Are stabilized methods a reliable tool for suppressing spurious oscillations?

Petr Knobloch
Charles University, Prague, Czech Republic

joint work with

Volker John
Universität des Saarlandes, Saarbrücken, Germany

19th Chemnitz FEM Symposium 2006
Chemnitz, September 25–27, 2006

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)

– can the methods remove spurious oscillations without smearing the layers?

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)

– can the methods remove spurious oscillations without smearing the layers?

– what is the best method?

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)

– can the methods remove spurious oscillations without smearing the layers?

– what is the best method?

– how to choose optimal parameters?

Contents/Motivation

– comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)

– can the methods remove spurious oscillations without smearing the layers?

– what is the best method?

– how to choose optimal parameters?

– are the nonlinear problems easy to solve?

Contents/Motivation

- comparison of the so–called discontinuity–capturing methods (developed to obtain discrete solutions of convection–diffusion equations without spurious oscillations on meshes which are not layer–adapted; important e.g. for chemical reactions)
- can the methods remove spurious oscillations without smearing the layers?
- what is the best method?
- how to choose optimal parameters?
- are the nonlinear problems easy to solve?
- are the methods reliable?

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b} \cdot \nabla u = f \quad \text{in } \Omega,$$

Ω . . . bounded polygon

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{in } \Omega,$$

$$u = u_b \quad \text{on } \Gamma^D, \qquad\qquad \varepsilon\,\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{on } \Gamma^N.$$

Ω ... bounded polygon; $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, $\mathrm{meas}_1(\Gamma^D) > 0$

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{in } \Omega,$$

$$u = u_b \quad \text{on } \Gamma^D, \qquad\qquad \varepsilon\,\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{on } \Gamma^N.$$

Ω ... bounded polygon; $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, $\mathrm{meas}_1(\Gamma^D) > 0$

Finite element method

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{in } \Omega,$$

$$u = u_b \quad \text{on } \Gamma^D, \qquad\qquad \varepsilon\,\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{on } \Gamma^N.$$

Ω ... bounded polygon; $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, $\mathrm{meas}_1(\Gamma^D) > 0$

Finite element method

$\mathcal{T}_h$ ... triangulation of Ω consisting of elements K

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{in } \Omega,$$

$$u = u_b \quad \text{on } \Gamma^D, \qquad\qquad \varepsilon\,\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{on } \Gamma^N.$$

Ω ... bounded polygon; $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, $\mathrm{meas}_1(\Gamma^D) > 0$

Finite element method

$\mathscr{T}_h$ ... triangulation of Ω consisting of elements K

V_h ... finite element space approximating the space

$$\{v \in H^1(\Omega)\,;\ v = 0 \text{ on } \Gamma^D\}$$

Scalar 2D steady convection–diffusion equation

$$-\varepsilon\,\Delta u + \mathbf{b}\cdot\nabla u = f \quad \text{in } \Omega,$$

$$u = u_b \quad \text{on } \Gamma^D, \qquad\qquad \varepsilon\,\frac{\partial u}{\partial \mathbf{n}} = g \quad \text{on } \Gamma^N.$$

Ω ... bounded polygon; $\partial\Omega = \overline{\Gamma^D \cup \Gamma^N}$, $\mathrm{meas}_1(\Gamma^D) > 0$

Finite element method

$\mathscr{T}_h$ ... triangulation of Ω consisting of elements K

V_h ... finite element space approximating the space

$$\{v \in H^1(\Omega)\,;\; v = 0 \text{ on } \Gamma^D\}$$

u_{bh} ... function defined on Ω approximating u_b on Γ^D

Galerkin FEM

$$u_h \in u_{bh} + V_h\,,$$

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

Galerkin FEM

$$u_h \in u_{bh} + V_h\,,$$

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

inappropriate if $\varepsilon \ll |\mathbf{b}|$!!!

Galerkin FEM

$$u_h \in u_{bh} + V_h \,,$$

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b} \cdot \nabla u_h, v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

inappropriate if $\varepsilon \ll |\mathbf{b}|$!!!

solution globally polluted by spurious oscillations

Galerkin FEM

$$u_h \in u_{bh} + V_h \,,$$

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

inappropriate if $\varepsilon \ll |\mathbf{b}|$!!!

solution globally polluted by spurious oscillations

SUPG method

Brooks, Hughes, CMAME (1982)

Galerkin FEM

$$u_h \in u_{bh} + V_h\,,$$

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

inappropriate if $\varepsilon \ll |\mathbf{b}|$!!!

solution globally polluted by spurious oscillations

SUPG method

Brooks, Hughes, CMAME (1982)

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \tau\,\mathbf{b}\cdot\nabla v_h)$$

$$= (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

$$R_h(u) = -\varepsilon\,\Delta_h u + \mathbf{b}\cdot\nabla u - f$$

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

possible remedy: add a suitable artificial diffusion term to the SUPG method

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

possible remedy: add a suitable artificial diffusion term to the SUPG method (discontinuity capturing or shock capturing)

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

possible remedy: add a suitable artificial diffusion term to the SUPG method (discontinuity capturing or shock capturing)

– isotropic artificial diffusion

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

possible remedy: add a suitable artificial diffusion term to the SUPG method (discontinuity capturing or shock capturing)

- isotropic artificial diffusion
- crosswind artificial diffusion

Choice of τ

$$\tau|_K = \frac{h_K}{2\,|\mathbf{b}|\,p}\left(\coth \mathrm{Pe}_K - \frac{1}{\mathrm{Pe}_K}\right) \quad \text{with} \quad \mathrm{Pe}_K = \frac{|\mathbf{b}|\,h_K}{2\,\varepsilon\,p}$$

optimal in 1D for P_1^c

in 2D still spurious oscillations localized in narrow regions along sharp layers (the SUPG method is not monotone)

possible remedy: add a suitable artificial diffusion term to the SUPG method (discontinuity capturing or shock capturing)

- isotropic artificial diffusion
- crosswind artificial diffusion
- edge stabilization

Methods adding isotropic artificial diffusion

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \boldsymbol{\tau}\,\mathbf{b}\cdot\nabla v_h)$$

$$= (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

Methods adding isotropic artificial diffusion

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \boldsymbol{\tau}\,\mathbf{b}\cdot\nabla v_h)$$

$$+ (\widetilde{\varepsilon}\,\nabla u_h, \nabla v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

Methods adding isotropic artificial diffusion

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \tau\,\mathbf{b}\cdot\nabla v_h)$$

$$+ (\widetilde{\varepsilon}\,\nabla u_h, \nabla v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

Hughes, Mallet, Mizukami, CMAME (1986)

Tezduyar, Park, CMAME (1986)

Galeão, do Carmo, CMAME (1988)

Johnson, CMAME (1990)

do Carmo, Galeão, CMAME (1991)

Almeida, Silva, CMAME (1997)

Knopp, Lube, Rapin, CMAME (2002)

do Carmo, Alvarez, CMAME (2003)

Methods adding crosswind artificial diffusion

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \tau\,\mathbf{b}\cdot\nabla v_h)$$

$$+ (\widetilde{\varepsilon}\,D\,\nabla u_h, \nabla v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

$$D = I - \frac{\mathbf{b}\otimes\mathbf{b}}{|\mathbf{b}|^2}$$

Methods adding crosswind artificial diffusion

$$\varepsilon\,(\nabla u_h, \nabla v_h) + (\mathbf{b}\cdot\nabla u_h, v_h) + (R_h(u_h), \tau\,\mathbf{b}\cdot\nabla v_h)$$

$$+ (\widetilde{\varepsilon}\,D\,\nabla u_h, \nabla v_h) = (f, v_h) + (g, v_h)_{\Gamma^N} \quad \forall\, v_h \in V_h$$

$$D = I - \frac{\mathbf{b}\otimes\mathbf{b}}{|\mathbf{b}|^2}$$

Johnson, Schatz, Wahlbin, Math. Comput. (1987)

Codina, CMAME (1993)

Knopp, Lube, Rapin, CMAME (2002)

Burman, Ern, CMAME (2002)

Examples of $\widetilde{\varepsilon}$

do Carmo, Galeão (1991)

$$\widetilde{\varepsilon}|_K = \max\left\{0, \frac{h_K\,|R_h(u_h)|}{2\,|\nabla u_h|} - \frac{h_K}{2\,|\mathbf{b}|}\,\frac{|R_h(u_h)|^2}{|\nabla u_h|^2}\right\}$$

Johnson (1990)

$$\widetilde{\varepsilon}|_K = \max\left\{0, C\,[\mathrm{diam}(K)]^2\,|R_h(u_h)| - \varepsilon\right\}$$

Codina (1993) (modified)

$$\widetilde{\varepsilon}|_K = \max\left\{0, C\,\frac{\mathrm{diam}(K)\,|R_h(u_h)|}{2\,|\nabla u_h|} - \varepsilon\right\}$$

$C = 0.7$ recommended

Burman, Ern (2002) (modified)

$$\widetilde{\varepsilon}|_K = \frac{h_K\,|R_h(u_h)|}{2\,|\nabla u_h|}\,\frac{1}{1 + \frac{|R_h(u_h)|}{|\mathbf{b}|\,|\nabla u_h|}}$$

Edge stabilization methods

additional term

$$\sum_{K\in\mathscr{T}_h}\int_{\partial K}\Psi_K(u_h)\,\mathrm{sign}(\mathbf{t}_{\partial K}\cdot\nabla(u_h|_K))\,\mathbf{t}_{\partial K}\cdot\nabla(v_h|_K)\,\mathrm{d}\sigma$$

Edge stabilization methods

additional term

$$\sum_{K\in\mathscr{T}_h}\int_{\partial K}\Psi_K(u_h)\,\mathrm{sign}(\mathbf{t}_{\partial K}\cdot\nabla(u_h|_K))\,\mathbf{t}_{\partial K}\cdot\nabla(v_h|_K)\,\mathrm{d}\sigma$$

Burman, Hansbo, CMAME (2004)

Burman, Ern, Math. Comput. (2005)

Mizukami–Hughes method

Mizukami, Hughes, CMAME (1985), K. (2005)

Mizukami–Hughes method

Mizukami, Hughes, CMAME (1985), K. (2005)

nonlinear Petrov–Galerkin method defined for P_1^c elements

Mizukami–Hughes method

Mizukami, Hughes, CMAME (1985), K. (2005)

nonlinear Petrov–Galerkin method defined for P_1^c elements

Advantages:

– accurate discrete solutions

– discrete maximum principle always satisfied

– no stabilization parameters (method of upwind type)

Mizukami–Hughes method

Mizukami, Hughes, CMAME (1985), K. (2005)

nonlinear Petrov–Galerkin method defined for P_1^c elements

Advantages:

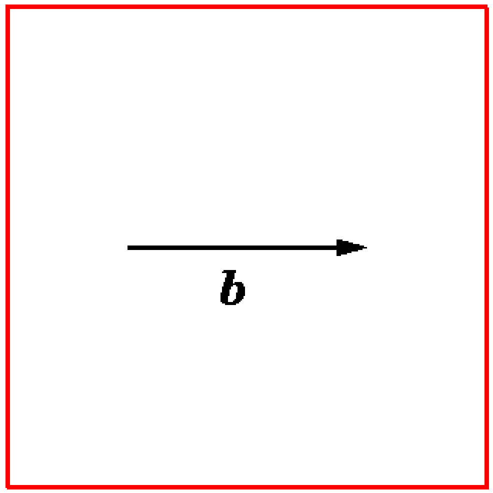
- accurate discrete solutions
- discrete maximum principle always satisfied
- no stabilization parameters (method of upwind type)

Disadvantages:

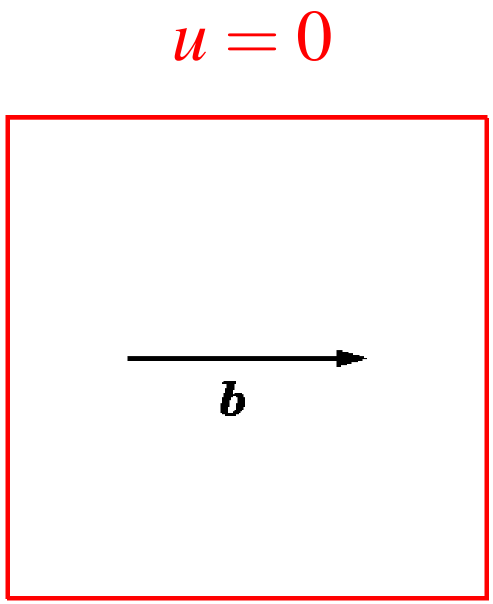
- a generalization to other types of finite elements not clear
- no results on existence, uniqueness and convergence of u_h
- difficult to apply to more complicated problems

Example 1 (convection with a source term)

$u = 0$



Example 1 (convection with a source term)

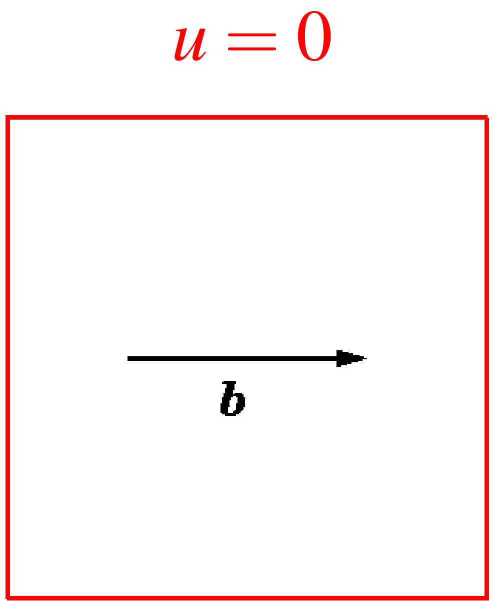


$$\varepsilon = 10^{-8}$$

$$|\mathbf{b}| = 1$$

$$f = 1$$

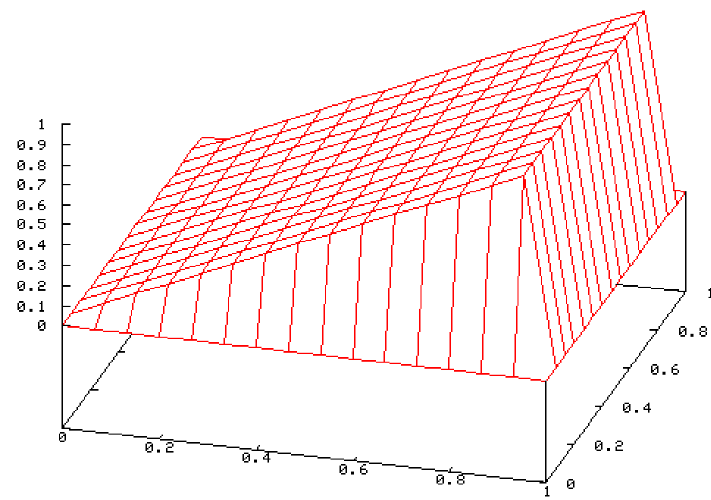
Example 1 (convection with a source term)



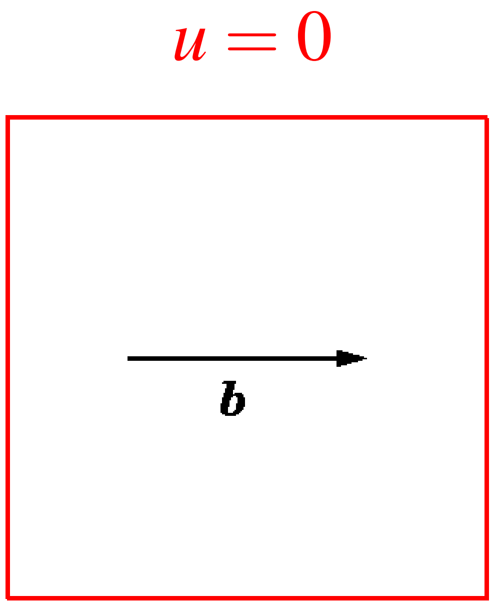
$$\varepsilon = 10^{-8}$$

$$|\mathbf{b}| = 1$$

$$f = 1$$



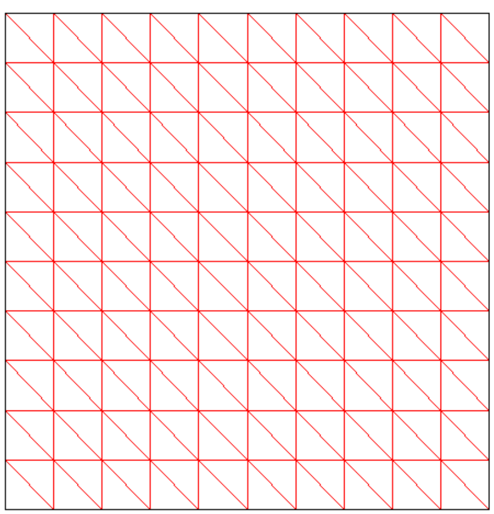
Example 1 (convection with a source term)

