An Explicit Mortar Flux Recovery Approach for Interface Coupling

K. Peterson, P. Bochev, P. Kuberry

Sandia National Laboratories

SIAM CSE

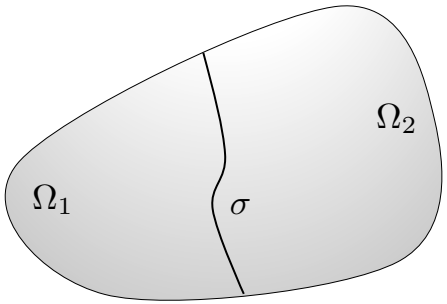
March 3, 2017

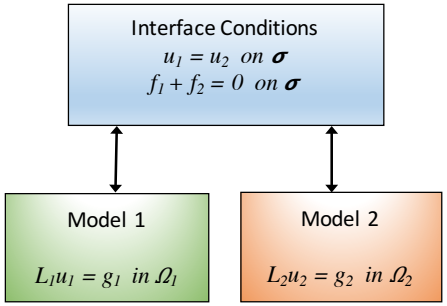
Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Partitioned Method for Interface Coupling

Consider two subdomains connected by an interface

- Objective is a method that enables solving independently on each subdomain
- Compute a flux or force along the interface to provide subdomain boundary conditions and enforce suitable interface conditions

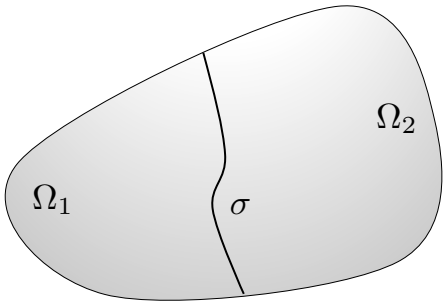




Partitioned Method for Interface Coupling

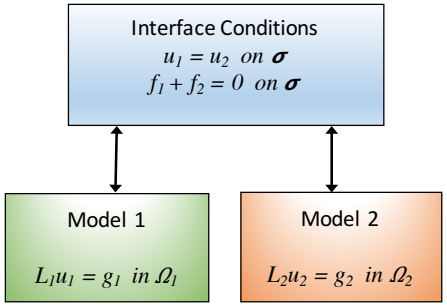
Consider two subdomains connected by an interface

- Objective is a method that enables solving independently on each subdomain
- Compute a flux or force along the interface to provide subdomain boundary conditions and enforce suitable interface conditions
- Assume scalar conservation equation as governing equations



$$\dot{\varphi}_1 - \nabla \cdot F_1(\varphi_1) = g_1 \quad \text{in} \quad \Omega_1$$
$$\dot{\varphi}_2 - \nabla \cdot F_2(\varphi_2) = g_2 \quad \text{in} \quad \Omega_2$$

$$\varphi_1 - \varphi_2 = 0 \quad \text{on} \quad \sigma$$
$$F_1 \cdot \mathbf{n_1} + F_2 \cdot \mathbf{n_2} = 0 \quad \text{on} \quad \sigma$$



Lagrange Multiplier Formulation

Begin with a system for the monolithic problem

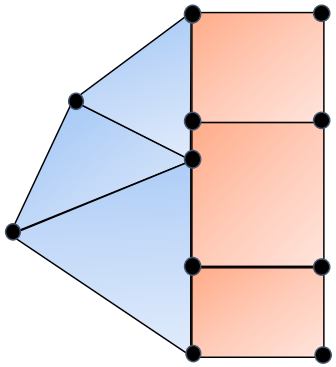
Weak system of equations

$$\int_{\Omega_1} \dot{\varphi}_1 \psi_1 \, d\Omega + \int_{\Omega_1} \left(F_1(\varphi_1) \cdot \nabla \psi_1 - g_1 \psi_1 \right) d\Omega = \int_\sigma \lambda \psi_1 \, dS \quad \forall \psi_1 \in V$$

$$\int_{\Omega_2} \dot{\varphi}_2 \psi_2 \, d\Omega + \int_{\Omega_2} \left(F_2(\varphi_2) \cdot \nabla \psi_2 - g_2 \psi_2 \right) d\Omega = \int_\sigma -\lambda \psi_2 \, dS \quad \forall \psi_2 \in W$$

$$\int_\sigma \left(\varphi_1 - \varphi_2 \right) \mu \, dS = 0 \quad \forall \mu \in M.$$

Discretize each subdomain separately with basis functions $\{N_{k,i}\}$



Mass matrix: $M_{k,ij} = \int_\Omega N_{k,i} N_{k,j} \, dS$

Coupling matrix: $G_{k,ij} = \int_\sigma N_{k,i} \hat{N}_j \, dS$

Semi-discrete system

$$M_1 \dot{\varphi}_1 + f_1(\varphi_1) = G_1^\mathsf{T} \lambda$$

$$M_2 \dot{\varphi}_2 + f_2(\varphi_2) = -G_2^\mathsf{T} \lambda$$

$$G_1 \varphi_1 - G_2 \varphi_2 = 0$$

Lagrange Multiplier Formulation

Begin with a system for the monolithic problem

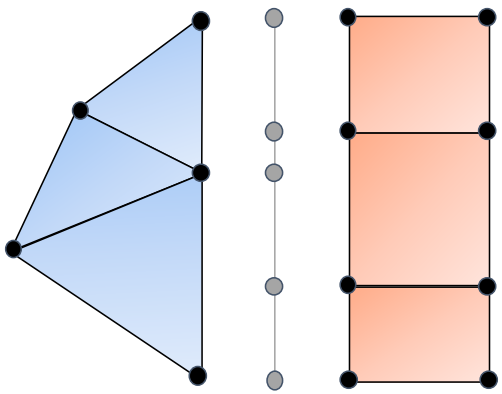
Weak system of equations

$$\int_{\Omega_1} \dot{\varphi}_1 \psi_1 d\Omega + \int_{\Omega_1} \left(F_1(\varphi_1) \cdot \nabla \psi_1 - g_1 \psi_1\right) d\Omega = \int_\sigma \lambda \psi_1 dS \quad \forall \psi_1 \in V$$

$$\int_{\Omega_2} \dot{\varphi}_2 \psi_2 d\Omega + \int_{\Omega_2} \left(F_2(\varphi_2) \cdot \nabla \psi_2 - g_2 \psi_2\right) d\Omega = \int_\sigma -\lambda \psi_2 dS \quad \forall \psi_2 \in W$$

$$\int_\sigma \left(\varphi_1 - \varphi_2\right) \mu dS = 0 \quad \forall \mu \in M.$$

Combined interface grid for Lagrange multiplier, with basis functions $\{\hat{N}_i\}$



Mass matrix: $M_{k,ij} = \int_\Omega N_{k,i} N_{k,j} dS$

Coupling matrix: $G_{k,ij} = \int_\sigma N_{k,i} \hat{N}_j dS$

Semi-discrete system

$$M_1 \dot{\varphi}_1 + f_1(\varphi_1) = G_1^\mathsf{T} \lambda$$

$$M_2 \dot{\varphi}_2 + f_2(\varphi_2) = -G_2^\mathsf{T} \lambda$$

$$G_1 \varphi_1 - G_2 \varphi_2 = 0$$

Lagrange Multiplier Formulation

- Index 2 Differential Algebraic Equation (DAE)
- Requires careful integration in time due to hidden constraints
- Not compatible with explicit treatment of interface flux (λ)

Semi-discrete system

$$M_1\dot{\varphi}_1 + f_1(\varphi_1) = G_1^{\mathsf{T}}\lambda$$
$$M_2\dot{\varphi}_2 + f_2(\varphi_2) = -G_2^{\mathsf{T}}\lambda$$
$$G_1\varphi_1 - G_2\varphi_2 = 0$$

Lagrange Multiplier Formulation

- Index 2 Differential Algebraic Equation (DAE)
- Requires careful integration in time due to hidden constraints
- Not compatible with explicit treatment of interface flux (λ)
- We replace the original constraint with $G_1\dot{\varphi}_1 - G_2\dot{\varphi}_2 = 0$
- Assuming the initial data satisfies $G_1\varphi_1(0) - G_2\varphi_2(0) = 0$, the two constraints are equivalent
- The new system enables a fully explicit treatment of λ

Semi-discrete system

$$M_1\dot{\varphi}_1 + f_1(\varphi_1) = G_1^{\mathsf{T}}\lambda$$
$$M_2\dot{\varphi}_2 + f_2(\varphi_2) = -G_2^{\mathsf{T}}\lambda$$
$$G_1\varphi_1 - G_2\varphi_2 = 0$$

Modified semi-discrete system

$$M_1\dot{\varphi}_1 + f_1(\varphi_1) = G_1^{\mathsf{T}}\lambda$$
$$M_2\dot{\varphi}_2 + f_2(\varphi_2) = -G_2^{\mathsf{T}}\lambda$$
$$G_1\dot{\varphi}_1 - G_2\dot{\varphi}_2 = 0$$

Mortar Flux Recovery (MFR)

Expanding the system in terms of interface (σ) and internal (I) degrees of freedom

Modified semi-discrete linear system

$$\begin{pmatrix} M_{1,II} & M_{1,I\sigma} & 0 & 0 & 0 \\ M_{1,\sigma I} & M_{1,\sigma\sigma} & 0 & 0 & G_1^\mathsf{T} \\ 0 & 0 & M_{2,II} & M_{2,I\sigma} & 0 \\ 0 & 0 & M_{2,\sigma I} & M_{2,\sigma\sigma} & -G_2^\mathsf{T} \\ 0 & G_1 & 0 & -G_2 & 0 \end{pmatrix} \begin{pmatrix} \dot{\varphi}_{1,I} \\ \dot{\varphi}_{1,\sigma} \\ \dot{\varphi}_{2,I} \\ \dot{\varphi}_{2,\sigma} \\ \lambda \end{pmatrix} = \begin{pmatrix} f_{1,I}(\varphi_1) \\ f_{1,\sigma}(\varphi_1) \\ f_{2,I}(\varphi_2) \\ f_{2,\sigma}(\varphi_2) \\ 0 \end{pmatrix}$$

Interface linear system

$$\begin{pmatrix} A_1 & 0 & G_1^\mathsf{T} \\ 0 & A_2 & -G_2^\mathsf{T} \\ G_1 & -G_2 & 0 \end{pmatrix} \begin{pmatrix} \dot{\varphi}_{1,\sigma} \\ \dot{\varphi}_{2,\sigma} \\ \lambda \end{pmatrix} = \begin{pmatrix} \tilde{f}_1(\varphi_1) \\ \tilde{f}_2(\varphi_2) \\ 0 \end{pmatrix}$$

$$A_1 = M_{1,\sigma\sigma} - M_{1,\sigma I} M_{1,II}^{-1} M_{1,I\sigma}$$

$$A_2 = M_{2,\sigma\sigma} - M_{2,\sigma I} M_{2,II}^{-1} M_{2,I\sigma}$$

$$\tilde{f}_1 = f_{1,\sigma} - M_{1,\sigma I} M_{1,II}^{-1} f_{1,I}$$

$$\tilde{f}_2 = f_{2,\sigma} - M_{2,\sigma I} M_{2,II}^{-1} f_{2,I}$$

Mortar Flux Recovery (MFR)

Expanding the system in terms of interface (σ) and internal (I) degrees of freedom

Modified semi-discrete linear system

$$\begin{pmatrix} M_{1,II} & M_{1,I\sigma} & 0 & 0 & 0 \\ M_{1,\sigma I} & M_{1,\sigma\sigma} & 0 & 0 & G_1^\mathsf{T} \\ 0 & 0 & M_{2,II} & M_{2,I\sigma} & 0 \\ 0 & 0 & M_{2,\sigma I} & M_{2,\sigma\sigma} & -G_2^\mathsf{T} \\ 0 & G_1 & 0 & -G_2 & 0 \end{pmatrix} \begin{pmatrix} \dot{\varphi}_{1,I} \\ \dot{\varphi}_{1,\sigma} \\ \dot{\varphi}_{2,I} \\ \dot{\varphi}_{2,\sigma} \\ \lambda \end{pmatrix} = \begin{pmatrix} f_{1,I}(\varphi_1) \\ f_{1,\sigma}(\varphi_1) \\ f_{2,I}(\varphi_2) \\ f_{2,\sigma}(\varphi_2) \\ 0 \end{pmatrix}$$

