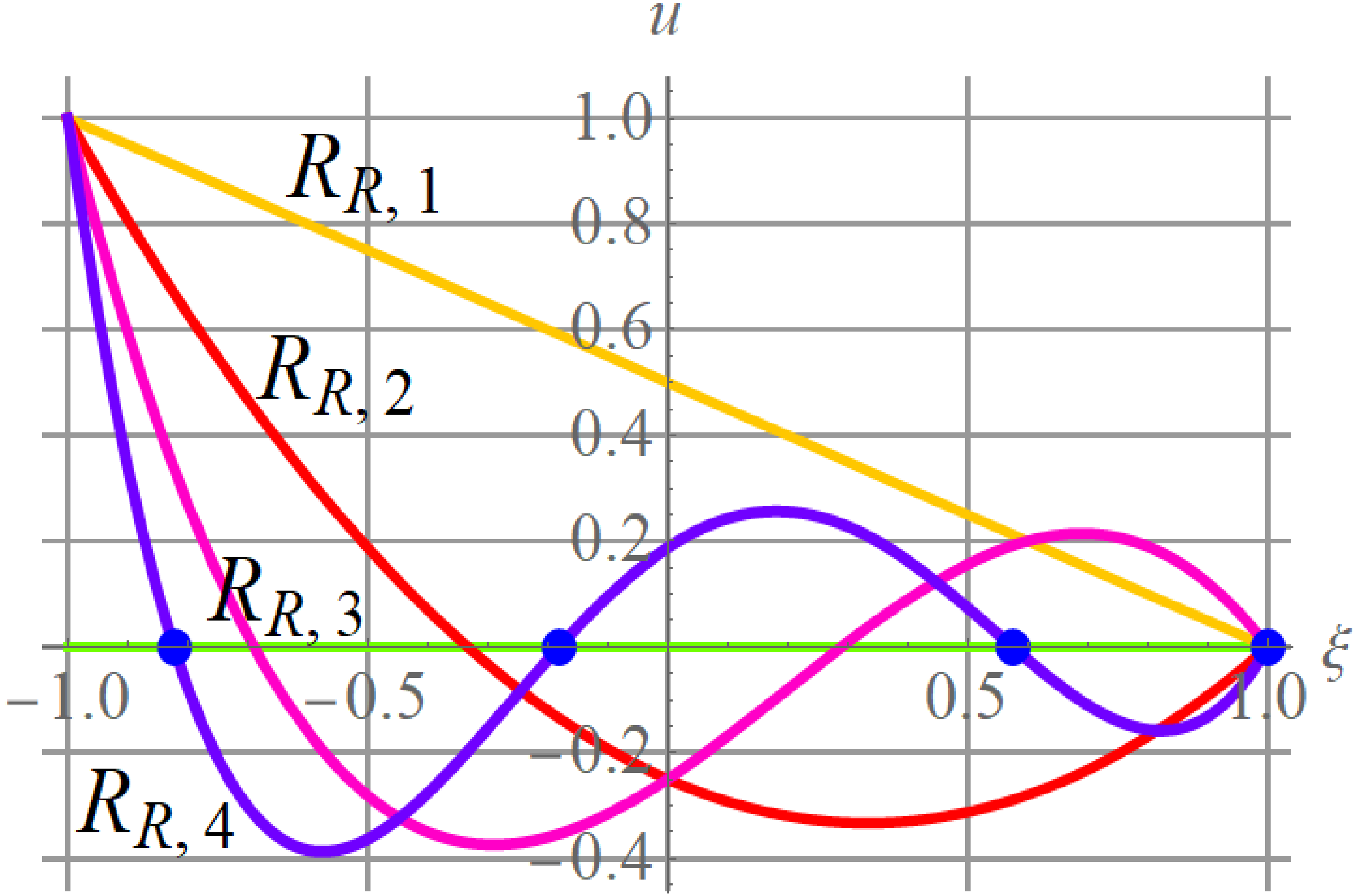
Left Radau Polynomials

$$R_{L,k} = \frac{1}{2}(L_k + L_{k-1}).$$

Then $R_{L,k}(-1) = 0$, and $R_{L,k}(1) = 1$.



Right Radau Polynomials



Derivatives of Right Radau Polynomials

$$R_{R,k+1}{}' =$$

$$\frac{1}{2}[-L_0 + 3L_1 - 5L_2$$

$$+ 7L_3 + \cdots$$

$$+(-1)^{k+1}(2k+1)L_k].$$

Polynomial Approximation

Approximate the solution in each cell by a degree p polynomial

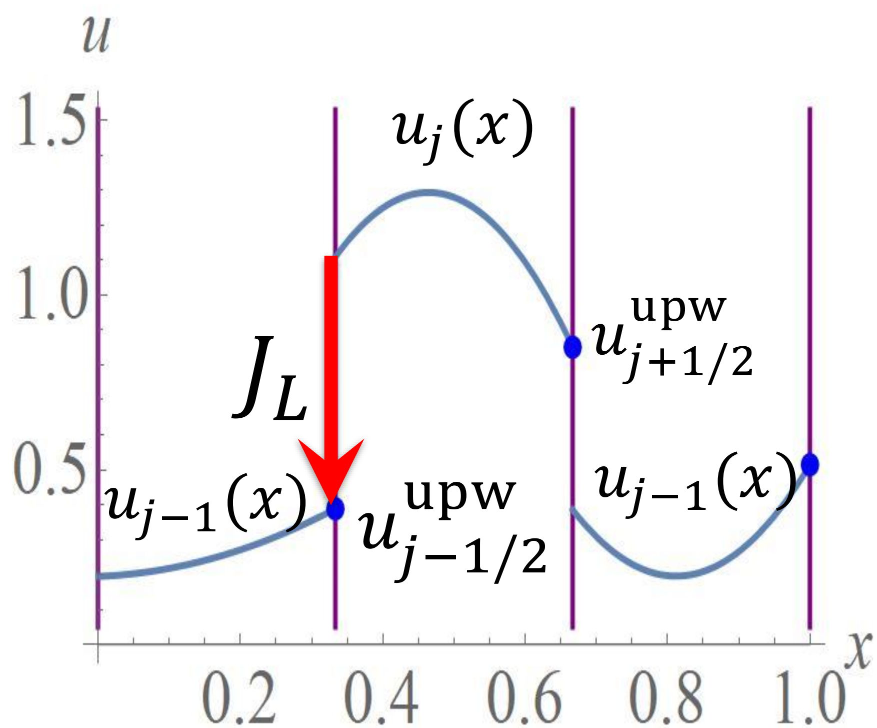
$$u_j(\xi) = \sum_{k=0}^{p} u_{j,k}\, L_k(\xi)\,.$$

At the right interface $\xi = 1$,

$$u_j(1) = \sum_{k=0}^{p} u_{j,k} = u_{j,0} + u_{j,1} + u_{j,2} + u_{j,3} + \dots$$

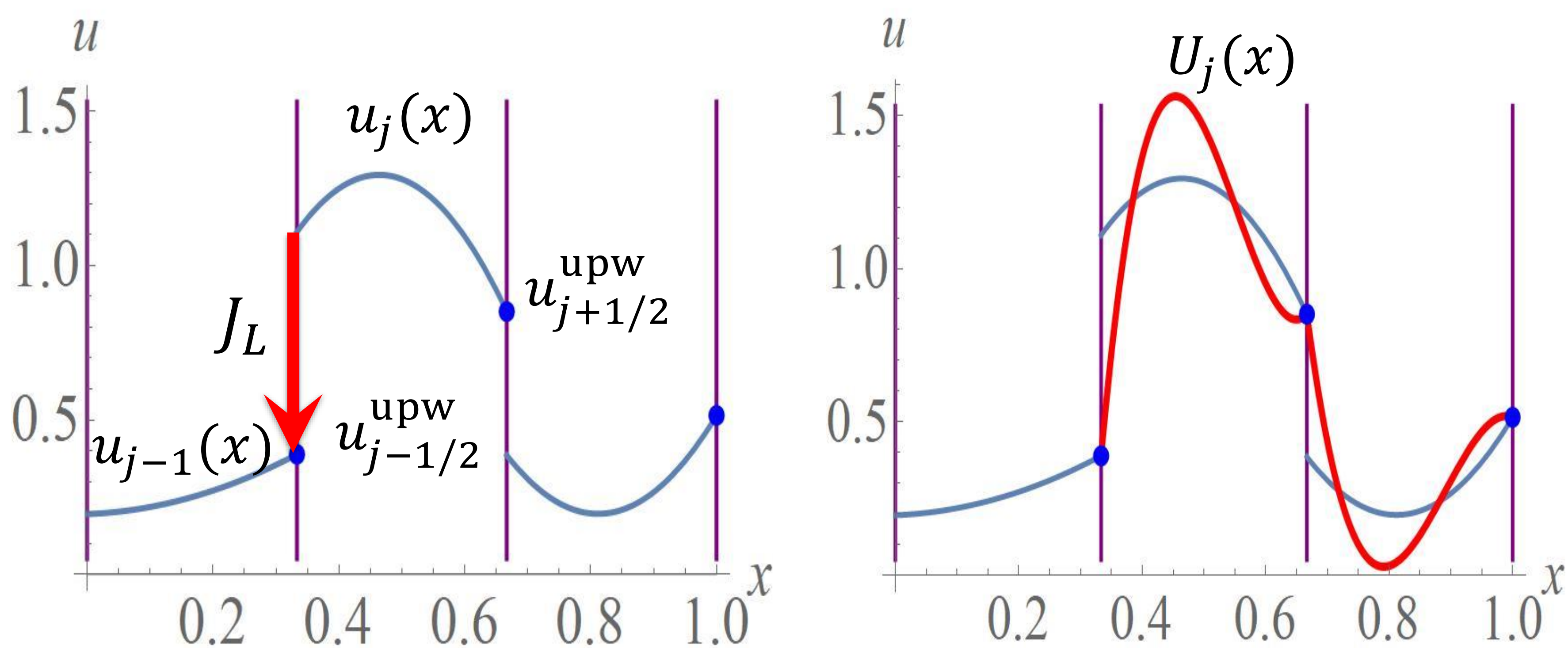
DG via FR framework for Advection

$$u_t + a\,u_x = 0.$$



Example: $u_j(x)$ are **quadratics** $(p = 2)$.

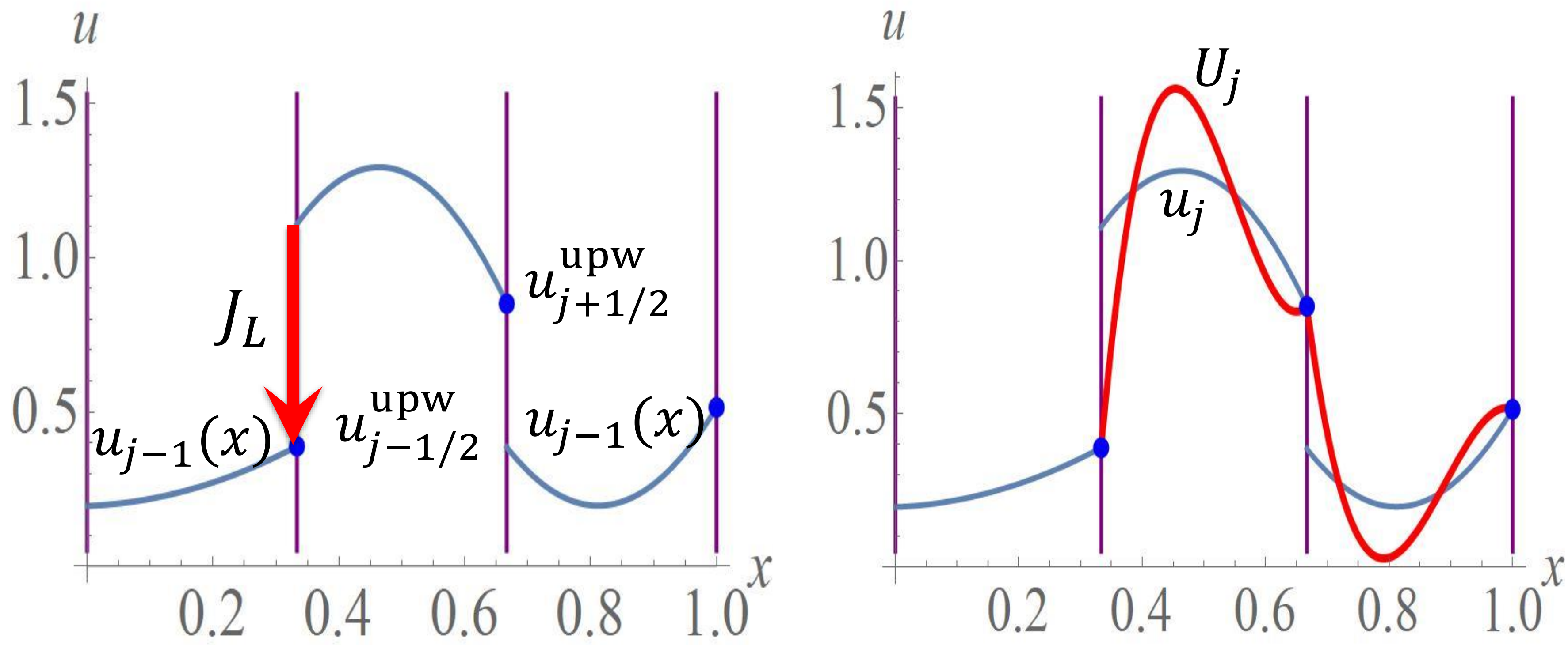
DG via FR framework



Define cubic $U_j(x)$ by (4 conditions):

- $U_j(x_{j-1/2}) = u^{\text{upw}}_{j-1/2}$,
- U_j interpolates u_j at the 3 right Radau points. Thus, $U_j(x_{j+1/2}) = u^{\text{upw}}_{j+1/2}$.

