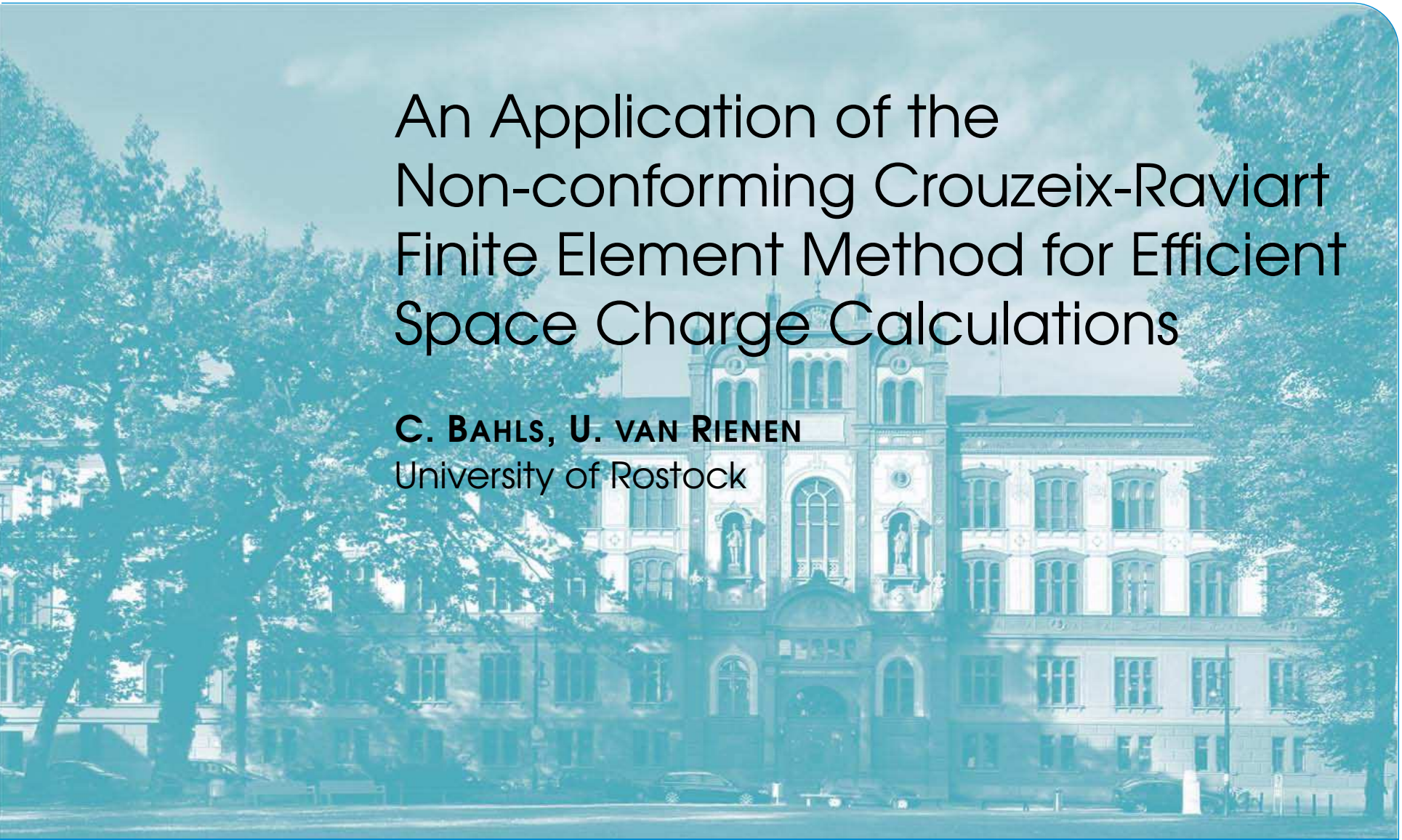
Universität Rostock

Traditio et Innovatio



An Application of the Non-conforming Crouzeix-Raviart Finite Element Method for Efficient Space Charge Calculations

C. Bahls, U. van Rienen
University of Rostock

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta u(x) = f(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$u(x) = g_D(x), \qquad \forall x \in \partial\Omega_D.$$
$$\nabla u(x) \cdot n(x) = g_N(x), \qquad \forall x \in \partial\Omega_N.$$

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta u(x) = f(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$u(x) = g_D(x), \qquad \forall x \in \partial\Omega_D.$$
$$\nabla u(x) \cdot n(x) = g_N(x), \qquad \forall x \in \partial\Omega_N.$$

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta u(x) = f(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$u(x) = g_D(x), \qquad \forall x \in \partial\Omega_D.$$
$$\nabla u(x) \cdot n(x) = g_N(x), \qquad \forall x \in \partial\Omega_N.$$

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta u(x) = f(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$u(x) = g_D(x), \qquad \forall x \in \partial\Omega_D.$$
$$\nabla u(x) \cdot n(x) = g_N(x), \qquad \forall x \in \partial\Omega_N.$$

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta u(x) = f(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$u(x) = g_D(x), \qquad \forall x \in \partial\Omega_D,$$
$$\nabla u(x) \cdot n(x) = g_N(x), \qquad \forall x \in \partial\Omega_N.$$

Space Charge Calculation

- Current and future accelerator design requires efficient 3D space charge calculations. These computations should be as efficient as possible
- One possible approach is Particle-in-Cell (PIC), especially the Particle-Mesh method which calculates the potential in the rest-frame of the bunch
- This computation usually is done by solving Poisson's equation

$$-\Delta\, \mathrm{u}(x) \quad = \quad \mathrm{f}(x), \qquad \forall x \in \Omega.$$

subject to some boundary conditions:

$$\begin{aligned} \mathrm{u}(x) &= g_D(x), && \forall x \in \partial\Omega_D, \\ \nabla \mathrm{u}(x) \cdot n(x) &= g_N(x), && \forall x \in \partial\Omega_N. \end{aligned}$$

Space Charge Calculations Revisited

- We are aiming at computing the self-field of the bunch.
- So we are estimating a solution to Gauss' Law:

$$\operatorname{div} \mathbf{D} = \rho,$$

where $\mathbf{D}$ denotes the dielectric flux and ρ the charge density

- There are infinitely many solutions to that equation. (a very large subspace of all vectorial functions)
- One can add any divergence-free (curl) field to a solution without changing the divergence of the field (the divergence of a curl is zero).

Space Charge Calculations Revisited

- We are aiming at computing the self-field of the bunch.
- So we are estimating a solution to Gauss' Law:

$$\operatorname{div} \mathbf{D} = \rho,$$

where $\mathbf{D}$ denotes the dielectric flux and ρ the charge density

- There are infinitely many solutions to that equation. (a very large subspace of all vectorial functions)
- One can add any divergence-free (curl) field to a solution without changing the divergence of the field (the divergence of a curl is zero).

Space Charge Calculations Revisited

- We are aiming at computing the self-field of the bunch.
- So we are estimating a solution to Gauss' Law:

$$\operatorname{div} \mathbf{D} = \rho,$$

where $\mathbf{D}$ denotes the dielectric flux and ρ the charge density

- There are infinitely many solutions to that equation. (a very large subspace of all vectorial functions)
- One can add any divergence-free (curl) field to a solution without changing the divergence of the field (the divergence of a curl is zero).

Space Charge Calculations Revisited

- We are aiming at computing the self-field of the bunch.
- So we are estimating a solution to Gauss' Law:

$$\operatorname{div} \mathbf{D} = \rho,$$

  where $\mathbf{D}$ denotes the dielectric flux and ρ the charge density
- There are infinitely many solutions to that equation. (a very large subspace of all vectorial functions)
- One can add any divergence-free (curl) field to a solution without changing the divergence of the field (the divergence of a curl is zero).

Space Charge Calculations Revisited

- We are aiming at computing the self-field of the bunch.
- So we are estimating a solution to Gauss' Law:

$$\operatorname{div} \mathbf{D} = \rho,$$

  where $\mathbf{D}$ denotes the dielectric flux and ρ the charge density
- There are infinitely many solutions to that equation. (a very large subspace of all vectorial functions)
- One can add any divergence-free (curl) field to a solution without changing the divergence of the field (the divergence of a curl is zero).

Space Charge Calculations Revisited #2

- We are only interested in curl-free solutions of Gauss' law
- For this we will use fields which are gradients of a scalar function.

$$\boldsymbol{\Psi} = -\text{grad}\, \mathrm{u}\ .$$

- Then our equations become

$$\begin{aligned} \text{grad}\, \mathrm{u}(x) + \boldsymbol{\Psi}(x) &= \mathbf{0} \\ \text{div}\, \varepsilon(x)\boldsymbol{\Psi}(x) &= \rho(x), \end{aligned}$$

- Which usually get shortened to:

$$-\text{div}\ \varepsilon(x)\ \text{grad}\, \mathrm{u}(x) = \rho(x).$$

- Or if $\varepsilon(x)$ is isotropic or constant: $-\Delta \mathrm{u}(x) = \varepsilon^{-1}\rho(x)$.

Space Charge Calculations Revisited #2

- We are only interested in curl-free solutions of Gauss' law
- For this we will use fields which are gradients of a scalar function.

$$\boldsymbol{\Psi} = -\mathrm{grad}\,\mathrm{u}\ .$$

- Then our equations become

$$\begin{aligned} \mathrm{grad}\,\mathrm{u}(x) + \boldsymbol{\Psi}(x) &= \mathbf{0} \\ \mathrm{div}\,\varepsilon(x)\boldsymbol{\Psi}(x) &= \rho(x), \end{aligned}$$

- Which usually get shortened to:

$$-\mathrm{div}\ \varepsilon(x)\ \mathrm{grad}\,\mathrm{u}(x) = \rho(x).$$

- Or if $\varepsilon(x)$ is isotropic or constant: $-\Delta \mathrm{u}(x) = \varepsilon^{-1}\rho(x)$.

Space Charge Calculations Revisited #2

- We are only interested in curl-free solutions of Gauss' law
- For this we will use fields which are gradients of a scalar function.

$$\boldsymbol{\Psi} = -\operatorname{grad} \mathrm{u} \ .$$

- Then our equations become

$$\begin{aligned} \operatorname{grad} \mathrm{u}(x) + \boldsymbol{\Psi}(x) &= \mathbf{0} \\ \operatorname{div} \varepsilon(x) \boldsymbol{\Psi}(x) &= \rho(x), \end{aligned}$$

- Which usually get shortened to:

$$-\operatorname{div} \varepsilon(x) \operatorname{grad} \mathrm{u}(x) = \rho(x).$$

- Or if $\varepsilon(x)$ is isotropic or constant: $-\Delta \mathrm{u}(x) = \varepsilon^{-1} \rho(x)$.