Interface linear system

$$\begin{pmatrix} A_1 & 0 & G_1^\mathsf{T} \\ 0 & A_2 & -G_2^\mathsf{T} \\ G_1 & -G_2 & 0 \end{pmatrix} \begin{pmatrix} \dot{\varphi}_{1,\sigma} \\ \dot{\varphi}_{2,\sigma} \\ \lambda \end{pmatrix} = \begin{pmatrix} \tilde{f}_1(\varphi_1) \\ \tilde{f}_2(\varphi_2) \\ 0 \end{pmatrix}$$

$$A_1 = M_{1,\sigma\sigma} - M_{1,\sigma I} M_{1,II}^{-1} M_{1,I\sigma}$$

$$A_2 = M_{2,\sigma\sigma} - M_{2,\sigma I} M_{2,II}^{-1} M_{2,I\sigma}$$

$$\tilde{f}_1 = f_{1,\sigma} - M_{1,\sigma I} M_{1,II}^{-1} f_{1,I}$$

$$\tilde{f}_2 = f_{2,\sigma} - M_{2,\sigma I} M_{2,II}^{-1} f_{2,I}$$

$$\Downarrow$$

$$\lambda(\varphi_1, \varphi_2) = \left(G_1 A_1^{-1} G_1^\mathsf{T} + G_2 A_2^{-1} G_2^\mathsf{T}\right)^{-1} \left(G_1 A_1^{-1} \tilde{f}_1(\varphi_1) - G_2 A_2^{-1} \tilde{f}_2(\varphi_2)\right)$$

Mortar Flux Recovery (MFR)

MFR algorithm

1. Solve for λ

$$\lambda(\varphi_1^n, \varphi_2^n) = \left(G_1 A_1^{-1} G_1^\mathsf{T} + G_2 A_2^{-1} G_2^\mathsf{T}\right)^{-1} \left(G_1 A_1^{-1} \tilde{f}_1(\varphi_1^n) - G_2 A_2^{-1} \tilde{f}_2(\varphi_2^n)\right)$$

2. Solve each subdomain equation independently

$$M_1(\varphi_1^{n+1} - \varphi_1^n) = \Delta t \left(f_1(\varphi_1^n) + G_1^\mathsf{T} \lambda(\varphi_1^n, \varphi_2^n)\right)$$
$$M_2(\varphi_2^{n+1} - \varphi_2^n) = \Delta t \left(f_2(\varphi_2^n) - G_2^\mathsf{T} \lambda(\varphi_1^n, \varphi_2^n)\right)$$

- MFR is an explicit method to estimate Neumann flux on interface boundary
- Equation for λ is similar to interface equation in FETI method*
  - domain decomposition method with goal to increase concurrency
  - developed originally for elliptic problems
  - for time dependent problems, discretize in time then solve the monolithic problem implicitly

* C. Farhat, F. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, IJNME 32, 1991
* C. Farhat, L. Crivelli, F. Roux, A transient FETI methodology for large-scale parallel implicit computations in structural mechanics, IJNME 37, 1994
* A. Toselli, FETI domain decomposition methods for scalar advection-diffusion problems, CMAME 190, 2001

Mortar Flux Recovery (MFR)

MFR algorithm

1. Solve for λ

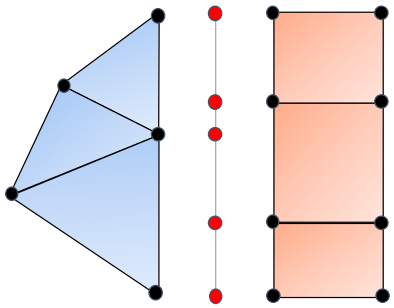
$$\lambda(\varphi_1^n, \varphi_2^n) = \left(G_1 A_1^{-1} G_1^\mathsf{T} + G_2 A_2^{-1} G_2^\mathsf{T}\right)^{-1} \left(G_1 A_1^{-1} \tilde{f}_1(\varphi_1^n) - G_2 A_2^{-1} \tilde{f}_2(\varphi_2^n)\right)$$

2. Solve each subdomain equation independently

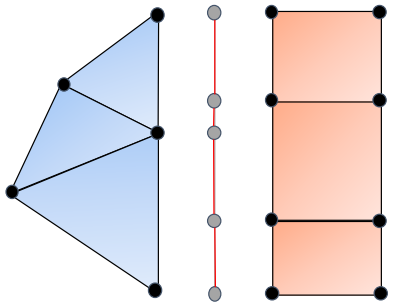
$$M_1(\varphi_1^{n+1} - \varphi_1^n) = \Delta t \left(f_1(\varphi^n) + G_1^\mathsf{T} \lambda(\varphi_1^n, \varphi_2^n)\right)$$
$$M_2(\varphi_2^{n+1} - \varphi_2^n) = \Delta t \left(f_2(\varphi^n) - G_2^\mathsf{T} \lambda(\varphi_1^n, \varphi_2^n)\right)$$

Two variations:

MFR 1 - $\{\hat{N}_i\}$ on nodes



MFR 2 - $\{\hat{N}_i\}$ on faces



Direct Flux Recovery (DFR) Approach

For comparison, we consider an alternative algorithm based on an L^2 projection of the fluxes:

- Want to approximate ζ_i, the Neumann flux along interface

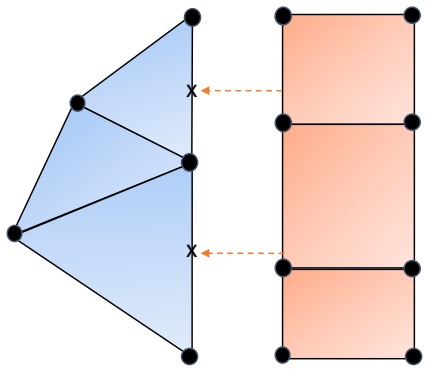
$$M_1\dot{\varphi}_1 = f_1(\varphi_1) + \zeta_1$$
$$M_2\dot{\varphi}_2 = f_2(\varphi_2) + \zeta_2$$

- Minimize the difference between flux density t_i and value from opposite side

$$||t_1 - F_2(\varphi_2)\cdot\mathbf{n}||_\sigma \quad ||t_2 - F_1(\varphi_1)\cdot\mathbf{n}||_\sigma$$

- Define interface mass matrices $M_{\sigma,i}$, then

$$\zeta_1 = M_{\sigma,1}t_1 = \int_\sigma F_2(\varphi_2)\cdot\mathbf{n}\psi_1 dS$$
$$\zeta_2 = M_{\sigma,2}t_2 = \int_\sigma F_1(\varphi_1)\cdot\mathbf{n}\psi_2 dS$$



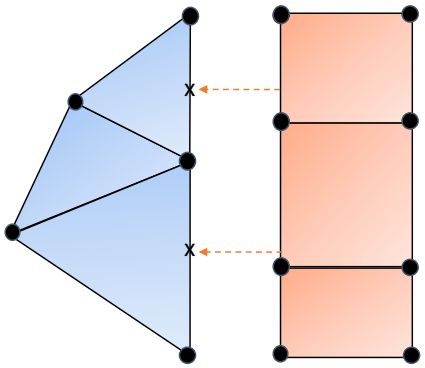
$$\zeta_1 = \int_\sigma F_2(\varphi_2)\cdot\mathbf{n}\psi_1 dS$$

R. K. Jaiman, X. Jiao, P. H. Geubelle, E. Loth, Assessment of conservative load transfer for fluid-solid interface with non-matching meshes. IJNME 64, 2005

Direct Flux Recovery (DFR) Approach

Three variations on the method:

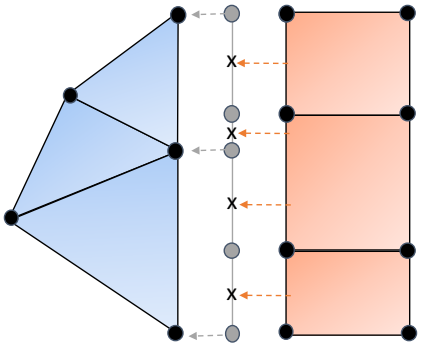
DFR 1



Evaluate $F_2(\varphi_2)$ at integration points on side 1 and integrate

$$\zeta_1 = \int_\sigma F_2(\varphi_2) \cdot \mathbf{n} \psi_1 \, dS$$

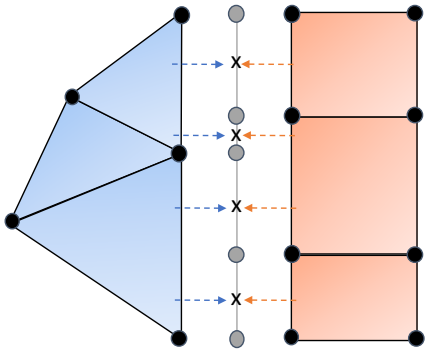
DFR 2



Compute flux vector on combined interface $(\hat{\zeta}_1)$ and L^2 project back to side 1

$$\hat{\zeta}_1 = \int_\sigma F_2(\varphi_2) \cdot \mathbf{n} \hat{\psi} \, dS$$
$$\zeta_1 = M_{\sigma,1} \left(\hat{M}_\sigma^{-1} \hat{\zeta}_1 \right)$$

DFR 3



Compute flux vectors on combined interface $(\hat{\zeta}_1, \hat{\zeta}_2)$, average, then L^2 project back to each side

$$\hat{\zeta} = (\hat{\zeta}_1 - \hat{\zeta}_2)/2$$
$$\zeta_1 = M_{\sigma,1} \left(\hat{M}_\sigma^{-1} \hat{\zeta} \right)$$
$$\zeta_2 = -M_{\sigma,2} \left(\hat{M}_\sigma^{-1} \hat{\zeta} \right)$$

Numerical Results: Patch Test

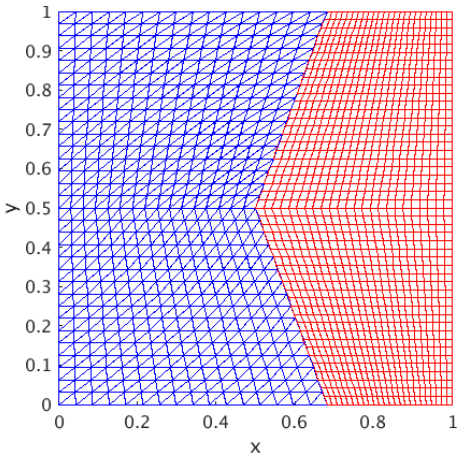
Manufactured solution: $\varphi_k(\mathbf{x}, t) = x + y$

$$\dot{\varphi}_k - \nabla \cdot (\epsilon \nabla \varphi_k - \mathbf{v}\varphi_k) = g_k$$

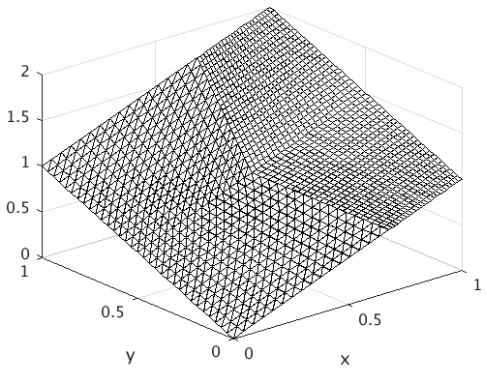
with $\epsilon = 0.001$, $\mathbf{v} = (-\sin(\pi/6), \cos(\pi/6))$

Stabilize with SUPG

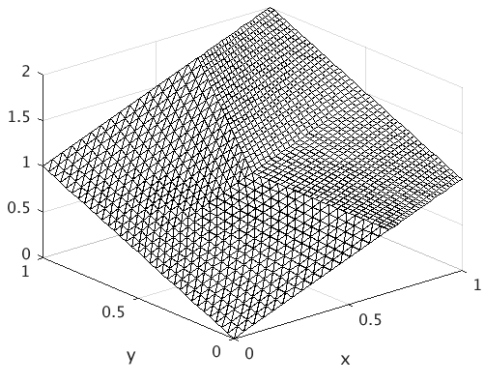
Mesh



MFR $t = 1$



DFR $t = 1$



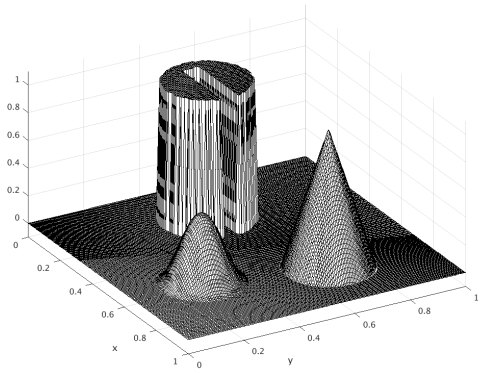
Both methods recover linear manufactured solution to machine precision