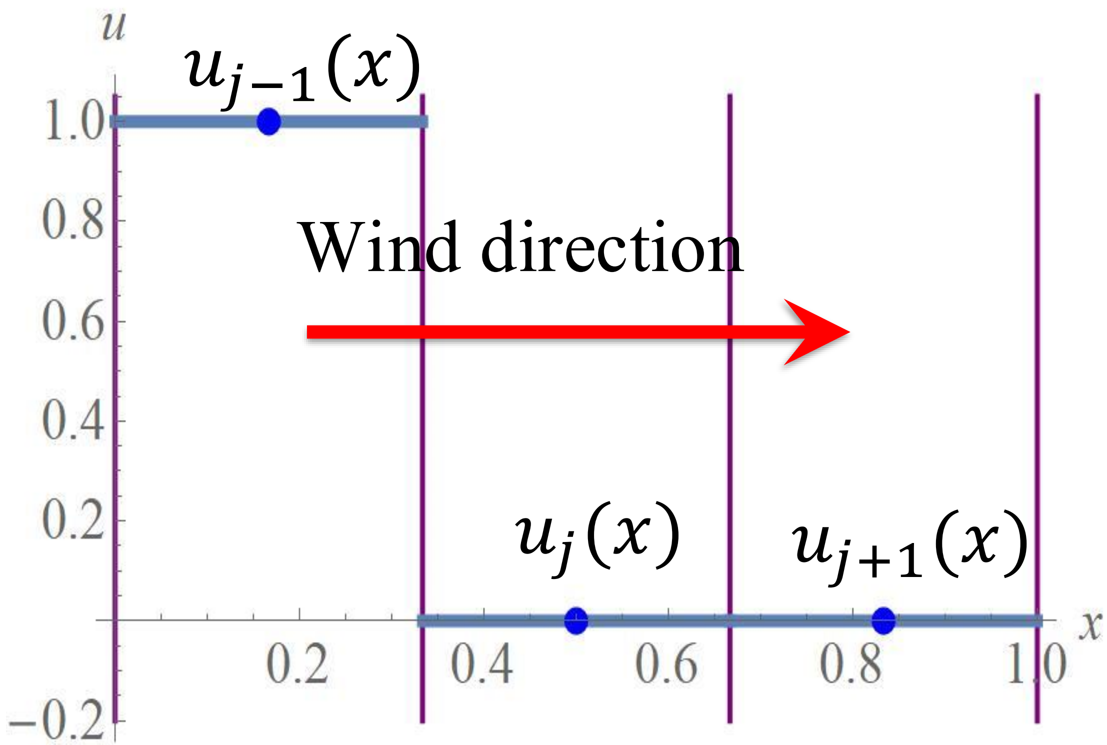
DG via FR framework



Use right Radau polynomial $R_{R,p+1}$:

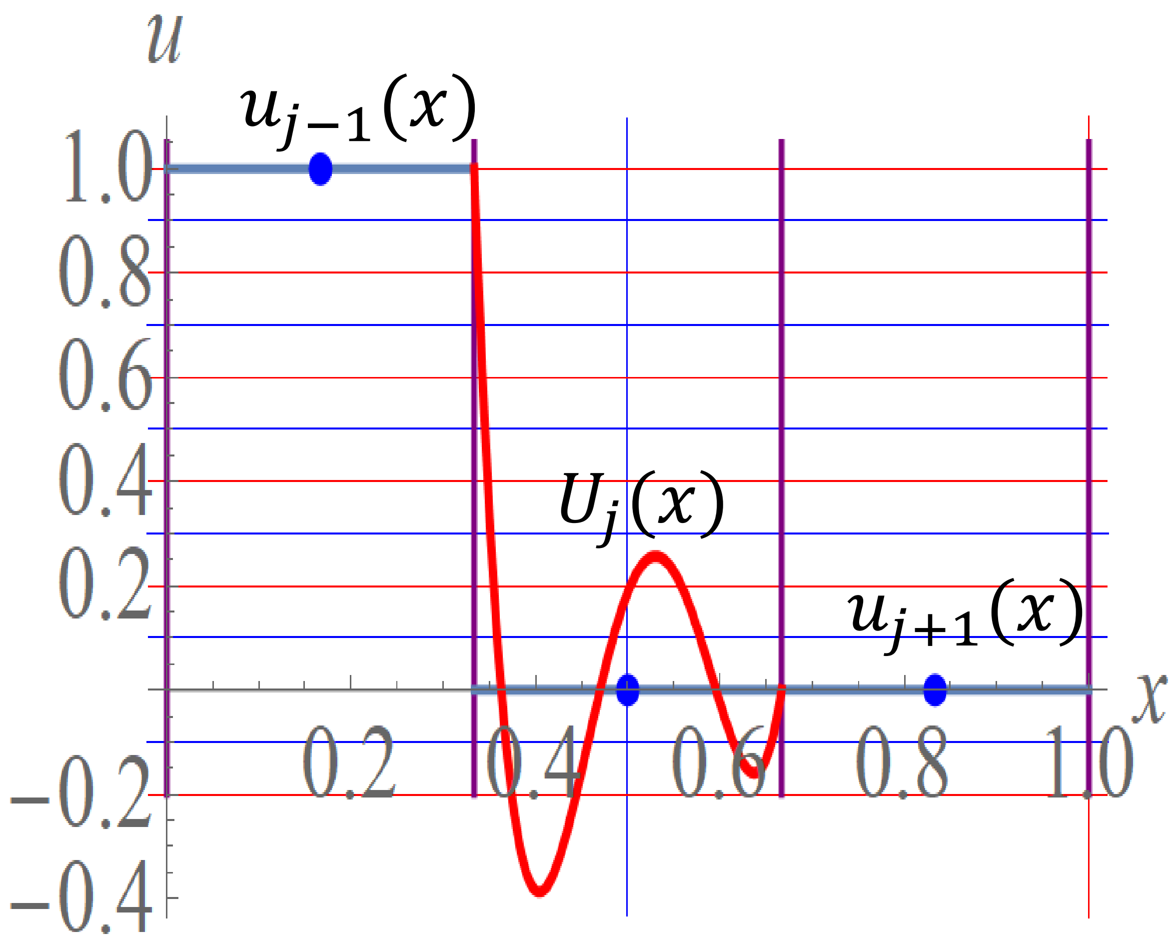
$$U_j = u_j + J_L R_{R,p+1}$$

Behavior of DG Method at a Discontinuity



$$u_t + a\, u_x = 0.$$

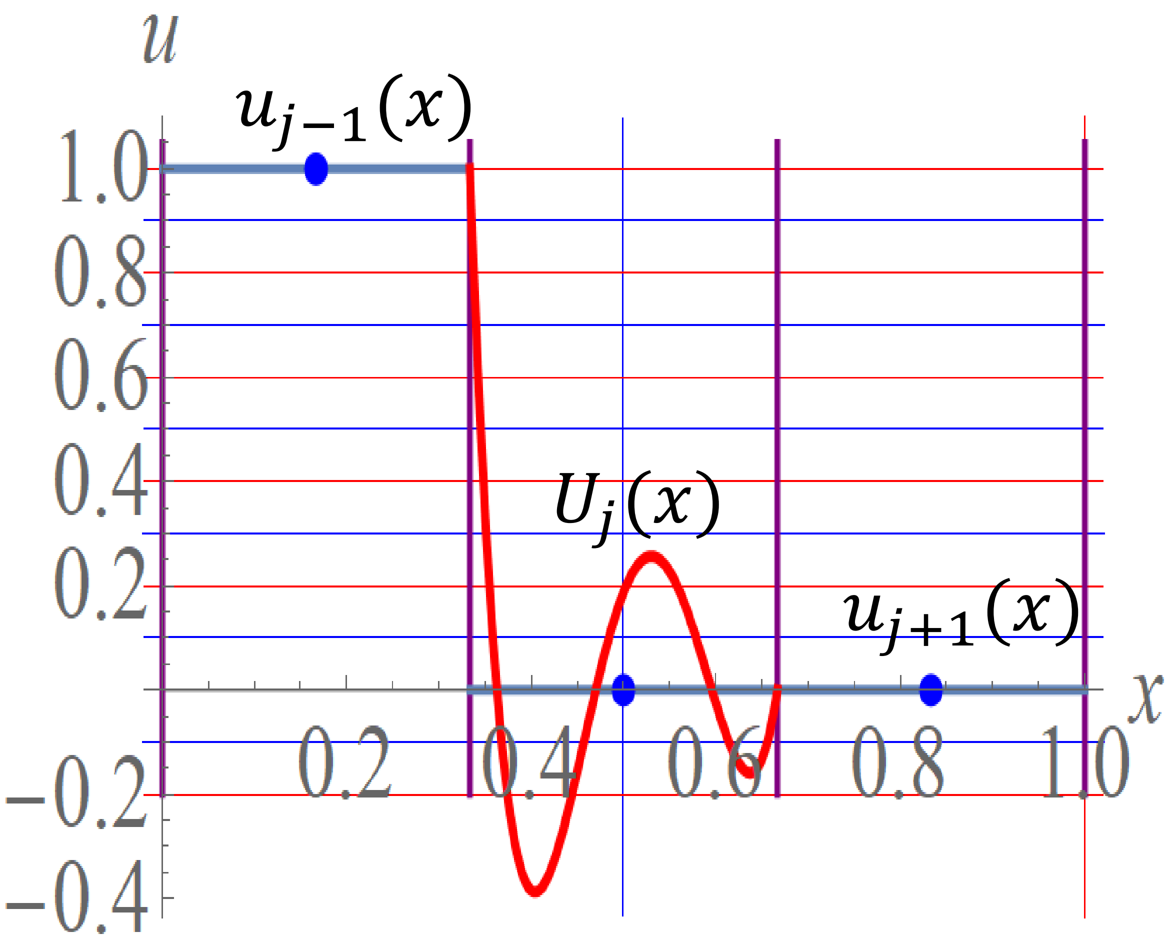
Behavior of **DG3** Method at a Discontinuity



