Optimal analysis-aware parameterization of computational domain in isogeometric analysis

Gang Xu, Bernard Mourrain, Régis Duvigneau, André Galligo

GALAAD and OPALE Team, INRIA Sophia-Antipolis

University of Nice Sophia-Antipolis

gxu@sophia.inria.fr

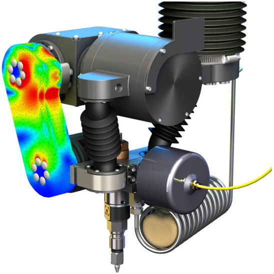
June 16th, 2010

Outline

1. *Introduction of isogeometric analysis (IGA)*
2. *Parameterization of computational domain*
3. *Linear and easy-to-check sufficient condition for injective B-spline planar parameterization*
4. *Shape optimization method for problems with exact solution*
   - Main idea and approach
   - Results and comparison
5. *Conclusion and future work*

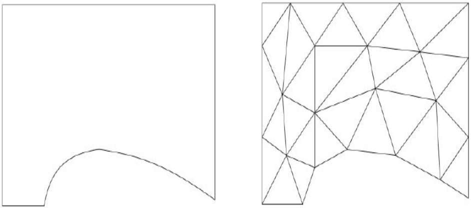
Finite element analysis (FEA)

- FEA is the most popular method for product analysis
- FEA was first developed in 1943 by R. Courant, who utilized the Ritz method of numerical analysis and minimization of variational calculus to obtain approximate solutions to vibration systems.

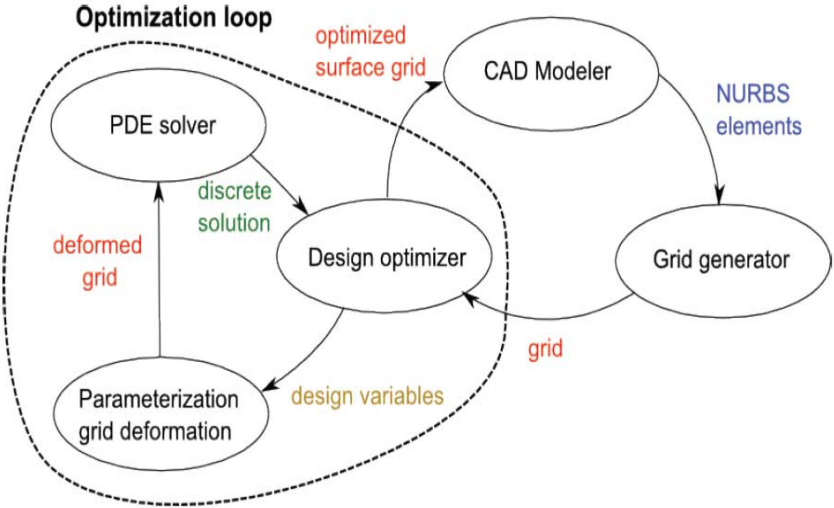


Two issues of FEA

- mesh(grid) generation: approximate the physical domain by discrete domain(triangular, quad, ...)
- mesh generation accounts for more than 80 % of overall analysis time and is the major bottleneck
- choice of basis function: Lagrange interpolation polynomial

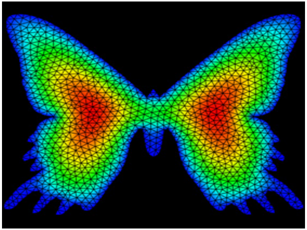


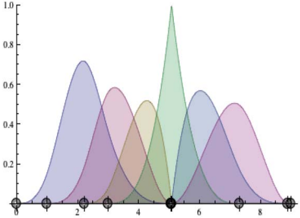
FEA-based design cycle



Gap between CAD and FEA

- computational domain:
  - FEA: discrete mesh
  - CAD: smooth domain
- basis function:
  - FEA: Lagrange interpolation polynomial
  - CAD: B-spline basis function





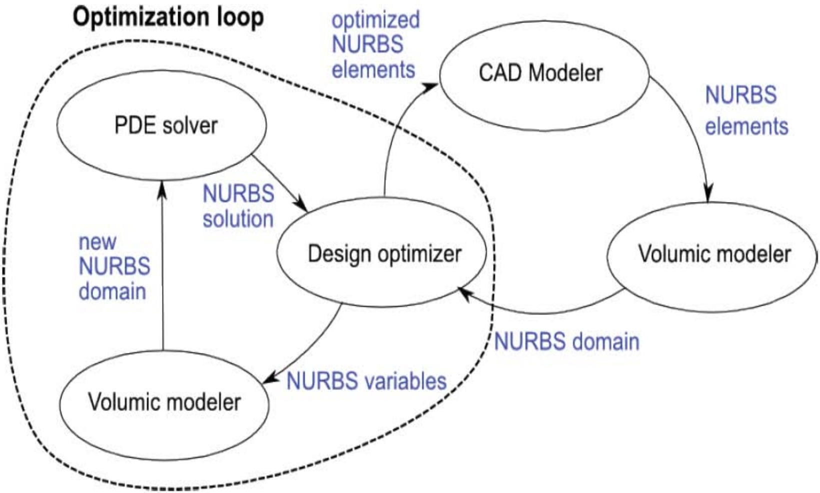
Isogeometric analysis (IGA)

- IGA is an isoparametric, exact geometry approach, which is recently providing very promising results as an alternative to FEA.
- proposed by Prof. T. Hughes et al. from University of Texas at Austin in 2005
- motivation:
  - seamless integration of CAD and CAE.
  - avoid geometry approximations of mesh generation in FEA
  - high regularity and refinement of B-spline functions.
- basic idea: use the same standard mathematical representation as in CAD systems (such as NURBS) for both the geometry and the solution field (such as thermal conduction).

Comparison of FEA and IGA

Finite Element Analysis	*Isogeometric Analysis*
Nodal points	Control points
Nodal variables	Control variables
Mesh	Knots
Basis interpolates nodal points and variables	Basis does *not* interpolate control points and variables
Approximate geometry	Exact geometry
Polynomial basis	NURBS basis
Gibbs phenomena	Variation diminishing
Subdomains	Patches

IGA based design cycle



Related work

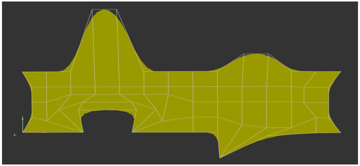
- application in various simulation problems : [Y. Bazilevs et al.2006],[J.A. Cottrell et al.2006], [Y. Bazilevs et al.2008]...
- application of various geometric modeling tools in IGA
    - NURBS : [T. Hughes et al., 2005]
    - T-spline: [M. Dörfel et al., 2010],[Y. Bazilevs et al.2010]
    - Subdivision surface: [F. Cirak et al., 2000]
    - PHT-spline: [Tian et al., 2010]
    - