

A simple goal-oriented error estimator for flux-limited Galerkin schemes

Dmitri Kuzmin¹ and Matthias Möller²

¹Department of Mathematics
University of Houston, USA
kuzmin@math.uh.edu

²Institute of Applied Mathematics, LS III
Dortmund University of Technology, Germany
matthias.moeller@math.tu-dortmund.de

Hirschegg Workshop on Conservation Laws 2009

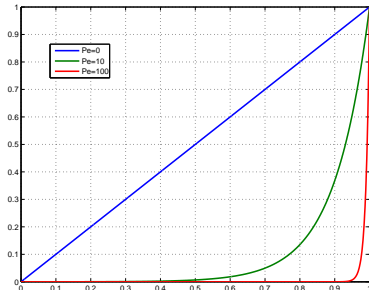
- 1 Goal-oriented error estimation
 - A general review of the duality argument
 - Error estimates for steady transport problems
- 2 Mesh adaptation algorithm
 - Refinement, recoarsening, data structures
- 3 Conclusions & Outlook

Convection-diffusion in 1D

$$\begin{cases} \text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0 & \text{in } (0, 1) \\ u(0) = 0, \quad u(1) = 1 \end{cases}$$

$$\text{Pe} > 0, \quad u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}$$

$$x_i = ih, \quad u_i \approx u(x_i)$$

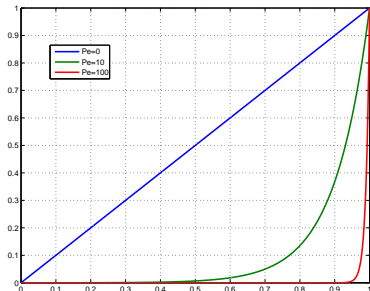


Convection-diffusion in 1D

$$\begin{cases} \text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0 & \text{in } (0, 1) \\ u(0) = 0, \quad u(1) = 1 \end{cases}$$

$$\text{Pe} > 0, \quad u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}$$

$$x_i = ih, \quad u_i \approx u(x_i)$$



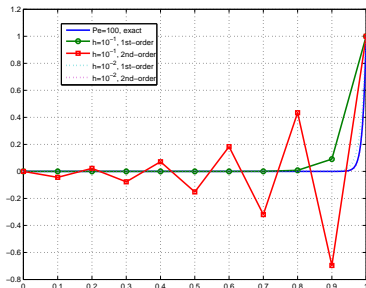
- Exact solution: nonnegativity, no internal maxima or minima

Convection-diffusion in 1D

$$\begin{cases} \text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0 & \text{in } (0, 1) \\ u(0) = 0, \quad u(1) = 1 \end{cases}$$

$$\text{Pe} > 0, \quad u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}$$

$$x_i = ih, \quad u_i \approx u(x_i)$$



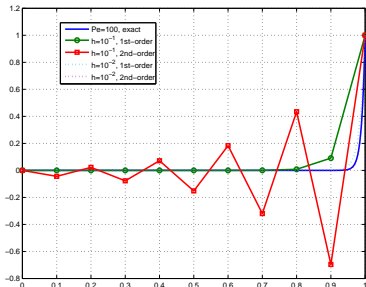
- Exact solution: nonnegativity, no internal maxima or minima
- Numerical approximations: **numerical diffusion** ↔ **spurious wiggles**

Convection-diffusion in 1D

$$\begin{cases} \text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0 & \text{in } (0, 1) \\ u(0) = 0, \quad u(1) = 1 \end{cases}$$

$$\text{Pe} > 0, \quad u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}$$

$$x_i = ih, \quad u_i \approx u(x_i)$$



- Exact solution: nonnegativity, no internal maxima or minima
- Numerical approximations: numerical diffusion \leftrightarrow spurious wiggles
- Way out: nonlinear combination of high- and low-order methods

Remarks on flux limiting schemes

- *Don't suppress the wiggles - They're telling you something.*

Gresho, Lee IJNMF 1979.

Remarks on flux limiting schemes

- *Don't suppress the wiggles - They're telling you something.*

Gresho, Lee IJNMF 1979.