Numerical Results: Pure Advection

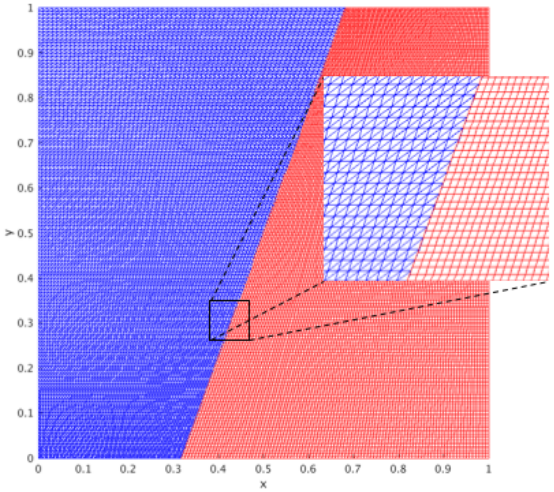
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

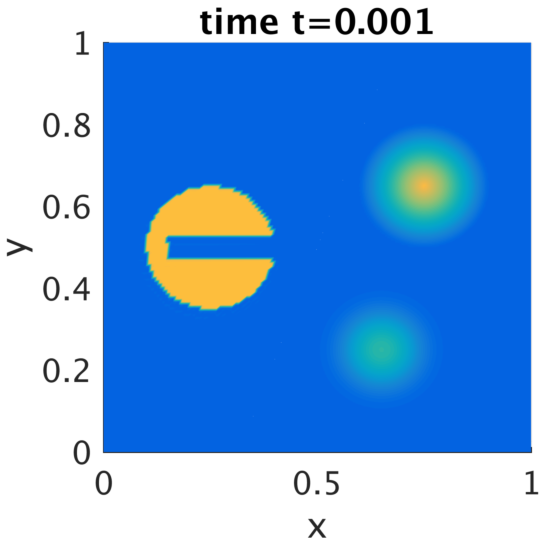
SUPG stabilization



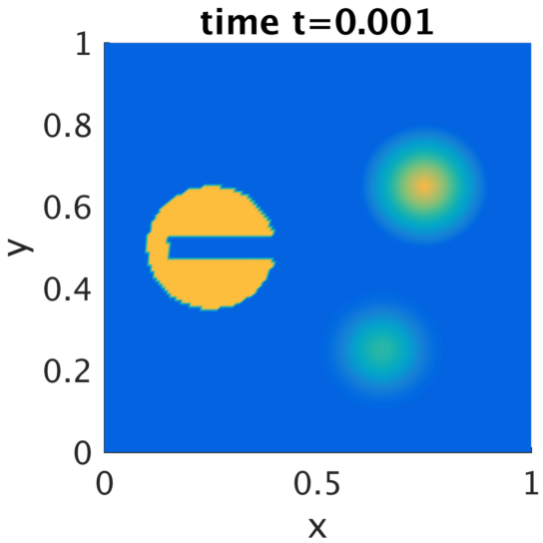
Mesh



DFR



MFR

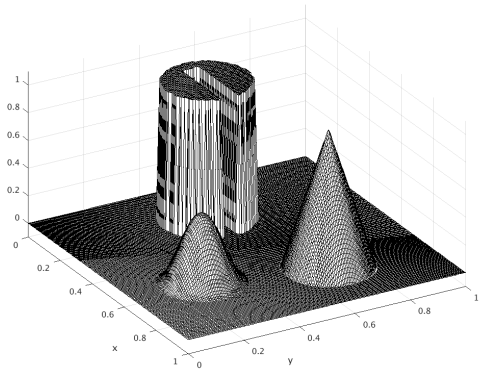


Numerical Results: Pure Advection

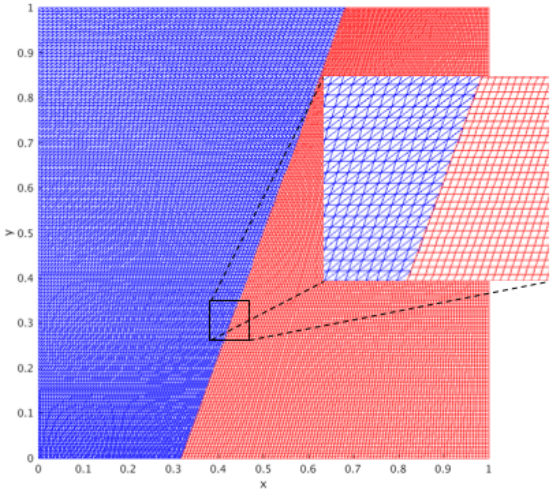
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

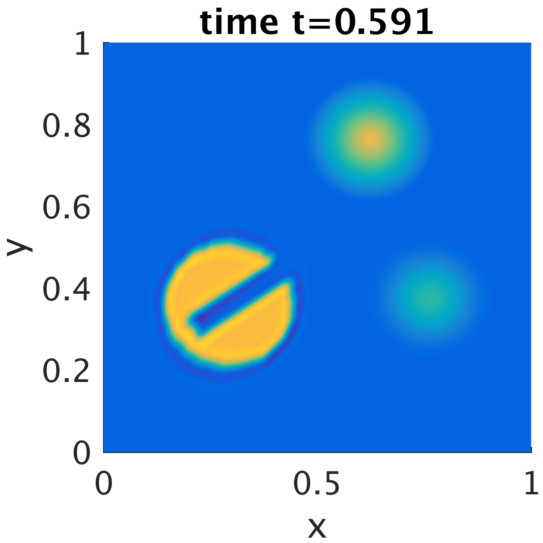
SUPG stabilization



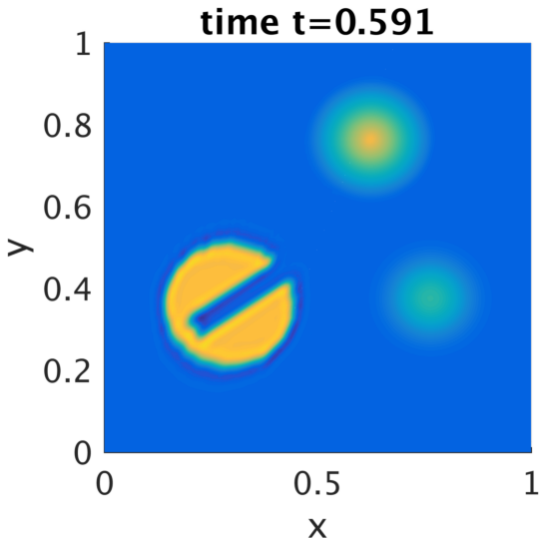
Mesh



DFR



MFR

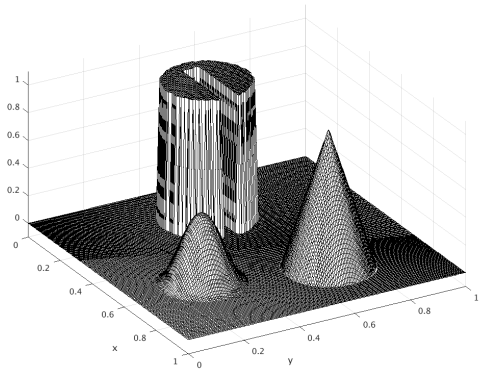


Numerical Results: Pure Advection

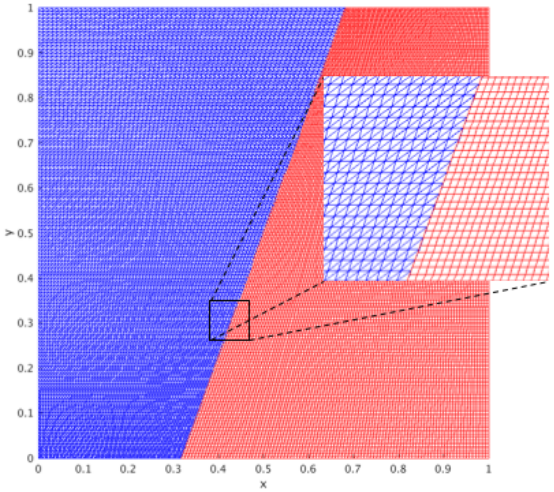
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

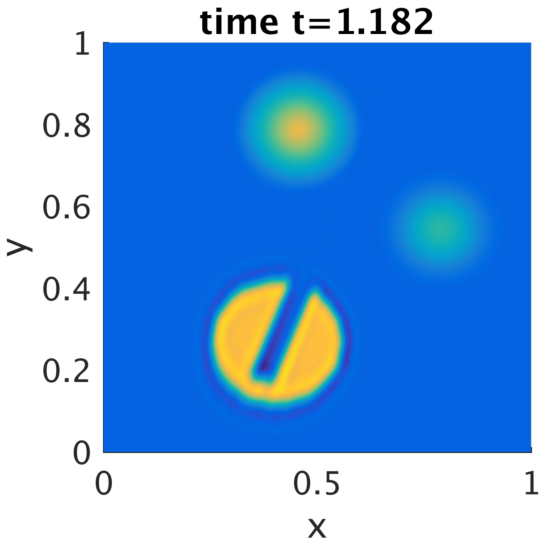
SUPG stabilization



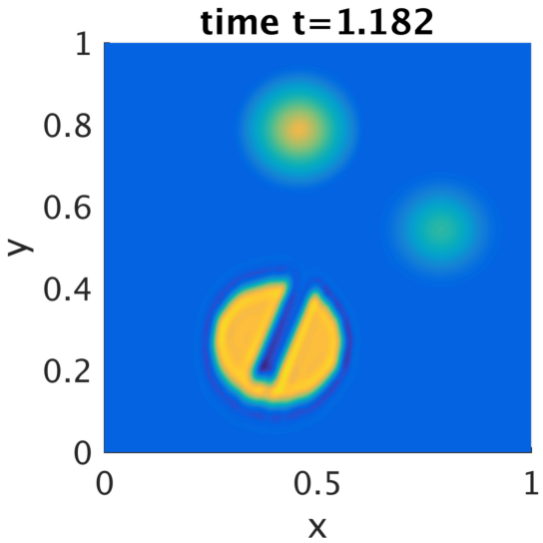
Mesh



DFR



MFR

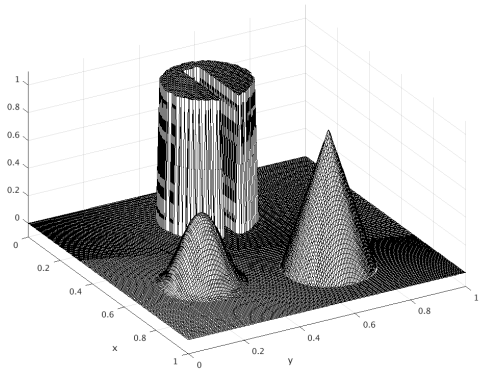


Numerical Results: Pure Advection

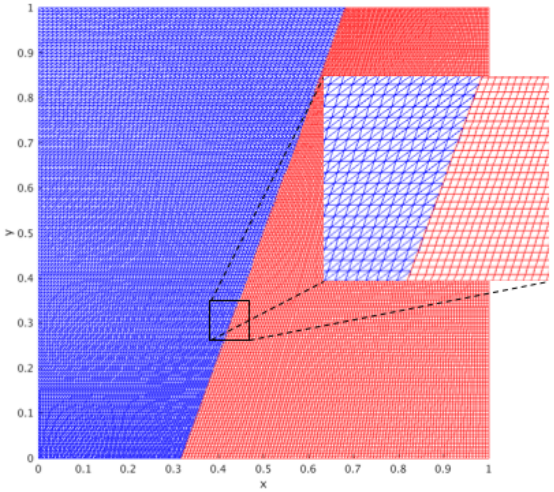
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