Space Charge Calculations Revisited #2

- We are only interested in curl-free solutions of Gauss' law
- For this we will use fields which are gradients of a scalar function.

$$\boldsymbol{\Psi} = -\operatorname{grad} \mathrm{u} \ .$$

- Then our equations become

$$\begin{aligned} \operatorname{grad} \mathrm{u}(x) + \boldsymbol{\Psi}(x) &= \mathbf{0} \\ \operatorname{div} \varepsilon(x) \boldsymbol{\Psi}(x) &= \rho(x), \end{aligned}$$

- Which usually get shortened to:

$$-\operatorname{div} \varepsilon(x) \operatorname{grad} \mathrm{u}(x) = \rho(x).$$

- Or if $\varepsilon(x)$ is isotropic or constant: $-\Delta \mathrm{u}(x) = \varepsilon^{-1} \rho(x)$.

Space Charge Calculations Revisited #2

- We are only interested in curl-free solutions of Gauss' law
- For this we will use fields which are gradients of a scalar function.

$$\boldsymbol{\Psi} = -\text{grad}\, \mathrm{u}\ .$$

- Then our equations become

$$\begin{aligned} \text{grad}\, \mathrm{u}(x) + \boldsymbol{\Psi}(x) &= \mathbf{0} \\ \text{div}\, \varepsilon(x)\boldsymbol{\Psi}(x) &= \rho(x), \end{aligned}$$

- Which usually get shortened to:

$$-\text{div}\ \varepsilon(x)\ \text{grad}\, \mathrm{u}(x) = \rho(x).$$

- Or if $\varepsilon(x)$ is isotropic or constant: $-\Delta \mathrm{u}(x) = \varepsilon^{-1}\rho(x)$.

Space Charge Calculations Revisited #3

- Our currently used numerical scheme (solving $-\Delta \mathrm{u}(x) = \rho(x)/\varepsilon_0$) using a Finite Difference approximation of the Laplace operator Δ is suboptimal for estimating the electric field.
- We are loosing one order of convergence ($O(h^2) \rightarrow O(h^1)$) by having to compute the gradient from the potential.
- The discretized solution u_h on an equidistant structured mesh approximates the solution u with an order of $O(h^2)$:

$$\mathrm{u}_h(x) = \mathrm{u}(x) + O(h^2).$$

- The gradient Ψ will than be approximated with an order of $O(h^1)$:

$$\Psi_h(x) = \Psi(x) + O(h^1).$$

Space Charge Calculations Revisited #3

- Our currently used numerical scheme (solving $-\Delta \mathrm{u}(x) = \rho(x)/\varepsilon_0$) using a Finite Difference approximation of the Laplace operator Δ is suboptimal for estimating the electric field.
- We are loosing one order of convergence ($O(h^2) \rightarrow O(h^1)$) by having to compute the gradient from the potential.
- The discretized solution u_h on an equidistant structured mesh approximates the solution u with an order of $O(h^2)$:

$$\mathrm{u}_h(x) = \mathrm{u}(x) + O(h^2).$$

- The gradient Ψ will than be approximated with an order of $O(h^1)$:

$$\Psi_h(x) = \Psi(x) + O(h^1).$$

Space Charge Calculations Revisited #3

- Our currently used numerical scheme (solving $-\Delta \mathrm{u}(x) = \rho(x)/\varepsilon_0$) using a Finite Difference approximation of the Laplace operator Δ is suboptimal for estimating the electric field.
- We are loosing one order of convergence ($O(h^2) \rightarrow O(h^1)$) by having to compute the gradient from the potential.
- The discretized solution $\mathrm{u_h}$ on an equidistant structured mesh approximates the solution u with an order of $O(h^2)$:

$$\mathrm{u}_h(x) = \mathrm{u}(x) + O(h^2).$$

- The gradient Ψ will than be approximated with an order of $O(h^1)$:

$$\Psi_h(x) = \Psi(x) + O(h^1).$$

Space Charge Calculations Revisited #3

- Our currently used numerical scheme (solving $-\Delta \mathrm{u}(x) = \rho(x)/\varepsilon_0$) using a Finite Difference approximation of the Laplace operator Δ is suboptimal for estimating the electric field.
- We are loosing one order of convergence ($O(h^2) \rightarrow O(h^1)$) by having to compute the gradient from the potential.
- The discretized solution $\mathrm{u_h}$ on an equidistant structured mesh approximates the solution u with an order of $O(h^2)$:

$$\mathrm{u}_h(x) = \mathrm{u}(x) + O(h^2).$$

- The gradient $\boldsymbol{\Psi}$ will than be approximated with an order of $O(h^1)$:

$$\boldsymbol{\Psi}_h(x) = \boldsymbol{\Psi}(x) + O(h^1).$$

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law $\operatorname{div} \mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field **E**, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field $\mathbf{E}$, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Rethinking Space Charge Calculations

- So lets ask: What do we actually need for our computations?
  $\Rightarrow$ The electric field $\mathbf{E}$, as it accelerates the charged particles.
- What has the Poisson-Equation originally been derived from?
  $\Rightarrow$ Gauss' law div $\mathbf{D} = \rho$, plus some Gauging
- In our Problem setting the potential seems somewhat arbitrary - it could be calculated as an integrated field strength from the boundary of the domain.
- So instead we want to discretize and solve for the vector field directly
- The Discretization used has to be curl-free and should somehow allow for a sane definition of the Divergence of the field (e.g. be conformal)

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

$$\Psi_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

(where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain Ω)

- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

$|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

$$\boldsymbol{\Psi}_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

(where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain $\boldsymbol{\Omega}$)

- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

$|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

$$\boldsymbol{\Psi}_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

(where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain $\boldsymbol{\Omega}$)

- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

$|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

$$\boldsymbol{\Psi}_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

(where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain $\boldsymbol{\Omega}$)

- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

$|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

$$\boldsymbol{\Psi}_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

  (where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain $\boldsymbol{\Omega}$)
- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

  $|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

Raviart-Thomas Ansatz-Space

- One suitable ansatz space is $\mathbf{RT}_0$ the Raviart-Thomas space of lowest order, whose vector functions have following element-wise linear expression:

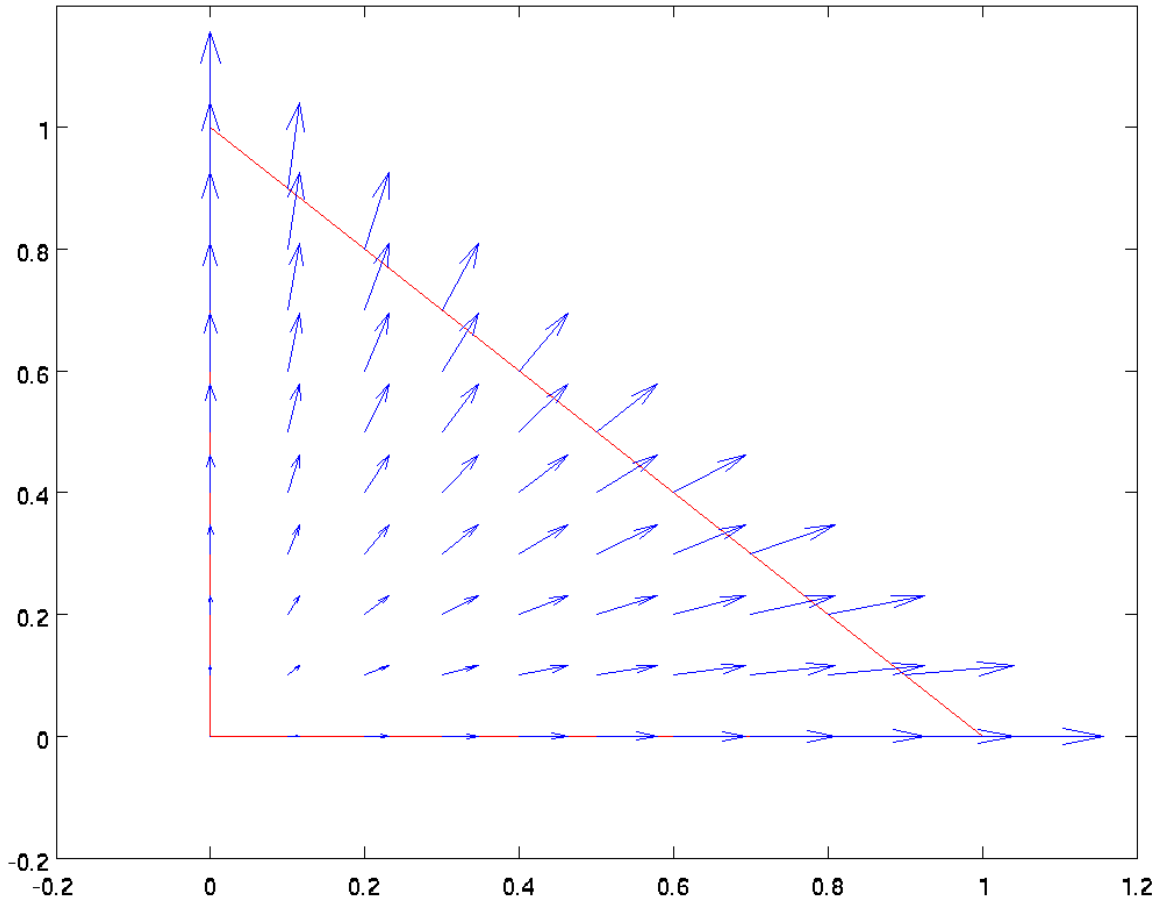
$$\boldsymbol{\Psi}_h(\mathbf{x}) \quad = \quad \mathbf{a}_k + b_k \mathbf{x},$$

  (where $\mathbf{x}$ is in the element T_k of the triangulation $\mathbf{T}$ of the domain $\boldsymbol{\Omega}$)
- For the discretisation to be conformal the normal Components of the Field have to be continuous at every inner Interface (Edges in 2D, Faces in 3D).
- So $\mathbf{RT}_0$ usually is represented by an Edge/Face-based discretization (as shown in the next few frames) using following local representation:

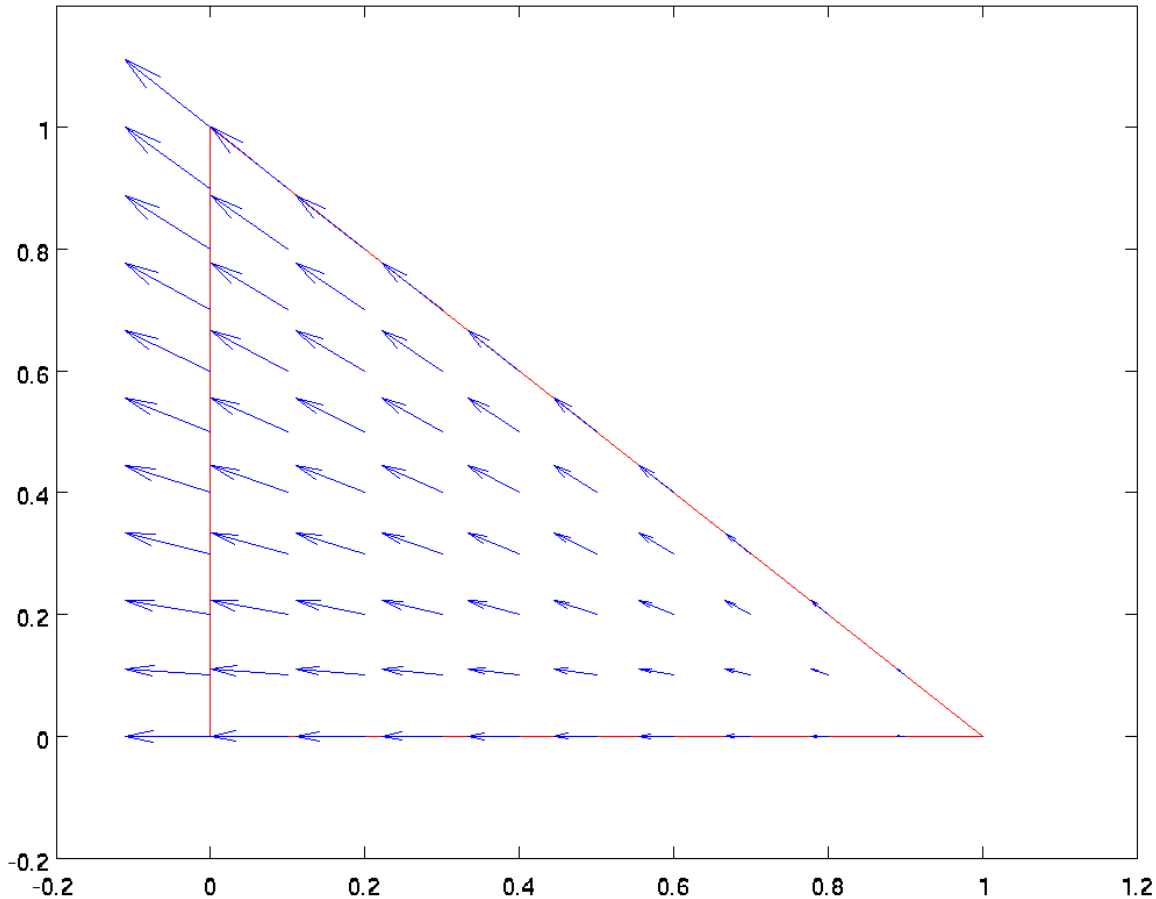
$$\psi_{E_j}(\mathbf{x}) = \sigma_j \frac{|E_j|}{2|T_k|}(\mathbf{x} - \mathbf{P}_j).$$

  $|T_k|$ is the area/volume of T_k, $|E_j|$ is the length/area of the edge/face E_j, σ_j indicates the orientation of E_j and $\mathbf{P}_j$ is the Vertex opposite to E_j.

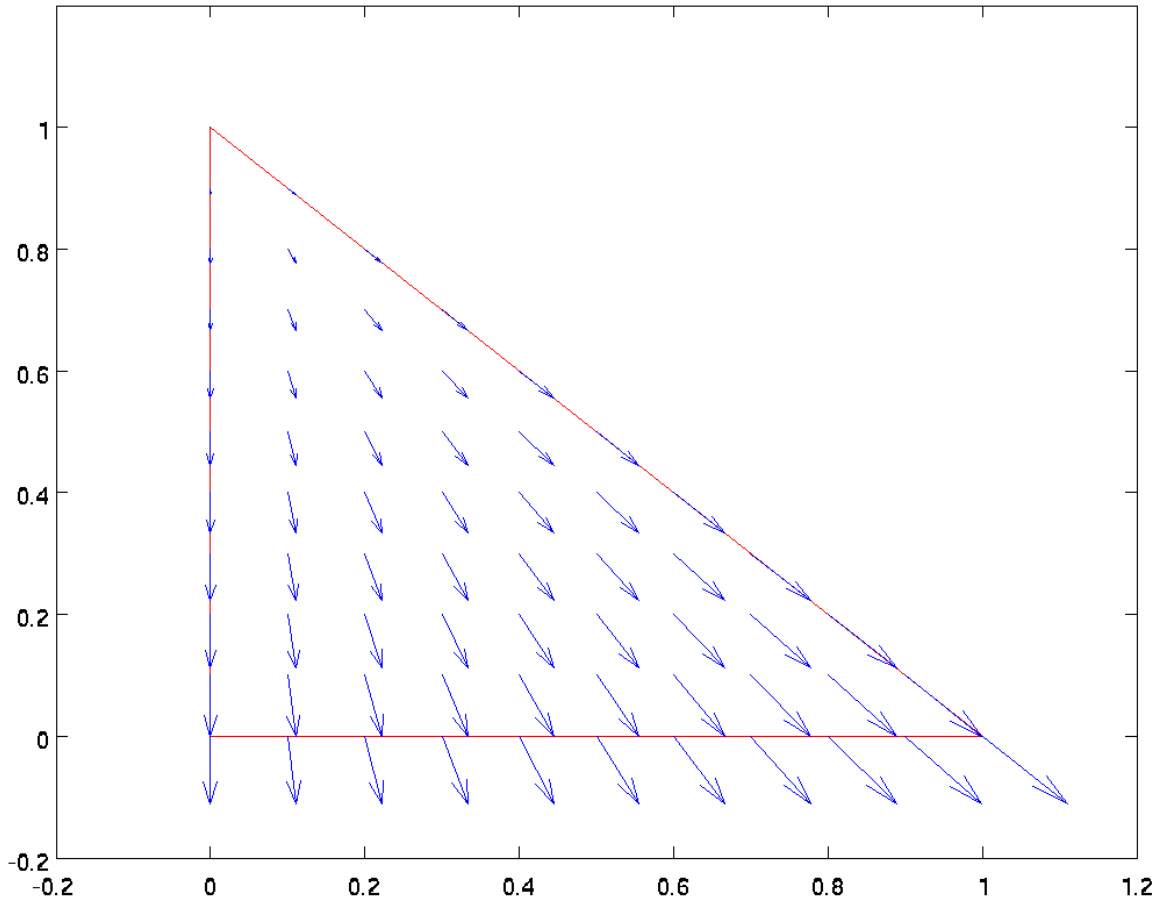
Raviart-Thomas Ansatz-Space – Basis-Element ψ_{E_1}



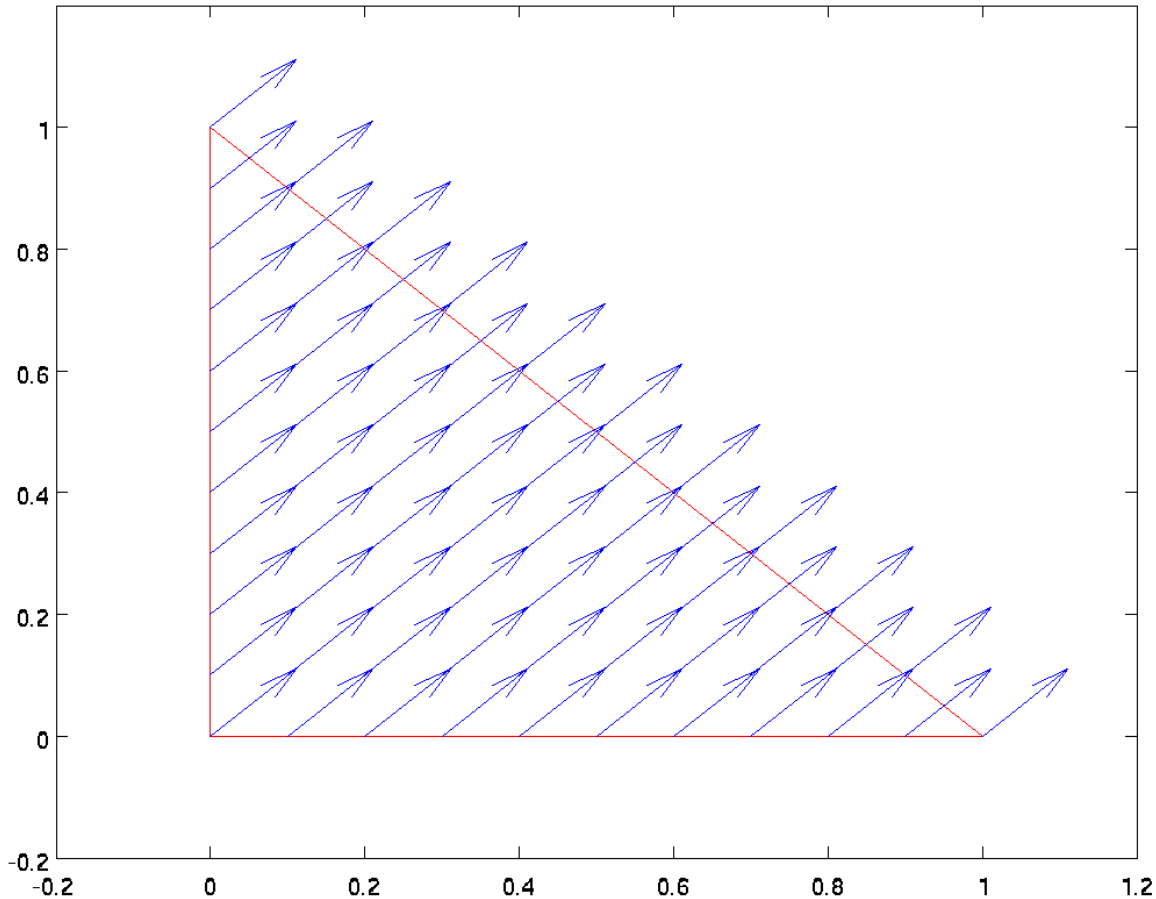
Raviart-Thomas Ansatz-Space – Basis-Element ψ_{E_2}