- Detection of underresolved regions at no additional cost

- mesh enrichment/deformation controlled by numerical viscosity

[Palmero et al. RR-1175, 1990]

- grid refinement based on correction factors [Geppert 1996, thesis]

Remarks on flux limiting schemes

- *Don't suppress the wiggles - They're telling you something.*

Gresho, Lee IJNMF 1979.

- Detection of underresolved regions at no additional cost

- mesh enrichment/deformation controlled by numerical viscosity

[Palmero et al. RR-1175, 1990]

- grid refinement based on correction factors [Geppert 1996, thesis]

- No error quantification/no information about error propagation

Remarks on flux limiting schemes

- *Don't suppress the wiggles - They're telling you something.*

Gresho, Lee IJNMF 1979.

- Detection of underresolved regions at no additional cost

- mesh enrichment/deformation controlled by numerical viscosity

[Palmero et al. RR-1175, 1990]

- grid refinement based on correction factors [Geppert 1996, thesis]

- No error quantification/no information about error propagation

- Violation of the Galerkin orthogonality property

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$u_h \approx u, \quad \rho(w, u_h) = b(w) - a(w, u_h), \quad \rho(w, u) = 0$$

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$u_h \approx u, \quad \rho(w, u_h) = b(w) - a(w, u_h), \quad \rho(w, u) = 0$$

■ Galerkin orthogonality $\rho(w_h, u_h) = 0, \quad \forall w_h \in V_h$

$$\text{Primal problem} \quad a(w, u) = b(w), \quad \forall w \in V$$

$$u_h \approx u, \quad \rho(w, u_h) = b(w) - a(w, u_h), \quad \rho(w, u) = 0$$

- Linear target functional $j(u)$, mean/point value, boundary flux

$$\text{Dual problem} \quad a(z, e) = j(e), \quad \forall e \in V$$

$$e = u - u_h, \quad j(e) = a(z, u) - a(z, u_h) = \rho(z, u_h)$$

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$u_h \approx u, \quad \rho(w, u_h) = b(w) - a(w, u_h), \quad \rho(w, u) = 0$$

- Linear target functional $j(u)$, mean/point value, boundary flux

Dual problem $a(z, e) = j(e), \quad \forall e \in V$

$$e = u - u_h, \quad j(e) = a(z, u) - a(z, u_h) = \rho(z, u_h)$$

- Error representation $j(e) = \rho(z - z_h, u_h) + \underbrace{\rho(z_h, u_h)}_{\text{computable}}, \quad z_h \approx z$

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$u_h \approx u, \quad \rho(w, u_h) = b(w) - a(w, u_h), \quad \rho(w, u) = 0$$

- Linear target functional $j(u)$, mean/point value, boundary flux

Dual problem $a(z, e) = j(e), \quad \forall e \in V$

$$e = u - u_h, \quad j(e) = a(z, u) - a(z, u_h) = \rho(z, u_h)$$

- Error representation $j(e) \approx \underbrace{\rho(\hat{z} - z_h, u_h)}_{\text{small !?}} + \underbrace{\rho(z_h, u_h)}_{\text{computable}}, \quad \hat{z} \approx z$

Primal problem

$$\begin{aligned}\nabla \cdot (\mathbf{v}u) &= s && \text{in } \Omega \\ u &= u_D && \text{on } \Gamma_{\text{in}}\end{aligned}$$

Dual problem

$$\begin{aligned}-\mathbf{v} \cdot \nabla z &= j && \text{in } \Omega \\ z &= h && \text{on } \Gamma_{\text{out}}\end{aligned}$$

Primal problem

$$\begin{aligned}\nabla \cdot (\mathbf{v}u) &= s && \text{in } \Omega \\ u &= u_D && \text{on } \Gamma_{\text{in}}\end{aligned}$$

Dual problem

$$\begin{aligned}-\mathbf{v} \cdot \nabla z &= j && \text{in } \Omega \\ z &= h && \text{on } \Gamma_{\text{out}}\end{aligned}$$