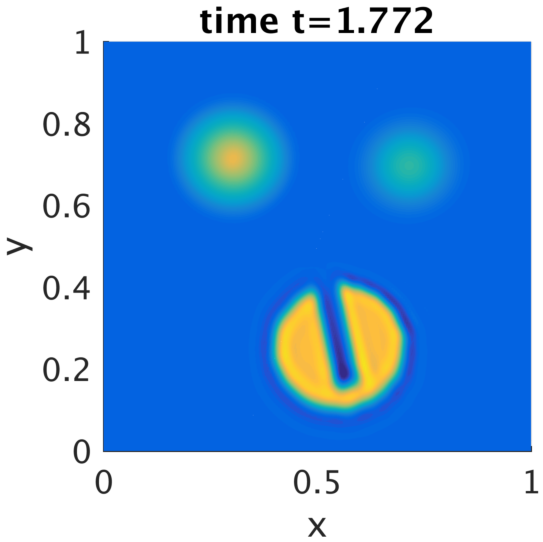
SUPG stabilization



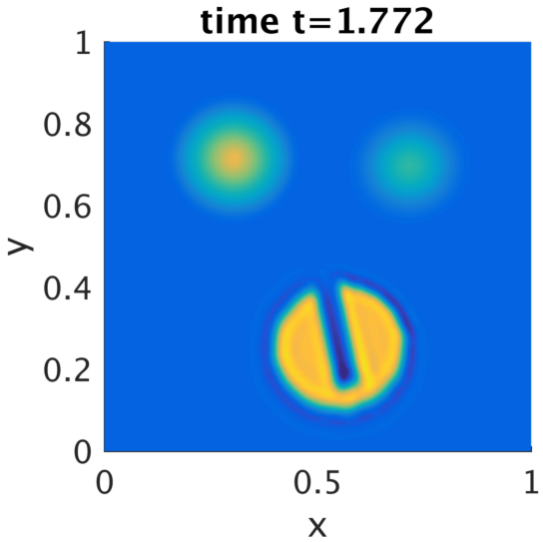
Mesh



DFR



MFR

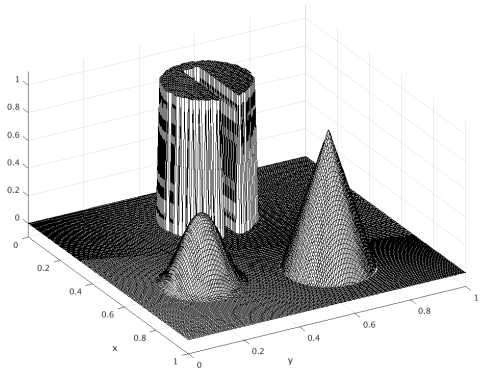


Numerical Results: Pure Advection

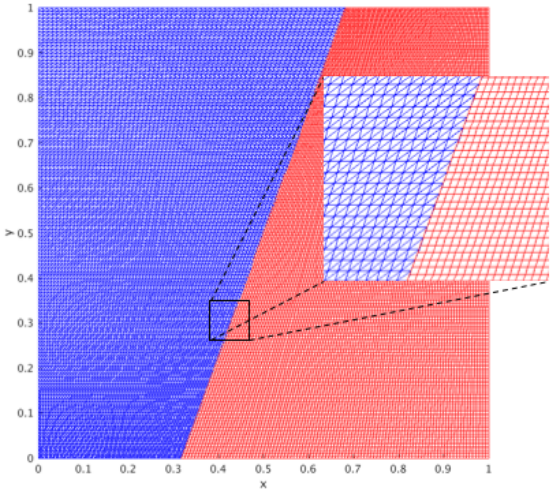
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

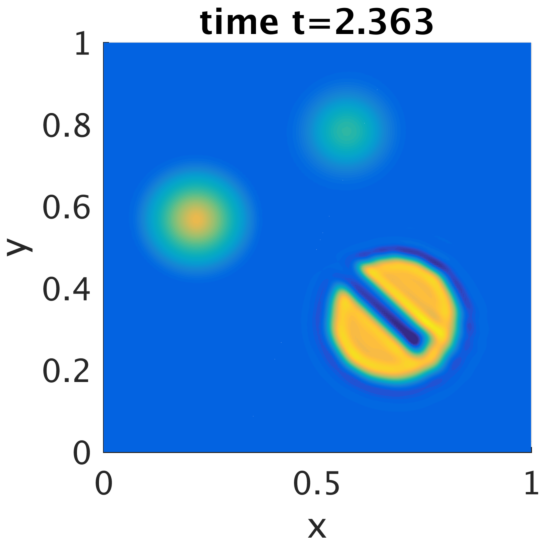
SUPG stabilization



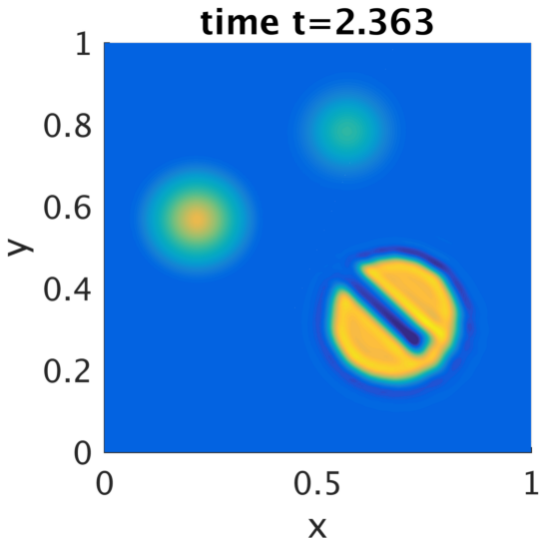
Mesh



DFR



MFR

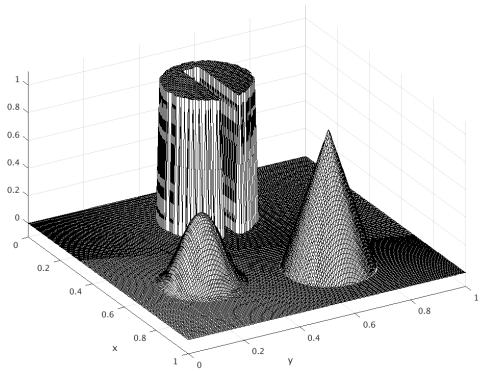


Numerical Results: Pure Advection

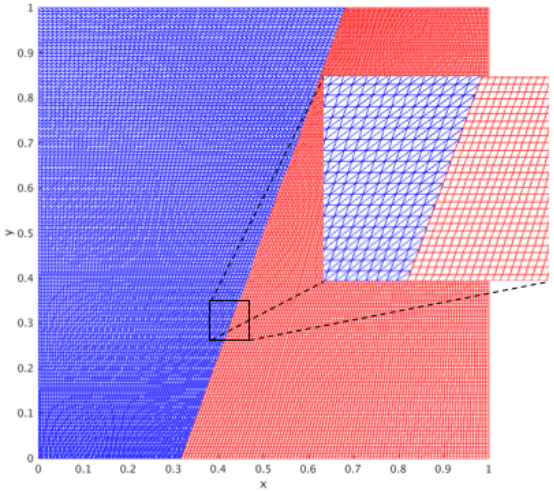
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

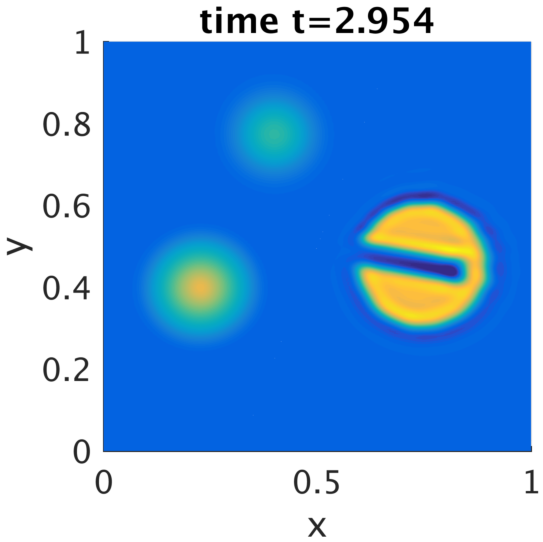
SUPG stabilization



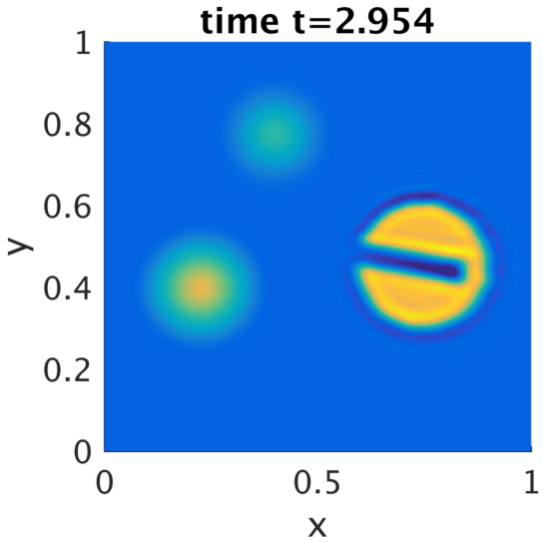
Mesh



DFR



MFR

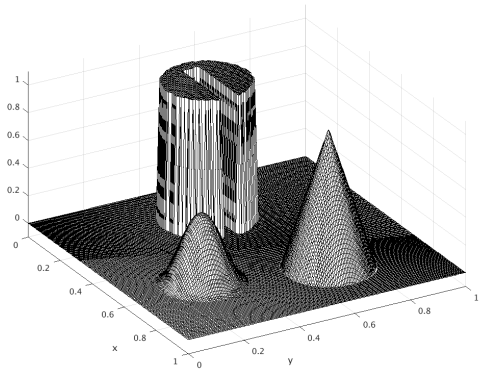


Numerical Results: Pure Advection

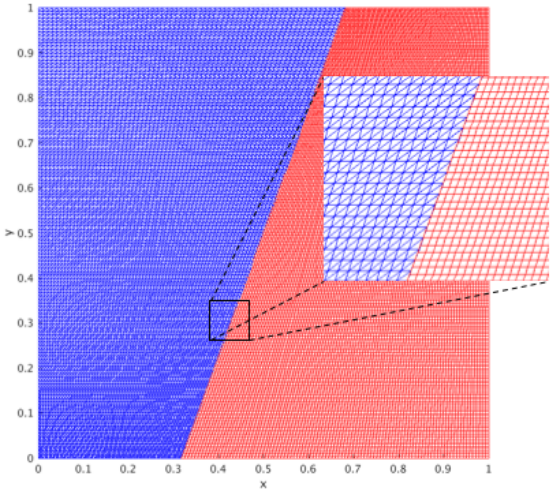
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

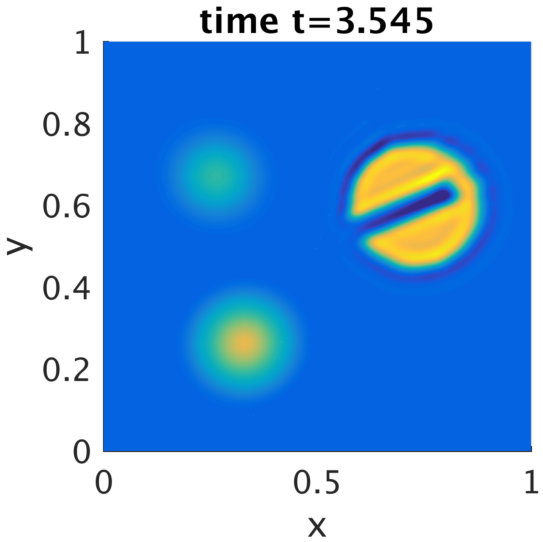
SUPG stabilization



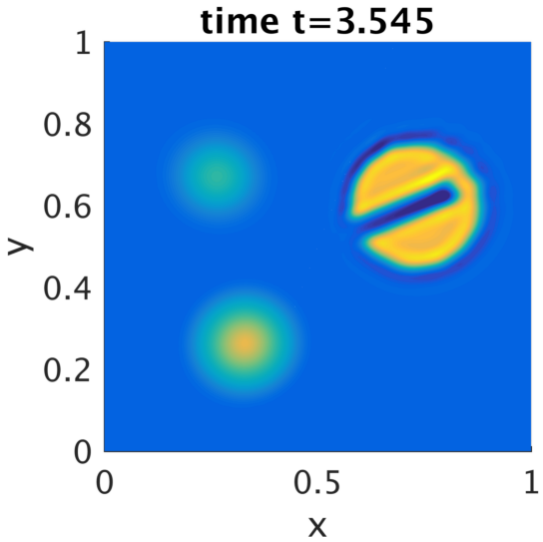
Mesh



DFR



MFR

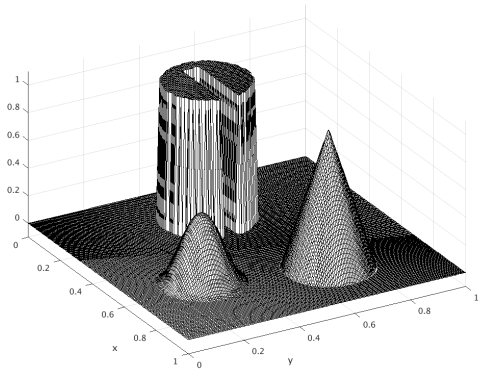


Numerical Results: Pure Advection

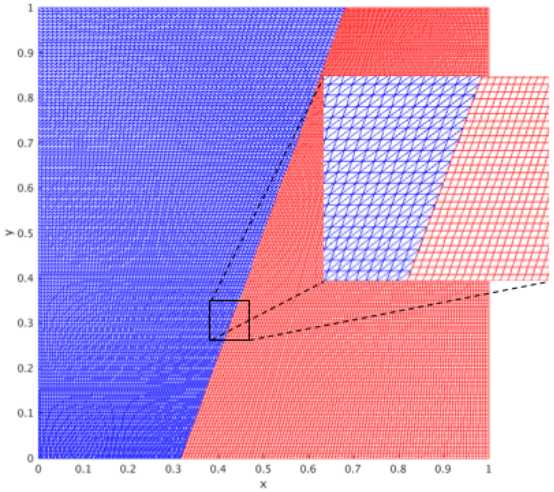
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