For the case of a cubic solution, set

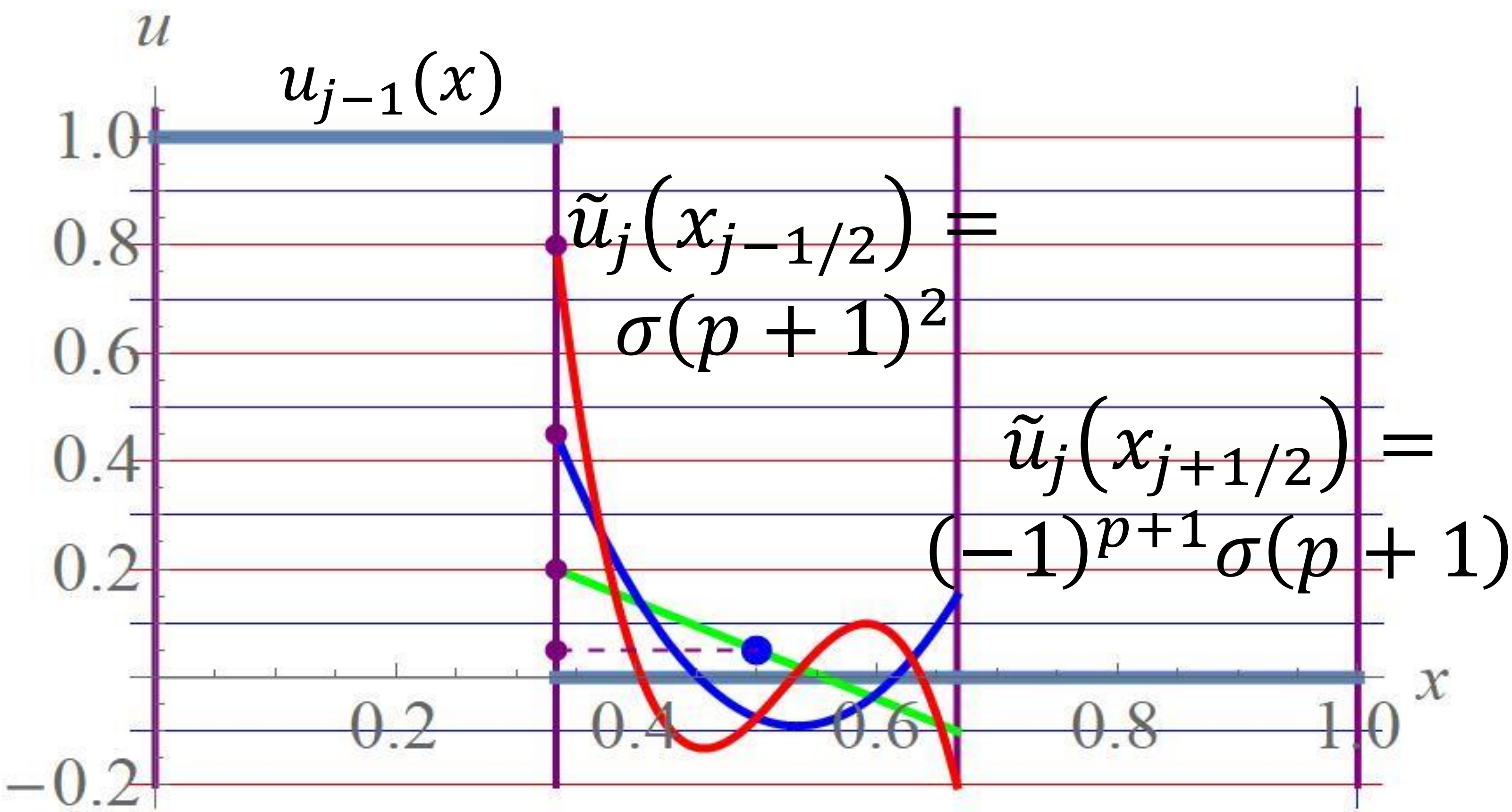
$$U_j = R_{R,4}.$$

Behavior of DG3 Method at a Discontinuity



Solution (Euler forward in time): $\tilde{u}_j = -2\sigma(R_{R,4})'$

Behavior of DG Method at a Discontinuity



$$\tilde{u}_j = -\sigma \left\{ \begin{matrix} -L_0 + 3L_1 - 5L_2 \ldots \\ +(-1)^{k+1}(2k+1)L_k \end{matrix} \right\}$$

Limiting

- The solution after an Euler forward time step (i.e., one stage of RK) is considered the data $u_{j,k}$ (might be oscillatory).
- We wish to impose constraints and reduce $|u_{j,k}|$, $k \geq 1$, to suppress oscillations (the cell average quantities $u_{j,0}$ should not be altered)

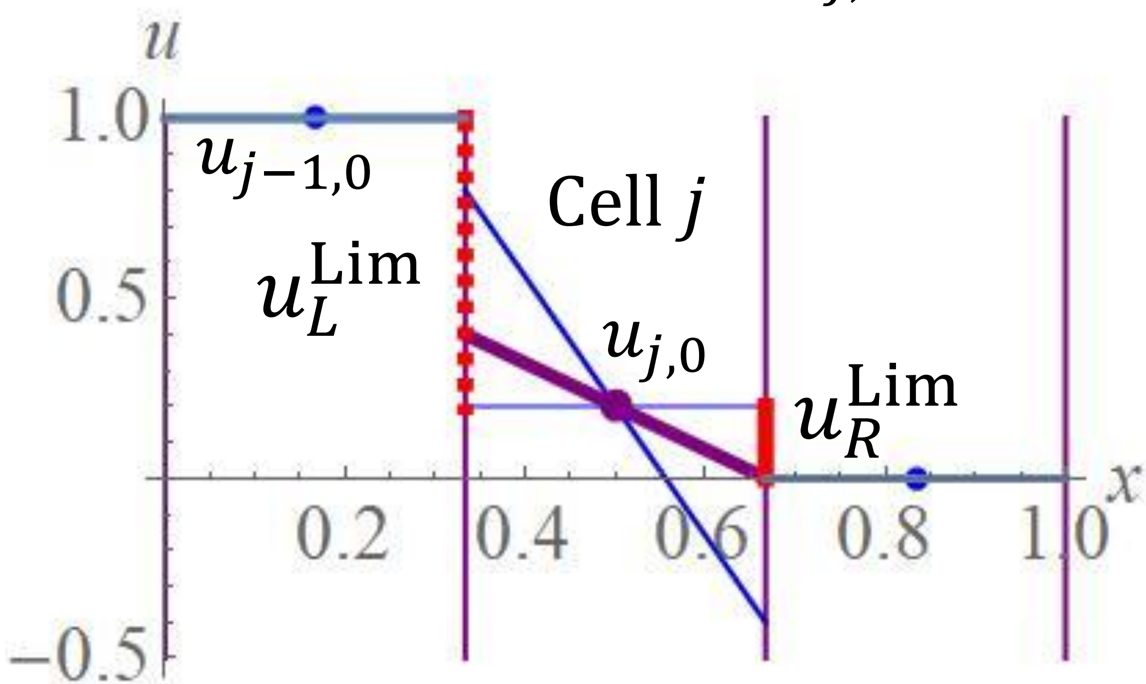
Limiting

For cell j, using the neighboring cell averages, set

$$u_L^{\text{Lim}} = \left|u_{j-1,0} - u_{j,0}\right|, \qquad u_R^{\text{Lim}} = \left|u_{j+1,0} - u_{j,0}\right|,$$

$$u^{\text{Lim}} = \min\left(u_L^{\text{Lim}}, u_R^{\text{Lim}}\right).$$

Constraint for **linear mode**: $u_{j,1} \le u^{\text{Lim}}$

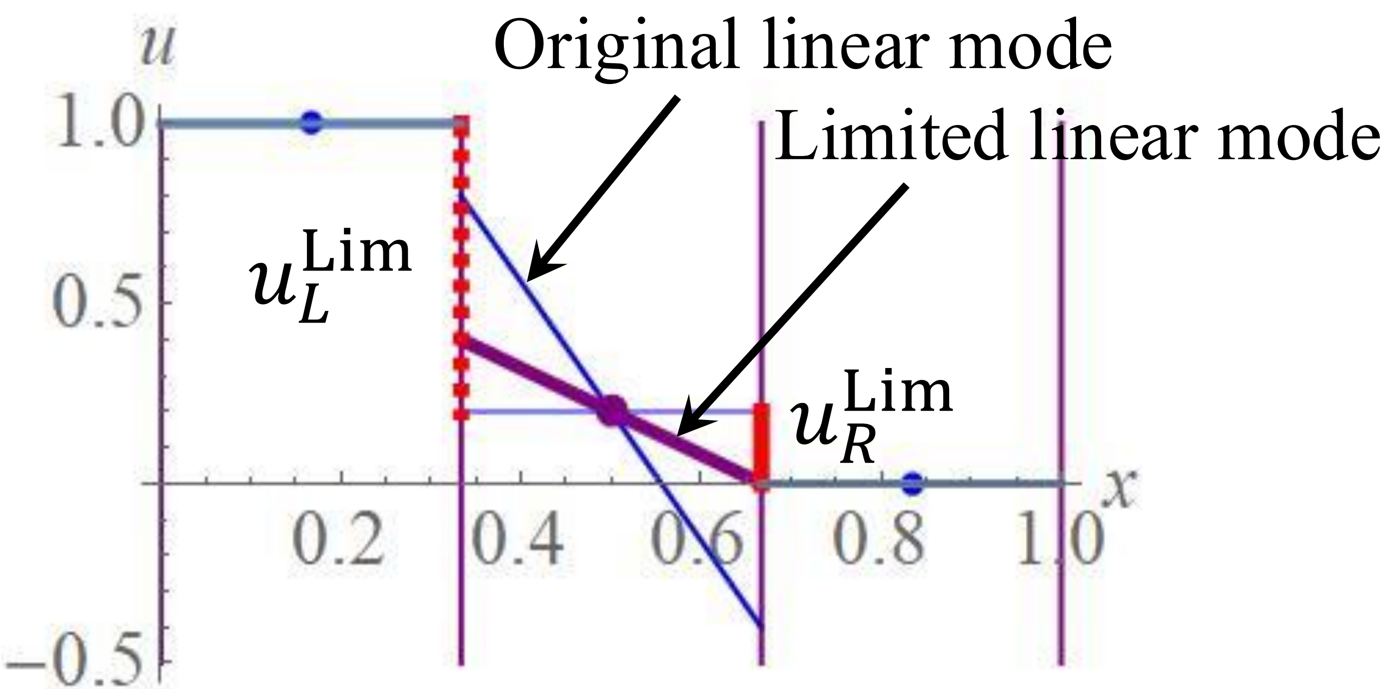


Limiting for Linear Mode

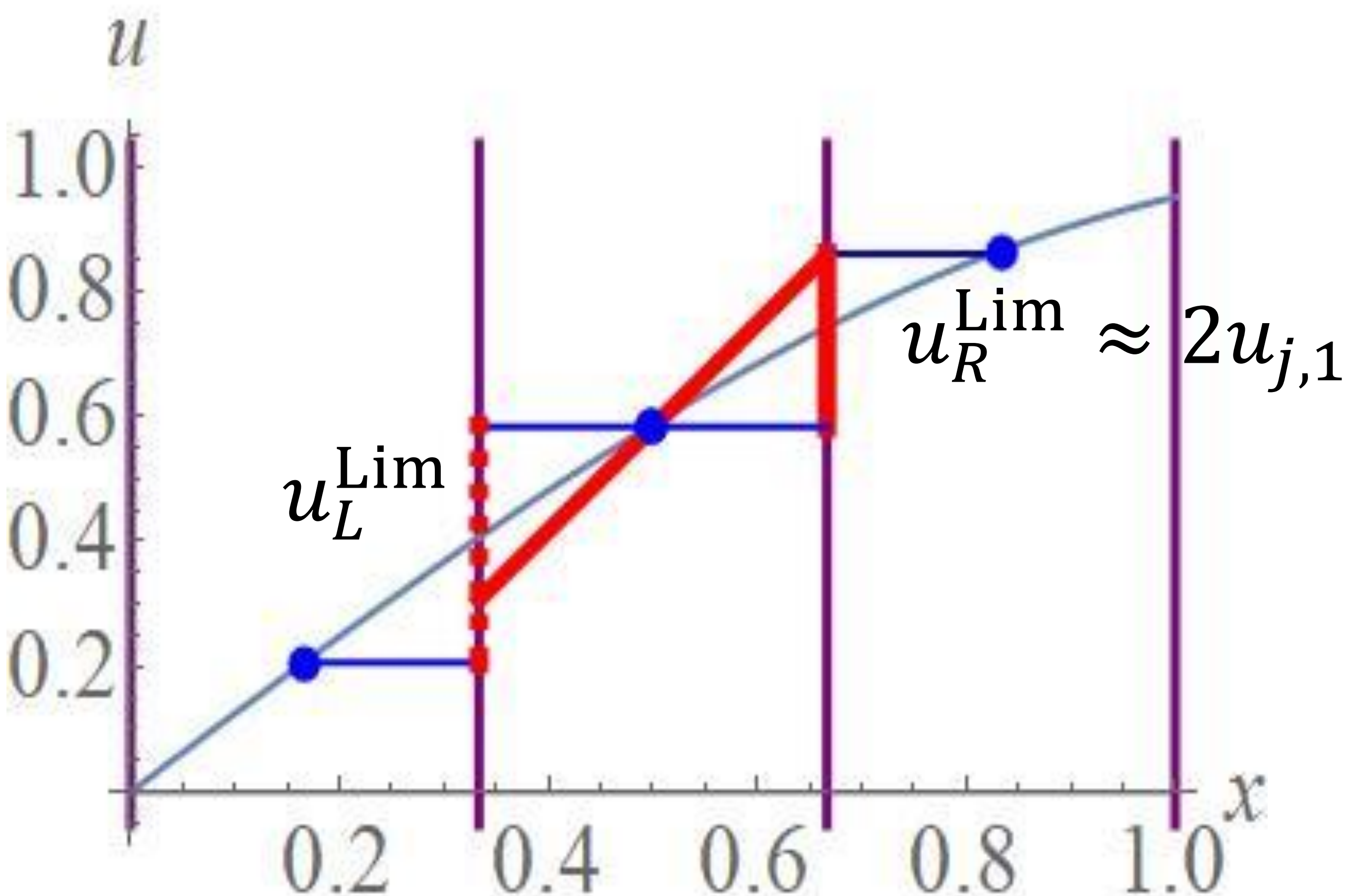
- With $u_1^{\text{Lim}} = u^{\text{Lim}} = \min\left(u_L^{\text{Lim}}, u_R^{\text{Lim}}\right)$, we require