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u \mathbf{v} \cdot \mathbf{n} \, ds = \int_{\Omega} w s \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u_D \mathbf{v} \cdot \mathbf{n} \, ds$$

Primal problem

$$\begin{aligned}\nabla \cdot (\mathbf{v}u) &= s && \text{in } \Omega \\ u &= u_D && \text{on } \Gamma_{\text{in}}\end{aligned}$$

Dual problem

$$\begin{aligned}-\mathbf{v} \cdot \nabla z &= j && \text{in } \Omega \\ z &= h && \text{on } \Gamma_{\text{out}}\end{aligned}$$

Primal problem $a(w, u) = b(w), \quad \forall w \in V$

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u \mathbf{v} \cdot \mathbf{n} \, ds = \int_{\Omega} w s \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u_D \mathbf{v} \cdot \mathbf{n} \, ds$$

■ Functional $j(u) = \int_{\Omega} g u \, d\mathbf{x} + \int_{\Gamma_{\text{out}}} h u \mathbf{v} \cdot \mathbf{n} \, ds, \quad g, h(\mathbf{x}) \in \{0, 1\}$

Primal problem

$$\begin{aligned}\nabla \cdot (\mathbf{v}u) &= s && \text{in } \Omega \\ u &= u_D && \text{on } \Gamma_{\text{in}}\end{aligned}$$

Dual problem

$$\begin{aligned}-\mathbf{v} \cdot \nabla z &= j && \text{in } \Omega \\ z &= h && \text{on } \Gamma_{\text{out}}\end{aligned}$$

$$\text{Primal problem} \quad a(w, u) = b(w), \quad \forall w \in V$$

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u \mathbf{v} \cdot \mathbf{n} \, ds = \int_{\Omega} w s \, d\mathbf{x} - \int_{\Gamma_{\text{in}}} w u_D \mathbf{v} \cdot \mathbf{n} \, ds$$

■ Functional $j(u) = \int_{\Omega} g u \, d\mathbf{x} + \int_{\Gamma_{\text{out}}} h u \mathbf{v} \cdot \mathbf{n} \, ds, \quad g, h(\mathbf{x}) \in \{0, 1\}$

$$\text{Dual problem} \quad a(z, w) = a^*(w, z) = j(w), \quad \forall w \in V$$

$$-\int_{\Omega} w \mathbf{v} \cdot \nabla z \, d\mathbf{x} + \int_{\Gamma_{\text{out}}} w z \mathbf{v} \cdot \mathbf{n} \, ds = \int_{\Omega} g w \, d\mathbf{x} + \int_{\Gamma_{\text{out}}} h w \mathbf{v} \cdot \mathbf{n} \, ds$$

Primal problem $\nabla \cdot (\mathbf{v}u - \epsilon \nabla u) = s$ + b.c.

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} + \int_{\Omega} \epsilon \nabla w \cdot \nabla u \, d\mathbf{x} = \int_{\Omega} w s \, d\mathbf{x}$$

Primal problem $\nabla \cdot (\mathbf{v}u - \epsilon \nabla u) = s \quad + \quad \text{b.c.}$

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} + \int_{\Omega} \epsilon \nabla w \cdot \nabla u \, d\mathbf{x} = \int_{\Omega} w s \, d\mathbf{x}$$

- Residual weighted by the dual error $w = \hat{z} - z_h$

$$\rho(w, u_h) = \int_{\Omega} w (s - \nabla \cdot (\mathbf{v}u_h)) \, d\mathbf{x} - \int_{\Omega} \epsilon \nabla w \cdot \nabla u_h \, d\mathbf{x}$$

Primal problem $\nabla \cdot (\mathbf{v}u - \epsilon \nabla u) = s$ + b.c.