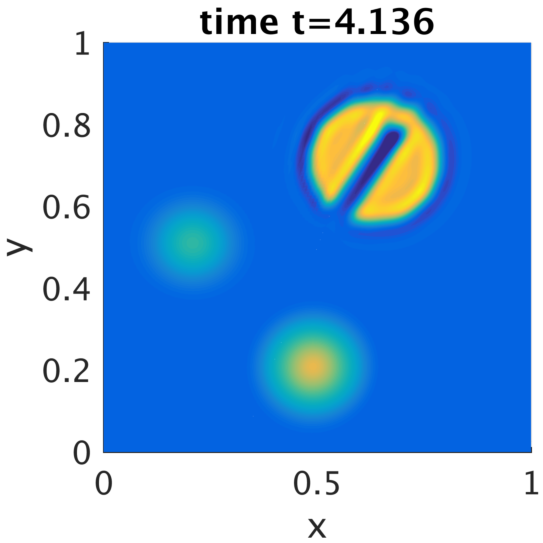
SUPG stabilization



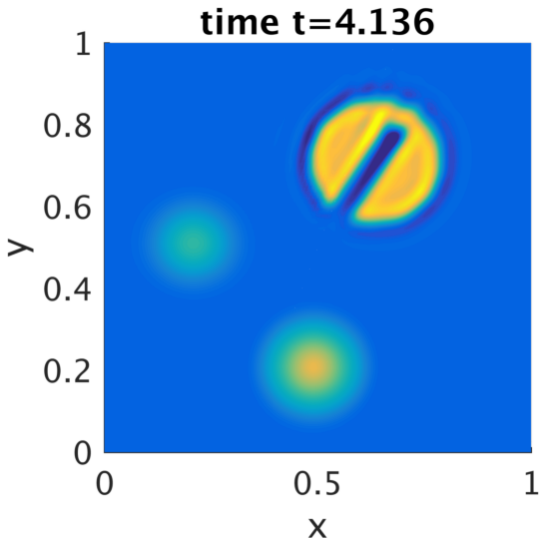
Mesh



DFR



MFR

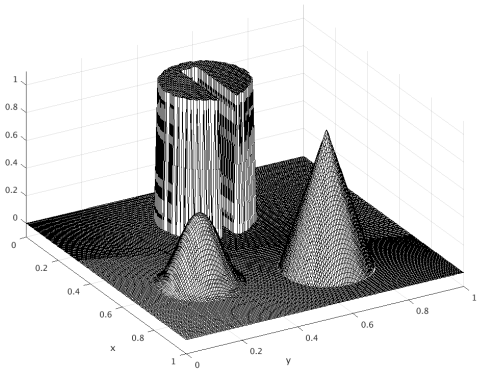


Numerical Results: Pure Advection

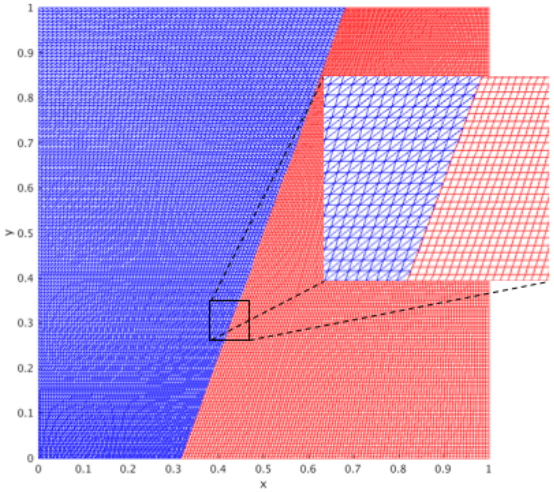
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

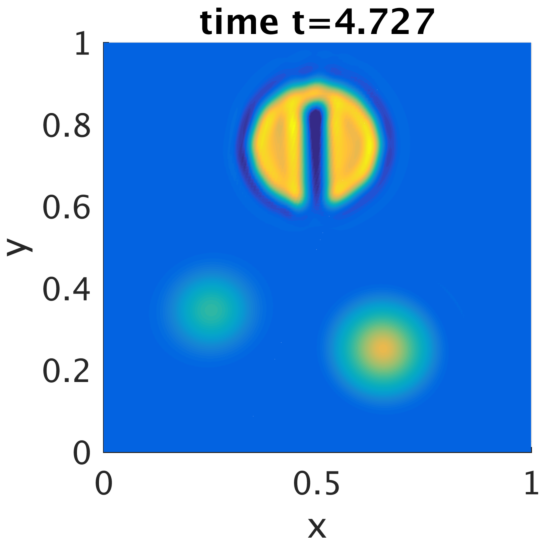
SUPG stabilization



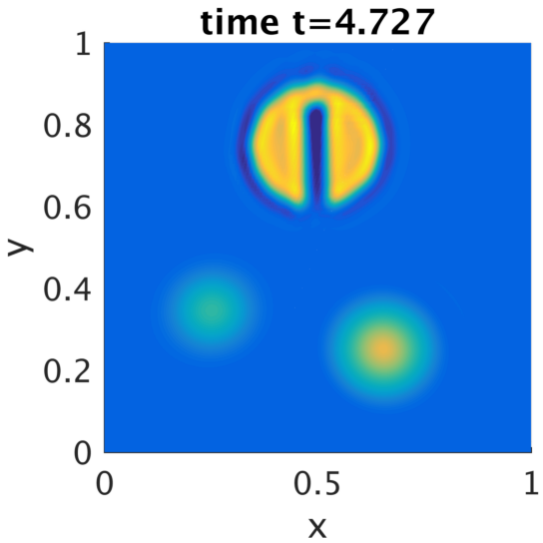
Mesh



DFR



MFR

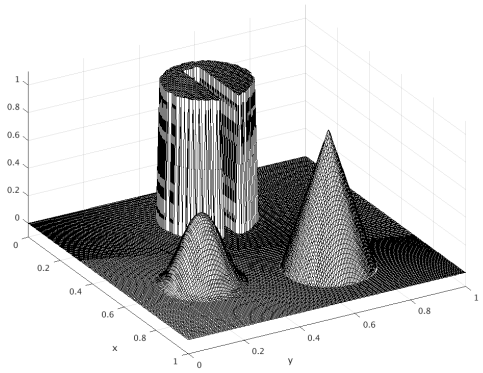


Numerical Results: Pure Advection

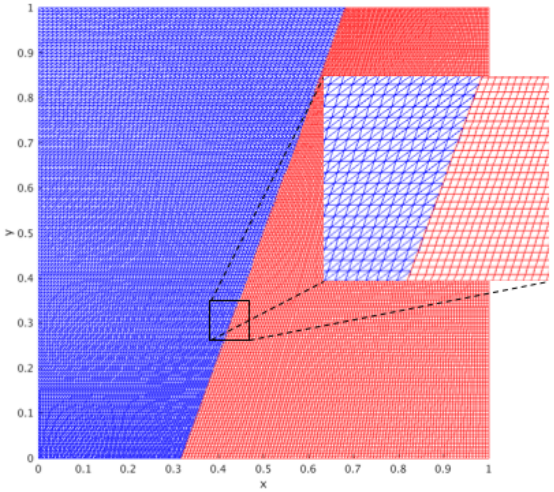
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

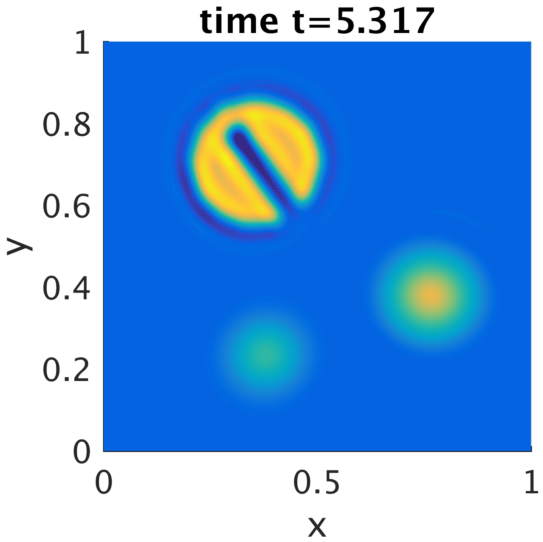
SUPG stabilization



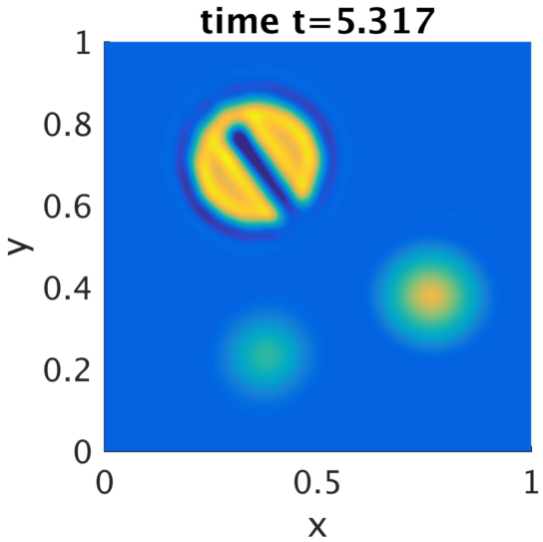
Mesh



DFR



MFR

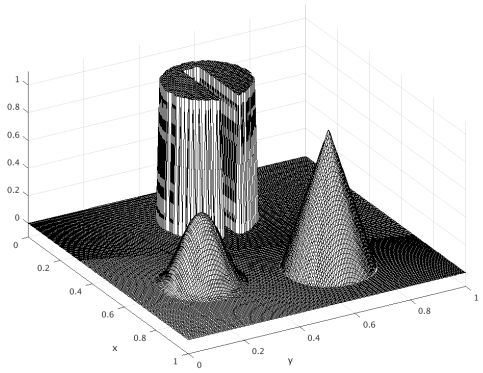


Numerical Results: Pure Advection

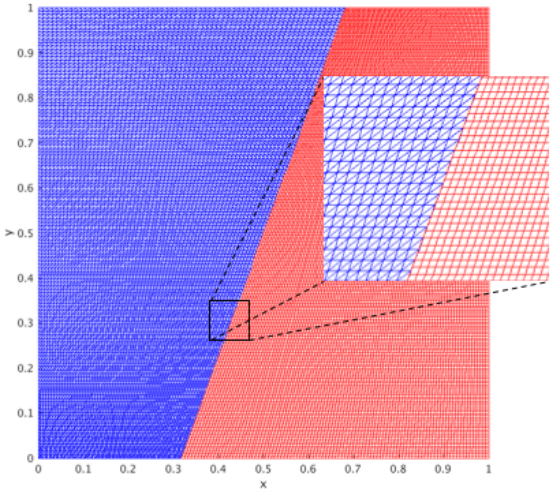
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

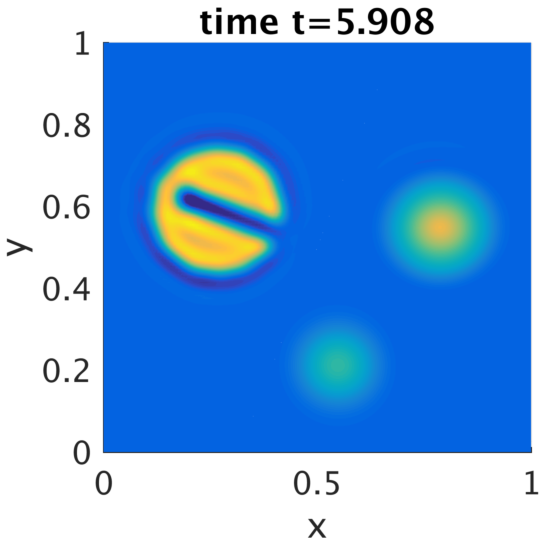
SUPG stabilization



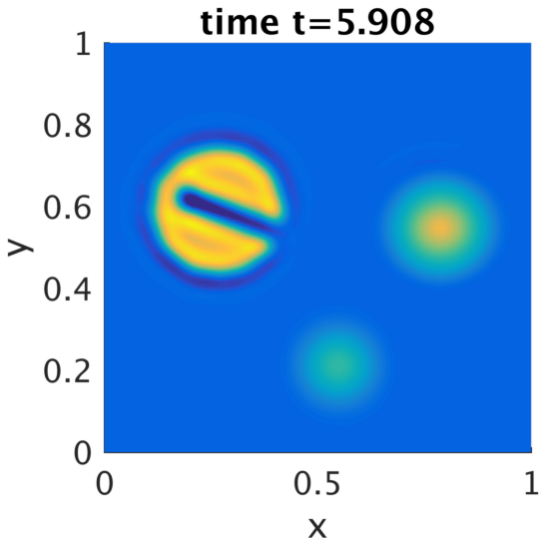
Mesh



DFR



MFR

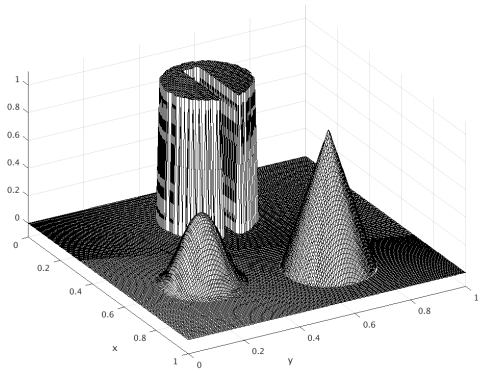


Numerical Results: Pure Advection

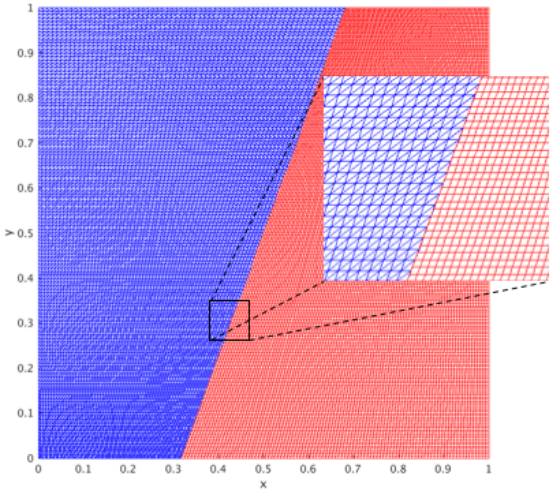
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