$$\left|u_{j,1}\right| \le u_1^{\text{Lim}}.$$

Set $$u_{j,1}^{\text{Ltd}} = \operatorname{sign}\left(u_{j,1}\right) \min\left(\left|u_{j,1}\right|, u_1^{\text{Lim}}\right).$$

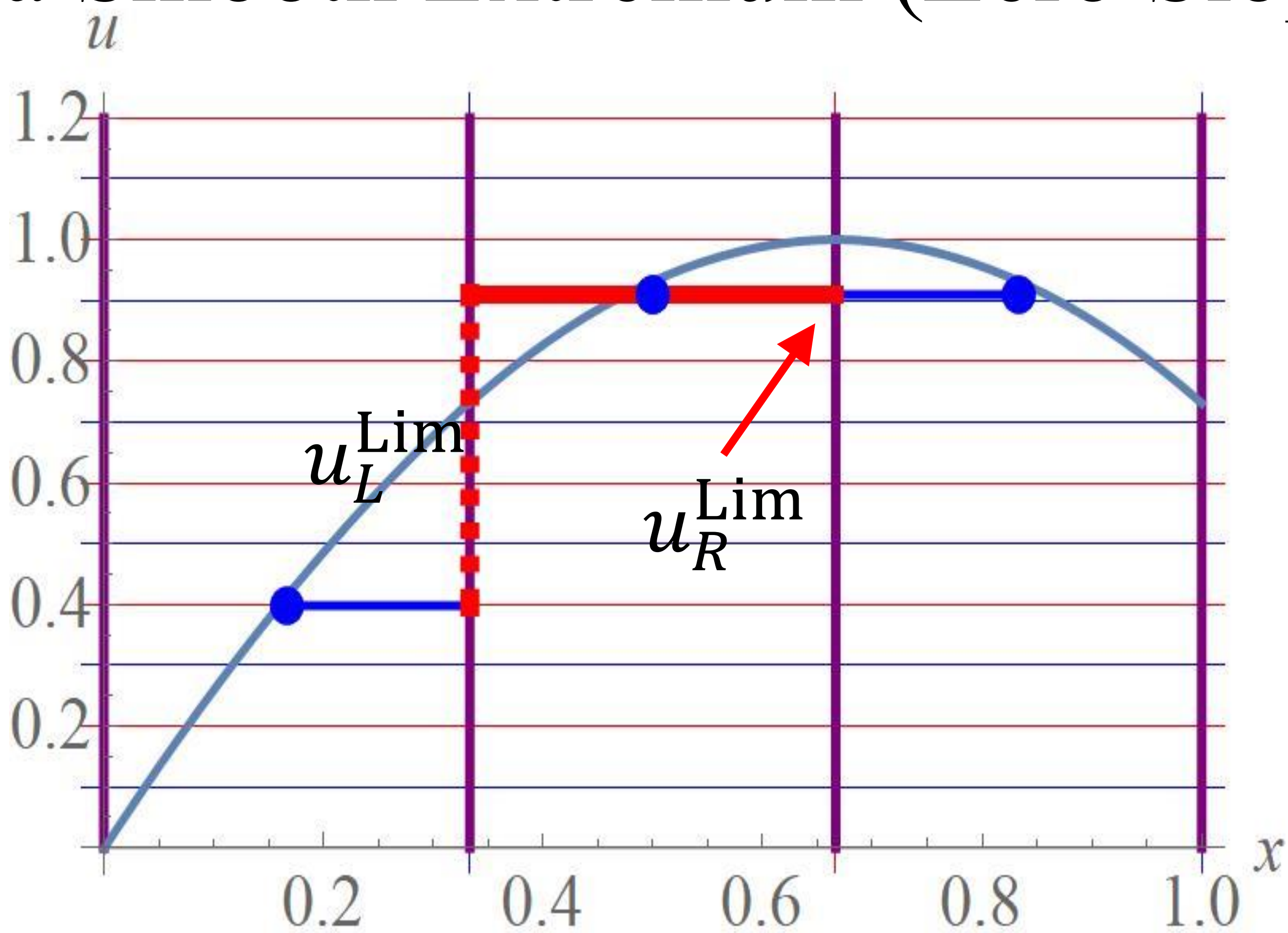


At a Smooth Region with Nonzero Slope



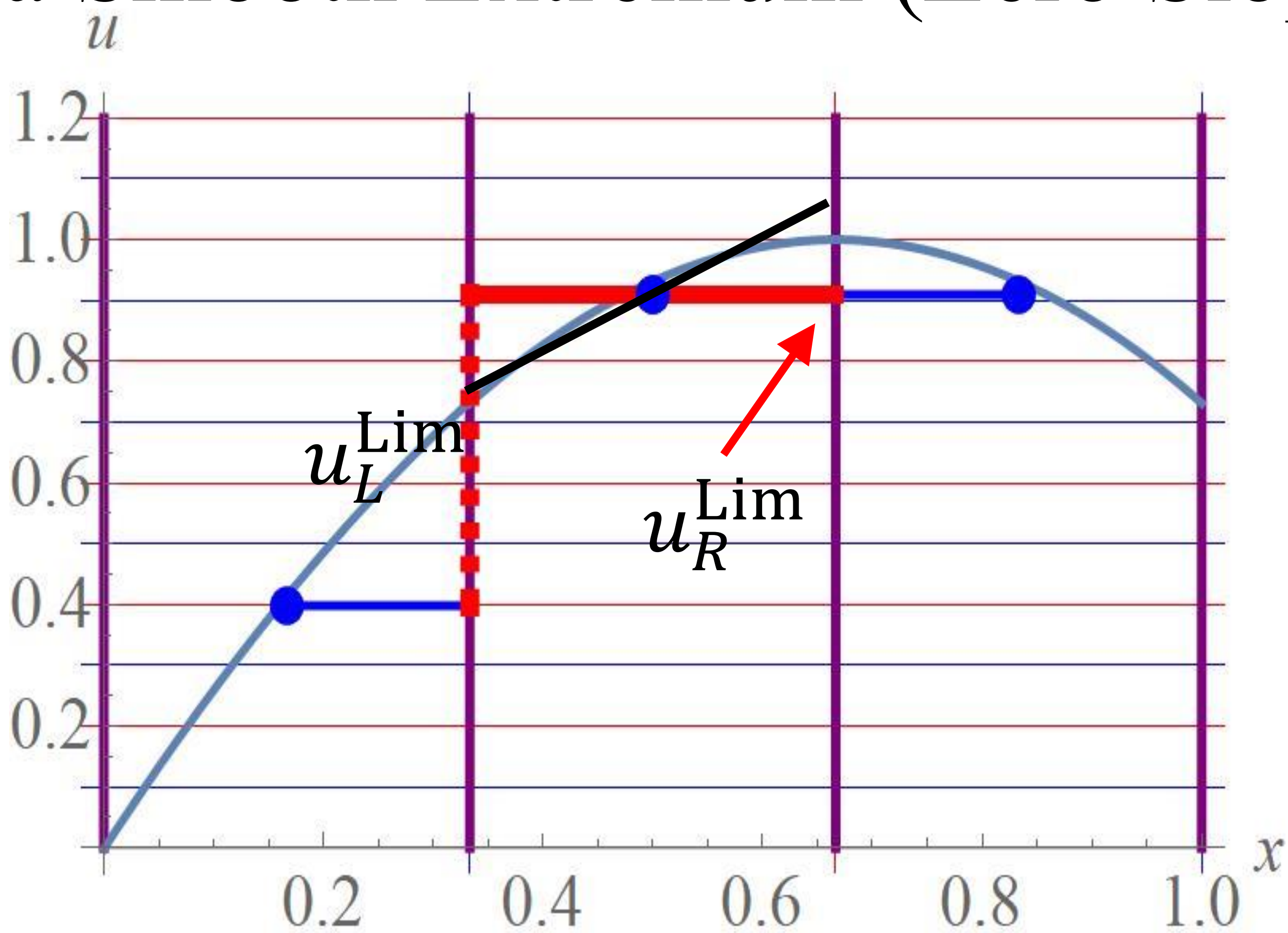
u^{Lim} provides lots of room, and leave the linear mode unchanged

At a Smooth Extremum (Zero Slope)



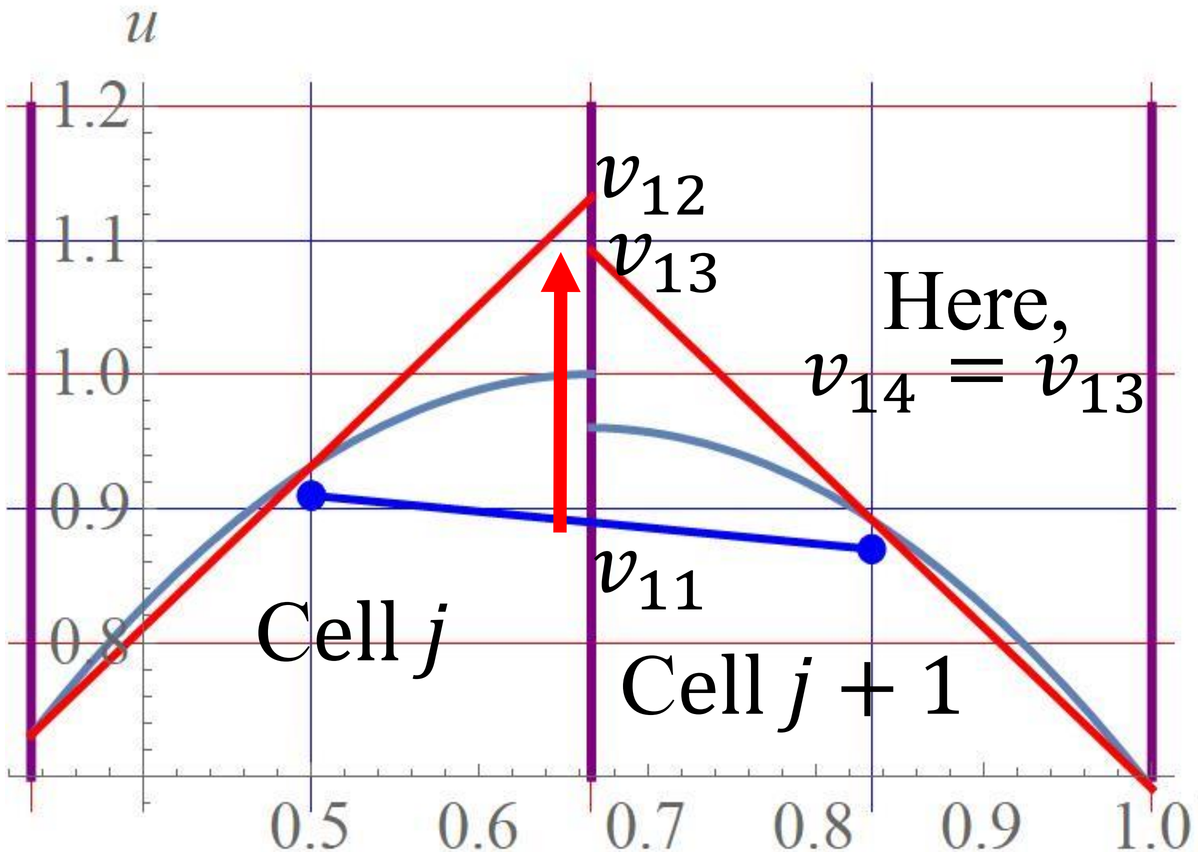
u_R^{Lim} provides no room

At a Smooth Extremum (Zero Slope)