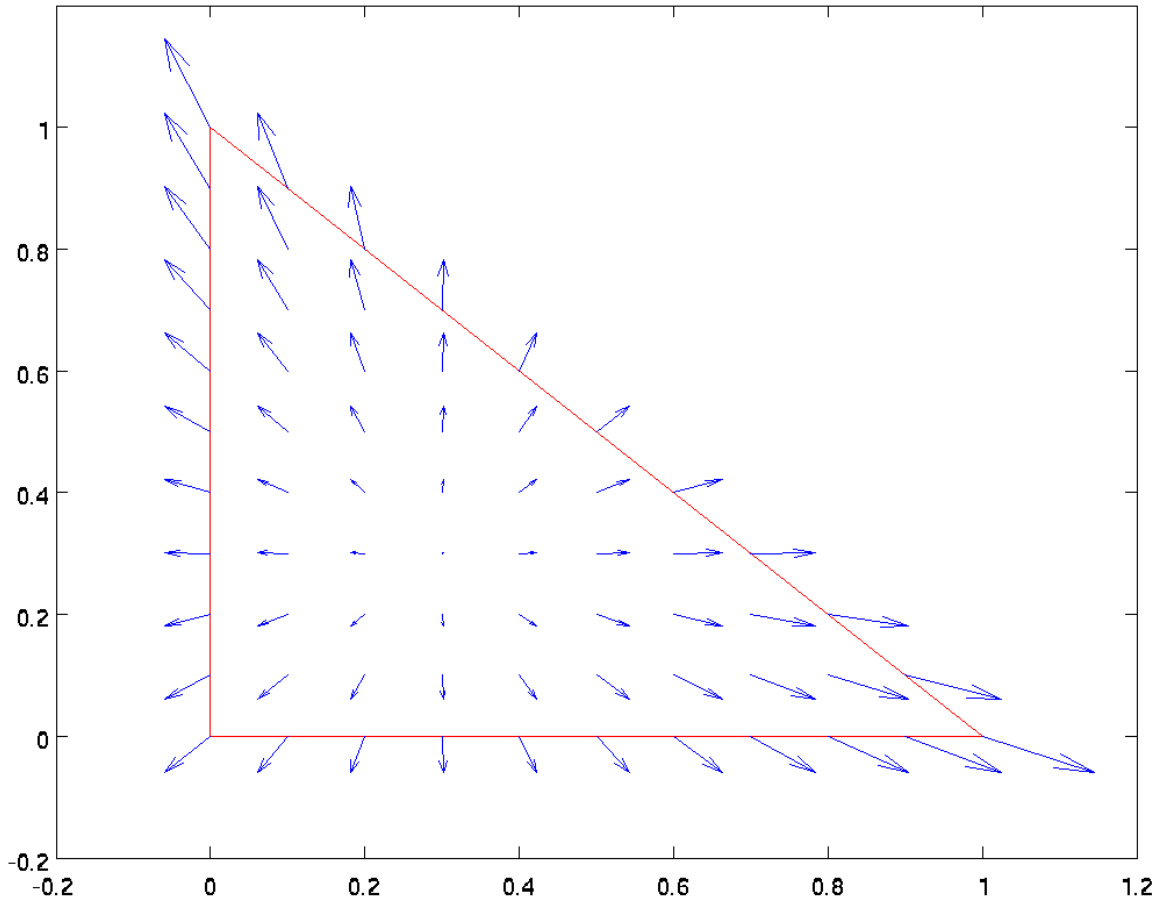
Raviart-Thomas Ansatz-Space – Basis-Element ψ_{E_3}



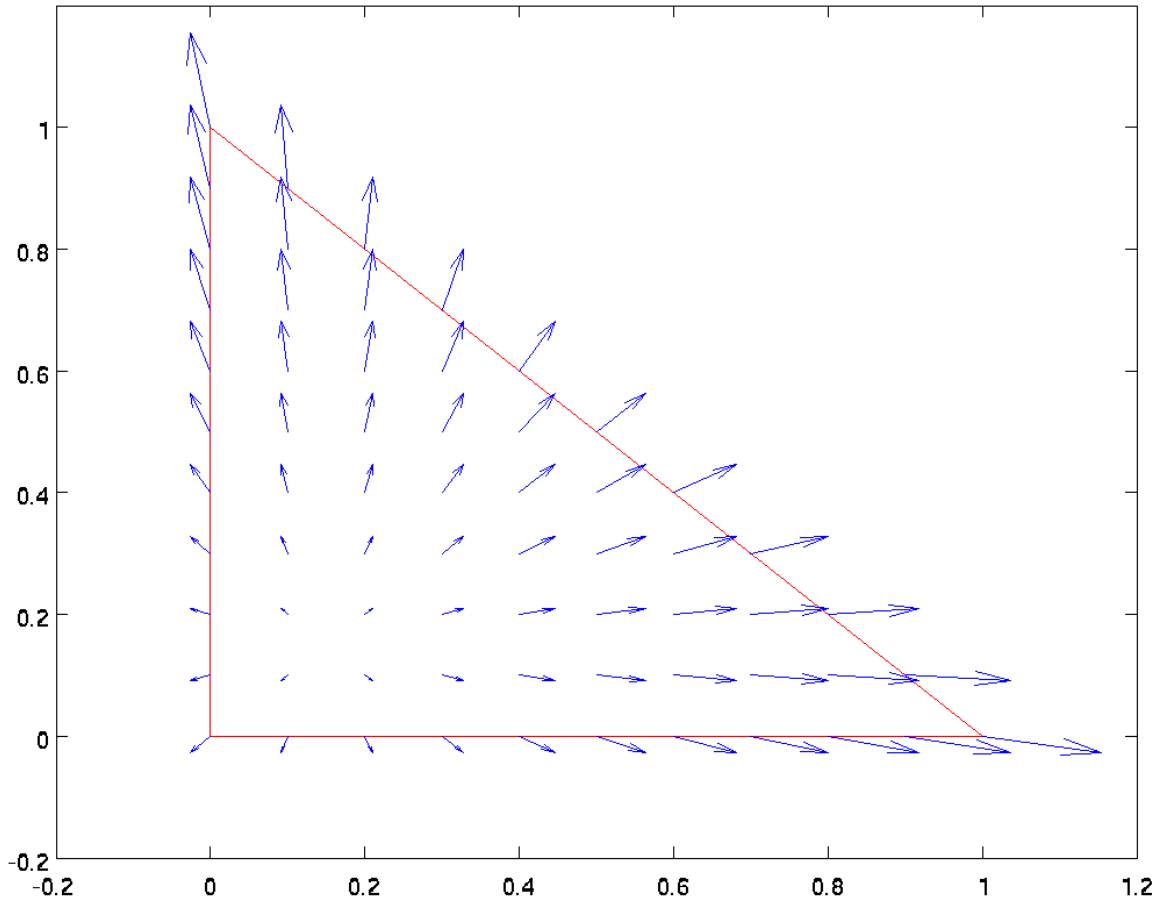
Raviart-Thomas Ansatz-Space – Constant Function



Raviart-Thomas Ansatz-Space – Bubble Function



Raviart-Thomas Ansatz-Space – Arbitrary Function



Mixed and Hybrid Formulation

- We are now using the canonical Galerkin approach for Mixed Finite Elements to compute approximate solutions for the field $\boldsymbol{\Psi}_h$ and the potential u_h:

$$\begin{aligned} \int_\Omega \boldsymbol{\tau} \cdot \boldsymbol{\Psi}_h + \int_\Omega \boldsymbol{\tau}\, \varepsilon \operatorname{grad} u_h &= 0 \quad \forall \boldsymbol{\tau} \in \mathbf{RT}_0 \\ \int_\Omega v \operatorname{div} \boldsymbol{\Psi}_h &= \int_\Omega v\, f \quad \forall v \in \mathbf{P}_1^{-1} \end{aligned}$$

- To later remove the flux variable from the system, we will relax the continuity requirement on the Ansatz-space and use the flux $\tilde{\boldsymbol{\Psi}}_h$ from the space $\mathbf{RT}_0^{-1}$ of discontinuous linear vector functions.

Mixed and Hybrid Formulation

- We are now using the canonical Galerkin approach for Mixed Finite Elements to compute approximate solutions for the field $\boldsymbol{\Psi}_h$ and the potential u_h:

$$\begin{aligned} \int_\Omega \boldsymbol{\tau} \cdot \boldsymbol{\Psi}_h + \int_\Omega \boldsymbol{\tau}\, \varepsilon \operatorname{grad} u_h &= 0 \quad \forall \boldsymbol{\tau} \in \mathbf{RT}_0 \\ \int_\Omega v \operatorname{div} \boldsymbol{\Psi}_h &= \int_\Omega v\, f \quad \forall v \in \mathbf{P}_1^{-1} \end{aligned}$$

- To later remove the flux variable from the system, we will relax the continuity requirement on the Ansatz-space and use the flux $\tilde{\boldsymbol{\Psi}}_h$ from the space $\mathbf{RT}_0^{-1}$ of discontinuous linear vector functions.

Mixed and Hybrid Formulation

- We are now using the canonical Galerkin approach for Mixed Finite Elements to compute approximate solutions for the field $\boldsymbol{\Psi}_h$ and the potential u_h:

$$\begin{aligned} \int_\Omega \boldsymbol{\tau} \cdot \boldsymbol{\Psi}_h + \int_\Omega \boldsymbol{\tau}\, \varepsilon \operatorname{grad} u_h &= 0 \quad \forall \boldsymbol{\tau} \in \mathbf{RT}_0 \\ \int_\Omega v \operatorname{div} \boldsymbol{\Psi}_h &= \int_\Omega v\, f \quad \forall v \in \mathbf{P}_1^{-1} \end{aligned}$$

- To later remove the flux variable from the system, we will relax the continuity requirement on the Ansatz-space and use the flux $\tilde{\boldsymbol{\Psi}}_h$ from the space $\mathbf{RT}_0^{-1}$ of discontinuous linear vector functions.

Mixed and Hybrid Formulation

- We are now using the canonical Galerkin approach for Mixed Finite Elements to compute approximate solutions for the field $\boldsymbol{\Psi}_h$ and the potential u_h:

$$\begin{aligned} \int_\Omega \boldsymbol{\tau} \cdot \boldsymbol{\Psi}_h + \int_\Omega \boldsymbol{\tau}\, \varepsilon \operatorname{grad} u_h &= 0 \quad \forall \boldsymbol{\tau} \in \mathbf{RT}_0 \\ \int_\Omega v \operatorname{div} \boldsymbol{\Psi}_h &= \int_\Omega v\, f \quad \forall v \in \mathbf{P}_1^{-1} \end{aligned}$$

- To later remove the flux variable from the system, we will relax the continuity requirement on the Ansatz-space and use the flux $\tilde{\boldsymbol{\Psi}}_h$ from the space $\mathbf{RT}_0^{-1}$ of discontinuous linear vector functions.