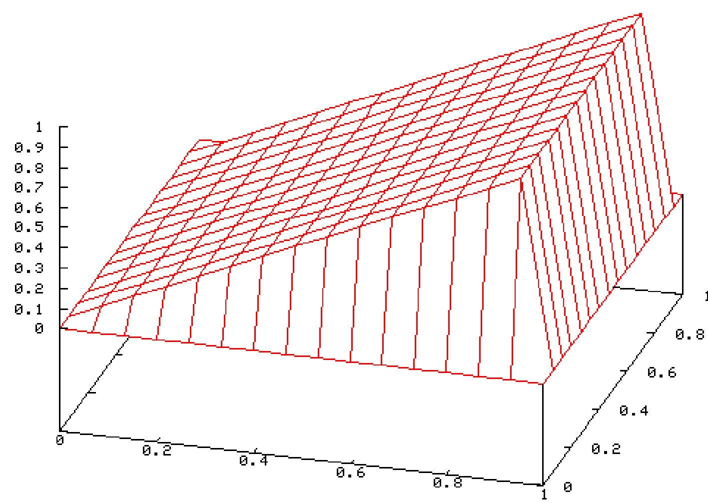


$$\varepsilon = 10^{-8}$$

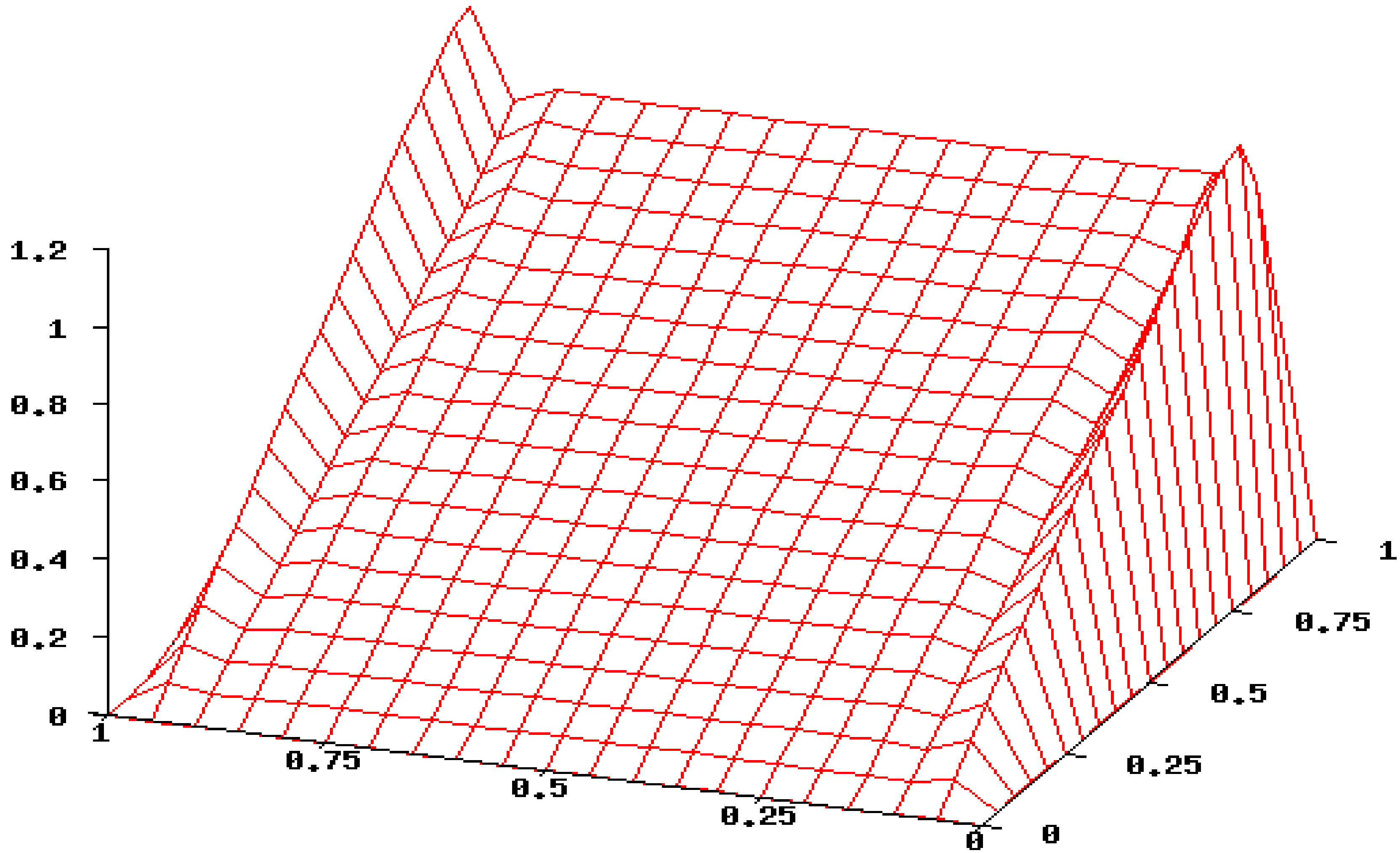
$$|\mathbf{b}| = 1$$

$$f = 1$$

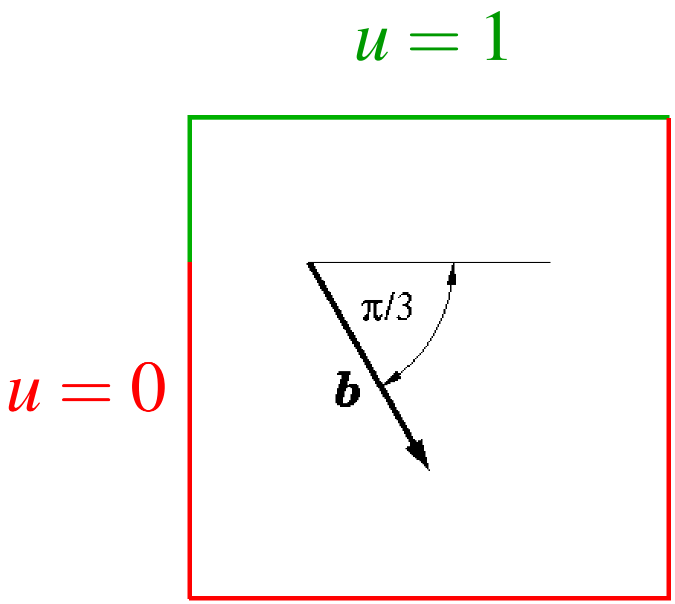




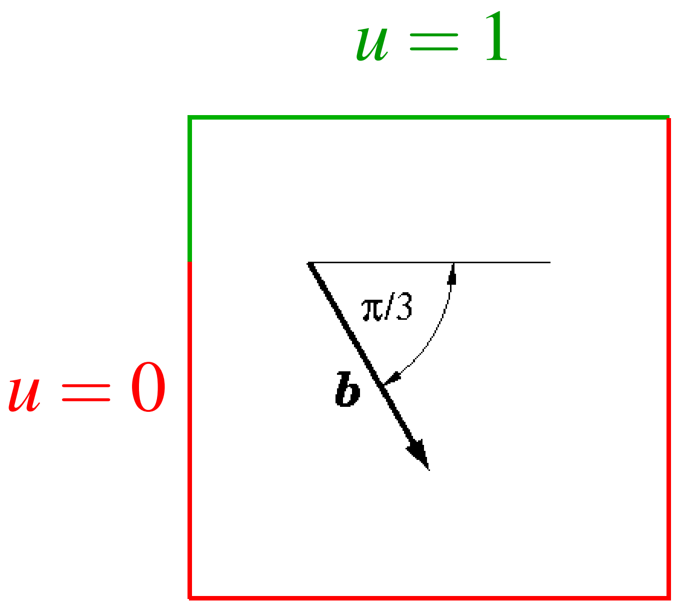
Example 1, SUPG method, mesh 21x21



Example 2 (convection skew to the mesh)



Example 2 (convection skew to the mesh)

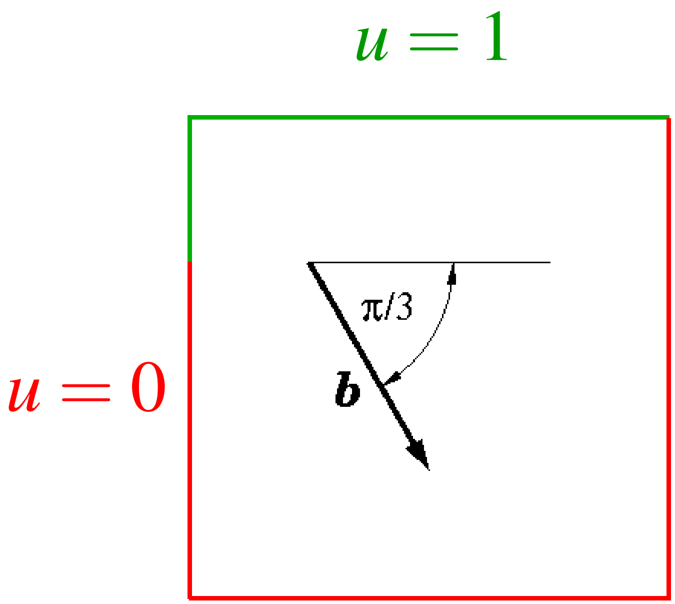


$$\varepsilon = 10^{-8}$$

$$|\mathbf{b}| = 1$$

$$f = 0$$

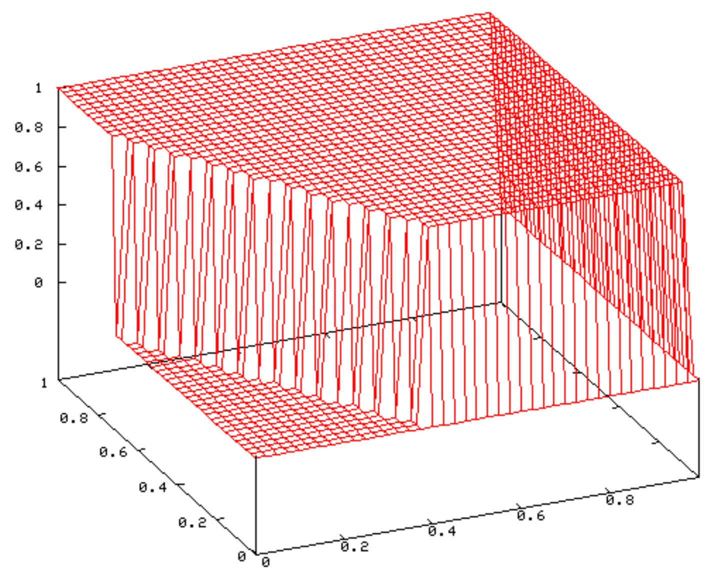
Example 2 (convection skew to the mesh)



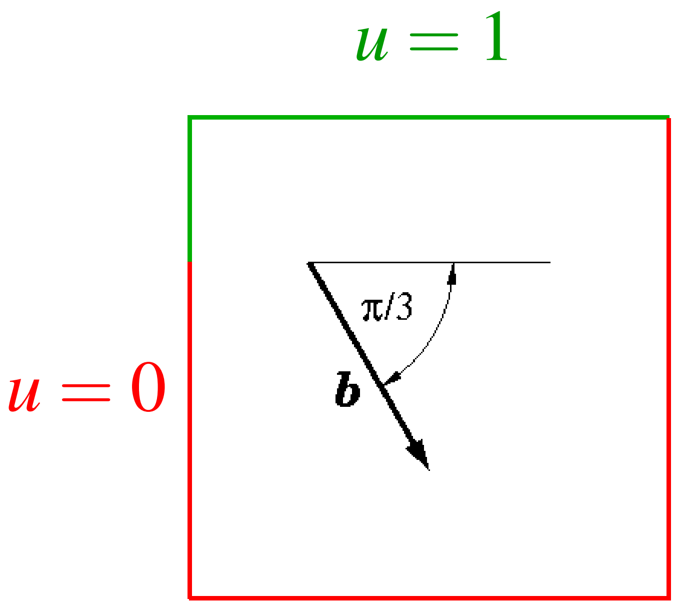
$$\varepsilon = 10^{-8}$$

$$|\mathbf{b}| = 1$$

$$f = 0$$



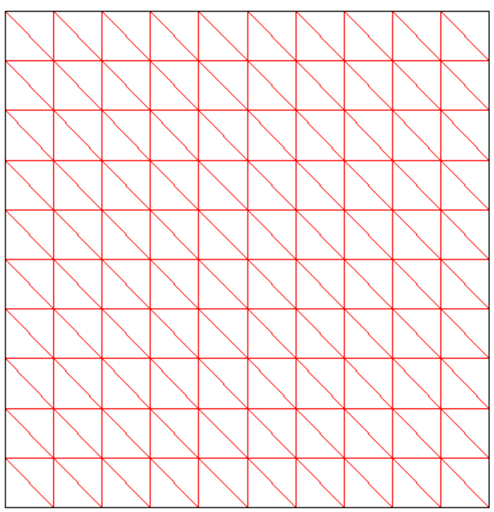
Example 2 (convection skew to the mesh)