u_R^{Lim} provides no room, and limiting flattens the data

Expand the Limit at a Smooth Extremum



$v_{11} = 0.5(u_{j,0} + u_{j+1,0}),$

$v_{12} = u_{j,0} + u_{j,1} - 2u_{j,2}$

$v_{13} = u_{j+1,0} - u_{j+1,1} - 2u_{j+1,2}$

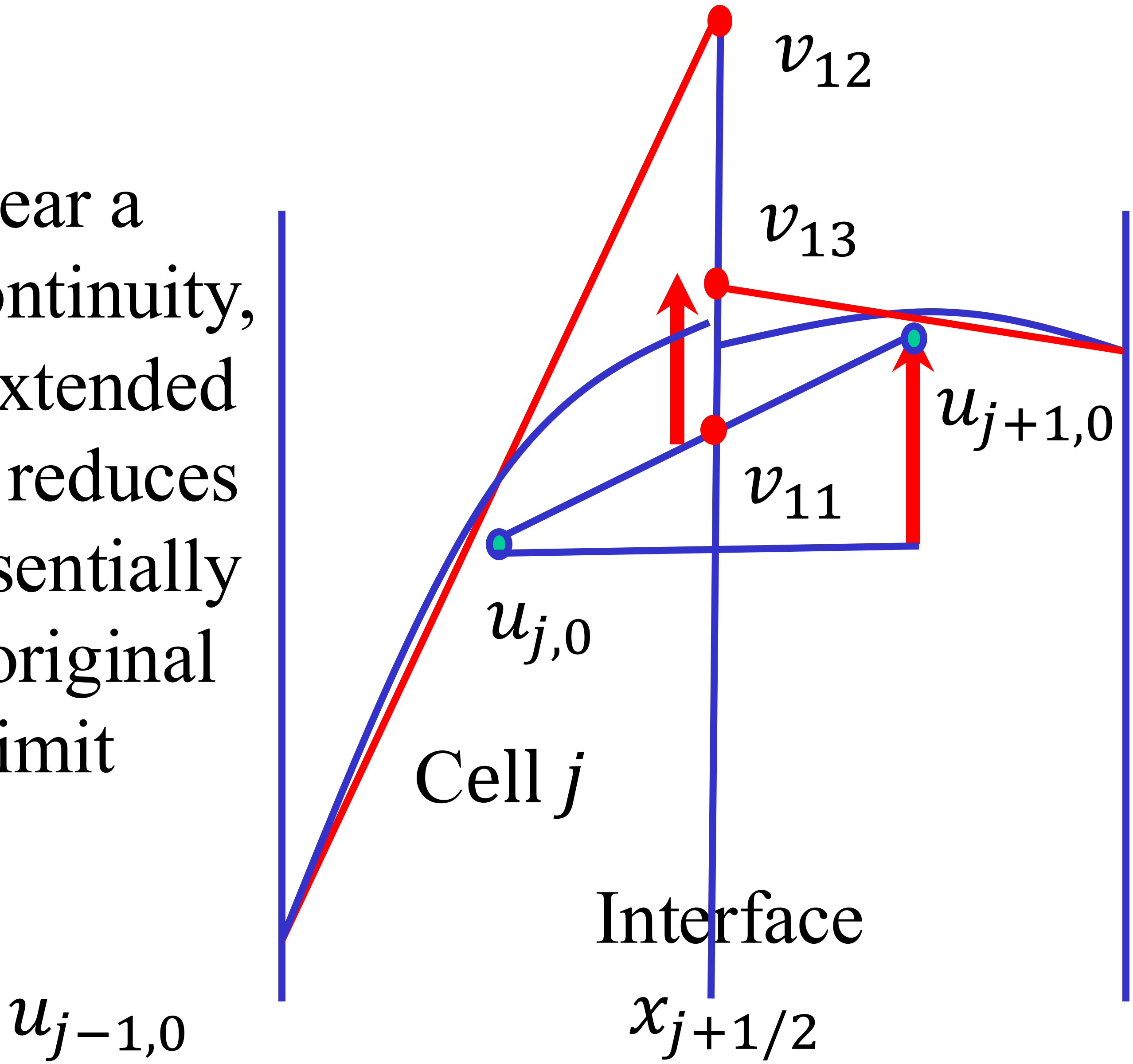
$v_{14} =$ median(v_{11}, v_{12}, v_{13})

At a Smooth Extremum (Slope near Zero)

$$u_R^{\mathrm{XLim}} = \mathrm{Max}(|u_{j+1,0} - u_{j,0}|, |v_{14} - u_{j,0}|)$$

Near a Discontinuity

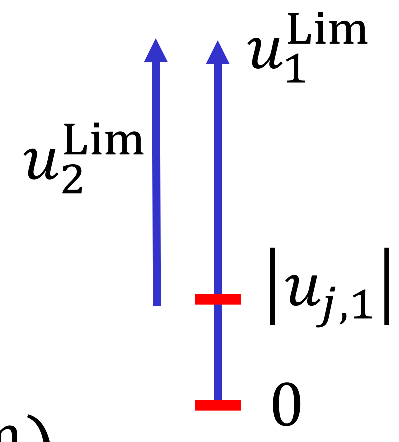
Near a Discontinuity, the extended limit reduces to essentially the original limit



Limiting for All Modes

- For all modes, we require

$$\sum_{k=1}^{p} |u_{j,k}| \leq u^{\text{Lim}} = u_1^{\text{Lim}}.$$



$$u_{j,1}^{\text{Ltd}} = \text{sign}(u_{j,1}) \min(|u_{j,1}|, u_1^{\text{Lim}}).$$

For $k \geq 2$, set

$$u_k^{\text{Lim}} = u_{k-1}^{\text{Lim}} - |u_{j,k-1}^{\text{Ltd}}|$$

And

$$u_{j,k}^{\text{Ltd}} = \text{sign}(u_{j,k}) \min(|u_{j,k}|, u_k^{\text{Lim}}).$$

Limiting Algorithm

$$\begin{cases} v_{11} = \frac{1}{2}\left(u_{j,0} + u_{j+1,0}\right); \quad v_{12} = u_{j,0} + u_{j,1} - 2u_{j,2}; \\ v_{13} = u_{j+1,0} - u_{j+1,1} - 2u_{j+1,2}; \; v_{14} = \text{median}(v_{11}, v_{12}, v_{13}); \\ u_R^{\text{XLim}} = \text{Max}(|u_{j+1,0} - u_{j,0}|, |v_{14} - u_{j,0}|); \end{cases}$$

$$\begin{cases} v_{21} = \frac{1}{2}\left(u_{j,0} + u_{j-1,0}\right); \quad v_{22} = u_{j,0} - u_{j,1} - 2u_{j,2}; \\ v_{23} = u_{j-1,0} + u_{j-1,1} - 2u_{j+1,2}; \; v_{24} = \text{median}(v_{21}, v_{22}, v_{23}); \\ u_L^{\text{XLim}} = \text{Max}(|u_{j-1,0} - u_{j,0}|, |v_{24} - u_{j,0}|); \end{cases}$$

$$u_1^{\text{Lim}} = u^{\text{XLim}} = \min\left(u_L^{\text{XLim}}, u_R^{\text{XLim}}\right);$$

$$\begin{cases} \text{For } k = 1, \dots, p, \quad \text{set } u_{j,k}^{\text{Ltd}} = \text{sign}\left(u_{j,k}\right) \min\left(\left|u_{j,k}\right|, u_k^{\text{Lim}}\right). \\ u_{k+1}^{\text{Lim}} = u_k^{\text{Lim}} - \left|u_{j,k}^{\text{Ltd}}\right| \end{cases}$$

Detecting "Good Cells"

- **Gap Criteria**

Define quadratic content in cell j, $c_j^Q = |u_{j,1}| + |u_{j,2}|$

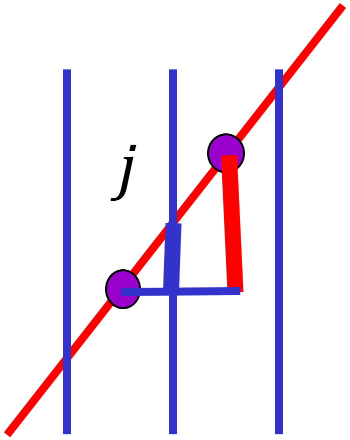
$$\text{Max}\left[\, |u_{j-1}(1) - u_j(-1)|, |u_{j+1}(-1) - u_j(1)| \,\right] \le \frac{1}{5} c_j^Q + 10^{-3}.$$

- **Curvature Criteria (Second Mode)**

$$\frac{4}{5} \le \frac{u_{j-1,2}}{u_{j,2}} \le \frac{5}{4} \quad \text{and} \quad \frac{4}{5} \le \frac{u_{j+1,2}}{u_{j,2}} \le \frac{5}{4}\,.$$

- **Slope Criteria** $(u_{j,1} \approx \frac{1}{2}\, u^{\text{Lim}})$

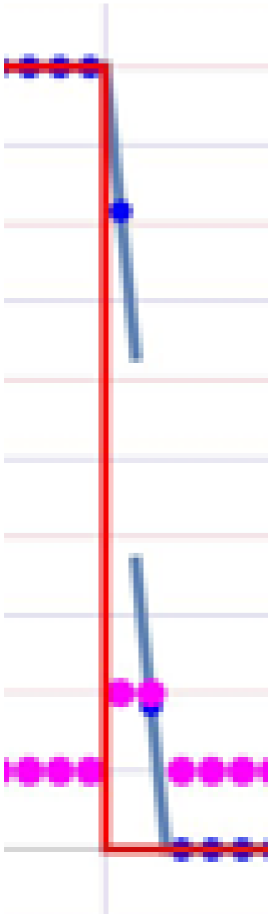
$$|u_{j,1}| \le \frac{3}{4}\, u^{\text{Lim}} + 10^{-4}.$$