Mixed and Hybrid Formulation #2

- We than enforce the continuity of the normal component of the flux on faces by the use of Lagrange multipliers $\lambda_h \in \mathbf{M}_1^{-1}$, leading to the system:

$$\begin{aligned}
\int_\Omega \tilde{\boldsymbol{\tau}} \cdot \tilde{\boldsymbol{\Psi}}_h + \int_\Omega \tilde{\boldsymbol{\tau}}\, \varepsilon \,\mathrm{grad}\, u_h + \int_{\delta\Omega} \lambda_h\, \mathbf{n}_T \cdot \tilde{\boldsymbol{\tau}} &= 0 && \forall \tilde{\boldsymbol{\tau}} \in \mathbf{RT}_0^{-1} \\
\int_\Omega v\, \mathrm{div}\, \tilde{\boldsymbol{\Psi}}_h &= \int_\Omega v\, f && \forall v \in \mathbf{P}_1^{-1} \\
\int_{\delta\Omega} \mu\, \mathbf{n}_T \cdot \tilde{\boldsymbol{\Psi}}_h &= 0 && \forall \mu \in \mathbf{M}_1^{-1}
\end{aligned}$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} \\ \mathbf{B}^T & & \\ \mathbf{C}^T & & \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\Psi}}_h \\ u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} 0 \\ f_h \\ 0 \end{pmatrix}$$

Mixed and Hybrid Formulation #2

- We than enforce the continuity of the normal component of the flux on faces by the use of Lagrange multipliers $\lambda_h \in \mathbf{M}_1^{-1}$, leading to the system:

$$\begin{aligned}
\int_\Omega \tilde{\boldsymbol{\tau}} \cdot \tilde{\boldsymbol{\Psi}}_h + \int_\Omega \tilde{\boldsymbol{\tau}}\, \varepsilon \,\mathrm{grad}\, u_h + \int_{\delta\Omega} \lambda_h\, \mathbf{n}_T \cdot \tilde{\boldsymbol{\tau}} &= 0 && \forall \tilde{\boldsymbol{\tau}} \in \mathbf{RT}_0^{-1} \\
\int_\Omega v\, \mathrm{div}\, \tilde{\boldsymbol{\Psi}}_h &= \int_\Omega v\, f && \forall v \in \mathbf{P}_1^{-1} \\
\int_{\delta\Omega} \mu\, \mathbf{n}_T \cdot \tilde{\boldsymbol{\Psi}}_h &= 0 && \forall \mu \in \mathbf{M}_1^{-1}
\end{aligned}$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} \\ \mathbf{B}^T & & \\ \mathbf{C}^T & & \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\Psi}}_h \\ u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} 0 \\ f_h \\ 0 \end{pmatrix}$$

Mixed and Hybrid Formulation #3

- The Submatrix **A** is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\Psi}_h$ can be computed by:

$$\tilde{\Psi}_h = -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

  This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

  This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

  This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

Mixed and Hybrid Formulation #3

- The Submatrix $\mathbf{A}$ is block-diagonal, so it is easily element-wise invertable, so the flux $\tilde{\boldsymbol{\Psi}}_h$ can be computed by:

$$\tilde{\boldsymbol{\Psi}}_h \quad = \quad -\mathbf{A}^{-1}(\mathbf{B}\, u_h + \mathbf{C}\, \lambda_h)$$

- leading to following linear system of equations:

$$\begin{pmatrix} \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{B}^T\mathbf{A}^{-1}\mathbf{C} \\ \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B} & \mathbf{C}^T\mathbf{A}^{-1}\mathbf{C} \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} -f_h \\ 0 \end{pmatrix}$$

  This method is called static condensation.

- Eliminating the u_h using a Schur complement we would arrive at a variant of the Crouzeix-Raviart Finite Element Method $\Rightarrow$ which can also be derived directly.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method

- In "*An inexpensive Method for the Evaluation of the Solution of the lowest order Raviart-Thomas Mixed Method*" Marini suggested to use the $\mathbf{P}_1$ nonconforming finite element spaces $\mathbf{P}_1^{\mathrm{NC}}$.
- This $\mathbf{P}_1^{\mathrm{NC}}$ are also called Crouzeix-Raviart or loof finite element spaces.
- These finite Elements have their degrees of freedom allocated to the barycenters of their Edges/Faces, rather than their Vertices.
- The function space $\mathbf{P}_1^{\mathrm{NC}}$ actually contains the space $\mathbf{P}_1$ of linear functions.
- So $\mathbf{P}_1^{\mathrm{NC}}$ can at least represent the continuous solutions from nodal $\mathbf{P}_1$-FEM.
- As the Ansatz space is larger there is room for additional contraints, e.g. for choosing a solution with a higher order approximation of the field.

The Crouzeix-Raviart Finite Element Method - Examples #1

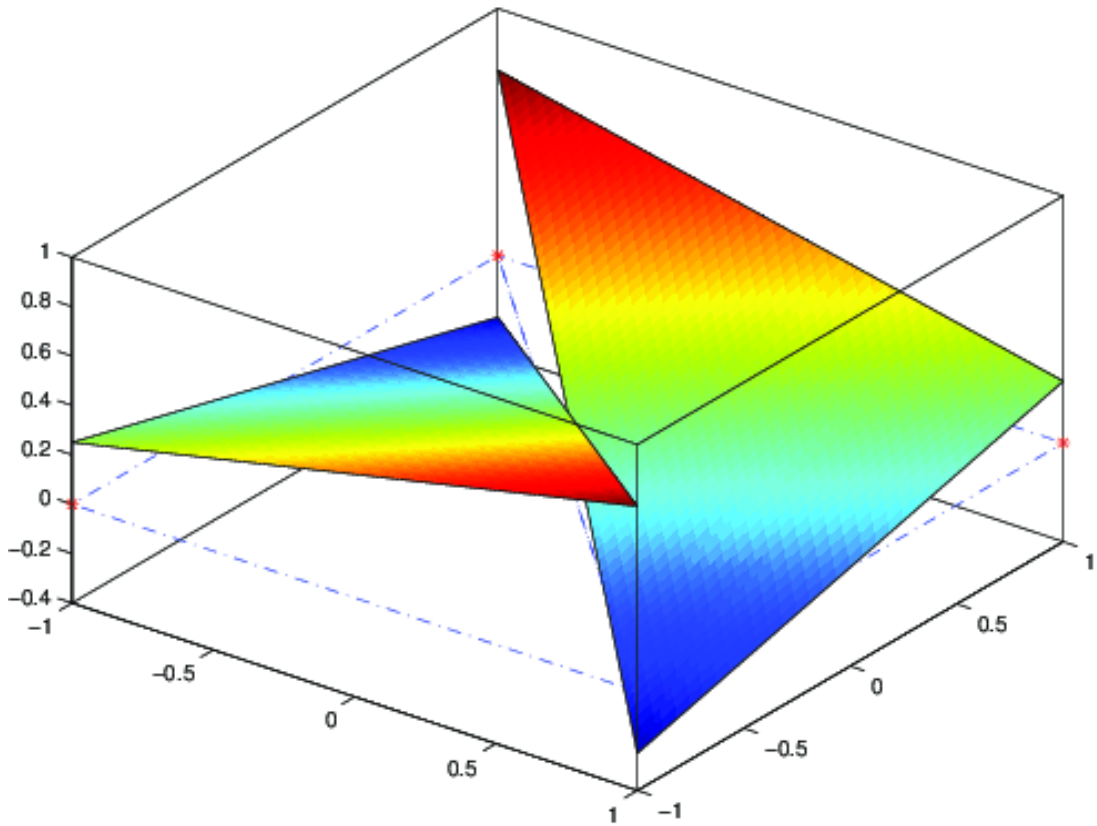


Figure: The ansatz-functions are only continuous at the midpoints of interfaces

The Crouzeix-Raviart Finite Element Method - Examples #2

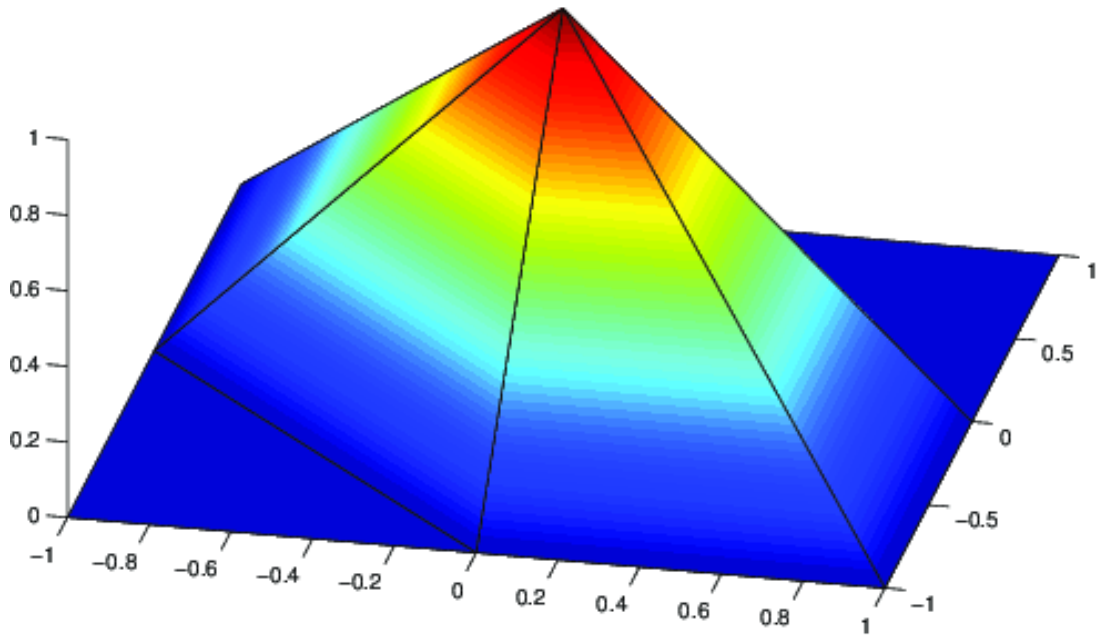


Figure: The space $\mathbf{P}_1$ of piecewise linear and continuous functions is contained in $\mathbf{P}_1^{\mathrm{NC}}$.

The Crouzeix-Raviart Finite Element Method - Examples #3

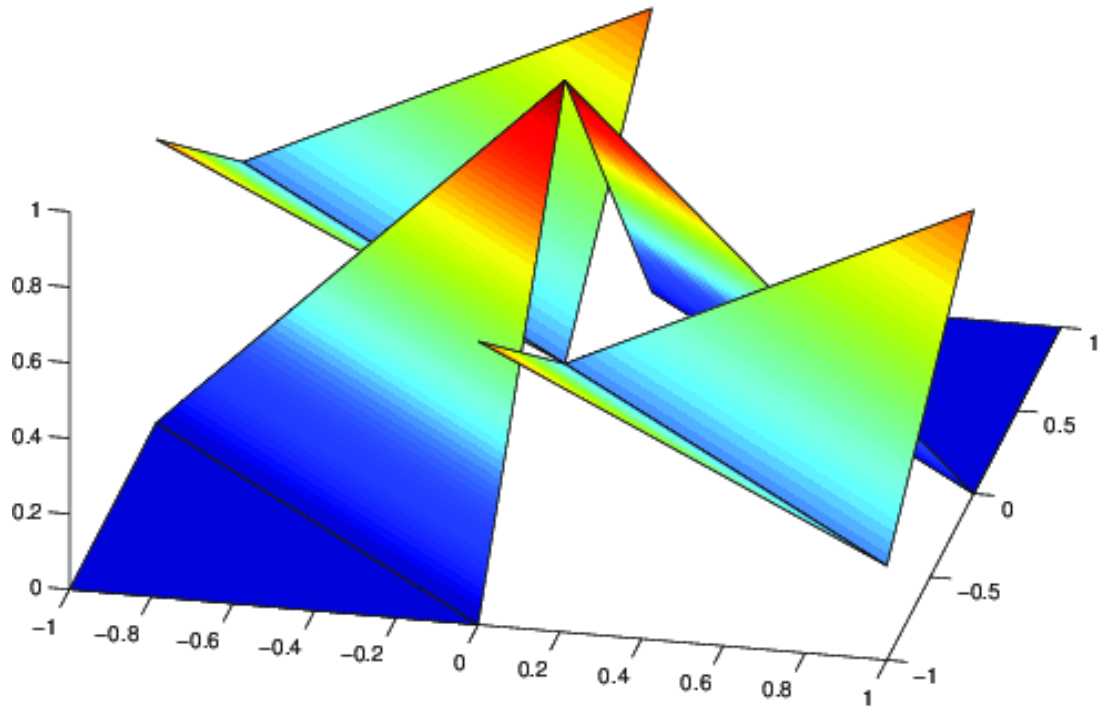


Figure: The space $\mathbf{P}_1^{\mathrm{NC}}$ also contains discontinuous functions.