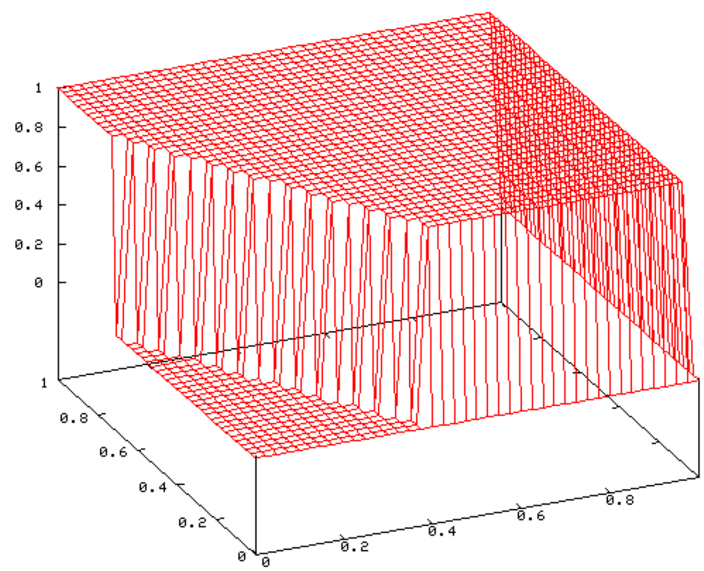


$$\varepsilon = 10^{-8}$$

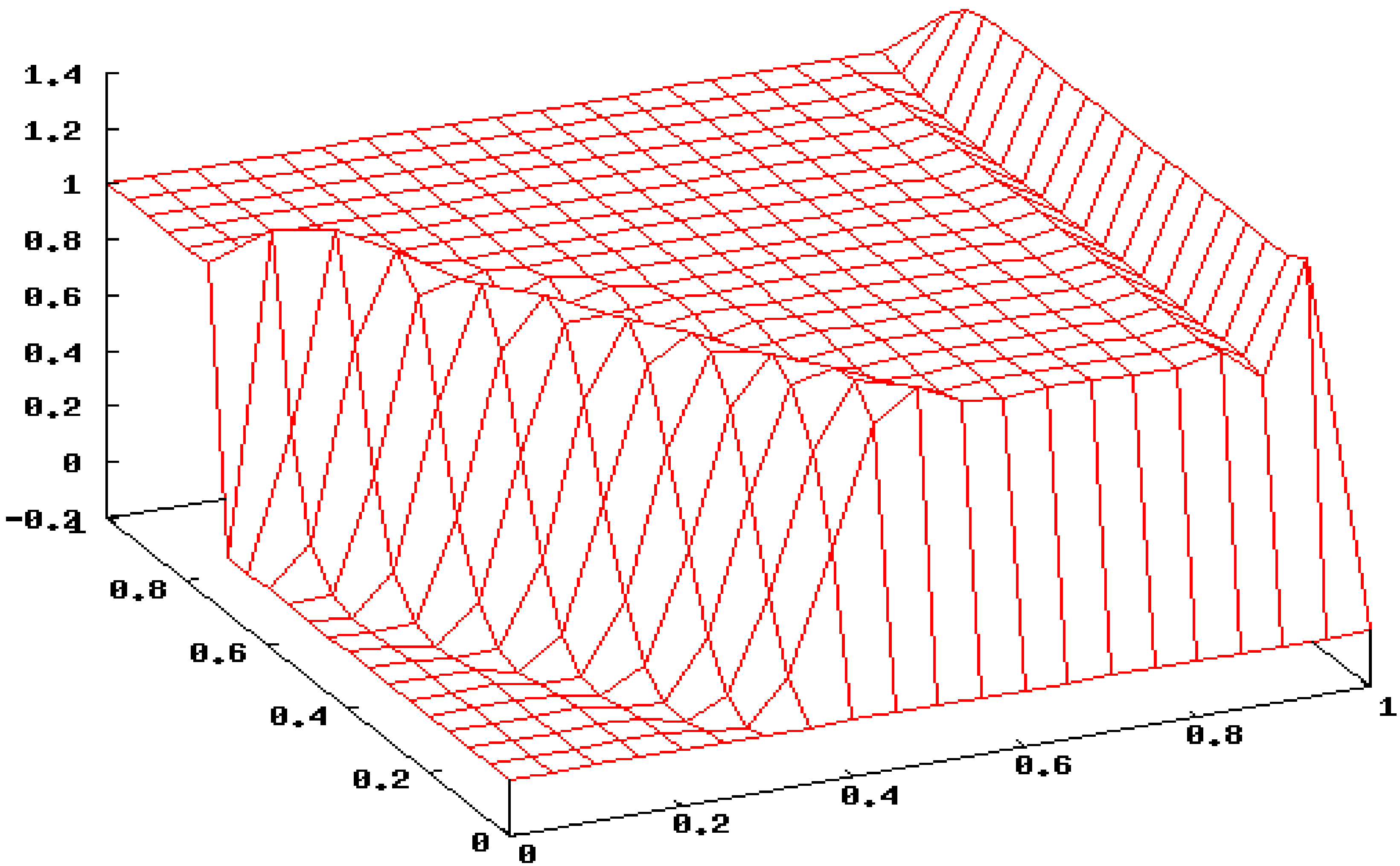
$$|\mathbf{b}| = 1$$

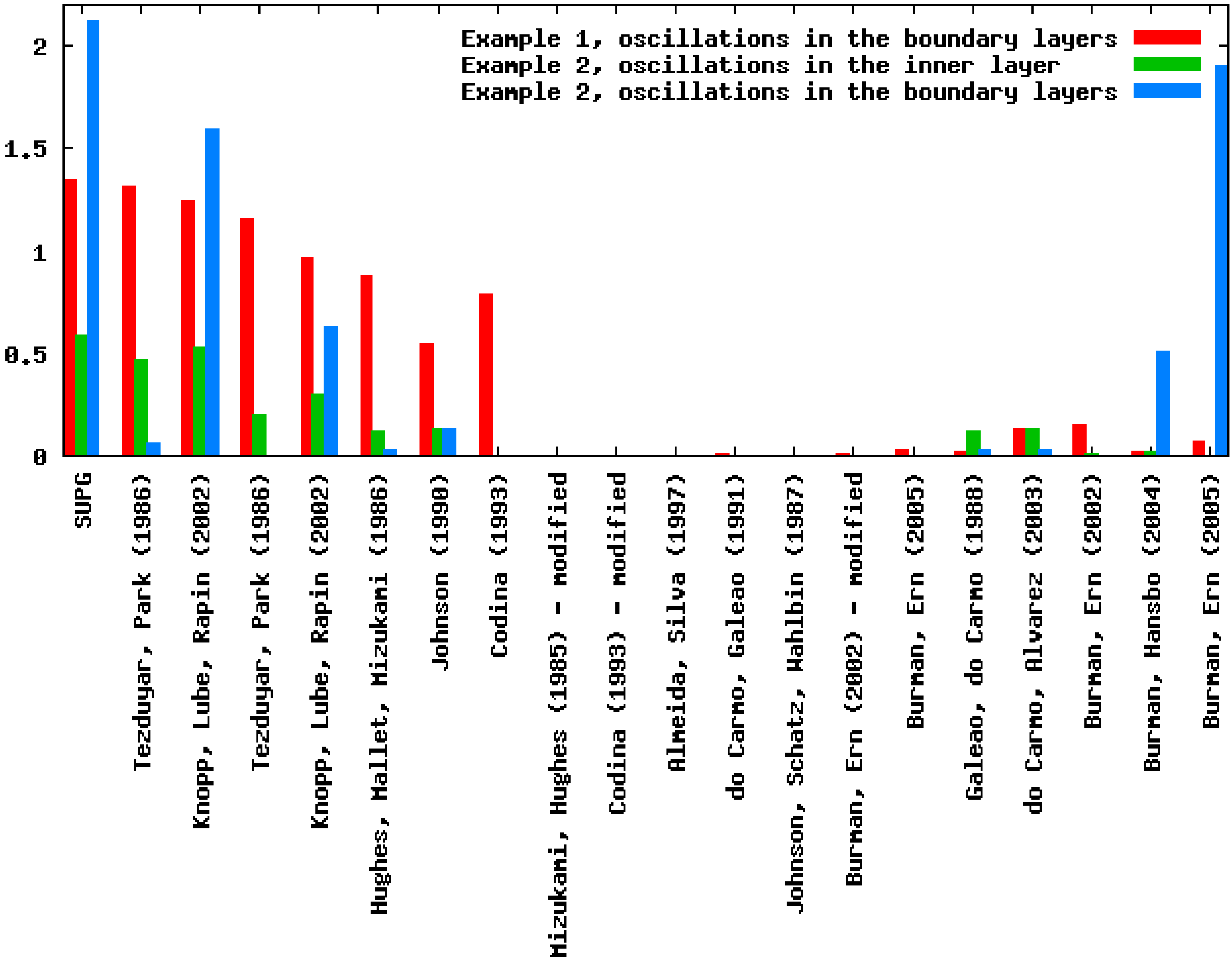
$$f = 0$$

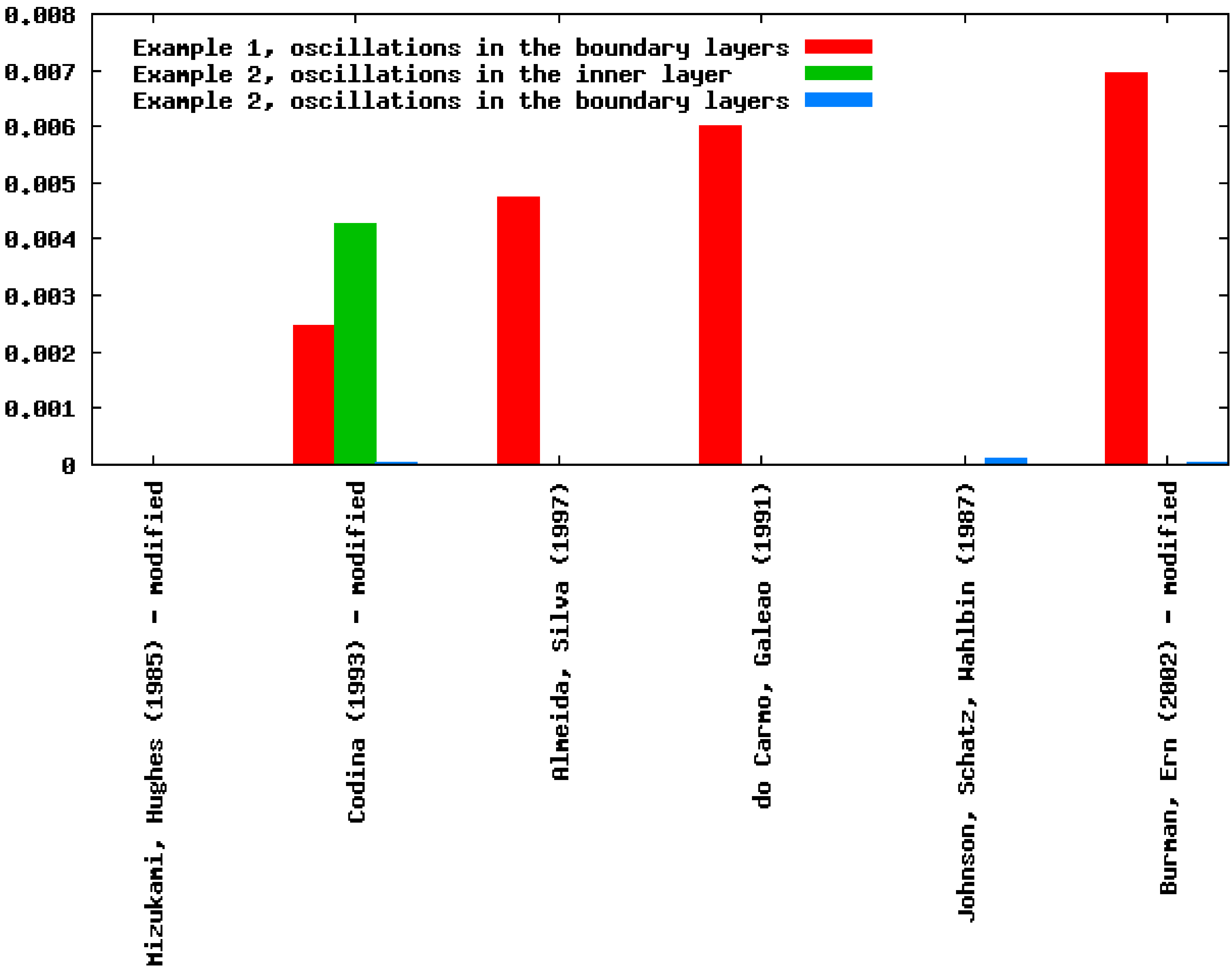




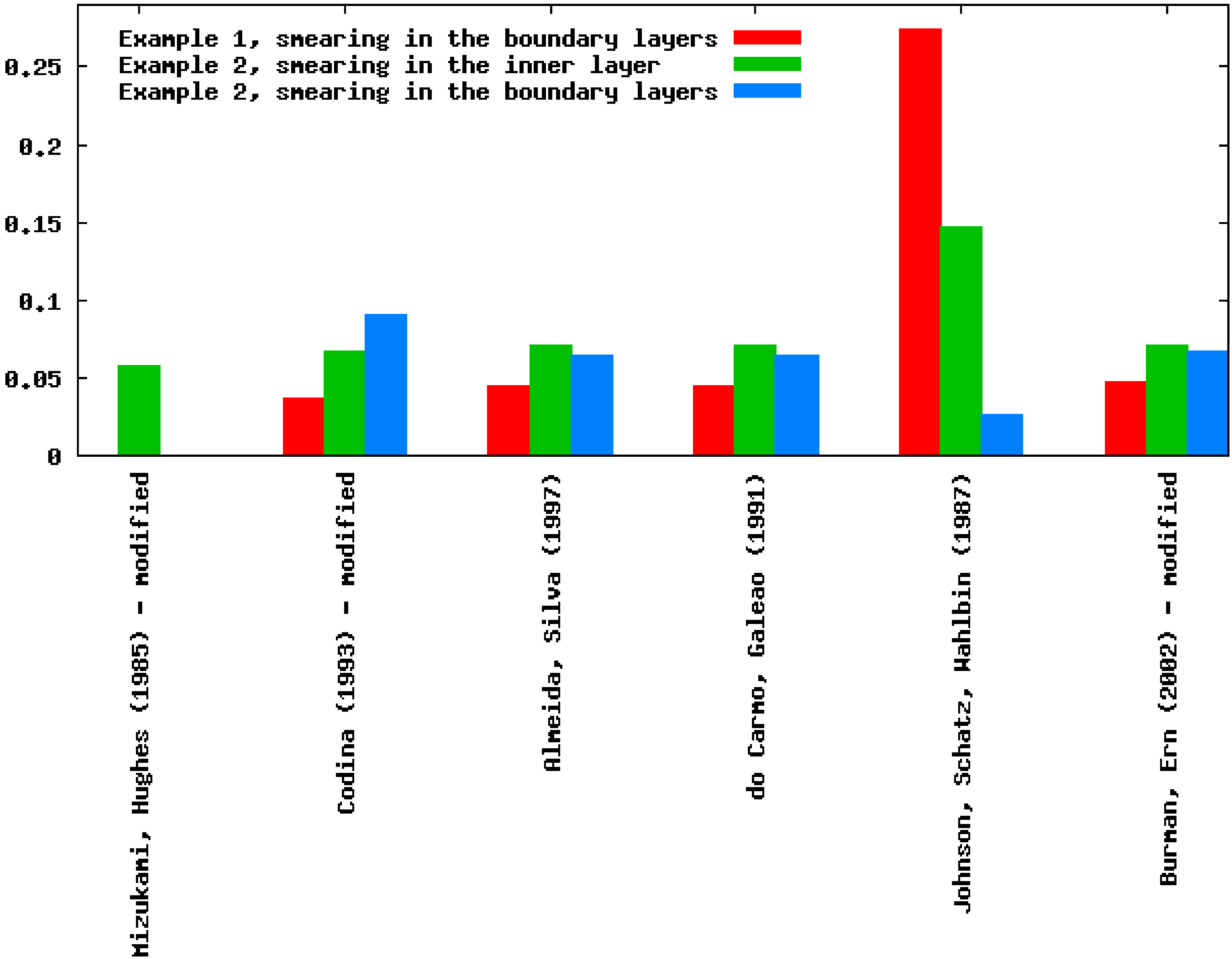
Example 2, SUPG method, mesh 21x21



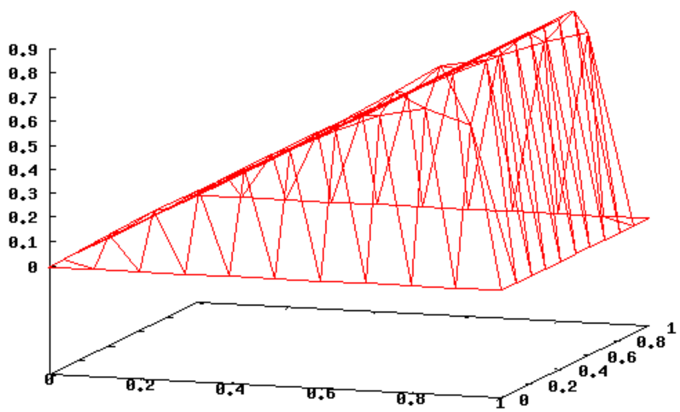

Example 1, oscillations in the boundary layers
Example 2, oscillations in the inner layer
Example 2, oscillations in the boundary layers
2
1,5
1
0,5
0
SUPG
Tezduyar, Park (1986)
Knopp, Lube, Rapin (2002)
Tezduyar, Park (1986)
Knopp, Lube, Rapin (2002)
Hughes, Mallet, Mizukami (1986)
Johnson (1990)
Codina (1993)
Mizukami, Hughes (1985) - modified
Codina (1993) - modified
Almeida, Silva (1997)
do Carmo, Galeao (1991)
Johnson, Schatz, Wahlbin (1987)
Burman, Ern (2002) - modified
Burman, Ern (2005)
Galeao, do Carmo (1988)
do Carmo, Alvarez (2003)
Burman, Ern (2002)
Burman, Hansbo (2004)
Burman, Ern (2005)