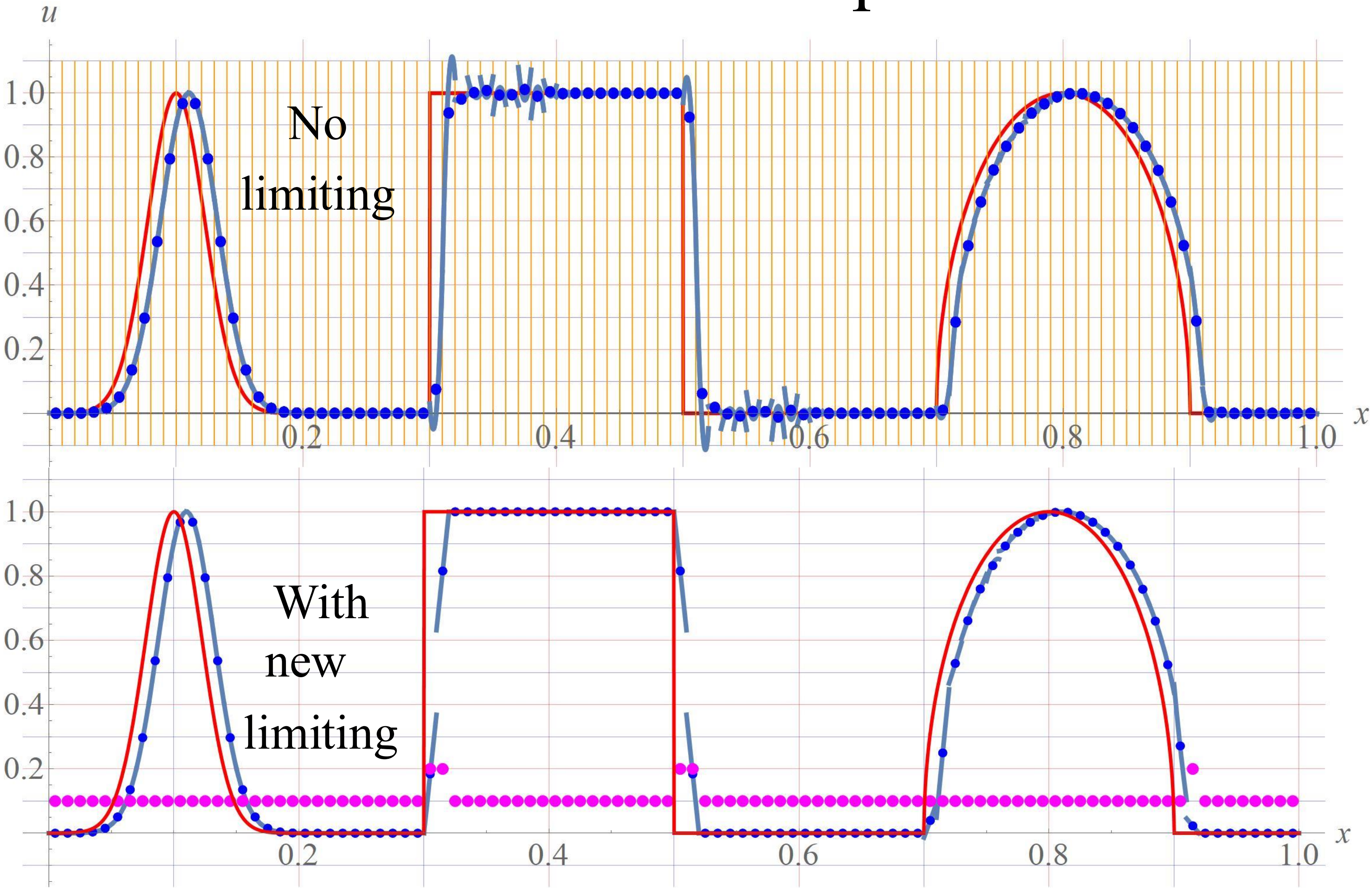
Numerical Examples

In all numerical examples, the dots represent the cell average values, and the curves represent the polynomial solution.

The magenta dots represent either "good cells" where no limiting is required, or "troubled cells" where limiting is needed (when they are bumped up).

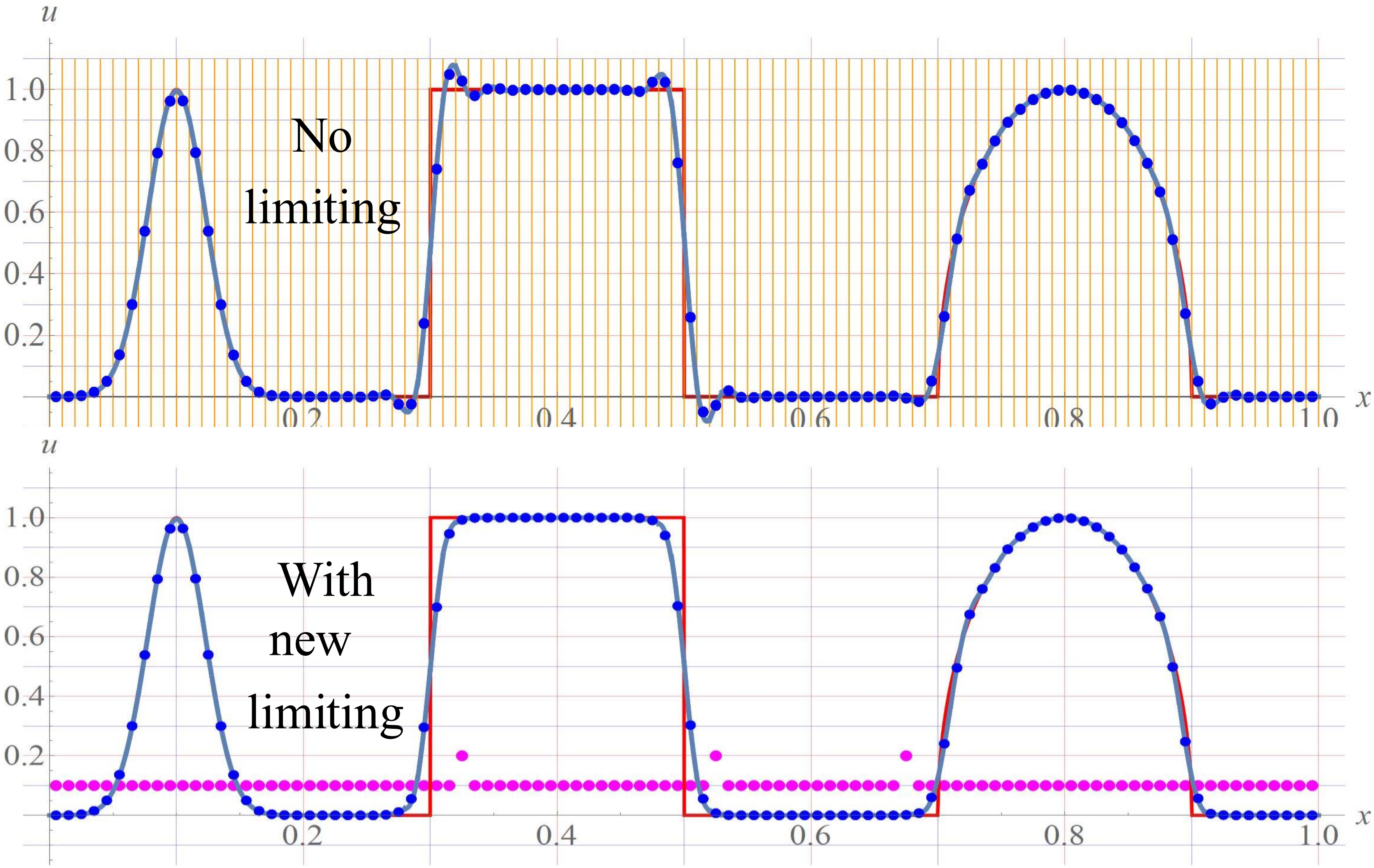


Numerical Examples



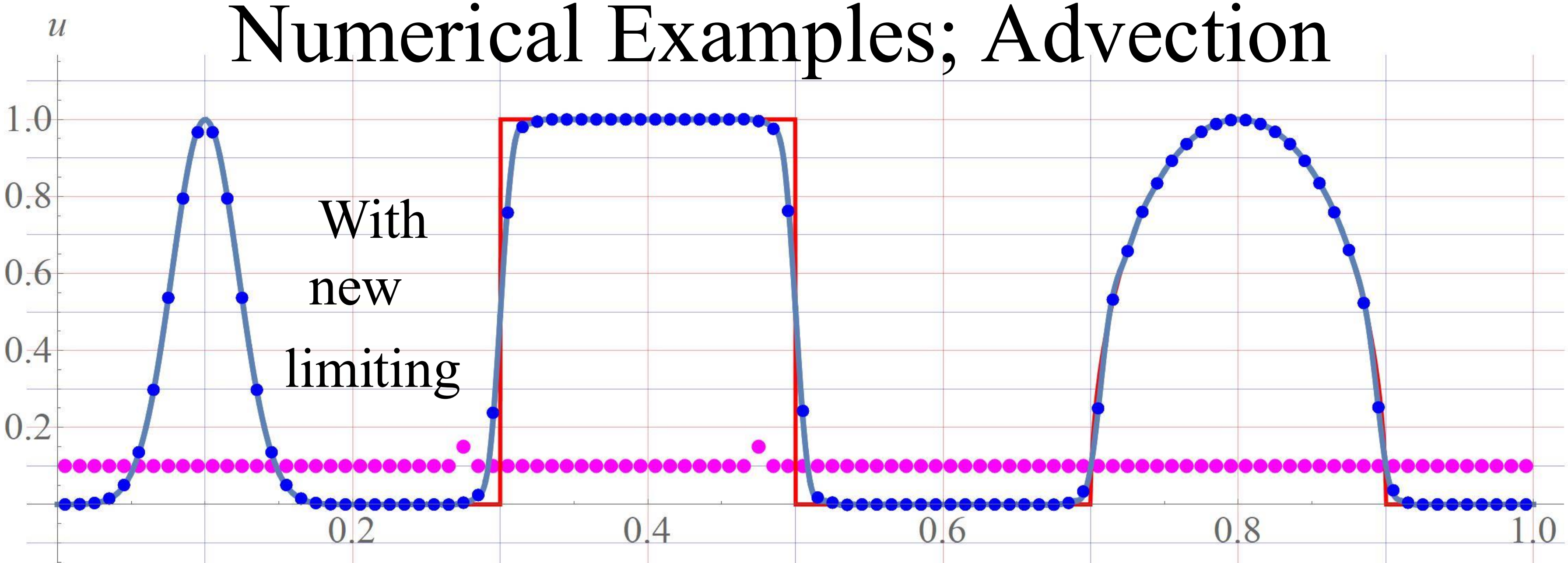
DG2 solution, $\sigma = 0.20$, after 5 time steps, distance 1 cell width