The Crouzeix-Raviart Finite Element Method #3

- Another (more straightforward) way to arrive at the Crouzeix-Raviart FEM is to apply the Galerkin approach to the nonconforming Ansatz space $\mathbf{P}_1^{\mathrm{NC}}$ directly:

$$\sum_{T_k \in \mathbf{T}} \int_{T_k} \varepsilon \operatorname{grad} u_h \cdot \operatorname{grad} v = \int_\Omega f_h v \quad \forall v \in \mathbf{P}_1^{\mathrm{NC}}.$$

- With decreasing mesh-size the num. solution u_h converges to u with $O(h^2)$.
- More interestingly using a special post-processing we can recover a flux Ψ_h of second order accuracy $O(h^2)$ using:

$$\Psi_h(\mathbf{x}) = \varepsilon \operatorname{grad} u_h - \frac{f_{T_k}}{n}(\mathbf{x} - \mathbf{x}_{T_k}), \qquad \mathbf{x} \in T_k,\ \mathbf{x}_{T_k} \text{ barycenter of } T_k.$$

- The normal component of Ψ_h is continuous at inter-element interfaces.

The Crouzeix-Raviart Finite Element Method #3

- Another (more straightforward) way to arrive at the Crouzeix-Raviart FEM is to apply the Galerkin approach to the nonconforming Ansatz space $\mathbf{P}_1^{\mathrm{NC}}$ directly:

$$\sum_{T_k \in \mathbf{T}} \int_{T_k} \varepsilon \operatorname{grad} u_h \cdot \operatorname{grad} v = \int_\Omega f_h v \quad \forall v \in \mathbf{P}_1^{\mathrm{NC}}.$$

- With decreasing mesh-size the num. solution u_h converges to u with $O(h^2)$.
- More interestingly using a special post-processing we can recover a flux Ψ_h of second order accuracy $O(h^2)$ using:

$$\Psi_h(\mathbf{x}) = \varepsilon \operatorname{grad} u_h - \frac{f_{T_k}}{n}(\mathbf{x} - \mathbf{x}_{T_k}), \qquad \mathbf{x} \in T_k,\ \mathbf{x}_{T_k} \text{ barycenter of } T_k.$$

- The normal component of Ψ_h is continuous at inter-element interfaces.

The Crouzeix-Raviart Finite Element Method #3

- Another (more straightforward) way to arrive at the Crouzeix-Raviart FEM is to apply the Galerkin approach to the nonconforming Ansatz space $\mathbf{P}_1^{\mathrm{NC}}$ directly:

$$\sum_{T_k \in \mathbf{T}} \int_{T_k} \varepsilon \operatorname{grad} \mathrm{u}_h \cdot \operatorname{grad} v = \int_\Omega f_h v \quad \forall v \in \mathbf{P}_1^{\mathrm{NC}}.$$

- With decreasing mesh-size the num. solution u_h converges to u with $O(h^2)$.
- More interestingly using a special post-processing we can recover a flux $\boldsymbol{\Psi}_h$ of second order accuracy $O(h^2)$ using:

$$\boldsymbol{\Psi}_h(\mathbf{x}) = \varepsilon \operatorname{grad} \mathrm{u}_h - \frac{f_{T_k}}{n}(\mathbf{x} - \mathbf{x}_{T_k}), \qquad \mathbf{x} \in T_k,\ \mathbf{x}_{T_k} \text{ barycenter of } T_k.$$

- The normal component of $\boldsymbol{\Psi}_h$ is continuous at inter-element interfaces.

The Crouzeix-Raviart Finite Element Method #3

- Another (more straightforward) way to arrive at the Crouzeix-Raviart FEM is to apply the Galerkin approach to the nonconforming Ansatz space $\mathbf{P}_1^{\mathrm{NC}}$ directly:

$$\sum_{T_k \in \mathbf{T}} \int_{T_k} \varepsilon \operatorname{grad} \mathrm{u}_h \cdot \operatorname{grad} v = \int_\Omega f_h v \quad \forall v \in \mathbf{P}_1^{\mathrm{NC}}.$$

- With decreasing mesh-size the num. solution u_h converges to u with $O(h^2)$.
- More interestingly using a special post-processing we can recover a flux $\boldsymbol{\Psi}_h$ of second order accuracy $O(h^2)$ using:

$$\boldsymbol{\Psi}_h(\mathbf{x}) = \varepsilon \operatorname{grad} \mathrm{u}_h - \frac{f_{T_k}}{n}(\mathbf{x} - \mathbf{x}_{T_k}), \qquad \mathbf{x} \in T_k,\ \mathbf{x}_{T_k} \text{ barycenter of } T_k.$$

- The normal component of $\boldsymbol{\Psi}_h$ is continuous at inter-element interfaces.