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} + \int_{\Omega} \epsilon \nabla w \cdot \nabla u \, d\mathbf{x} = \int_{\Omega} w s \, d\mathbf{x}$$

- Residual weighted by the dual error $w = \hat{z} - z_h$

$$\rho(w, u_h) = \int_{\Omega} w (s - \nabla \cdot (\mathbf{v}u_h)) \, d\mathbf{x} - \int_{\Omega} \epsilon \nabla w \cdot \nabla u_h \, d\mathbf{x}$$

- Approximation $\mathbf{g}_h \approx \nabla u$, $\int_{\Omega} w \nabla \cdot \mathbf{g}_h \, d\mathbf{x} + \int_{\Omega} \nabla w \cdot \mathbf{g}_h \, d\mathbf{x} = 0$

Primal problem $\nabla \cdot (\mathbf{v}u - \epsilon \nabla u) = s$ + b.c.

$$\int_{\Omega} w \nabla \cdot (\mathbf{v}u) \, d\mathbf{x} + \int_{\Omega} \epsilon \nabla w \cdot \nabla u \, d\mathbf{x} = \int_{\Omega} w s \, d\mathbf{x}$$

- Residual weighted by the dual error $w = \hat{z} - z_h$

$$\rho(w, u_h) = \int_{\Omega} w (s - \nabla \cdot (\mathbf{v}u_h)) \, d\mathbf{x} - \int_{\Omega} \epsilon \nabla w \cdot \nabla u_h \, d\mathbf{x}$$

- Approximation $\mathbf{g}_h \approx \nabla u$, $\int_{\Omega} w \nabla \cdot \mathbf{g}_h \, d\mathbf{x} + \int_{\Omega} \nabla w \cdot \mathbf{g}_h \, d\mathbf{x} = 0$

$$\rho(w, u_h) = \int_{\Omega} w (s - \nabla \cdot (\mathbf{v}u_h - \epsilon \mathbf{g}_h)) \, d\mathbf{x} \quad \text{residual error}$$

$$+ \int_{\Omega} \epsilon \nabla w \cdot (\mathbf{g}_h - \nabla u_h) \, d\mathbf{x} \quad \text{diffusive flux error}$$

- Finite element solutions $u_h \approx u, \quad z_h \approx z, \quad \hat{z} \approx z$
- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

Computable error bounds $|j(u - u_h)| \leq \eta$

$$\eta = \Phi + \Psi, \quad |\rho(\hat{z} - z_h, u_h)| \leq \Phi, \quad |\rho(z_h, u_h)| \leq \Psi$$

- Finite element solutions $u_h \approx u, \quad z_h \approx z, \quad \hat{z} \approx z$
- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

Computable error bounds $|j(u - u_h)| \leq \eta$

$$\eta = \Phi + \Psi, \quad |\rho(\hat{z} - z_h, u_h)| \leq \Phi, \quad |\rho(z_h, u_h)| \leq \Psi$$

■ Finite element solutions $u_h \approx u, \quad z_h \approx z, \quad \hat{z} \approx z$

■ Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

Computable error bounds $|j(u - u_h)| \leq \eta$

$$\eta = \Phi + \Psi, \quad |\rho(\hat{z} - z_h, u_h)| \leq \Phi, \quad |\rho(z_h, u_h)| \leq \Psi$$

■ Effectivity indices

$$I_{\text{eff}} = \frac{\eta}{|j(u - u_h)|}, \quad I_{\text{rel}} = \left| \frac{|j(u - u_h)| - \eta}{|j(u)|} \right|$$

Estimates for $\text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0$ in $(0, 1)$, $u(0) = 0$, $u(1) = 1$

$$u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}, \quad j(u) = \int_0^1 u \, dx, \quad z(x) = \frac{e^{\text{Pe}(1-x)} + x(e^{\text{Pe}} - 1) - e^{\text{Pe}}}{\text{Pe}(1 - e^{\text{Pe}})}$$

Estimates for $\text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0$ in $(0, 1)$, $u(0) = 0$, $u(1) = 1$

$$u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}, \quad j(u) = \int_0^1 u \, dx, \quad z(x) = \frac{e^{\text{Pe}(1-x)} + x(e^{\text{Pe}} - 1) - e^{\text{Pe}}}{\text{Pe}(1 - e^{\text{Pe}})}$$