Numerical Examples

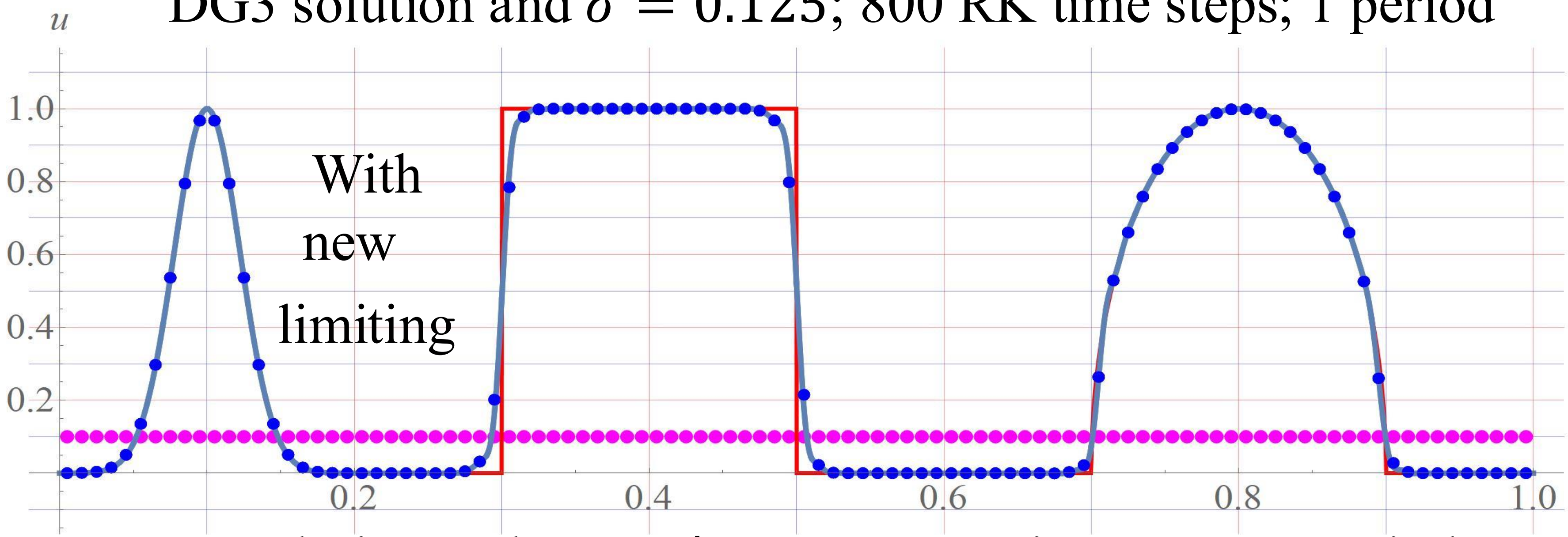


DG2 solution, $\sigma = 0.2$, 500 RK time steps, 1 period

Numerical Examples; Advection

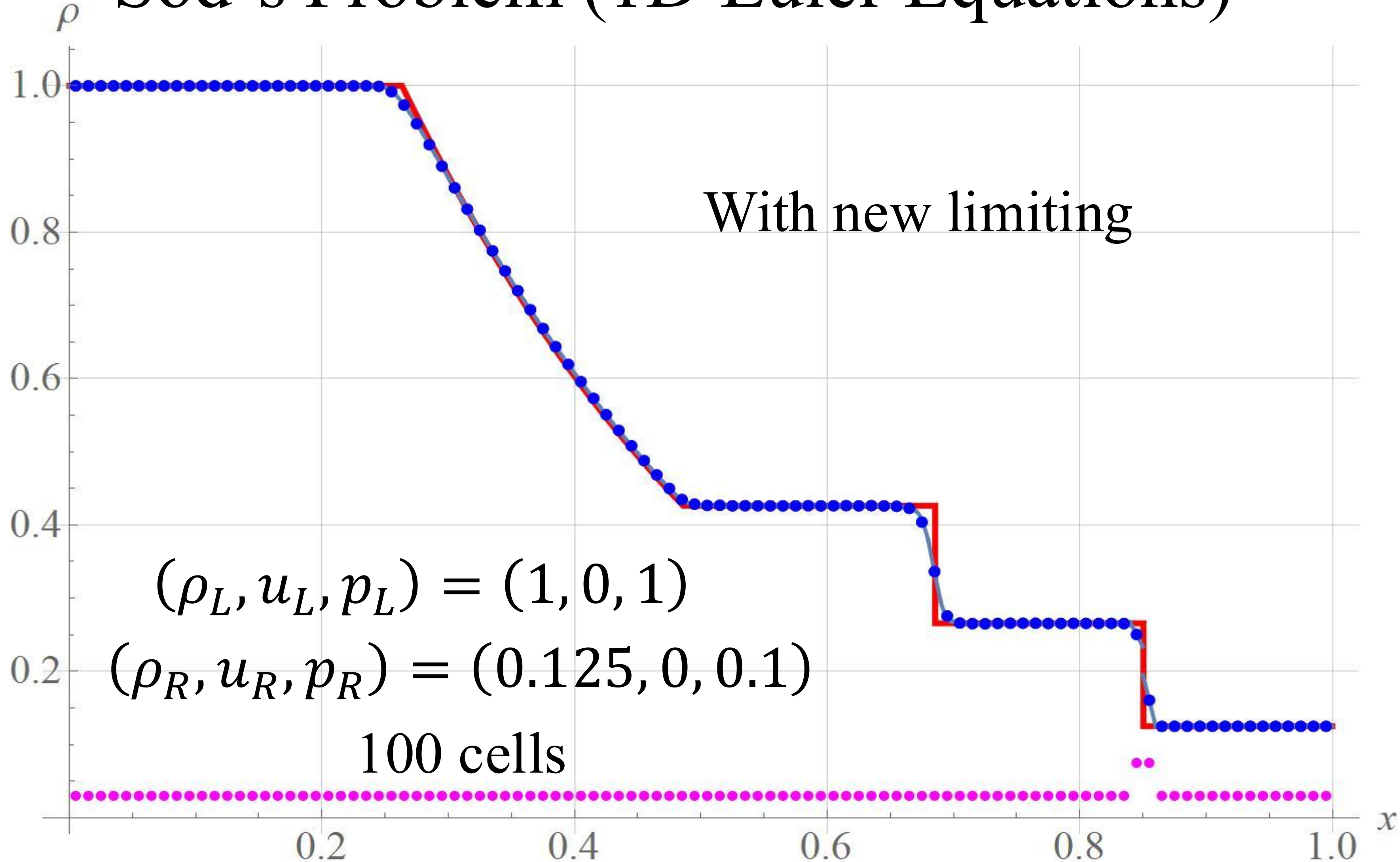


DG3 solution and $\sigma = 0.125$; 800 RK time steps; 1 period



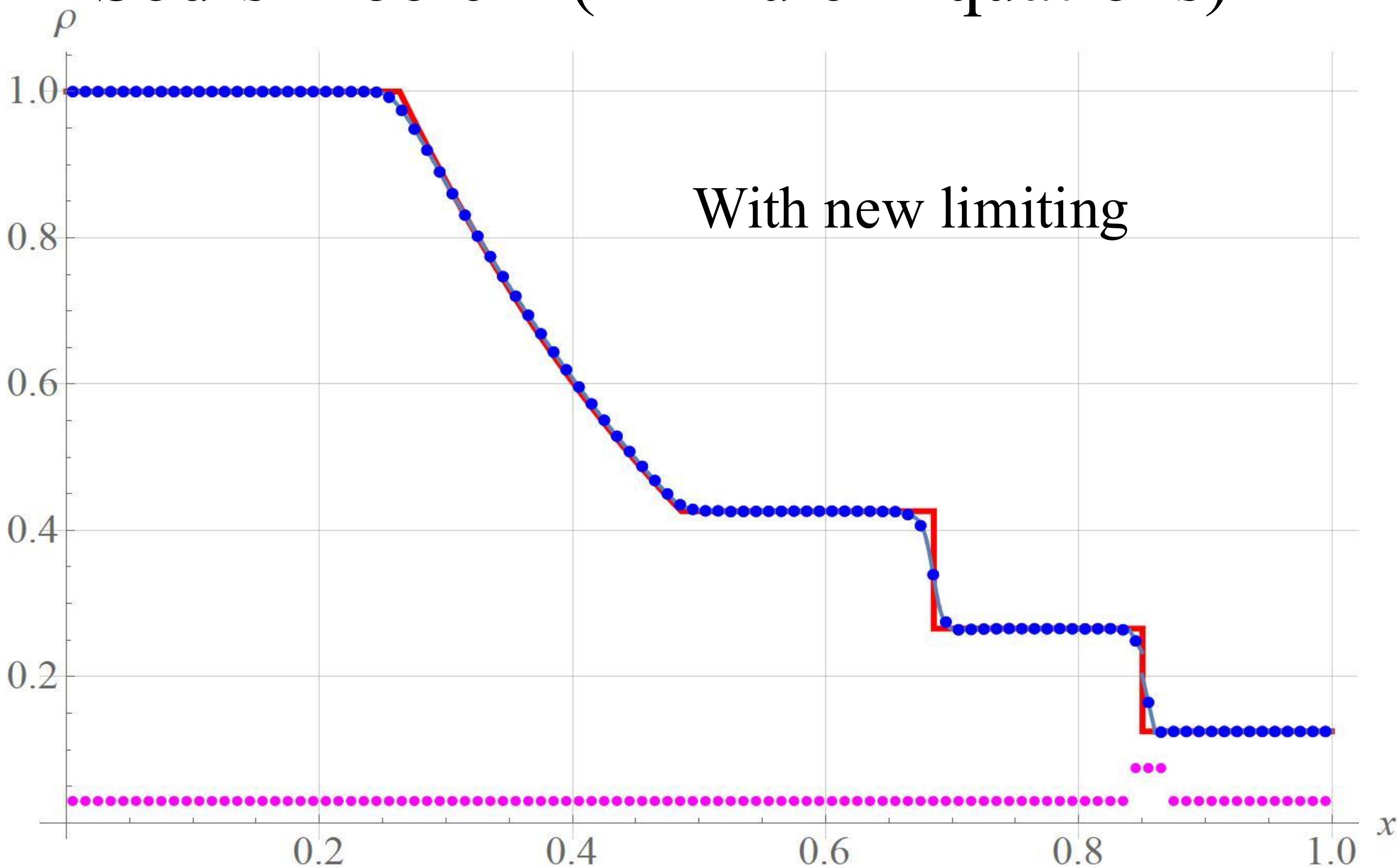
DG4 solution and $\sigma = 1/12$; 1200 RK time steps; 1 period

Sod's Problem (1D Euler Equations)



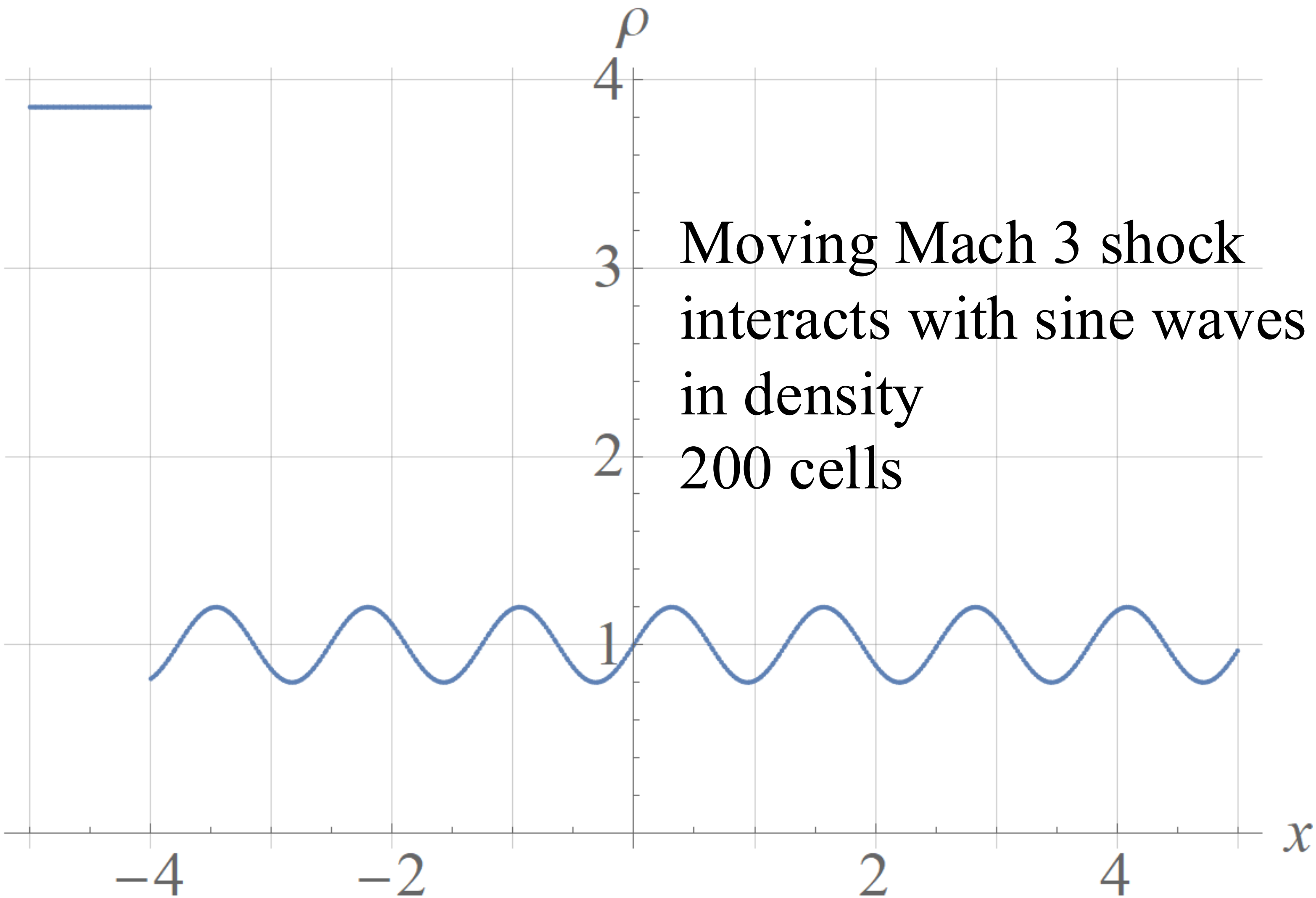
DG2 with $\Delta t = 0.001$, 200 time steps, final time 0.2

Sod's Problem (1D Euler Equations)