The Crouzeix-Raviart Finite Element Method – Numerical Displacement u_h

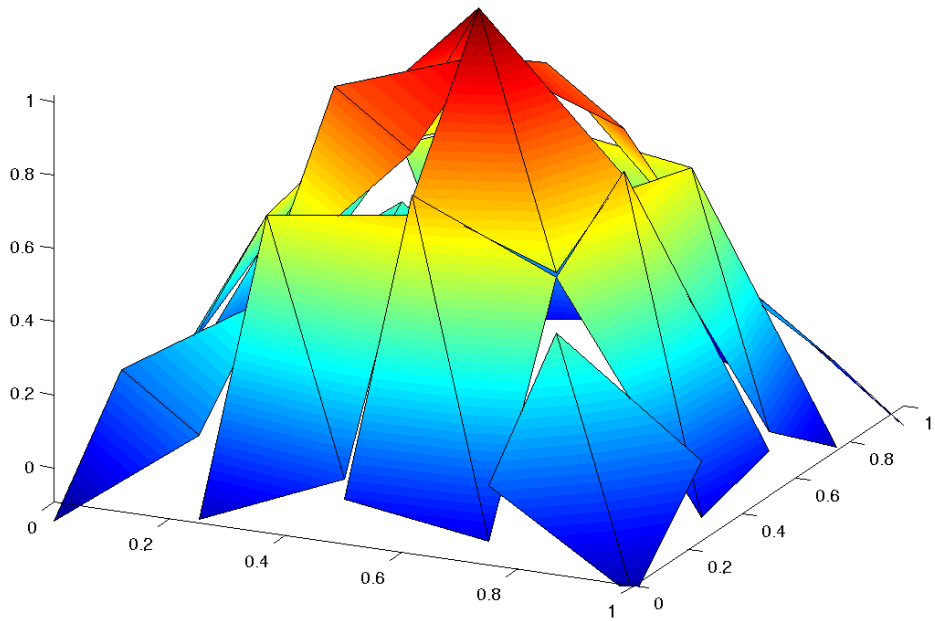


Figure: The Displacement is only continuous at the Midpoints of the Edges

The Crouzeix-Raviart Finite Element Method – Numerical Displacement u_h

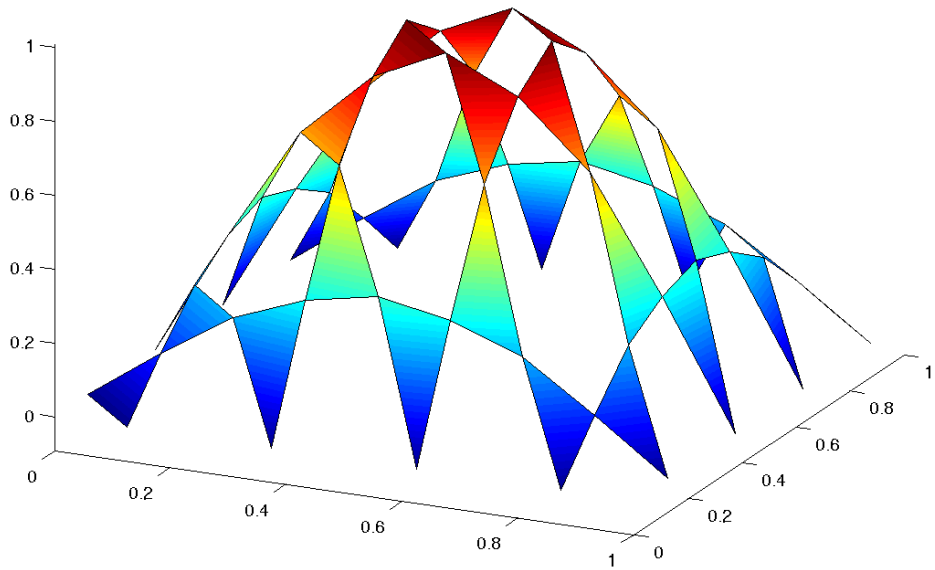


Figure: The Displacement is only continuous at the Midpoints of the Edges

The Crouzeix-Raviart Finite Element Method – Numerical Flux Ψ_h

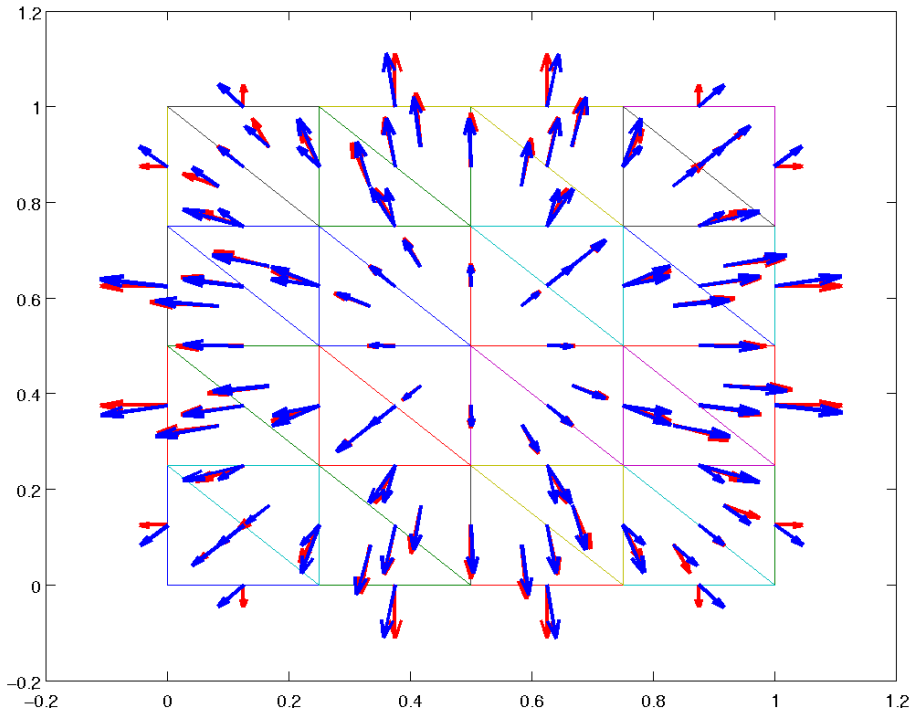


Figure: Plot of vector field at element interfaces and barycenters

The Crouzeix-Raviart Finite Element Method – Numerical Flux Ψ_h

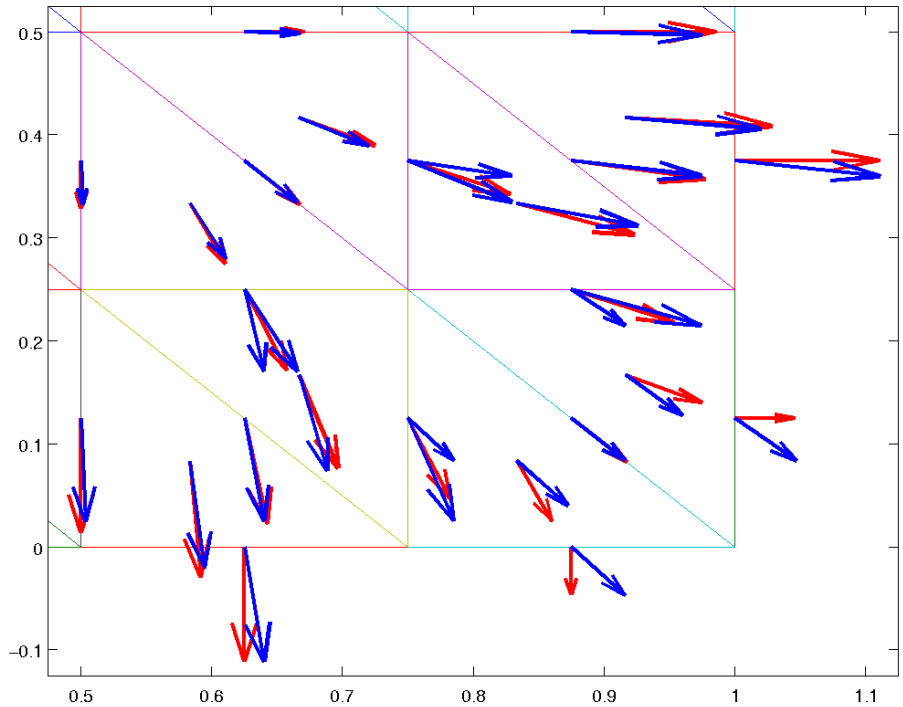


Figure: Plot of vector field at element interfaces and barycenters

Results – Problem setting

- For convergence studies we implemented the Crouzeix-Raviart Finite Element Method in MATLAB [6] for two and three dimensional simplical meshes.
- For benchmarking the robustness and the efficiency of this approach we used:

$$\begin{aligned} -\Delta u(x) &= 3\pi^2 \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3) \text{ in 3D,} \\ -\Delta u(x) &= 2\pi^2 \sin(\pi x_1)\sin(\pi x_2) \text{ in 2D.} \end{aligned}$$

- The analytic solutions for the scalar potential u is given by:

$$\begin{aligned} u(x) &= \sin(\pi x_1)\sin(\pi x_2)\sin(\pi x_3) \quad \text{in 3D} \\ u(x) &= \sin(\pi x_1)\sin(\pi x_2) \quad \text{in 2D respectively.} \end{aligned}$$

Results – Problem setting

- For convergence studies we implemented the Crouzeix-Raviart Finite Element Method in MATLAB [6] for two and three dimensional simplical meshes.
- For benchmarking the robustness and the efficiency of this approach we used:

$$\begin{aligned} -\Delta \mathrm{u}(x) &= 3\pi^2 \, sin(\pi x_1) \, sin(\pi x_2) \, sin(\pi x_3) \text{ in 3D,} \\ -\Delta \mathrm{u}(x) &= 2\pi^2 \, sin(\pi x_1) \, sin(\pi x_2) \text{ in 2D.} \end{aligned}$$

- The analytic solutions for the scalar potential u is given by:

$$\begin{aligned} \mathrm{u}(x) &= sin(\pi x_1) \, sin(\pi x_2) \, sin(\pi x_3) \quad \text{in 3D} \\ \mathrm{u}(x) &= sin(\pi x_1) \, sin(\pi x_2) \quad \text{in 2D respectively.} \end{aligned}$$

Results – Problem setting

- For convergence studies we implemented the Crouzeix-Raviart Finite Element Method in MATLAB [6] for two and three dimensional simplical meshes.
- For benchmarking the robustness and the efficiency of this approach we used:

$$\begin{aligned} -\Delta \mathrm{u}(x) &= 3\pi^2 \, sin(\pi x_1) \, sin(\pi x_2) \, sin(\pi x_3) \text{ in 3D,} \\ -\Delta \mathrm{u}(x) &= 2\pi^2 \, sin(\pi x_1) \, sin(\pi x_2) \text{ in 2D.} \end{aligned}$$

- The analytic solutions for the scalar potential u is given by:

$$\begin{aligned} \mathrm{u}(x) &= sin(\pi x_1) \, sin(\pi x_2) \, sin(\pi x_3) \quad \text{in 3D} \\ \mathrm{u}(x) &= sin(\pi x_1) \, sin(\pi x_2) \quad \text{in 2D respectively.} \end{aligned}$$

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Benchmark

- We benchmarked the convergence of the numerical solution for successively refined meshes (halving the element-diameter in each step)
- The convergence of the potential shows the expected behaviour (order $O(h^2)$ implying a reduction of the error by a factor of 4 with every step).
- The error in the approximated field distribution seems to be dominated by the error at the boundary (especially at the corners of the domain) first – only approaching order $O(h^2)$ on highly refined grids.

Results – Convergence 2D

K	t(sec)	$\|e_{u_h}\|_2$	$\|e_{u_h}\|_\infty$	$\|e_{\Psi_h}\|_\infty$
40	0.002	6.20e-2	1.20e-1	2.85e+0
176	0.004	1.49e-2	3.69e-2	1.28e+0
736	0.009	3.70e-3	9.59e-3	3.65e-1
3008	0.035	9.24e-4	2.40e-3	9.47e-2
12160	0.204	2.31e-4	6.01e-4	2.39e-2
48896	1.433	5.77e-5	1.50e-4	5.97e-3

Table: Convergence of successive refinements of the square $[0,1] \times [0,1]$; $\|e_{u_h}\|_2$ and $\|e_{u_h}\|_\infty$ are the L_2 and the maximum error of the potential u_h, while $\|e_{\Psi_h}\|_\infty$ indicates the maximum error of the approximated gradient Ψ_h at interface midpoints.

Results – Convergence 3D

K	t(sec)	$\|e_{u_h}\|_2$	$\|e_{u_h}\|_\infty$	$\|e_{\Psi_h}\|_\infty$
6	0.002	1.42e+0	9.74e-1	1.44e+1
72	0.002	7.14e-1	9.20e-1	1.32e+1
672	0.006	1.49e-1	3.20e-1	6.87e+0
5760	0.044	4.02e-2	1.13e-1	2.93e+0
47616	0.597	1.07e-2	3.21e-2	8.63e-1
387072	9.373	2.74e-3	8.58e-3	2.23e-1

Table: Convergence of successive refinements of the cube $[0,1] \times [0,1] \times [0,1]$; $\|e_{u_h}\|_2$ and $\|e_{u_h}\|_\infty$ are the L_2 and the maximum error of the potential u_h, while $\|e_{\Psi_h}\|_\infty$ indicates the maximum error of the approximated gradient Ψ_h at interface midpoints.

Results – Convergence in Domain

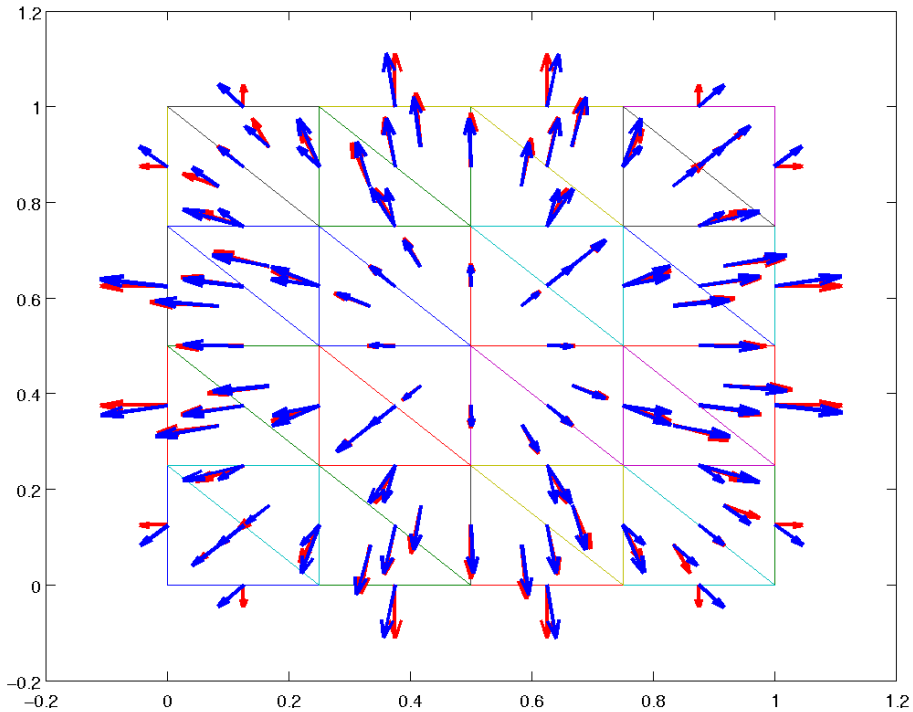


Figure: Plot of vector field at element interfaces and barycenters

Results – Convergence at Domain Boundaries

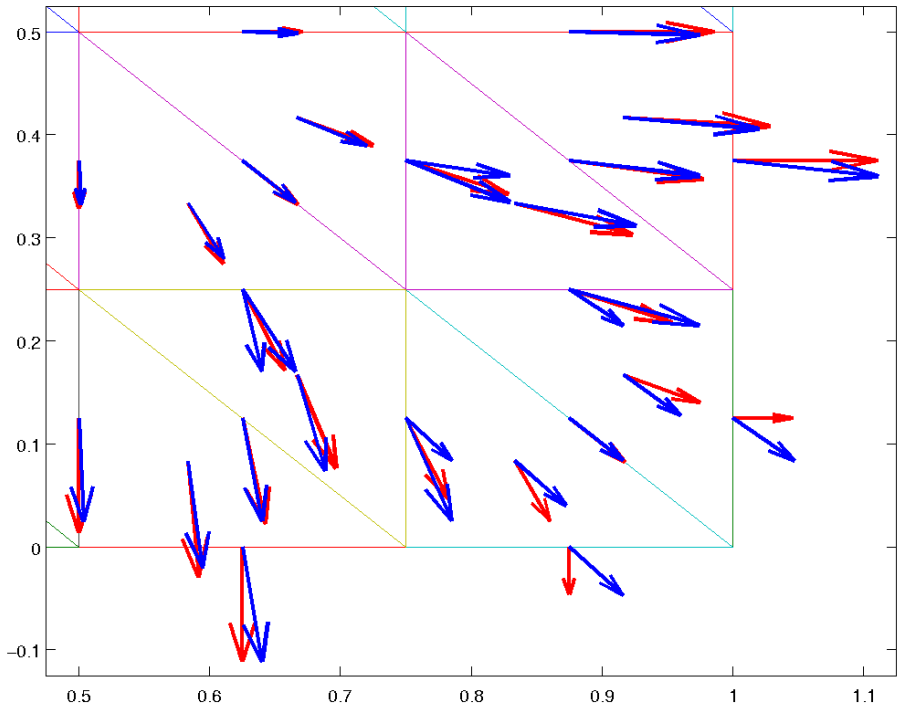


Figure: Convergence is not optimal at the corners of the domain

Conclusions

- Though the computation time could be reduced by a factor of 2 - using SSOR as a preconditioner - for our current applications and the accuracy sought the computational cost seems to be prohibitive.
- Besides using an more efficient preconditioner we want to explore the use of geometric multigrid for Crouzeix Raviart finite elements.
- Using the right prolongation and restriction operators is not completely straight-forward for the finite element spaces involved as the successively refined function spaces are not nested (see Figure 9).
- Exploring the approaches by Kraus, Margenov and Synka [4] and geometric multigrid as described by Braess, Dryja and Hackbusch [3] seems to be most promising at the moment.