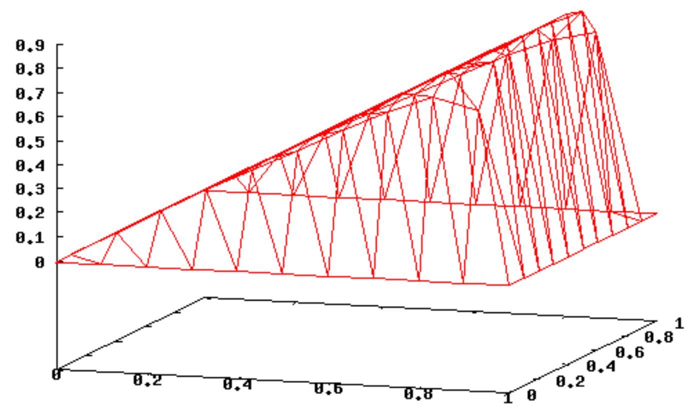
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0
Example 1, oscillations in the boundary layers
Example 2, oscillations in the inner layer
Example 2, oscillations in the boundary layers
Mizukami, Hughes (1985) - modified
Codina (1993) - modified
Almeida, Silva (1997)
do Carmo, Galeao (1991)
Johnson, Schatz, Wahlbin (1987)
Burman, Ern (2002) - modified


Example 1, smearing in the boundary layers
Example 2, smearing in the inner layer
Example 2, smearing in the boundary layers
0.25
0.2
0.15
0.1
0.05
0
Mizukami, Hughes (1985) - modified
Codina (1993) - modified
Almeida, Silva (1997)
do Carmo, Galeao (1991)
Johnson, Schatz, Wahlbin (1987)
Burman, Ern (2002) - modified

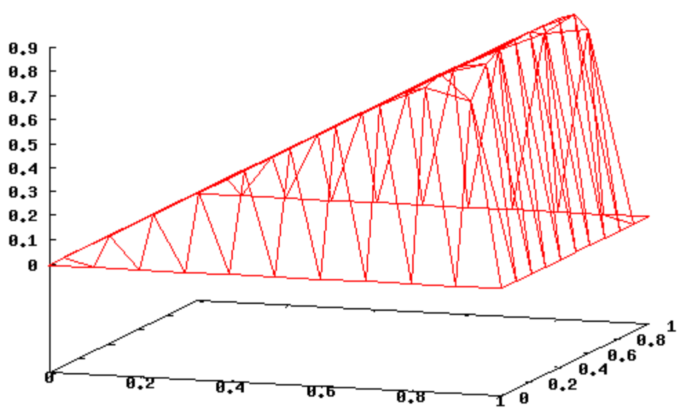
Example 1, P_1 element, mesh 11x11



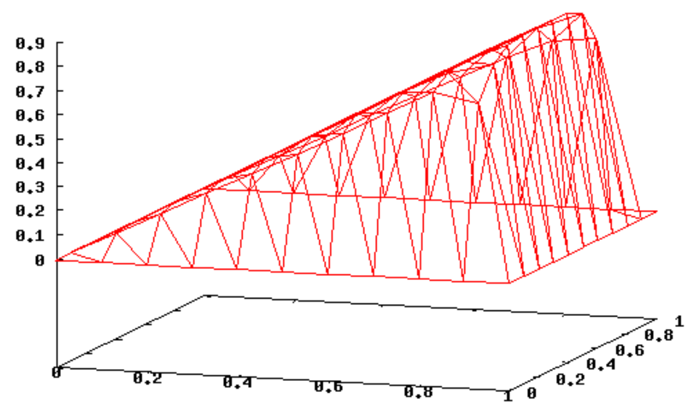
Almeida, Silva



modified Burman, Ern

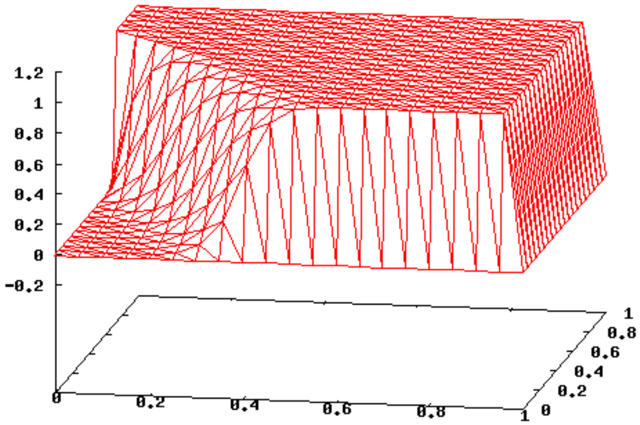


modified Codina C=0.465

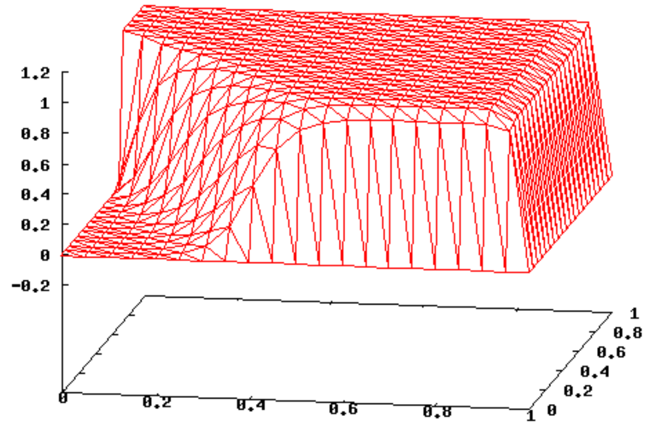


modified Codina C=0.6

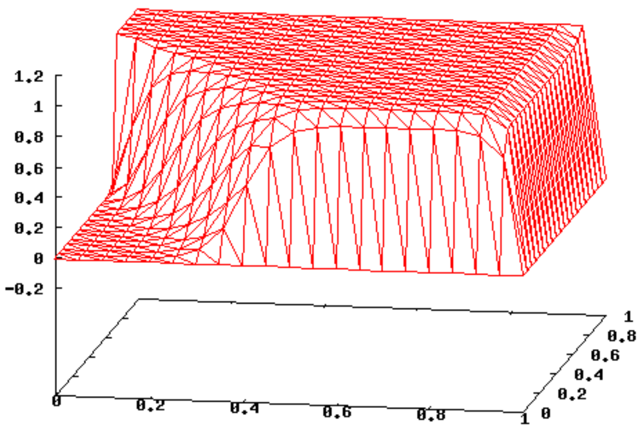
Example 2, P_1 element, mesh 21x21



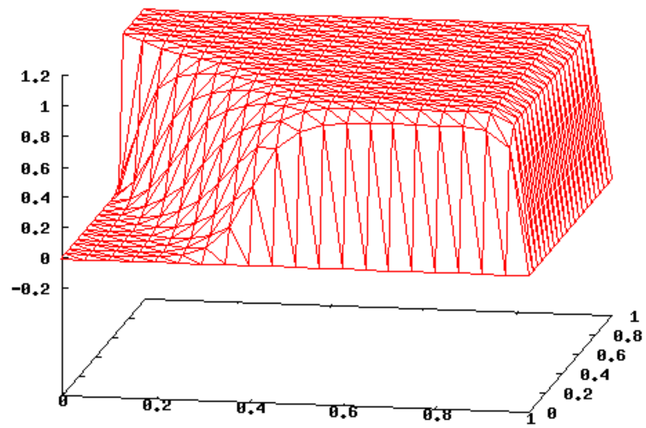
Mizukami, Hughes



do Carmo, Galeão (1991)