Discretization: central difference scheme, $h = 1/10$

Pe	$ j(u - u_h) $	Φ	Ψ	η	I_{rel}
1	7.67e-04	7.80e-04	4.09e-16	7.80e-04	3.05e-05
10	2.84e-05	4.10e-05	3.56e-18	4.10e-05	1.25e-04
100	–	–	–	–	–

Estimates for $\text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0$ in $(0, 1)$, $u(0) = 0$, $u(1) = 1$

$$u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}, \quad j(u) = \int_0^1 u \, dx, \quad z(x) = \frac{e^{\text{Pe}(1-x)} + x(e^{\text{Pe}} - 1) - e^{\text{Pe}}}{\text{Pe}(1 - e^{\text{Pe}})}$$

Discretization: upwind difference scheme, $h = 1/10$

Pe	$ j(u - u_h) $	Φ	Ψ	η	I_{rel}
1	4.52e-03	7.38e-04	3.58e-03	4.32e-03	4.79e-04
10	4.91e-02	3.06e-04	4.76e-02	4.79e-02	1.21e-02
100	5.00e-02	1.59e-09	5.00e-02	5.00e-02	1.21e-08

Estimates for $\text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0$ in $(0, 1)$, $u(0) = 0$, $u(1) = 1$

$$u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}, \quad j(u) = \int_0^1 u \, dx, \quad z(x) = \frac{e^{\text{Pe}(1-x)} + x(e^{\text{Pe}} - 1) - e^{\text{Pe}}}{\text{Pe}(1 - e^{\text{Pe}})}$$

Discretization: TVD scheme, MC limiter $h = 1/10$

Pe	$ j(u - u_h) $	Φ	Ψ	η	I_{rel}
1	1.03e-03	7.74e-04	2.60e-04	1.03e-03	1.34e-05
10	1.51e-02	9.12e-05	1.50e-02	1.51e-02	3.81e-05
100	4.51e-02	4.23e-09	4.51e-02	4.51e-02	1.97e-07

Estimates for $\text{Pe} \frac{du}{dx} - \frac{d^2u}{dx^2} = 0$ in $(0, 1)$, $u(0) = 0$, $u(1) = 1$

$$u(x) = \frac{e^{\text{Pe} x} - 1}{e^{\text{Pe}} - 1}, \quad j(u) = \int_0^1 u \, dx, \quad z(x) = \frac{e^{\text{Pe}(1-x)} + x(e^{\text{Pe}} - 1) - e^{\text{Pe}}}{\text{Pe}(1 - e^{\text{Pe}})}$$

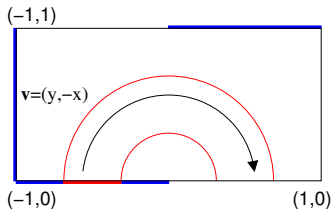
Discretization: TVD scheme, MC limiter $h = 1/10$

Pe	$ j(u - u_h) $	Φ	Ψ	η	I_{rel}
1	1.03e-03	7.74e-04	2.60e-04	1.03e-03	1.34e-05
10	1.51e-02	9.12e-05	1.50e-02	1.51e-02	3.81e-05
100	4.51e-02	4.23e-09	4.51e-02	4.51e-02	1.97e-07

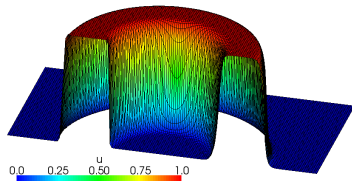
Galerkin orthogonality errors ($\hat{z} = z_h$) may be all you need *in practice*