DG3 solution, $\Delta t = 0.000625$, 320 time steps

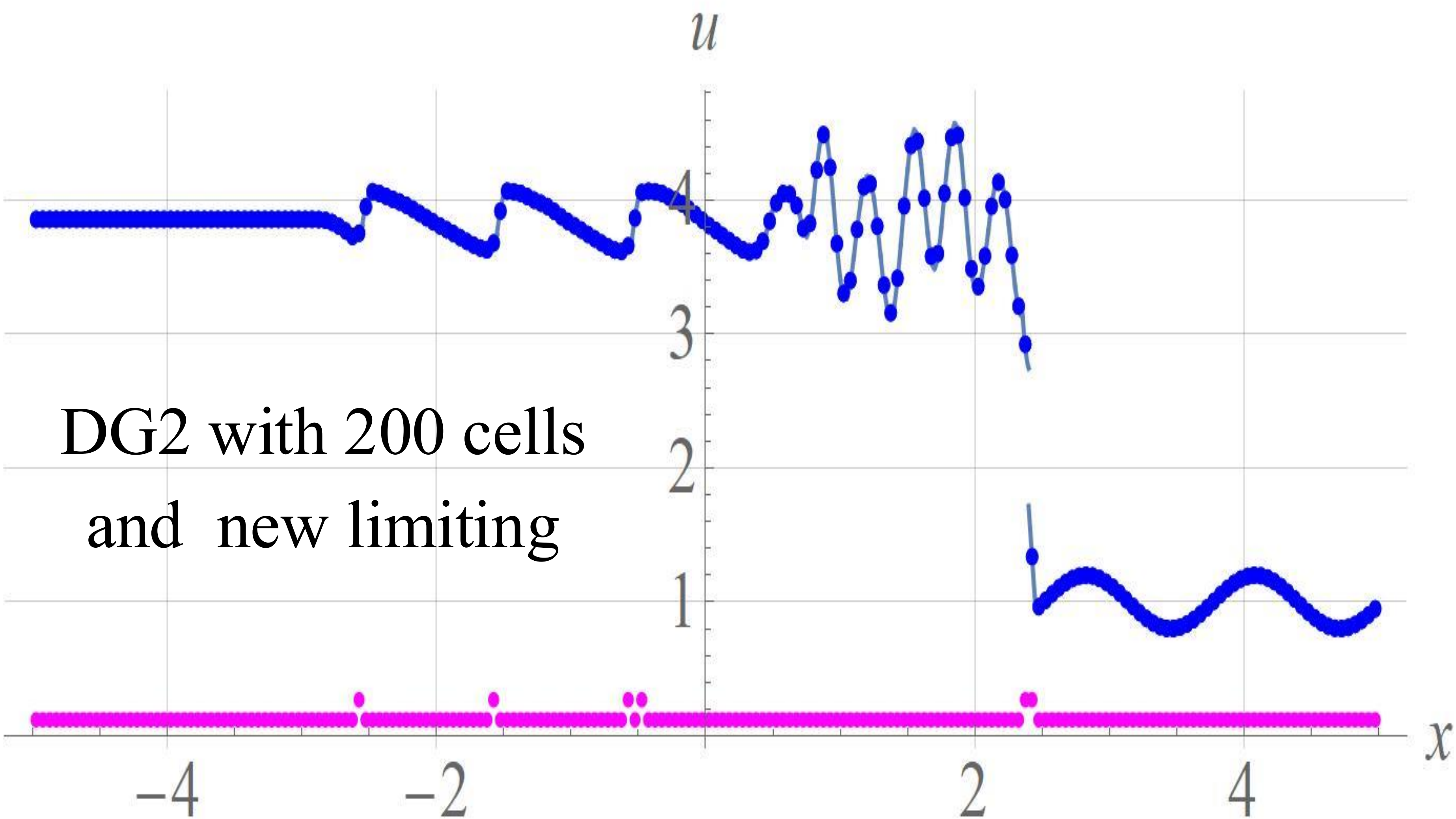
Shu's Problem (1D Euler Equations)



Moving Mach 3 shock
interacts with sine waves
in density
200 cells

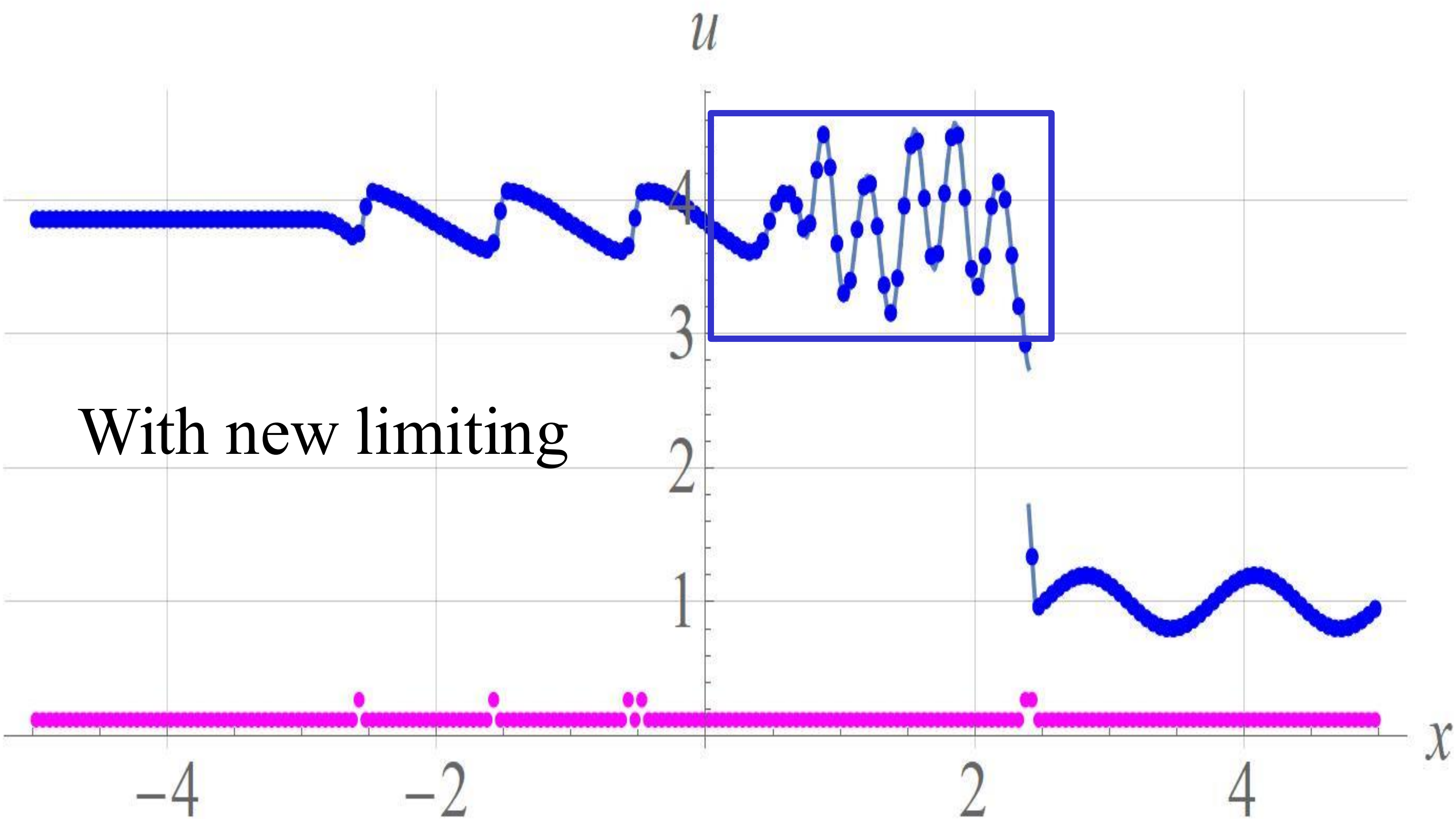
Shu's Problem (1D Euler Equations)

DG2 with 200 cells
and new limiting



DG2 after 860 time steps, each of size 0.002093

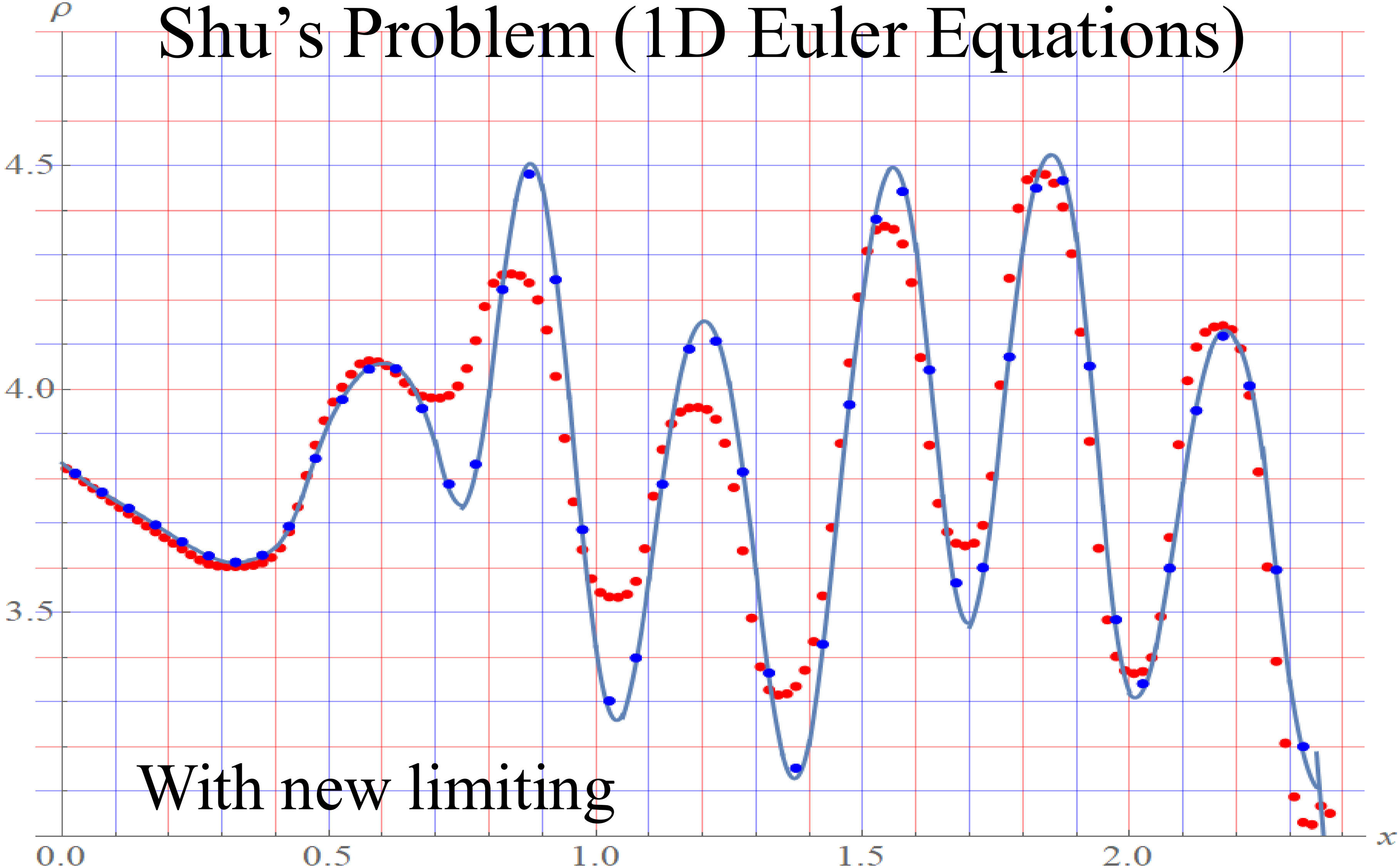
Shu's Problem (1D Euler Equations)



With new limiting

DG2 after 860 time steps, each of size 0.002093

Shu’s Problem (1D Euler Equations)



Close-up of DG2 solution. Red dots: Standard Second-order Shock-capturing solution

Summary

- Discussed oscillations caused by a discontinuity using the Radau polynomial
- Presented (a) a shock capturing method that suppresses oscillations near shocks and maintains accuracy near extrema and (b) a detection technique
- Numerical examples demonstrating the effectiveness of the new method

Future Research

Extend to 2D quad mesh as well as triangular mesh

Thank you