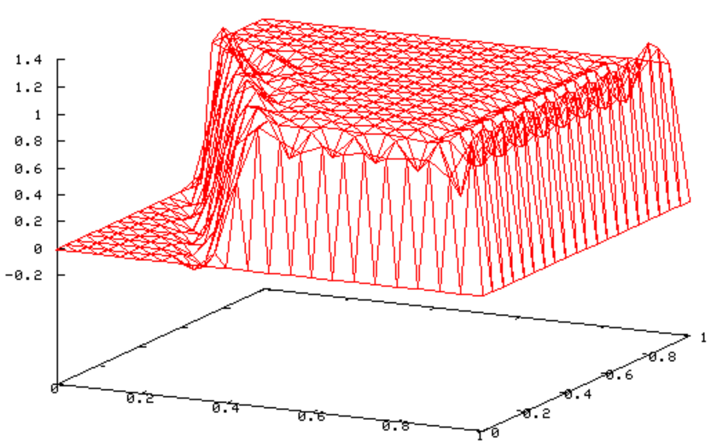


modified Codina

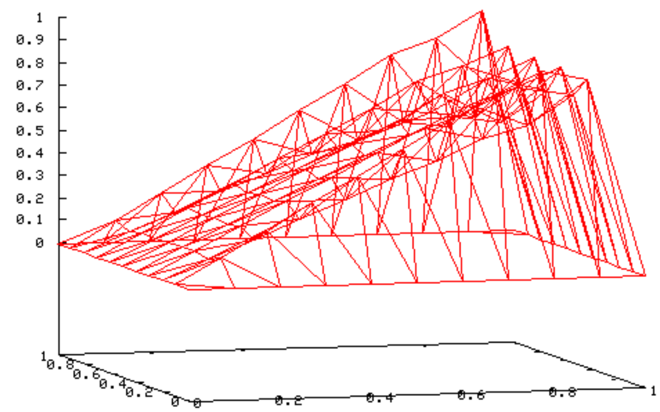


modified Burman, Ern

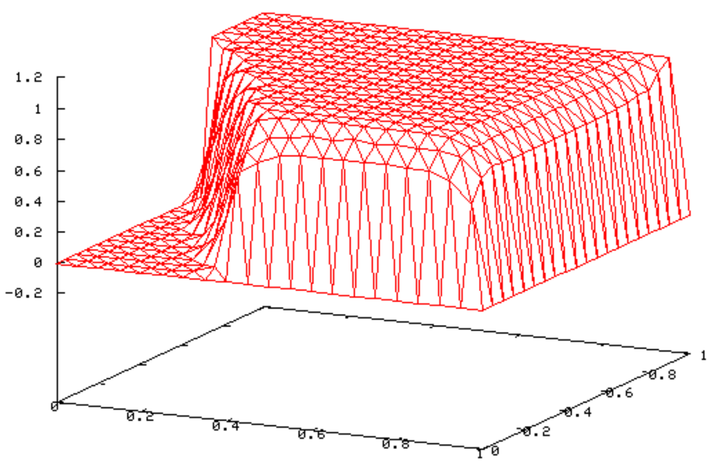
P_2 element



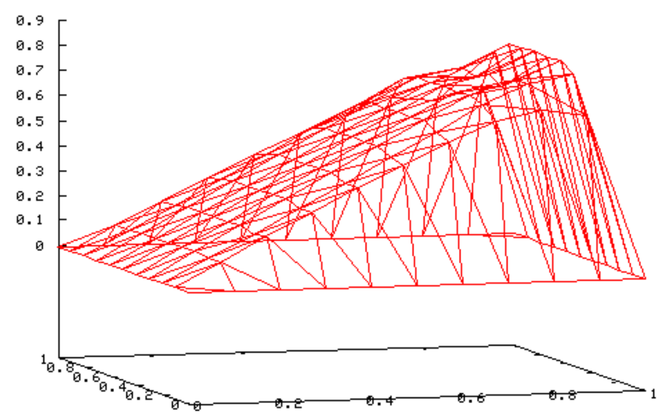
SUPG



SUPG

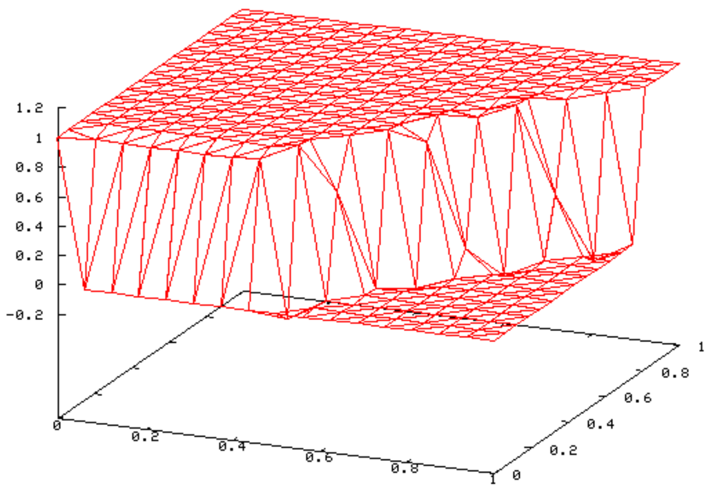


modified Codina C=0.35

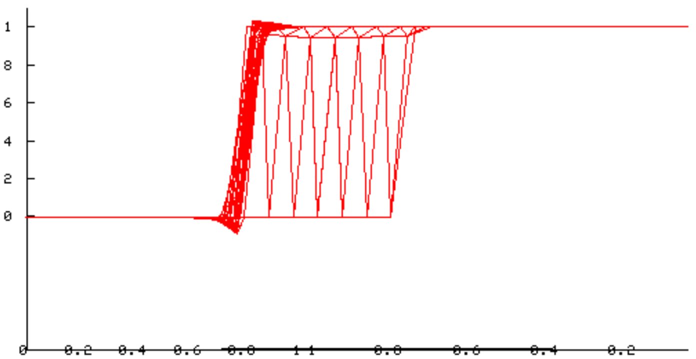


modified Burman, Ern

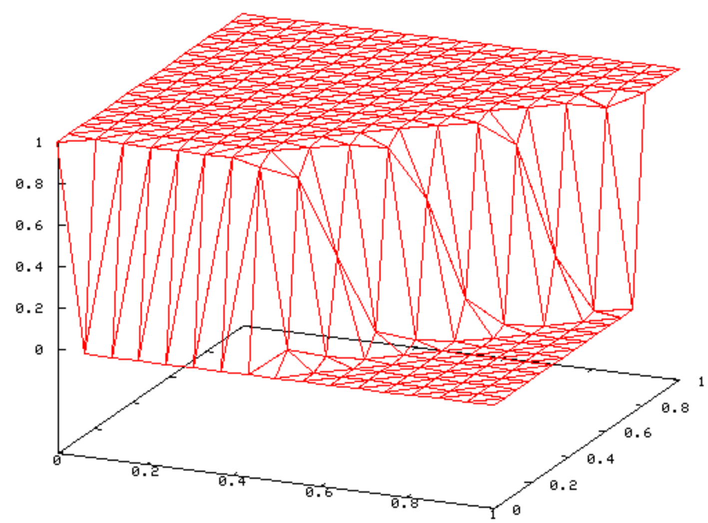
P_4 element



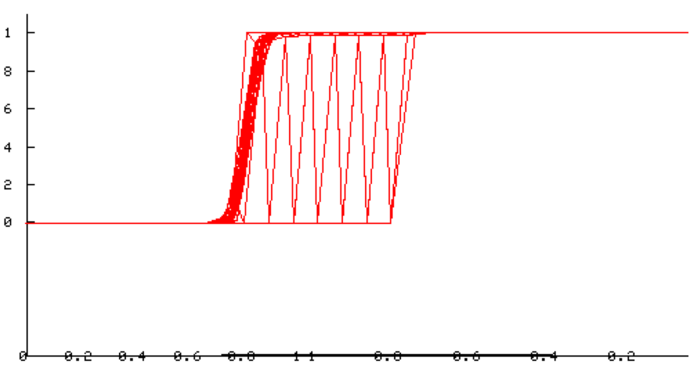
SUPG



SUPG

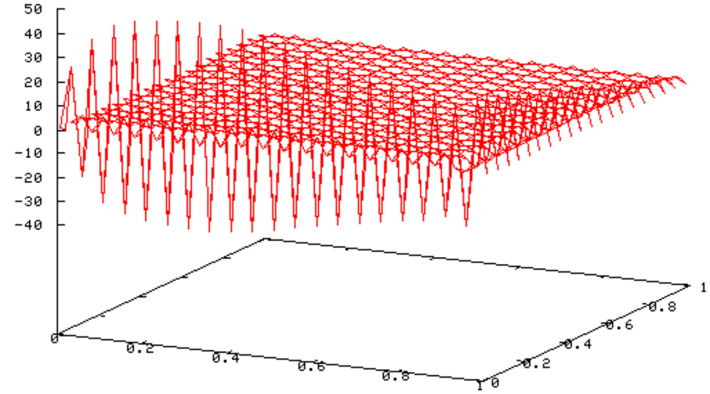


modified Codina C=0.2

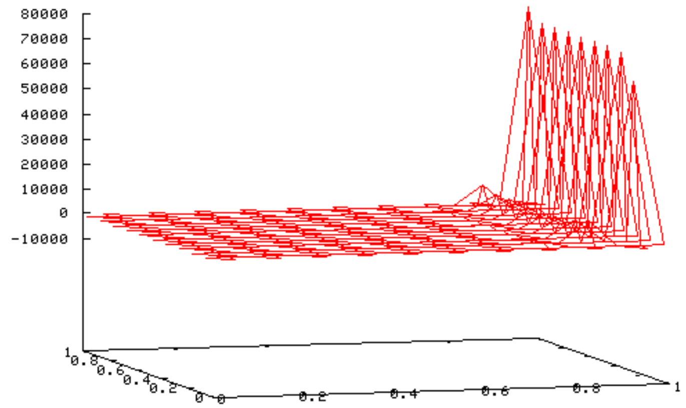


modified Codina C=0.2

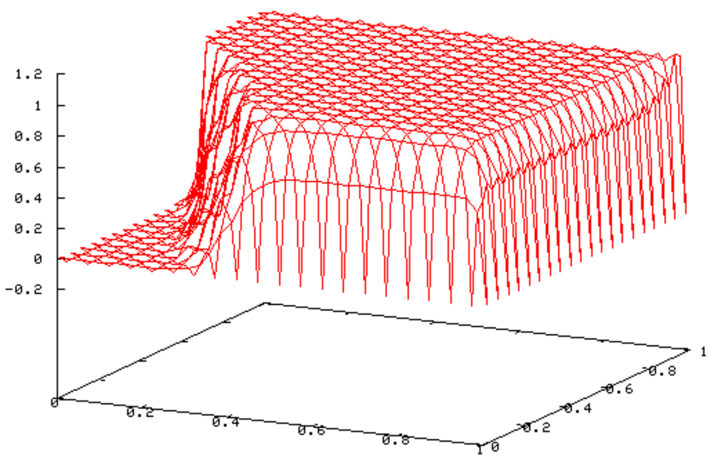
Crouzeix–Raviart element



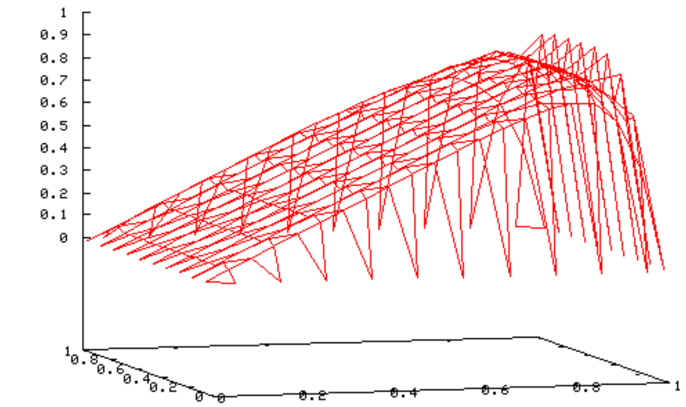
SUPG



SUPG



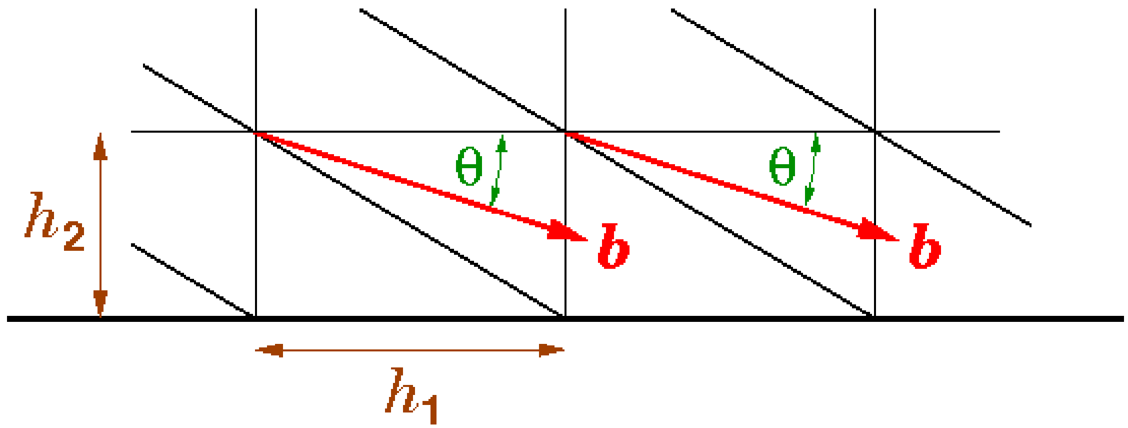
do Carmo, Galeão (1991)

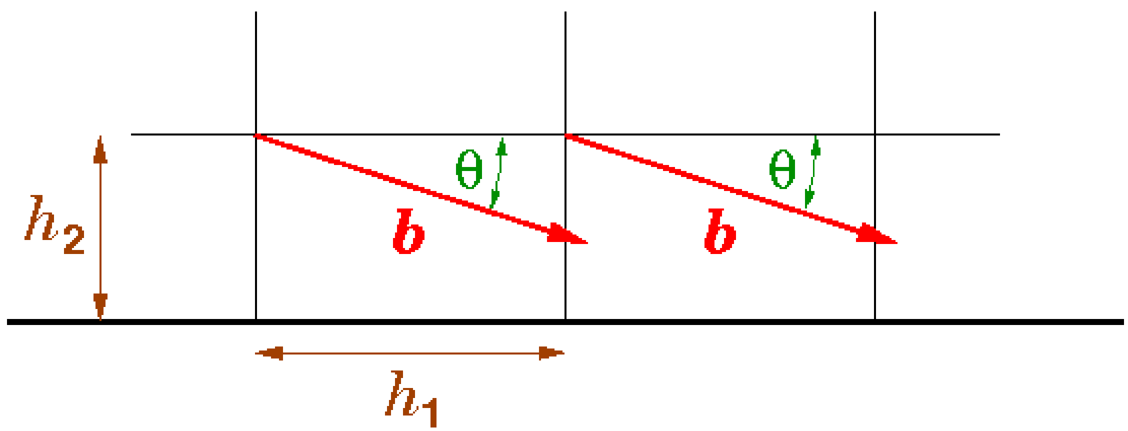


modified Codina C=0.6

Example 2: optimal C in boundary layers for P_1 and Q_1

$$C = \frac{h_2 \cos\theta - h_1 \sin\theta}{\sqrt{h_1^2 + h_2^2}\,(\cos\theta)^3}$$





Example 2: optimal C in boundary layers for P_1 and Q_1

$$C = \frac{h_2 \cos\theta - h_1 \sin\theta}{\sqrt{h_1^2 + h_2^2}\,(\cos\theta)^3}$$

– optimal C depends on the mesh and the data of the problem

Example 2: optimal C in boundary layers for P_1 and Q_1

$$C = \frac{h_2 \cos\theta - h_1 \sin\theta}{\sqrt{h_1^2 + h_2^2}\,(\cos\theta)^3}$$

– optimal C depends on the mesh and the data of the problem

– for $\theta = \pi/3$, the boundary layer at $x = 1$ and $h_1 = 2\,h_2$, we get $C \approx 0.85$

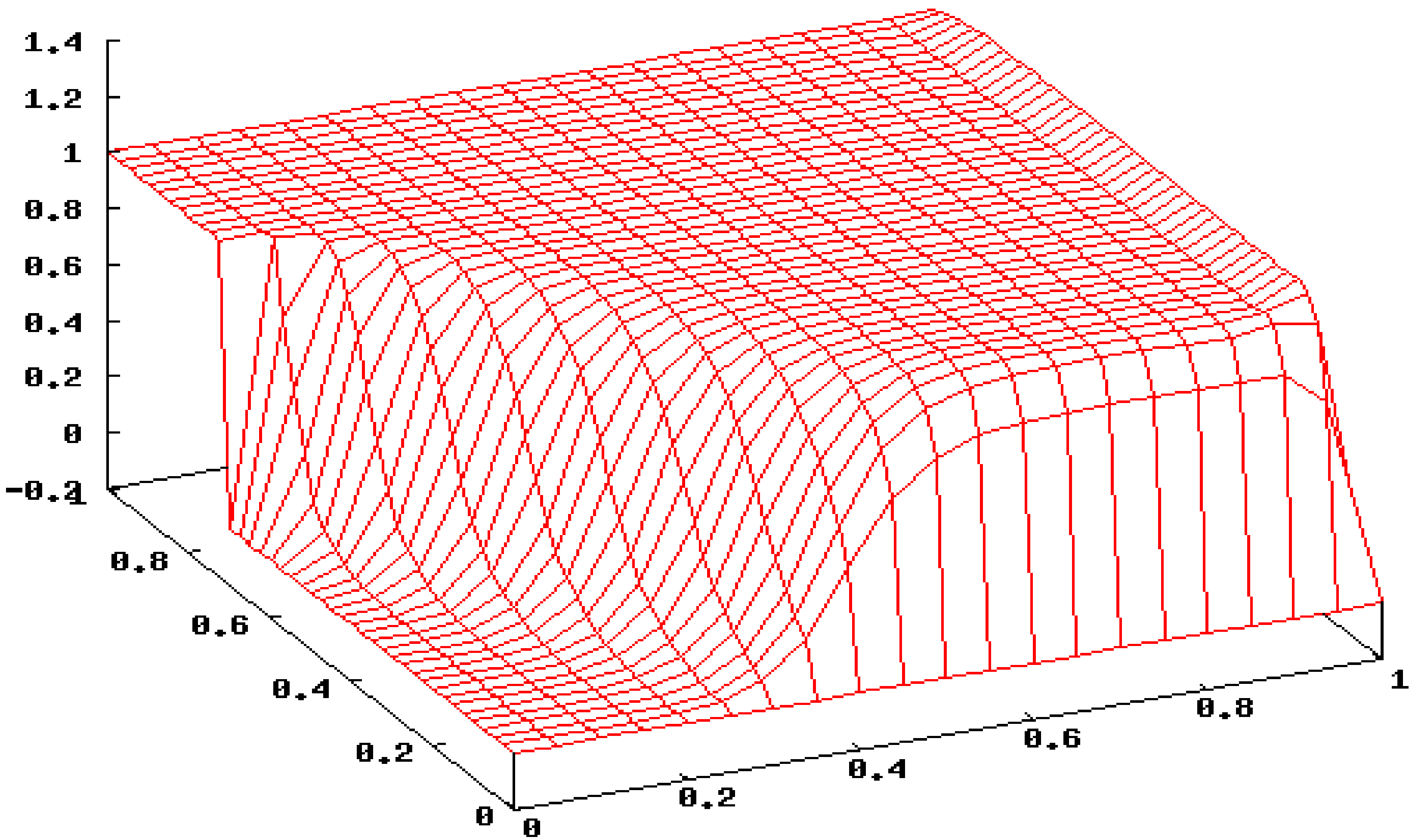
Example 2: optimal C in boundary layers for P_1 and Q_1

$$C = \frac{h_2 \cos\theta - h_1 \sin\theta}{\sqrt{h_1^2 + h_2^2}\,(\cos\theta)^3}$$

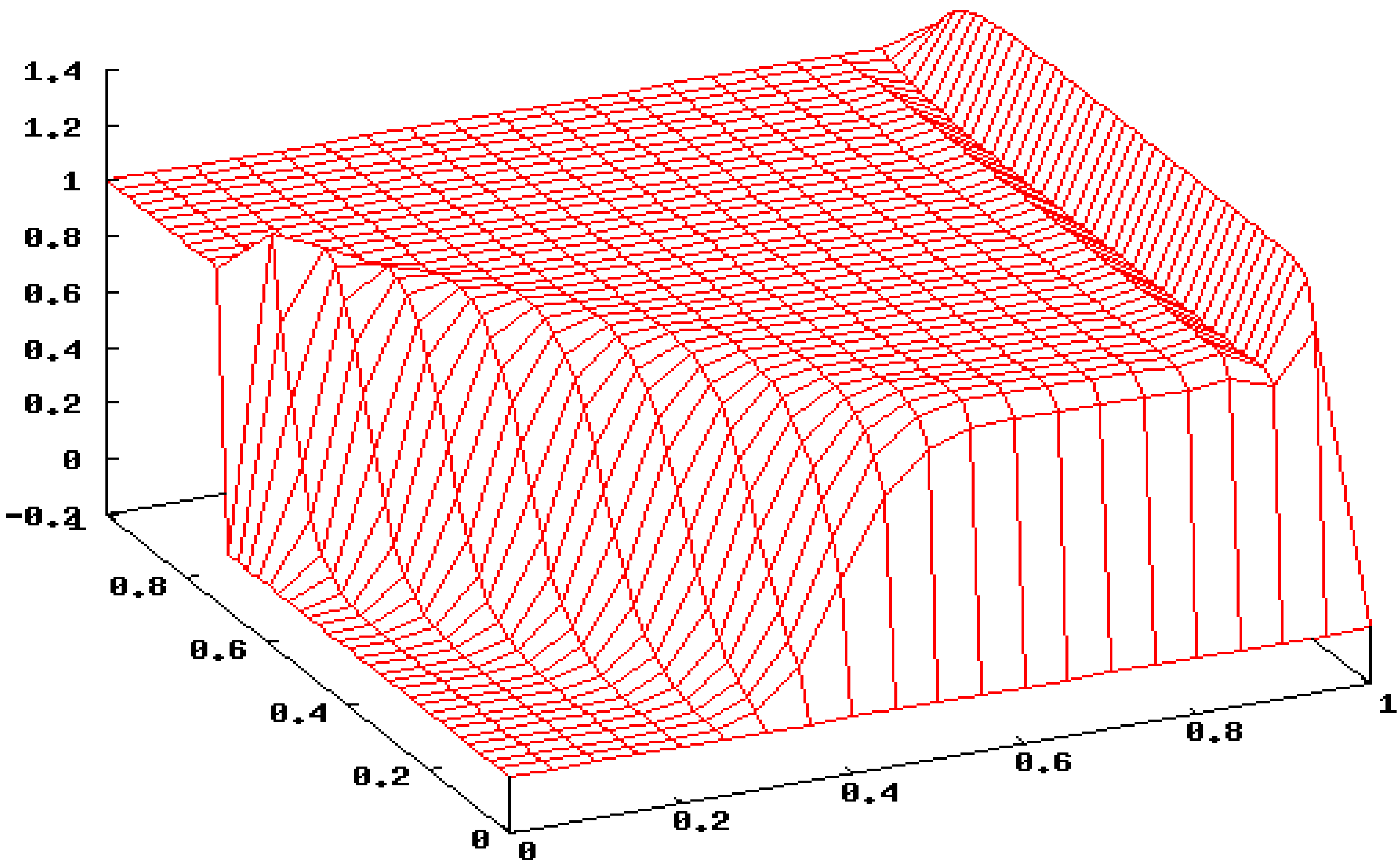
– optimal C depends on the mesh and the data of the problem