Conclusions

- Though the computation time could be reduced by a factor of 2 - using SSOR as a preconditioner - for our current applications and the accuracy sought the computational cost seems to be prohibitive.
- Besides using an more efficient preconditioner we want to explore the use of geometric multigrid for Crouzeix Raviart finite elements.
- Using the right prolongation and restriction operators is not completely straight-forward for the finite element spaces involved as the successively refined function spaces are not nested (see Figure 9).
- Exploring the approaches by Kraus, Margenov and Synka [4] and geometric multigrid as described by Braess, Dryja and Hackbusch [3] seems to be most promising at the moment.

Conclusions

- Though the computation time could be reduced by a factor of 2 - using SSOR as a preconditioner - for our current applications and the accuracy sought the computational cost seems to be prohibitive.
- Besides using an more efficient preconditioner we want to explore the use of geometric multigrid for Crouzeix Raviart finite elements.
- Using the right prolongation and restriction operators is not completely straight-forward for the finite element spaces involved as the successively refined function spaces are not nested (see Figure 9).
- Exploring the approaches by Kraus, Margenov and Synka [4] and geometric multigrid as described by Braess, Dryja and Hackbusch [3] seems to be most promising at the moment.

Conclusions

- Though the computation time could be reduced by a factor of 2 - using SSOR as a preconditioner - for our current applications and the accuracy sought the computational cost seems to be prohibitive.
- Besides using an more efficient preconditioner we want to explore the use of geometric multigrid for Crouzeix Raviart finite elements.
- Using the right prolongation and restriction operators is not completely straight-forward for the finite element spaces involved as the successively refined function spaces are not nested (see Figure 9).
- Exploring the approaches by Kraus, Margenov and Synka [4] and geometric multigrid as described by Braess, Dryja and Hackbusch [3] seems to be most promising at the moment.

Conclusions

- Though the computation time could be reduced by a factor of 2 - using SSOR as a preconditioner - for our current applications and the accuracy sought the computational cost seems to be prohibitive.
- Besides using an more efficient preconditioner we want to explore the use of geometric multigrid for Crouzeix Raviart finite elements.
- Using the right prolongation and restriction operators is not completely straight-forward for the finite element spaces involved as the successively refined function spaces are not nested (see Figure 9).
- Exploring the approaches by Kraus, Margenov and Synka [4] and geometric multigrid as described by Braess, Dryja and Hackbusch [3] seems to be most promising at the moment.

Questions and Answers

[Intentionally left empty]

Multigrid for Crouzeix Raviart FEM – Prolongation to Fine Grid

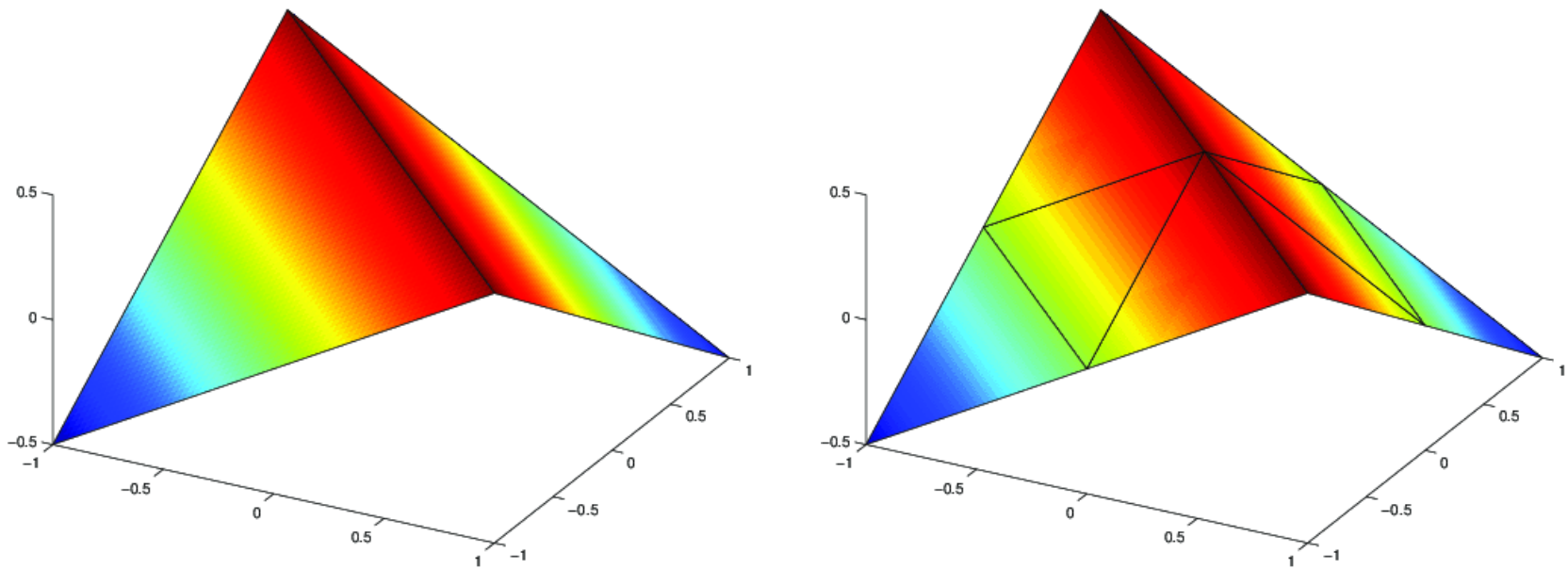


Figure: Continuous coarse-grid function is contained in fine grid

Multigrid for Crouzeix Raviart FEM – Prolongation to Fine Grid #2

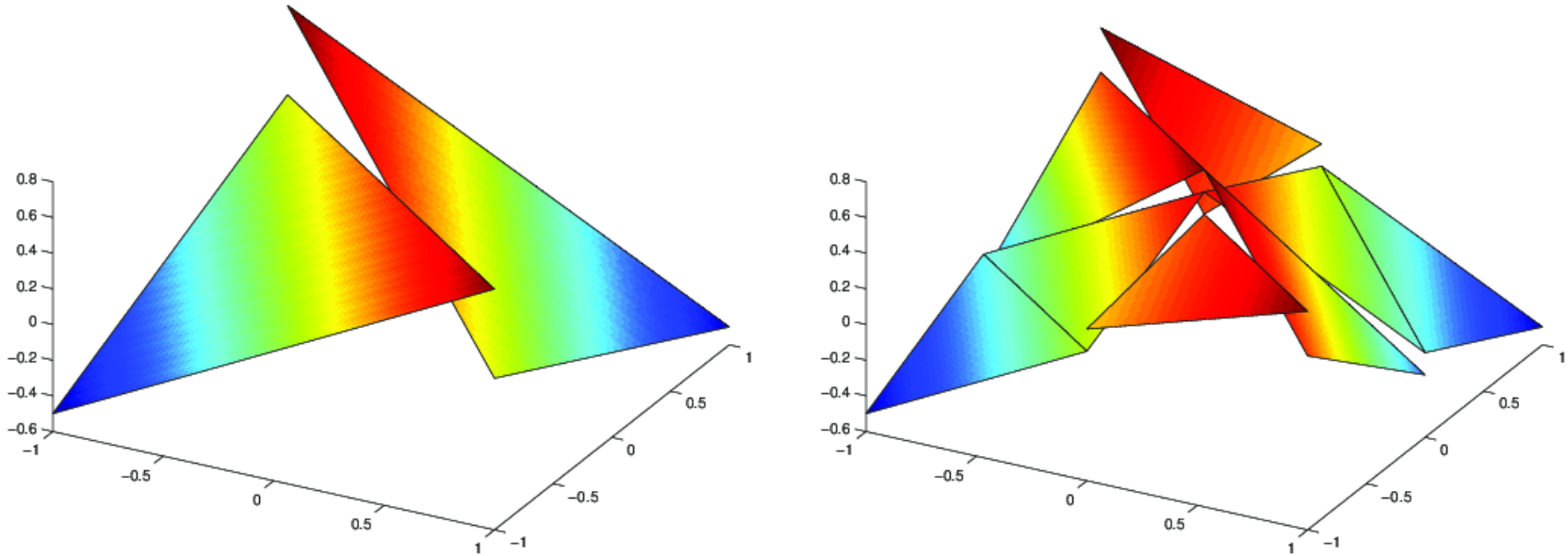


Figure: Discontinuous coarse-grid function is not contained in fine grid

Bibliography I

L. Marini.

An inexpensive method for the evaluation of the solution of the lowest order Raviart–Thomas mixed method.

SIAM Journal on Numerical Analysis, 22(3):493–496, 1985.

D. N. Arnold and F. Brezzi.

Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates.

Math. Anal. Numér, 19(1), 1985.

Bibliography II

D. Braess, M. Dryja, and W. Hackbusch.

A multigrid method for nonconforming FE-discretisations with application to non-matching grids.

Computing, 63(1):1–25, July 1999.

J. Kraus, S. Margenov, and J. Synka.

On the multilevel preconditioning of Crouzeix–Raviart elliptic problems.

Numerical Linear Algebra with Applications, 15(5):395–416, 2008.

Bibliography III

G. Pöplau and U. van Rienen.

An efficient 3D space charge routine with self-adaptive discretization.

In *Proceedings of ICAP 2009 (Proceedings of the 10th International Computational Accelerator Physics Conference), San Francisco, USA*, pages 23–26, 2010.

MATLAB.

Version 7.10.0 (R2010a).

The MathWorks Inc.