Circular convection $\nabla \cdot (\mathbf{v}u) = 0$ in $\Omega = (-1, 1) \times (0, 1)$

$$u(x, y) = \begin{cases} 1, & 0.35 \leq r \leq 0.65 \\ 0, & \text{otherwise} \end{cases} \quad r(x, y) = \sqrt{x^2 + y^2}$$



domain and velocity

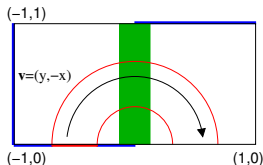


FEM-TVD, $h = 1/80$

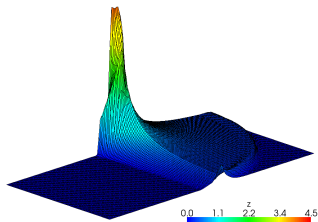
Target functional

$$j(u) = \int_{\Omega} gu \, dx + \int_{\Gamma_{\text{out}}} hu\mathbf{v} \cdot \mathbf{n} \, ds$$

$$h = \text{tr}(g), \quad g = 1 \text{ in } (-0.1, 0.1) \times (0, 1)$$



h	$j(e)$	Ψ	I_{eff}	I_{rel}
1/10	2.01e-3	2.16e-3	1.05	1.74e-3
1/20	4.40e-4	3.64e-4	0.82	1.26e-3
1/40	1.31e-4	1.03e-4	0.78	4.75e-4
1/80	4.28e-5	3.54e-5	0.82	1.24e-4
1/160	1.25e-5	1.07e-5	0.85	3.00e-5

FEM-TVD, $h = 1/80$

- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

$$|\rho(\hat{z} - z_h, u_h)| \leq \Phi = \sum_i \Phi_i, \quad |\rho(z_h, u_h)| \leq \Psi = \sum_i \Psi_i$$

- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

$$|\rho(\hat{z} - z_h, u_h)| \leq \Phi = \sum_i \Phi_i, \quad |\rho(z_h, u_h)| \leq \Psi = \sum_i \Psi_i$$

- Nodal localization $z_h = \sum_i z_i \varphi_i, \quad \Psi_i = |\rho(z_i \varphi_i, u_h)|$

$$\hat{z} - z_h = \sum_i w_i, \quad w_i = \varphi_i(\hat{z} - z_h), \quad \Phi_i = |\rho(w_i, u_h)|$$

- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

$$|\rho(\hat{z} - z_h, u_h)| \leq \Phi = \sum_i \Phi_i, \quad |\rho(z_h, u_h)| \leq \Psi = \sum_i \Psi_i$$

- Nodal localization $z_h = \sum_i z_i \varphi_i, \quad \Psi_i = |\rho(z_i \varphi_i, u_h)|$

$$\hat{z} - z_h = \sum_i w_i, \quad w_i = \varphi_i(\hat{z} - z_h), \quad \Phi_i = |\rho(w_i, u_h)|$$

Conversion to element contributions $\eta = \sum_K \eta_K = \Phi + \Psi$

$$\xi = \sum_i \xi_i \varphi_i, \quad \xi_i = \frac{\Phi_i + \Psi_i}{\int_{\Omega} \varphi_i \, dx}, \quad \eta_K = \int_K \xi \, dx$$

- Error representation $j(u - u_h) \approx \rho(\hat{z} - z_h, u_h) + \rho(z_h, u_h)$

$$|\rho(\hat{z} - z_h, u_h)| \leq \Phi = \sum_i \Phi_i, \quad |\rho(z_h, u_h)| \leq \Psi = \sum_i \Psi_i$$

- Nodal localization $z_h = \sum_i z_i \varphi_i, \quad \Psi_i = |\rho(z_i \varphi_i, u_h)|$

$$\hat{z} - z_h = \sum_i w_i, \quad w_i = \varphi_i(\hat{z} - z_h), \quad \Phi_i = |\rho(w_i, u_h)|$$

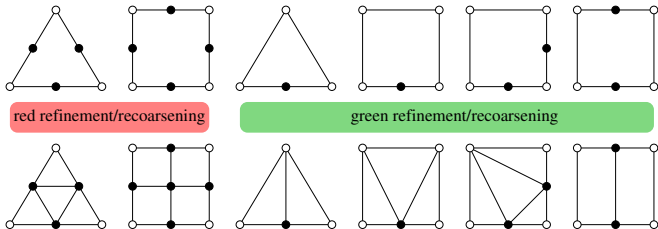
Conversion to element contributions $\eta = \sum_K \eta_K = \Phi + \Psi$

$$\xi = \sum_i \xi_i \varphi_i, \quad \xi_i = \frac{\Phi_i + \Psi_i}{\int_{\Omega} \varphi_i \, dx}, \quad \eta_K = \int_K \xi \, dx \stackrel{?}{>} tol$$