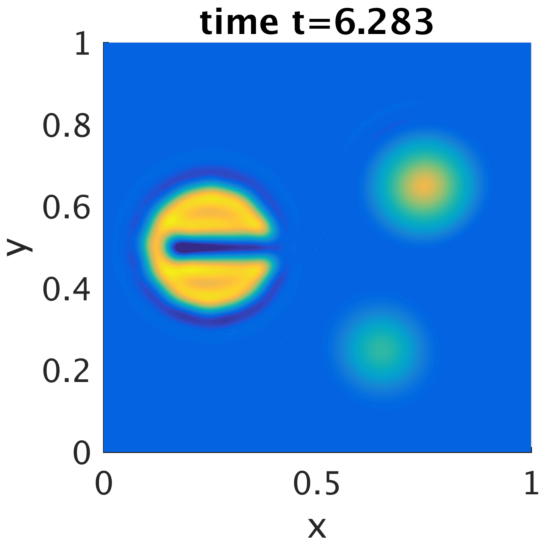
SUPG stabilization



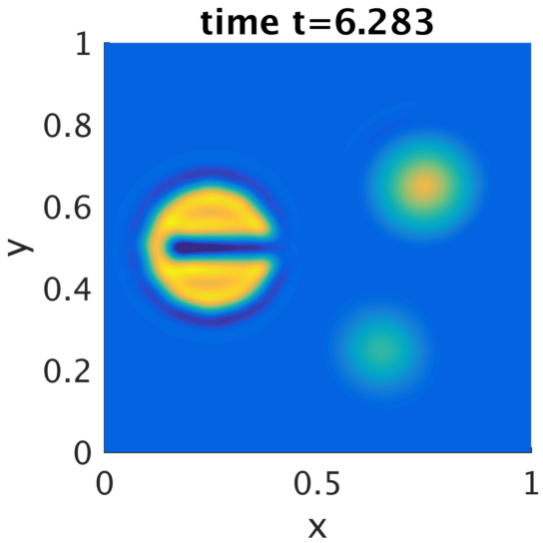
Mesh



DFR



MFR

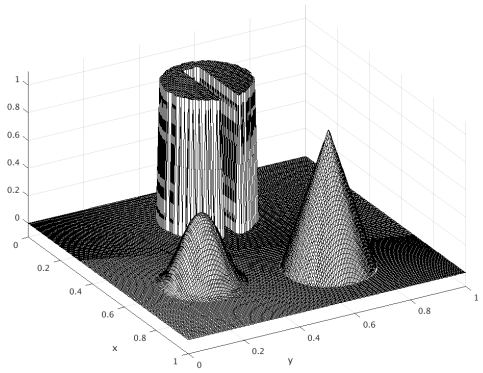


Numerical Results: Pure Advection

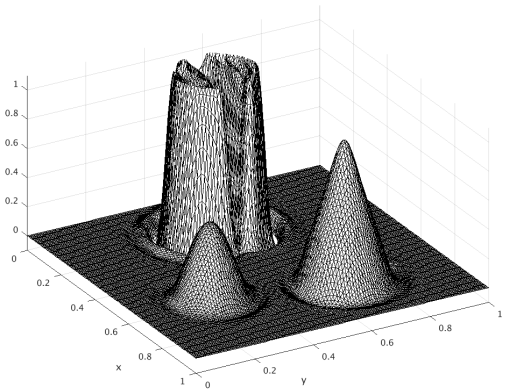
Rotating velocity field: $\mathbf{v} = (-y + \frac{1}{2}, x - \frac{1}{2})$

$$\dot{\varphi}_k + \nabla \cdot (\mathbf{v}\varphi_k) = 0$$

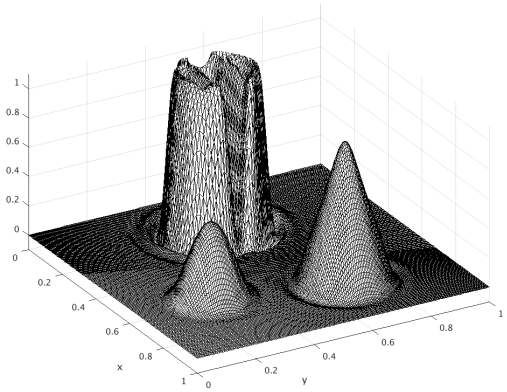
SUPG stabilization



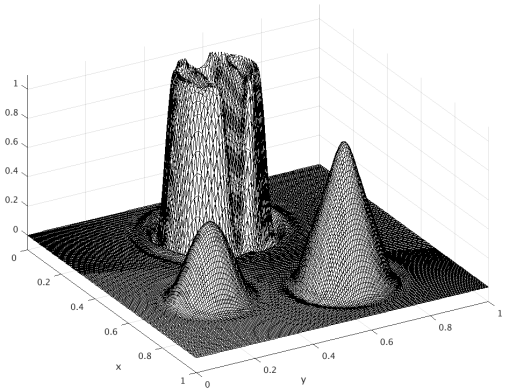
Single Domain



DFR



MFR

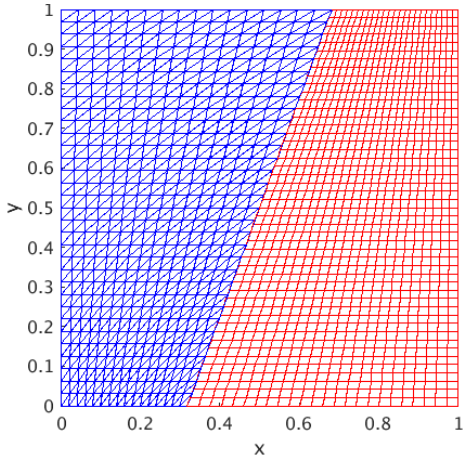


No artifacts due to interface, but some oscillations expected with SUPG stabilization

Numerical Results: Manufactured Solution

$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Mesh



$$\dot{\varphi}_k - \nabla \cdot (\epsilon \nabla \varphi_k - \mathbf{v} \varphi_k) = g_k$$

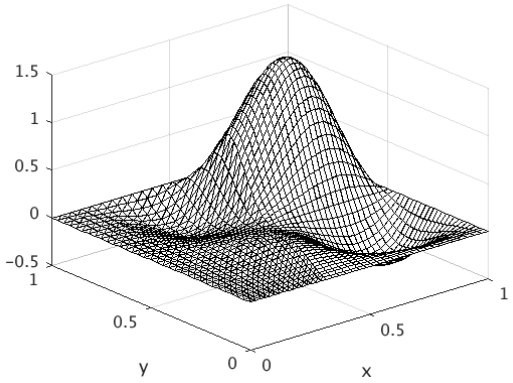
$$\mathbf{v} = \left(-\sin(\tfrac{\pi}{6}), \cos(\tfrac{\pi}{6}) \right)$$

$$\epsilon = 0.001$$

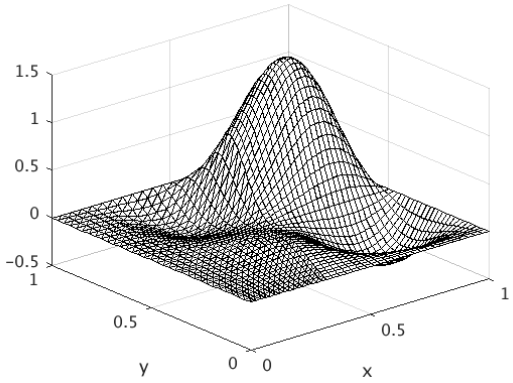
SUPG stabilization

In advection-dominated regime methods show expected convergence

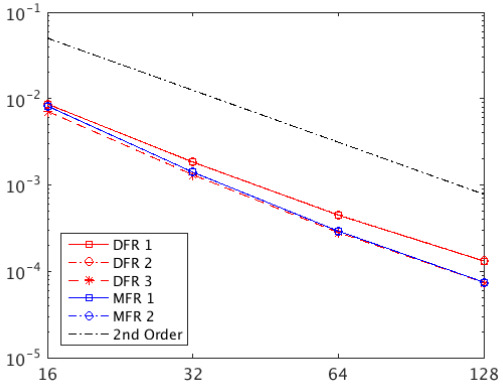
MFR $t = 1$



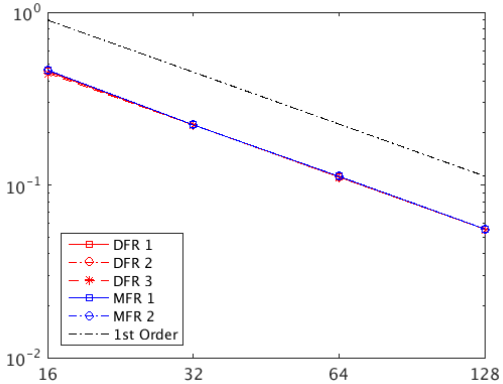
DFR $t = 1$



L^2 Error



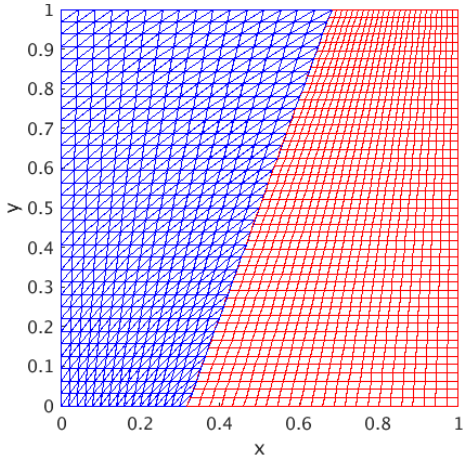
H^1 Error



Numerical Results: Manufactured Solution

$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Mesh

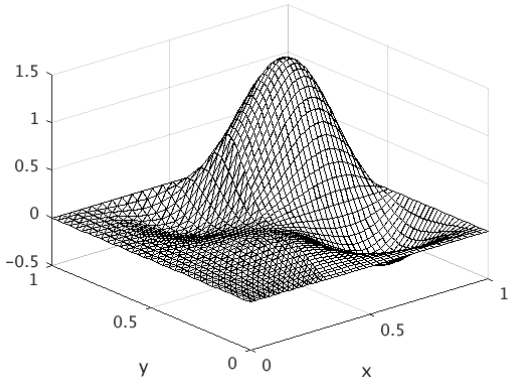


$$\dot{\varphi}_k - \nabla \cdot (\epsilon \nabla \varphi_k - \mathbf{v} \varphi_k) = g_k$$