– for $\theta = \pi/3$, the boundary layer at $x = 1$ and $h_1 = 2\,h_2$,
we get $C \approx 0.85 \Rightarrow$ spurious oscillations for $C = 0.7$

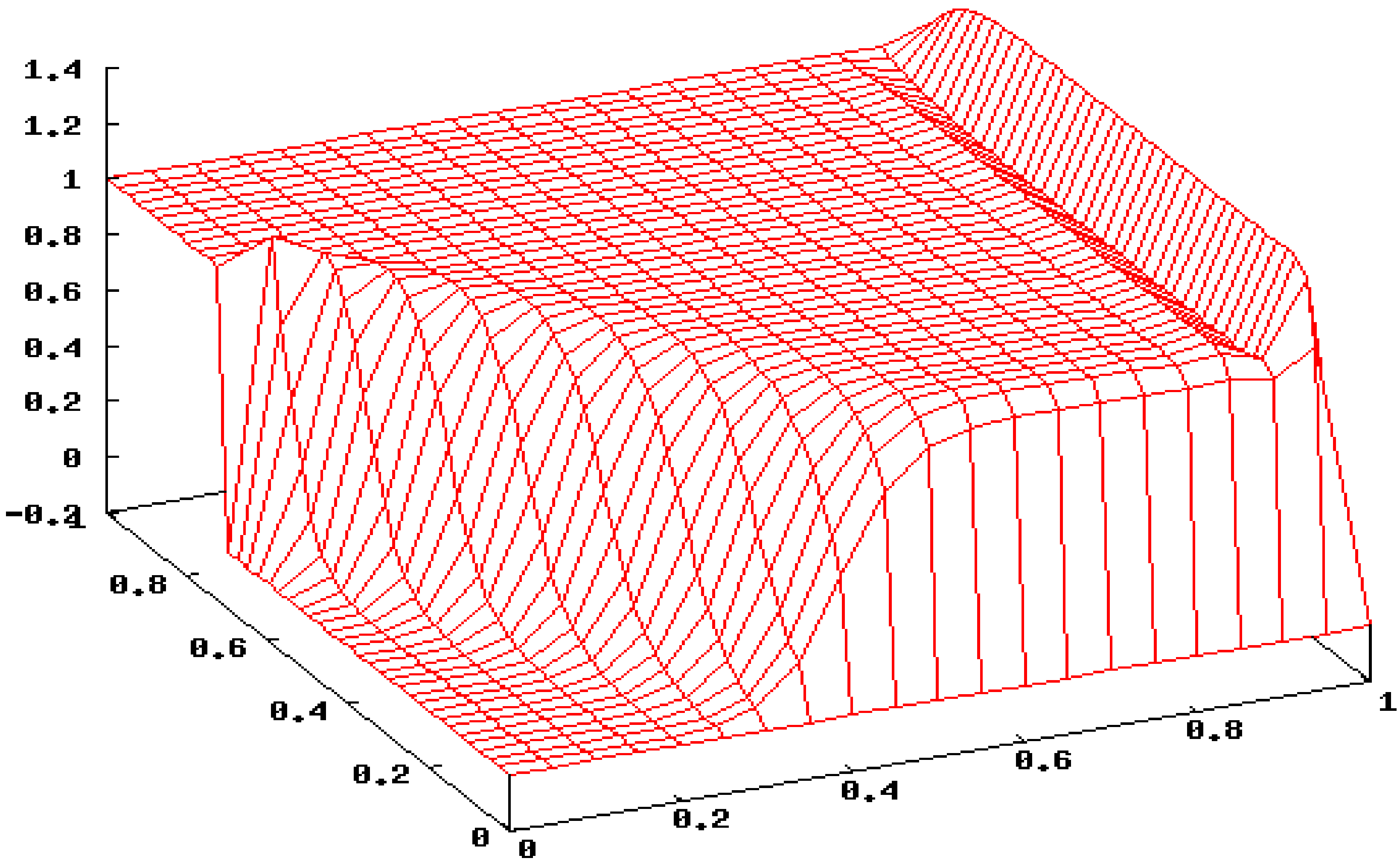
Example 2, Codina's method ($C = 0.7$), mesh 21x41



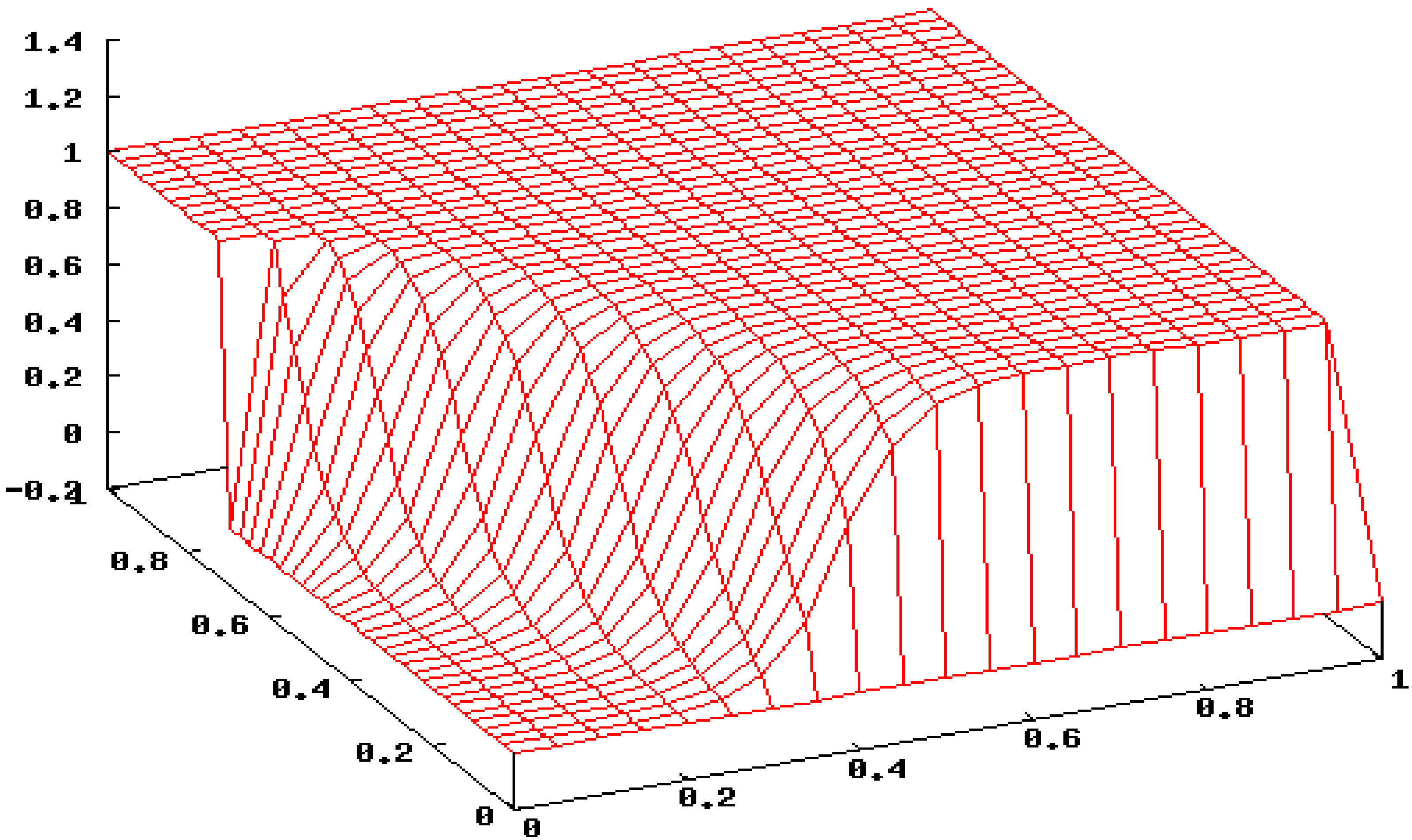
Example 2, do Carmo, Galeão (1991), mesh 21x41



Example 2, Burman, Ern (modified), mesh 21x41

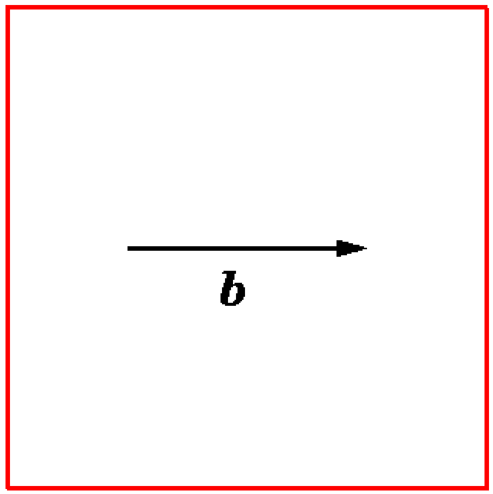


Example 2, Mizukami, Hughes, mesh 21x41



Example 3 (convection with a pw. linear source term)