Conformal refinement algorithm

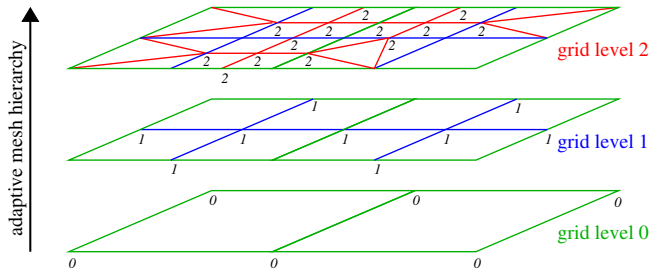
Bank, et al. '83

- 1 subdivide marked elements regularly (red rule)
- 2 eliminate 'hanging nodes' by transition cells (green rule)

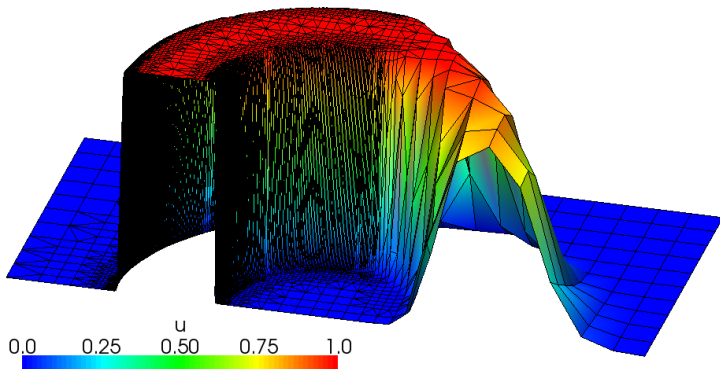


- Vertex-locking algorithm is used to reverse mesh refinement
- Nodal generation function provides all necessary information:
element type, inter-element relationship, refinement level, ...

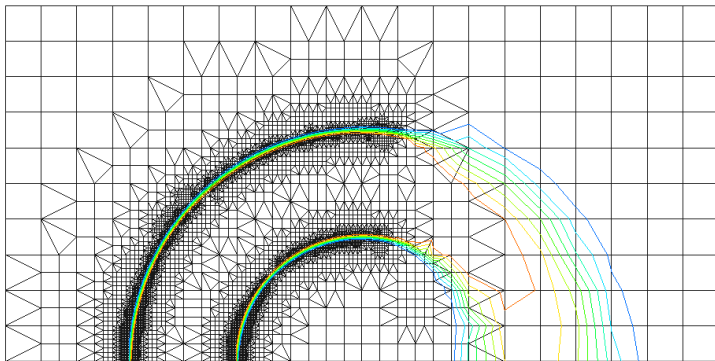
- Vertex-locking algorithm is used to reverse mesh refinement
- Nodal generation function provides all necessary information: *element type, inter-element relationship, refinement level, ...*



- Adaptive mesh refinement $h_{\min} = 1/320$



- Adaptive mesh refinement $h_{\min} = 1/320$



- Flux-limited Galerkin discretizations of AFC type
 - high resolution, violation of the Galerkin orthogonality
 - flux correction provides feedback for mesh adaptation
- Goal-oriented error estimation for transport problems
 - no dubious constants, control of local and transmitted errors
 - nodal decomposition of the error in the target functional
 - Galerkin orthogonality error is a handy criterion for refinement
 - Stabilization of diffusive fluxes using gradient recovery

- Time-dependent transport problems
 - high-resolution schemes of FCT type ✓
 - dynamic mesh adaptation algorithm ✓
 - error estimation in space *and* time

- Nonlinear systems of equations
 - compressible Euler equations ✓
 - conservative/characteristic limiters ✓
 - control of target quantities