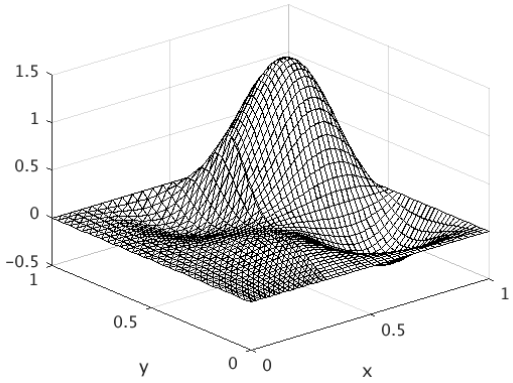
$$\mathbf{v} = 0 \; \epsilon = 0.001$$

For pure diffusion DFR loses accuracy

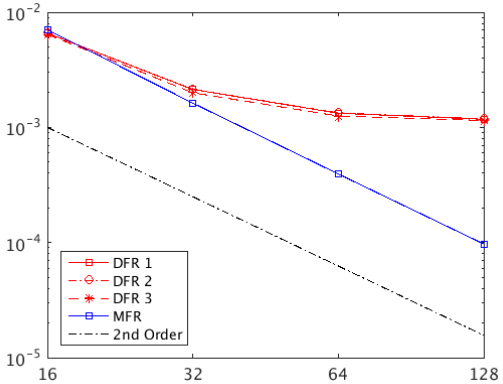
MFR $t = 1$



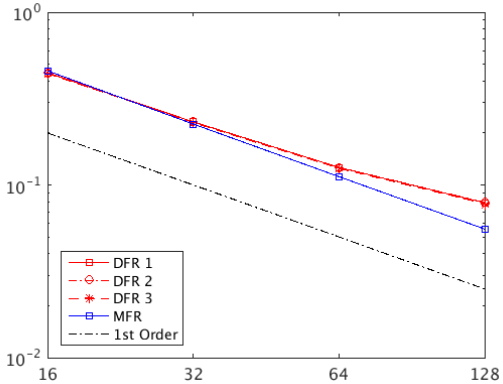
DFR $t = 1$



L^2 Error



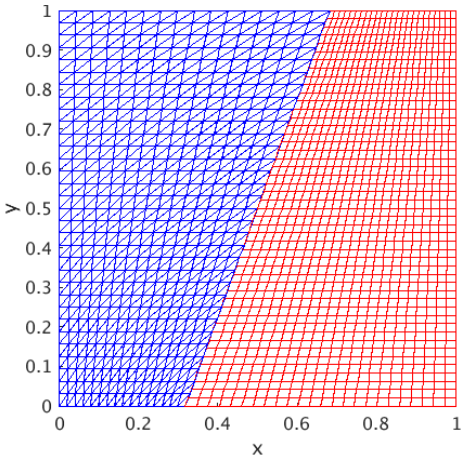
H^1 Error



Numerical Results: Manufactured Solution

$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Mesh

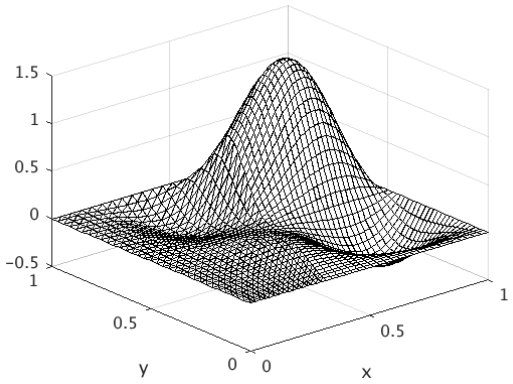


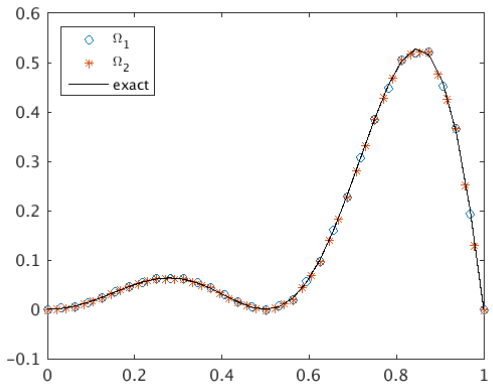
$$\dot{\varphi}_k - \nabla \cdot (\epsilon \nabla \varphi_k - \mathbf{v} \varphi_k) = g_k$$

$$\mathbf{v} = 0 \; \epsilon = 0.001$$

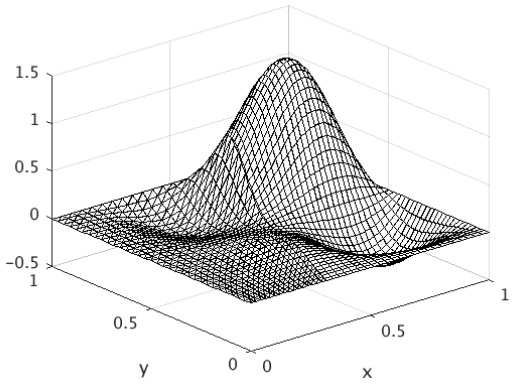
For pure diffusion DFR loses accuracy

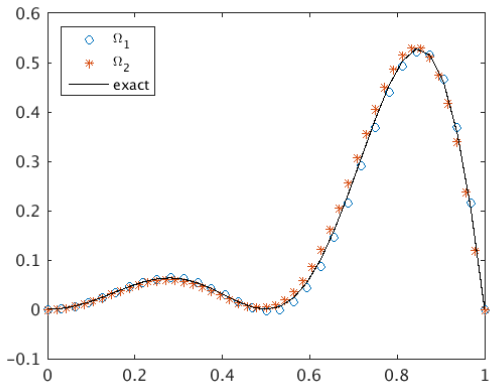
MFR $t = 1$





DFR $t = 1$

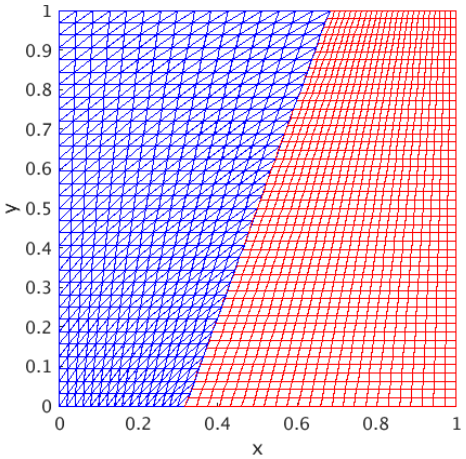




Numerical Results: Manufactured Solution

$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Mesh



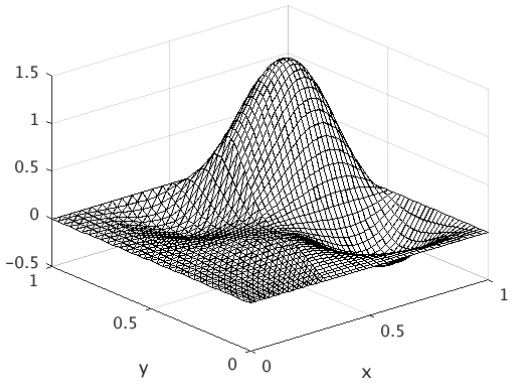
$$\dot{\varphi}_k - \nabla \cdot (\epsilon \nabla \varphi_k - \mathbf{v} \varphi_k) = g_k$$

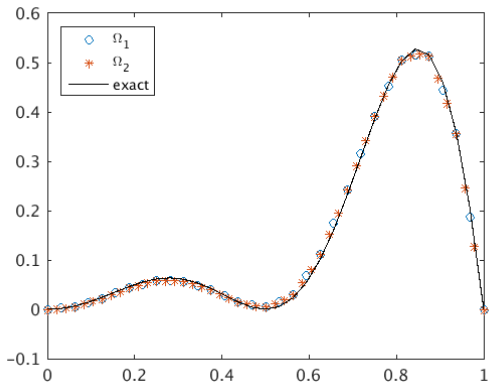
$$\mathbf{v} = \left(-\sin(\tfrac{\pi}{6}), \cos(\tfrac{\pi}{6}) \right)$$

$$\epsilon = 0.1$$

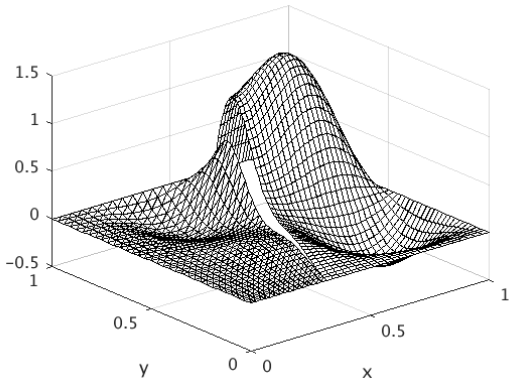
Error grows with diffusion coefficient

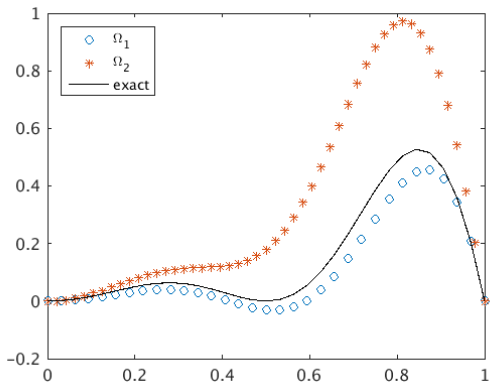
MFR $t = 1$





DFR $t = 1$



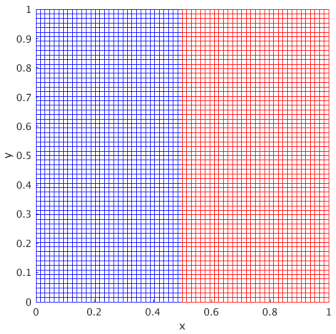


Numerical Results: Manufactured Solution

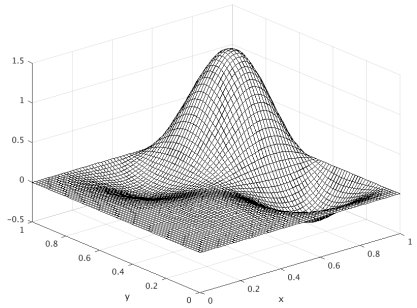
Matching grid comparison to single domain - advection dominated

$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

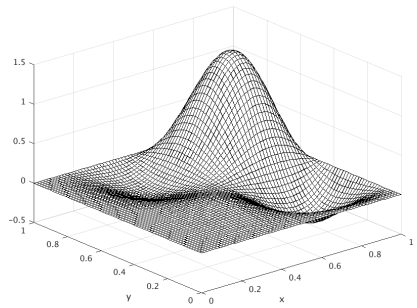
Mesh

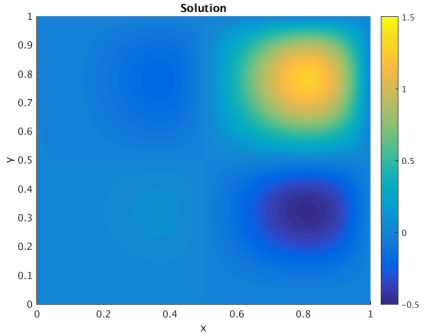


DFR

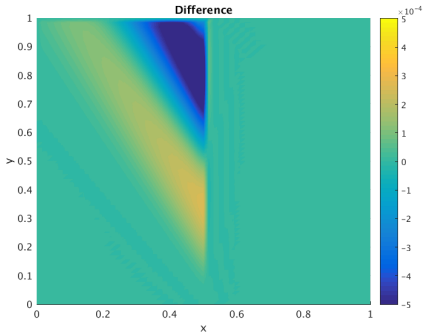


MFR

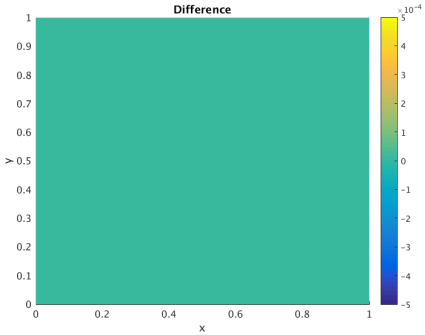




$(-0.5, 1.5)$



$(-5 \times 10^{-4}, 5 \times 10^{-4})$



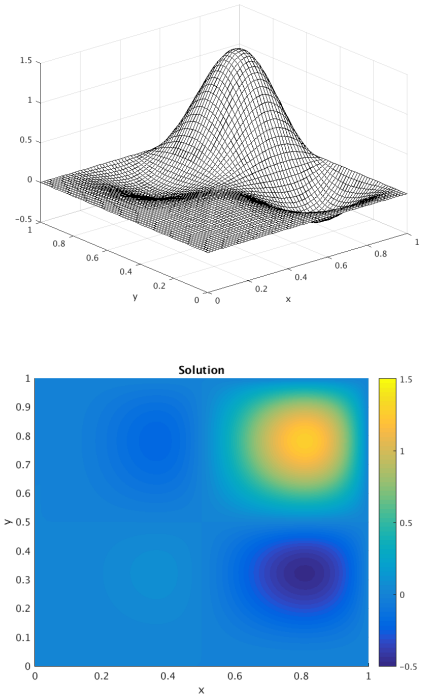
$(-5 \times 10^{-4}, 5 \times 10^{-4})$

Numerical Results: Manufactured Solution

Matching grid comparison to single domain - advection dominated

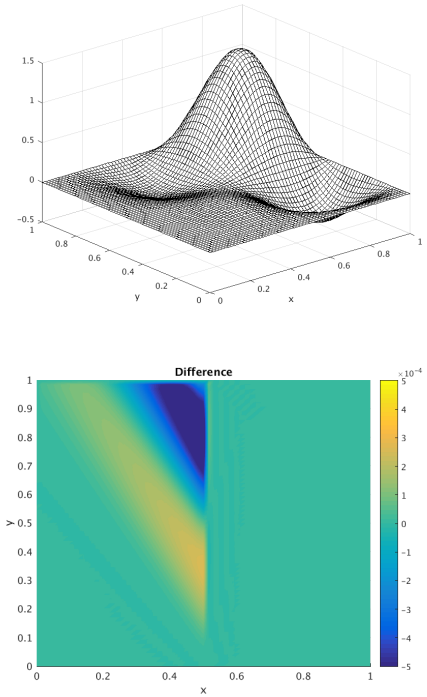
$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Single Domain