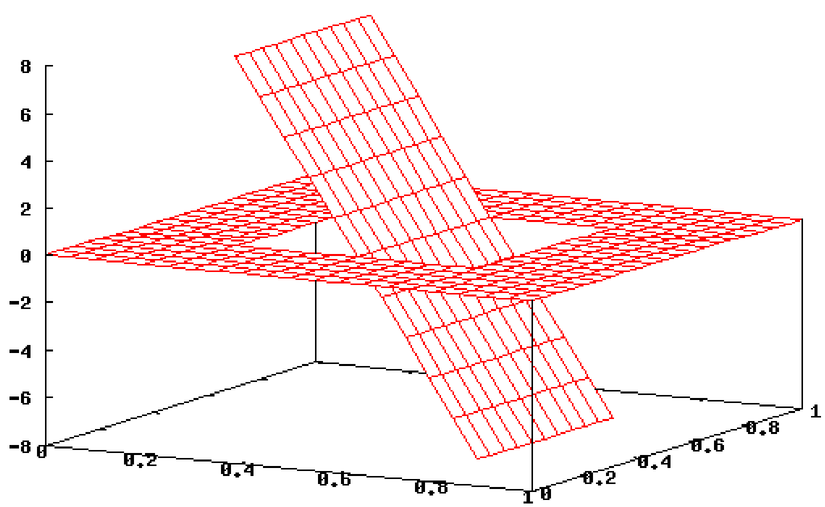
$u = 0$



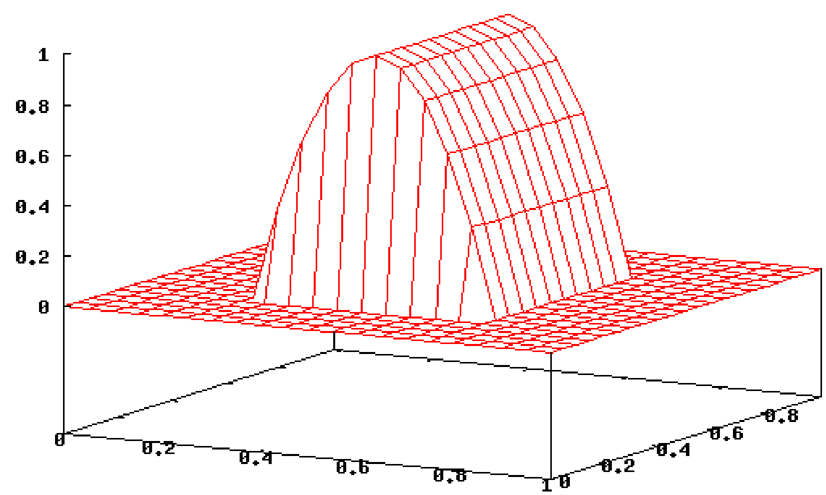
$\varepsilon = 10^{-8}$

$|\mathbf{b}| = 1$

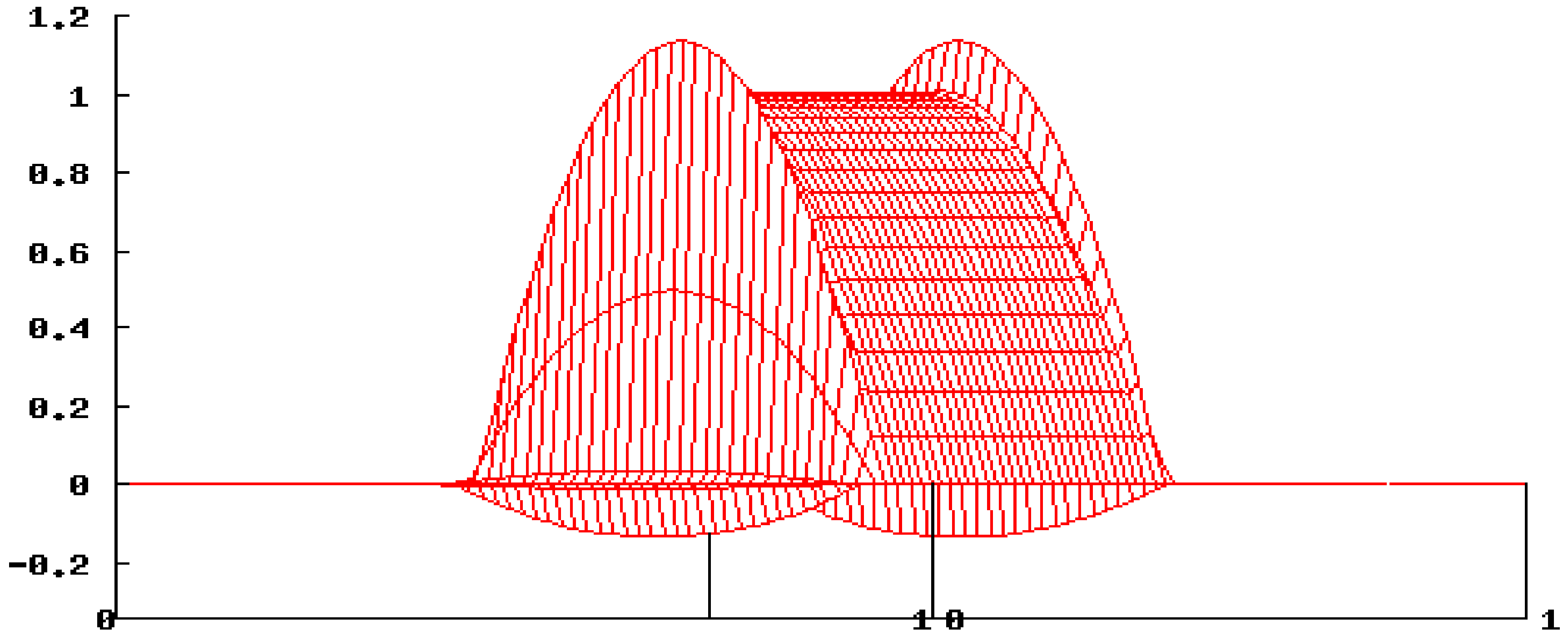
f ... pw. linear



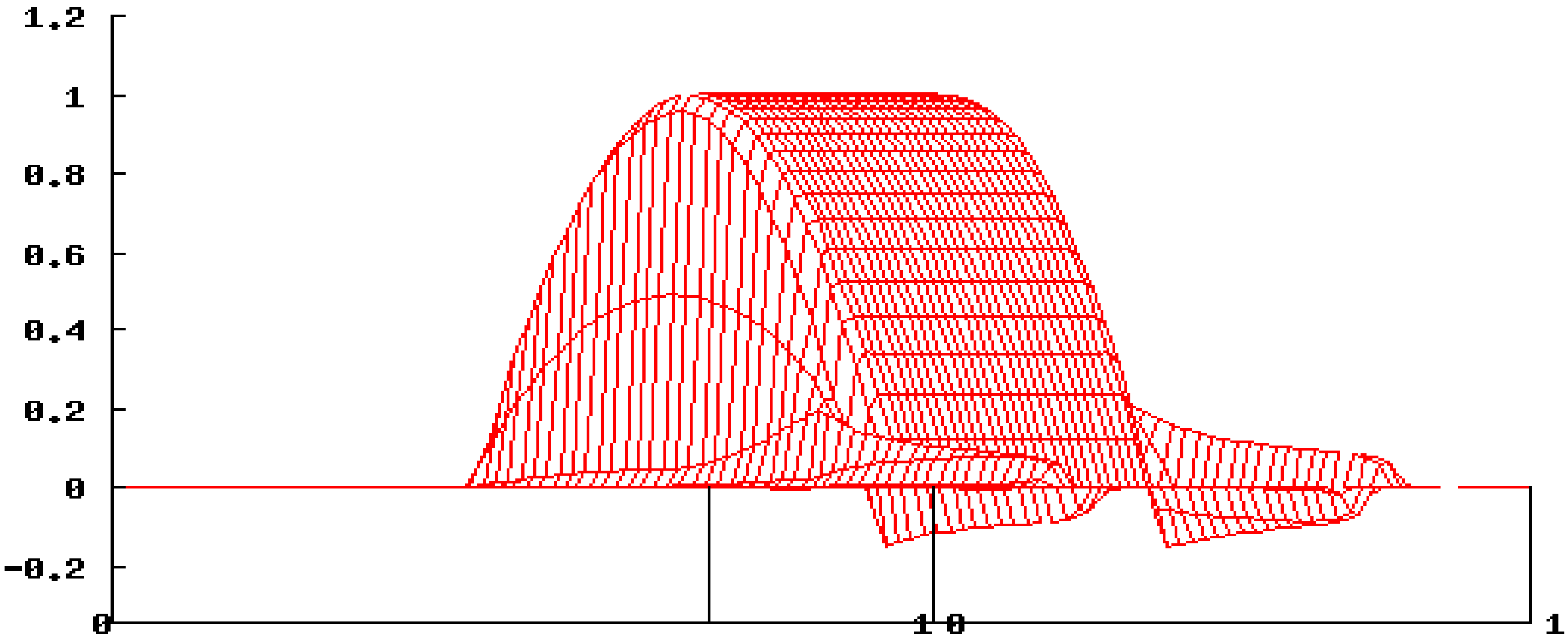
rhs f



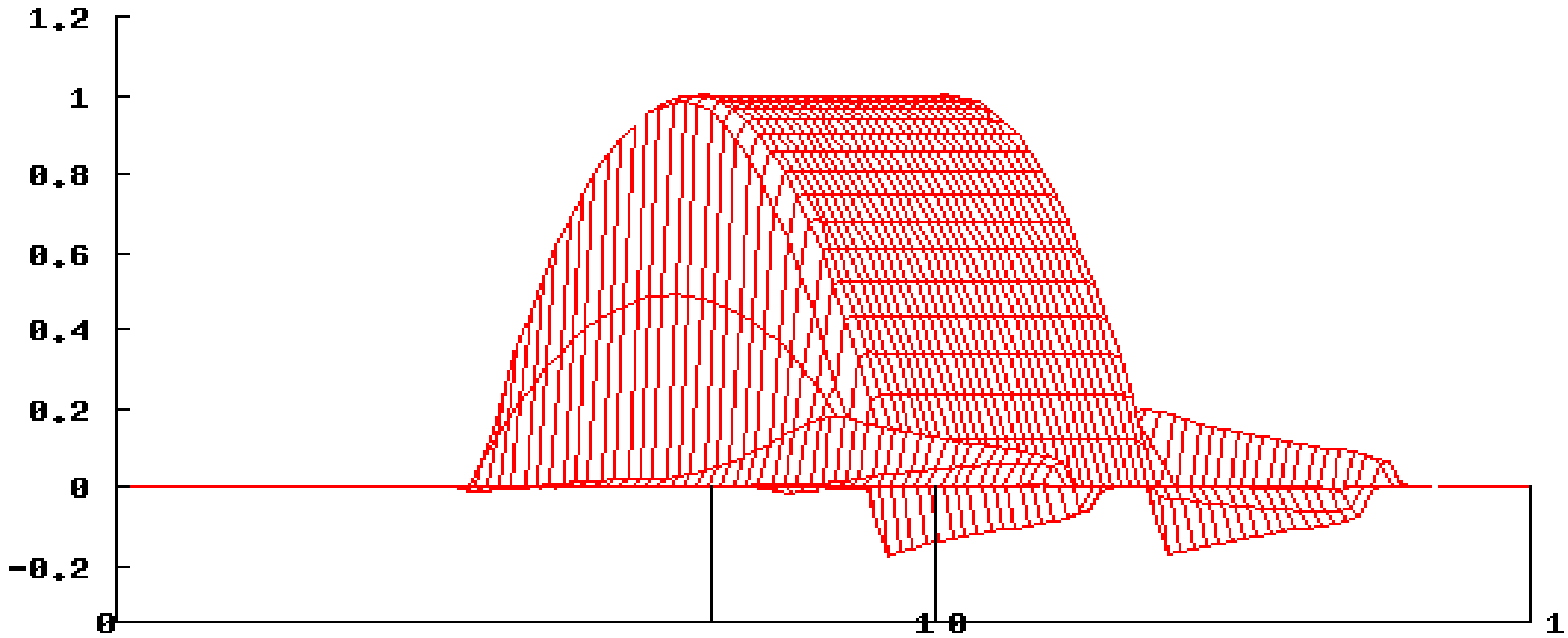
Example 3, SUPG method, mesh 65x65



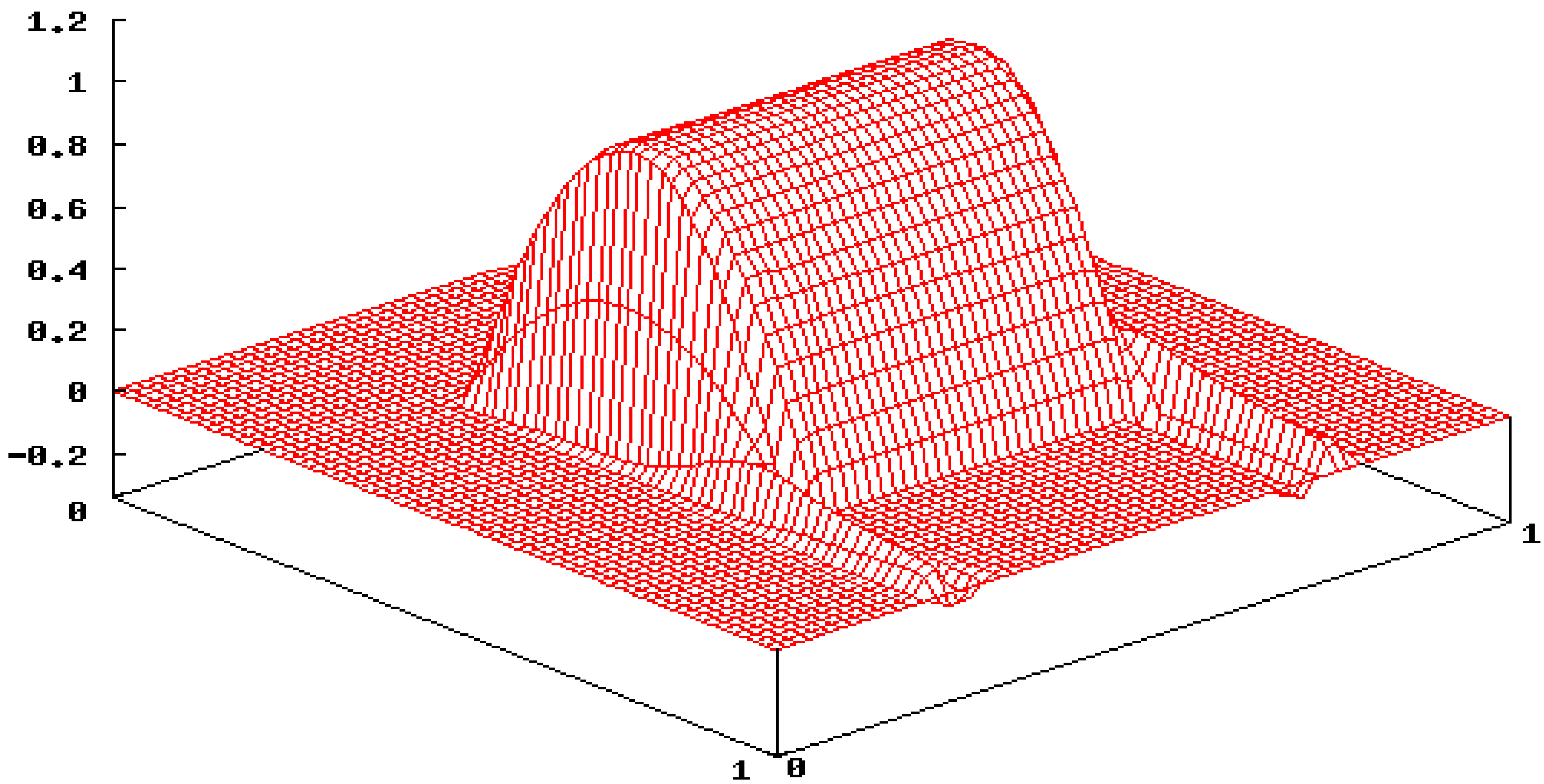
Example 3, Codina's method ($C = 0.7$), mesh 65x65