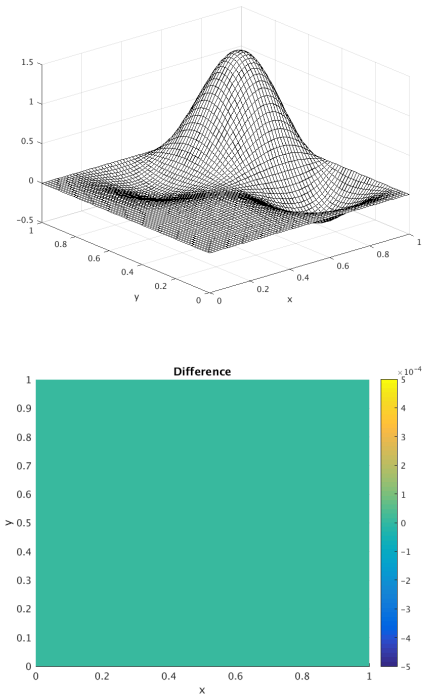
$(-0.5, 1.5)$

DFR



$(-5 \times 10^{-4}, 5 \times 10^{-4})$

MFR



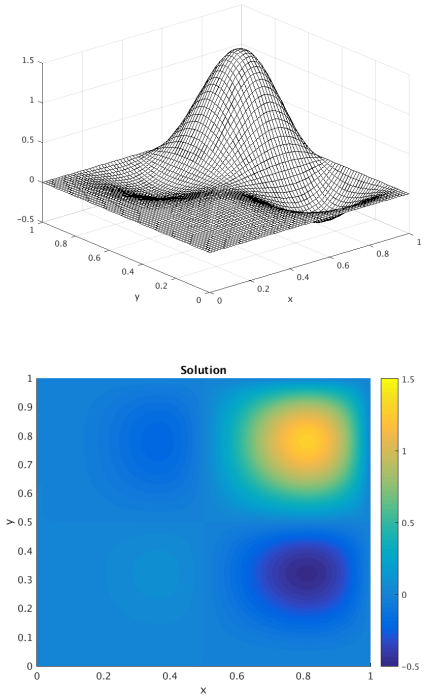
$(-5 \times 10^{-4}, 5 \times 10^{-4})$

Numerical Results: Manufactured Solution

Matching grid comparison to single domain - advection dominated

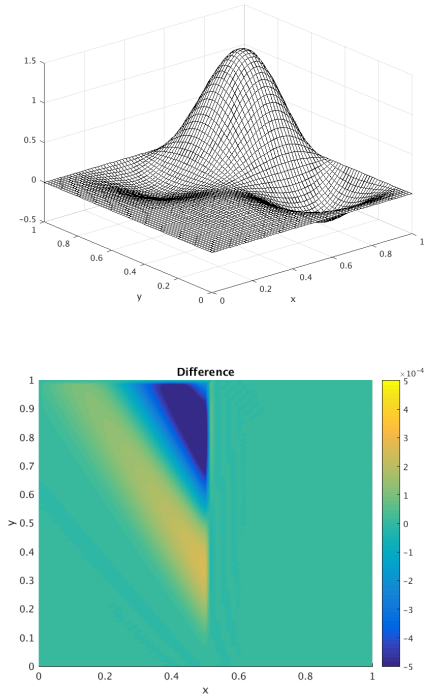
$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Single Domain



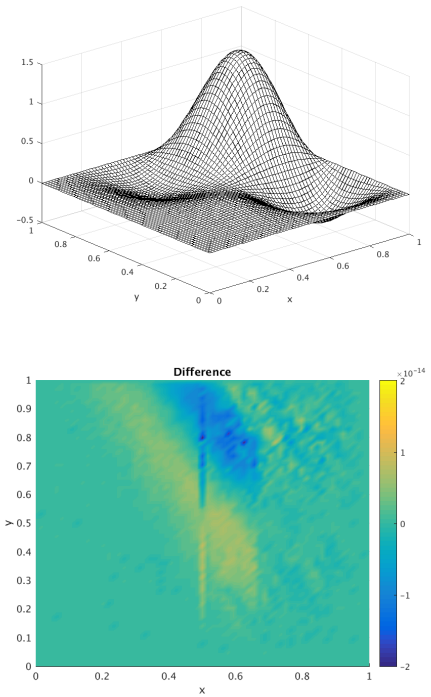
$(-0.5, 1.5)$

DFR



$(-5 \times 10^{-4}, 5 \times 10^{-4})$

MFR



$(-2 \times 10^{-14}, 2 \times 10^{-14})$

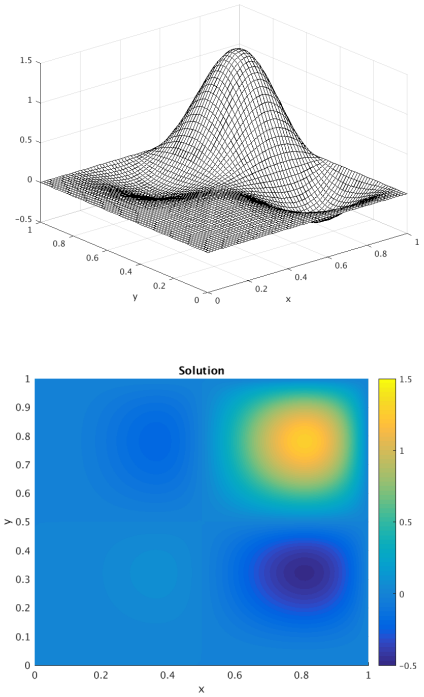
MFR recovers single domain solution

Numerical Results: Manufactured Solution

Matching grid comparison to single domain - pure diffusion

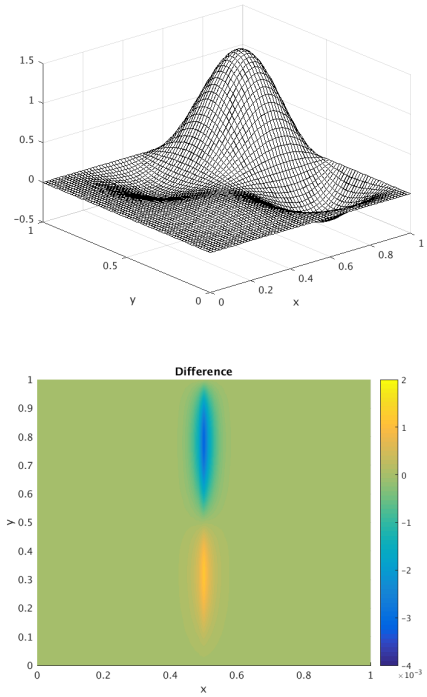
$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Single Domain



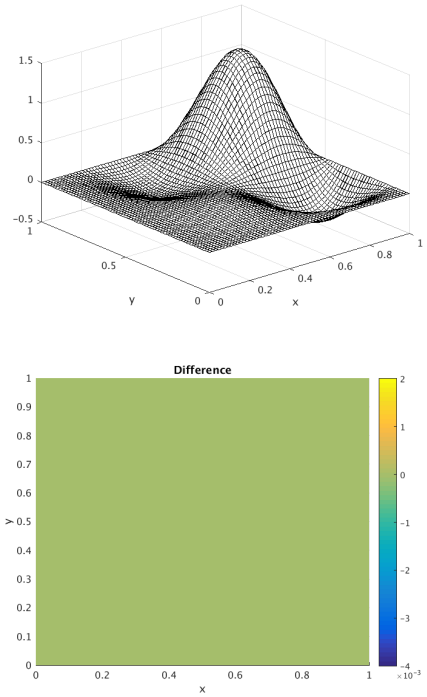
$(-0.5, 1.5)$

DFR



$(-4 \times 10^{-3}, 2 \times 10^{-3})$

MFR



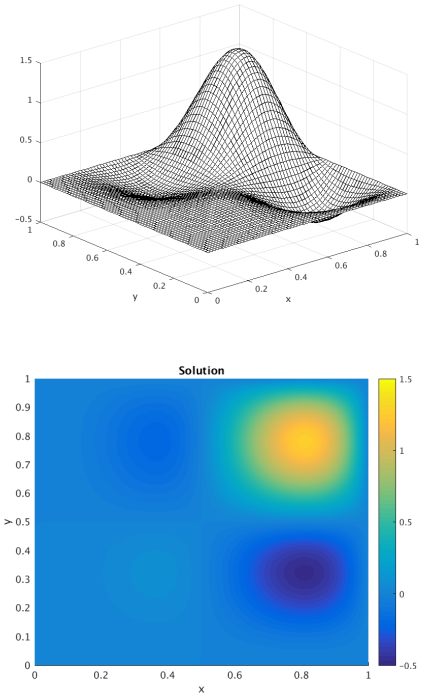
$(-4 \times 10^{-3}, 2 \times 10^{-3})$

Numerical Results: Manufactured Solution

Matching grid comparison to single domain - pure diffusion

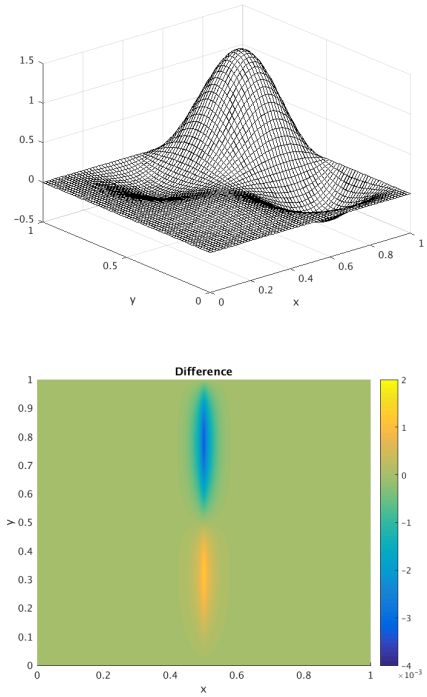
$$\varphi_k(\mathbf{x}, t) = x^2 y \sin(2\pi x) \sin(2\pi y) \exp(t)$$

Single Domain



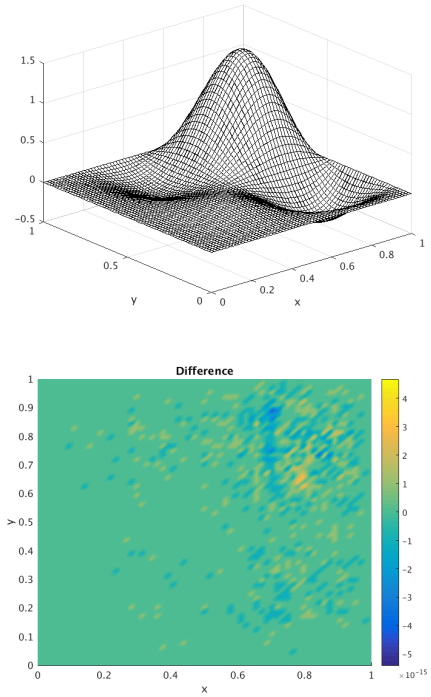
$(-0.5, 1.5)$

DFR



$(-4 \times 10^{-3}, 2 \times 10^{-3})$

MFR



$(-5 \times 10^{-15}, 4 \times 10^{-15})$

MFR recovers single domain solution

Conclusions

Presented method for approximating Neumann flux across a non-matching interface

- Confirmed the potential of Lagrange-multiplier formulation for partitioned solution of interface problems
- Key idea is to consider alternative constraint, which enables explicit treatment of Lagrange multiplier (MFR)
- Compared to a family of L^2-projection based methods described in the literature (DFR)
  - MFR comparable to DFR for advection-dominated and pure advection problems
  - DFR accuracy decreases for pure diffusion and diffusion-dominated problems