Example 3, do Carmo, Galeão (1991), mesh 65x65



Example 3, do Carmo, Galeão (1991), mesh 65x65

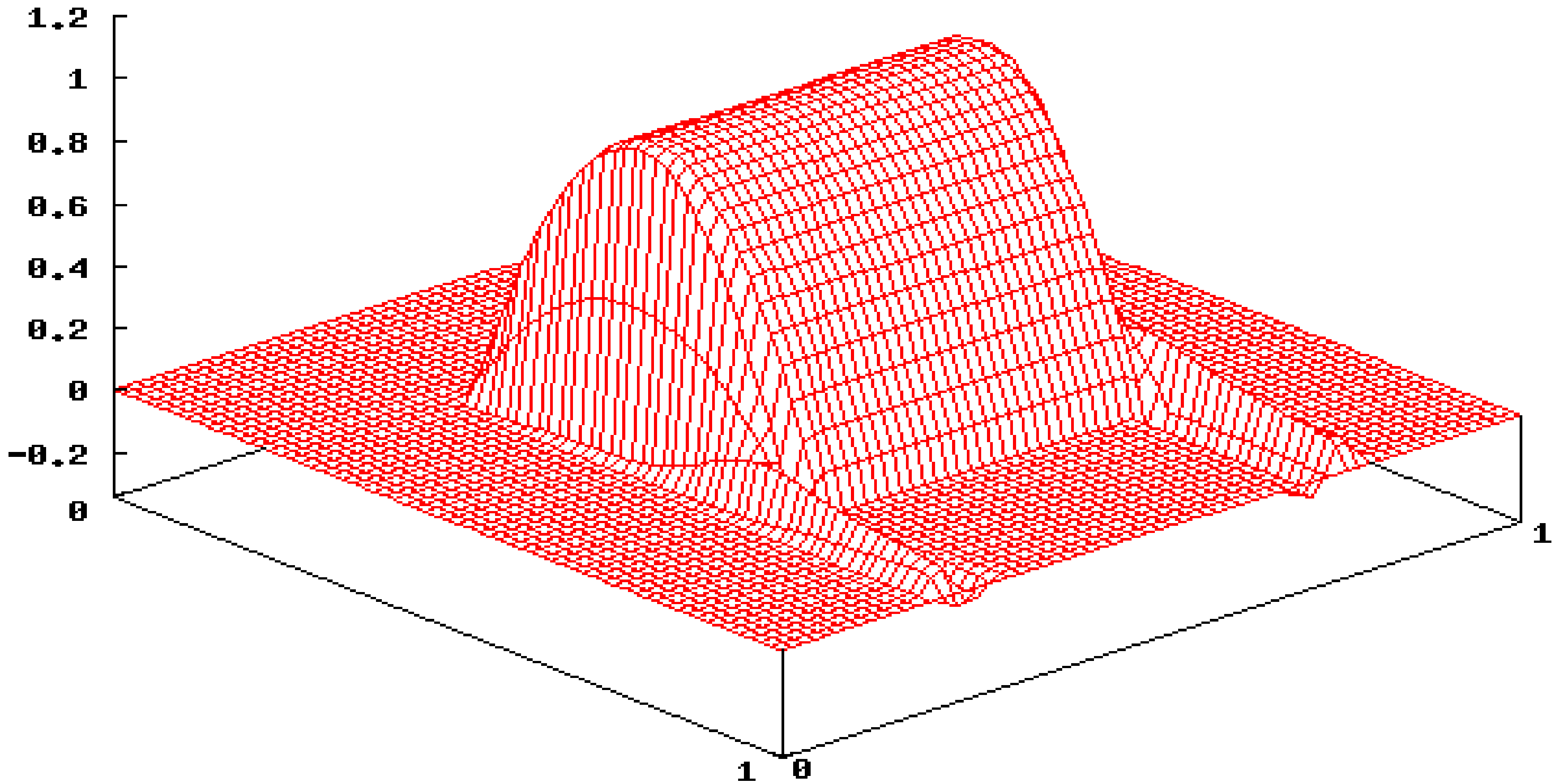


Example 3, do Carmo, Galeão (1991), mesh 65x65

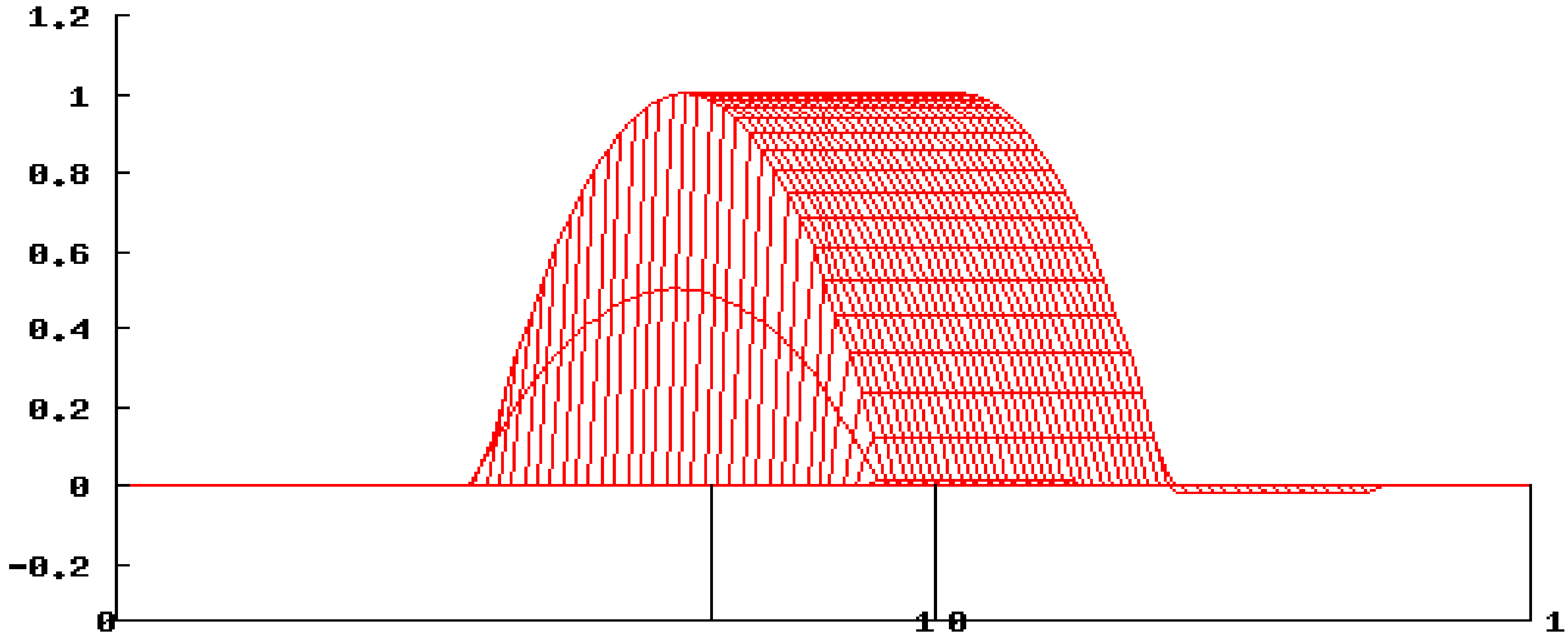
Such behaviour has to be expected from

any discontinuity–capturing method of the considered type

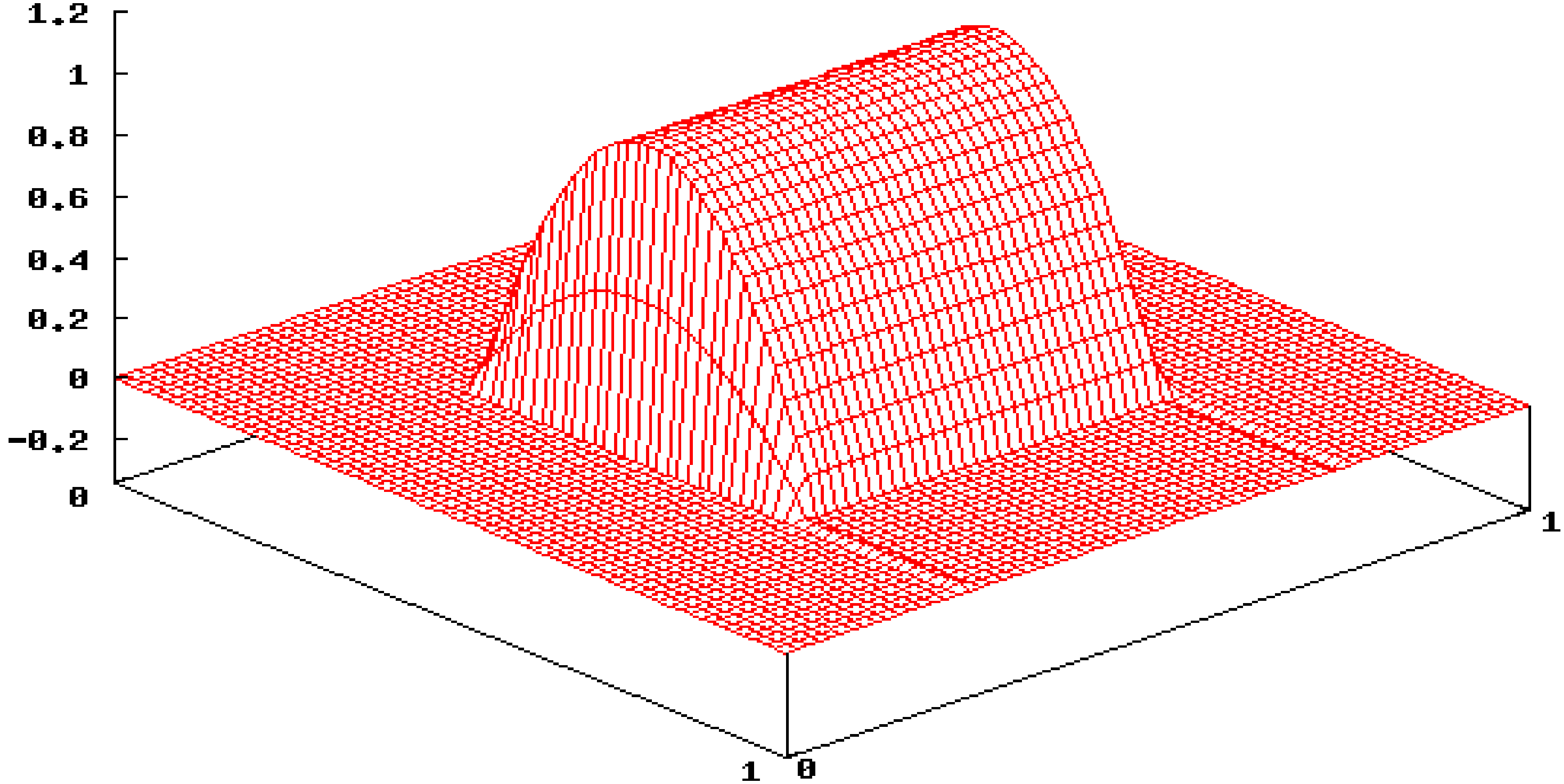
as soon as it removes the oscillations of the SUPG method.



Example 3, Mizukami, Hughes, mesh 65x65



Example 3, Mizukami, Hughes, mesh 65x65



Solution of the nonlinear problems

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

– choose u_h^0

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

– choose u_h^0

– for $k = 1, 2, \ldots$ compute

$$u_h^k \;:\; \qquad a_h(u_h^{k-1}; u_h^k \;, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

– choose u_h^0

– for $k = 1, 2, \ldots$ compute

$$u_h^{k,\star}: \qquad a_h(u_h^{k-1}; u_h^{k,\star}, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

– choose u_h^0

– for $k = 1, 2, \ldots$ compute

$$u_h^{k,\star}: \qquad a_h(u_h^{k-1}; u_h^{k,\star}, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

– damping: $\qquad u_h^k = u_h^{k-1} + \omega_k \, (u_h^{k,\star} - u_h^{k-1})$

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Simple iterations

– choose u_h^0

– for $k = 1, 2, \ldots$ compute

$$u_h^{k,\star}: \qquad a_h(u_h^{k-1}; u_h^{k,\star}, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

– damping: $\qquad u_h^k = u_h^{k-1} + \omega_k\, (u_h^{k,\star} - u_h^{k-1})$

$\omega_k \in (0, 1]$ chosen in a dynamic way

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Newton's method

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Newton's method

– difficult to implement (non–smooth operator)

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Newton's method

– difficult to implement (non–smooth operator)

– good initial approximation u_h^0 needed

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Newton's method

– difficult to implement (non–smooth operator)

– good initial approximation u_h^0 needed
  ($\longrightarrow$ use simple iterations first)

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

Newton's method

– difficult to implement (non–smooth operator)

– good initial approximation u_h^0 needed
(⟶ use simple iterations first)

– may help in a later stage of the iterative process

Solution of the nonlinear problems

Discrete solution u_h satisfies

$$a_h(u_h;u_h,v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

$a_h(u_h;\cdot,\cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

In some cases it is very difficult to compute the discrete solution

Solution of the nonlinear problems

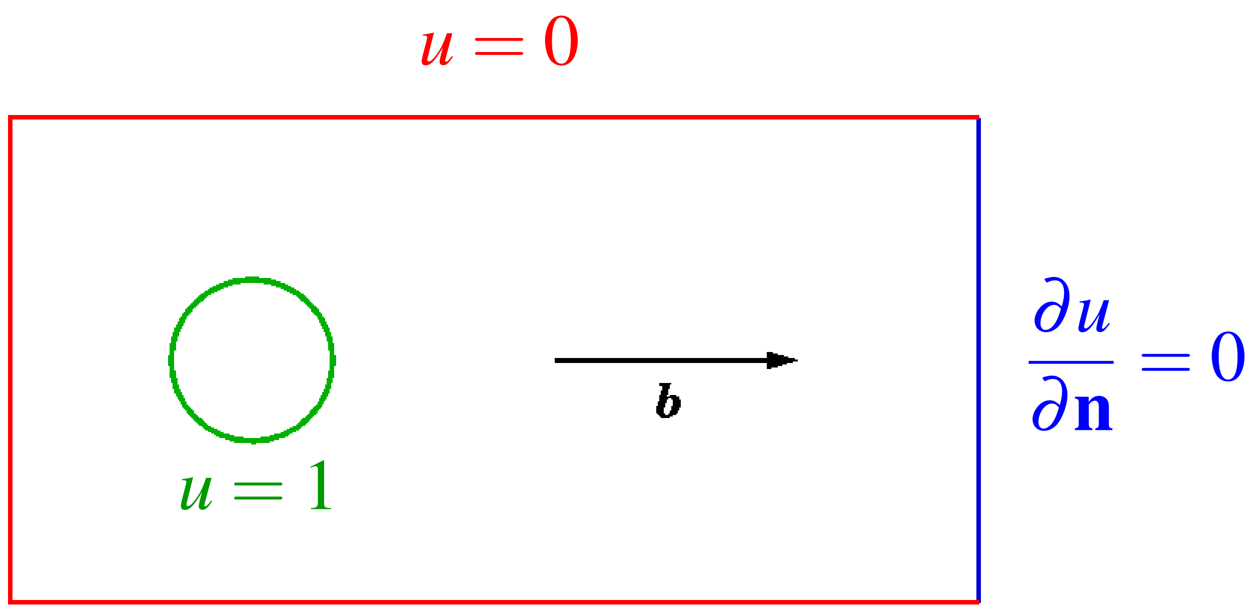
Discrete solution u_h satisfies

$$a_h(u_h; u_h, v_h) = \langle f, v_h \rangle \qquad \forall\, v_h \in V_h$$

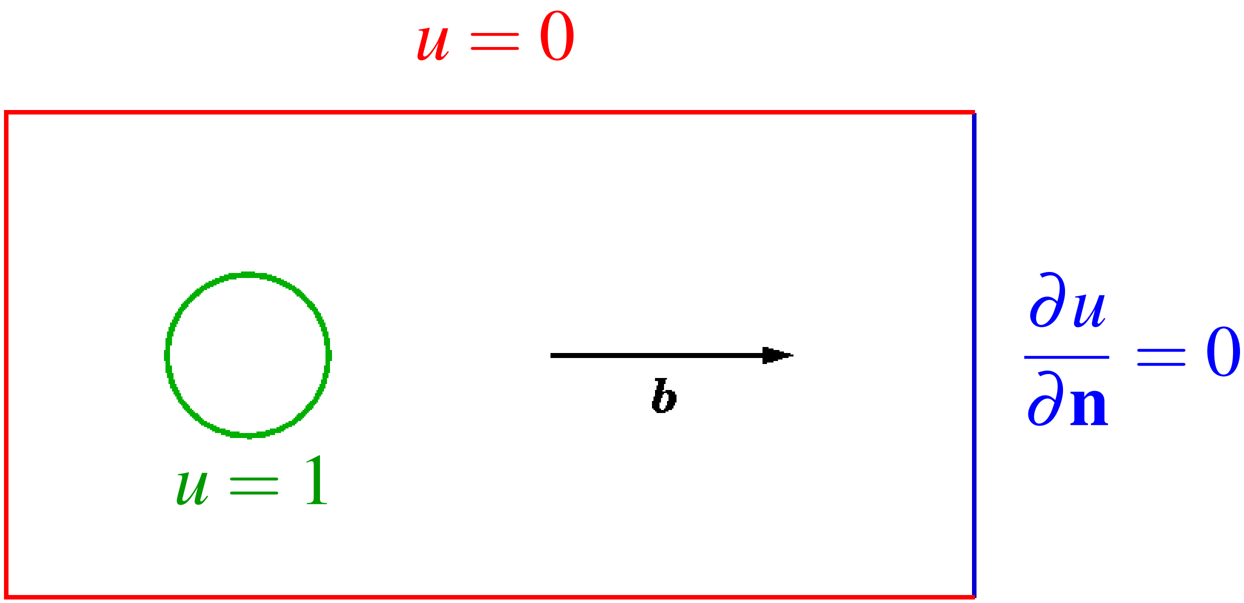
$a_h(u_h; \cdot, \cdot)$ is a bilinear form; the first argument used to define $\widetilde{\varepsilon}$

In some cases it is very difficult to compute the discrete solution (it depends on the problem, discretization, parameters, mesh).

Example 4 (P. Hemker's problem)



Example 4 (P. Hemker's problem)

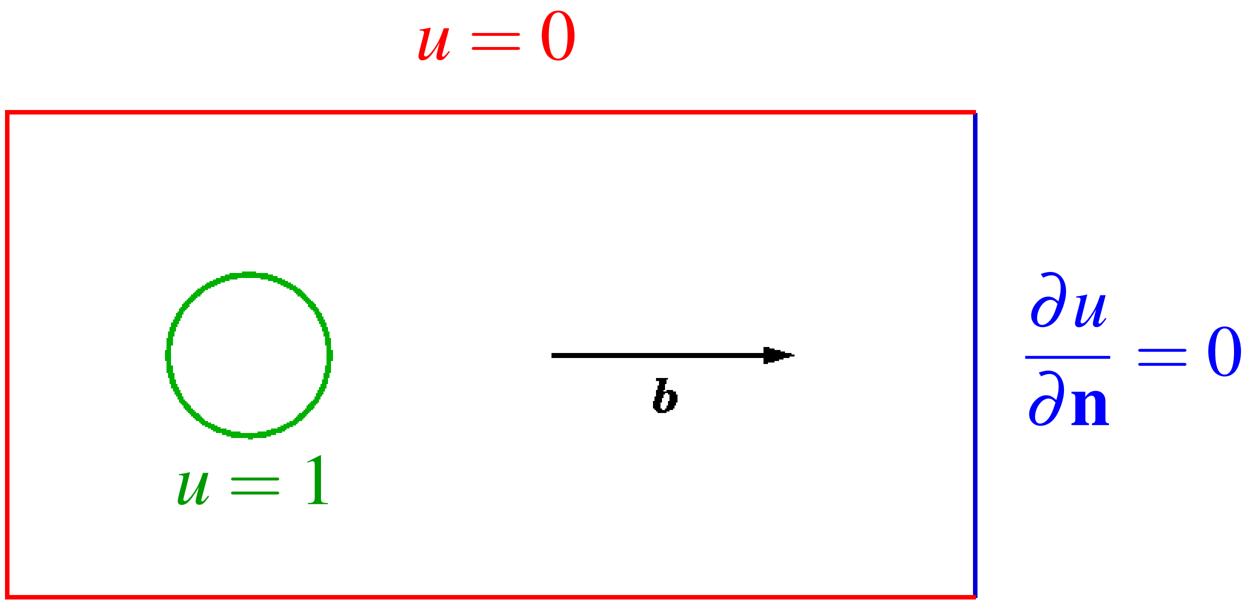


$$\varepsilon = 10^{-6}$$

$$|\mathbf{b}| = 1$$

$$f = 0$$

Example 4 (P. Hemker's problem)



$$\varepsilon = 10^{-6}$$

$$|\mathbf{b}| = 1$$

$$f = 0$$

For many of the methods, we were not able to compute the discrete solution!!!

Conclusions

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

– methods also work for higher order finite elements and for nonconforming finite elements

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

– methods also work for higher order finite elements and for nonconforming finite elements

– (modified) method of Codina is one of the best ones

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

– methods also work for higher order finite elements and for nonconforming finite elements

– (modified) method of Codina is one of the best ones

– not clear how to choose parameters (depend on data, mesh)

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

– methods also work for higher order finite elements and for nonconforming finite elements

– (modified) method of Codina is one of the best ones

– not clear how to choose parameters (depend on data, mesh)

– also for simple problems, any of the methods can give a solution with non–negligible oscillations

Conclusions

– many of the considered discontinuity–capturing methods often substantially reduce the oscillations appearing in the SUPG solutions without an excessive smearing of the layers

– methods also work for higher order finite elements and for nonconforming finite elements

– (modified) method of Codina is one of the best ones

– not clear how to choose parameters (depend on data, mesh)

– also for simple problems, any of the methods can give a solution with non–negligible oscillations

– sometimes very difficult to solve the nonlinear problems

Conclusions

– none of the methods is reliable

Conclusions

– none of the methods is reliable

– in general, it is still completely open how to obtain oscillation–free solutions using the considered class of discontinuity–capturing methods

Conclusions

– none of the methods is reliable

– in general, it is still completely open how to obtain oscillation–free solutions using the considered class of discontinuity–capturing methods

– new ideas necessary

Conclusions

– none of the methods is reliable

– in general, it is still completely open how to obtain oscillation–free solutions using the considered class of discontinuity–capturing methods

– new ideas necessary

– an alternative is the improved Mizukami–Hughes method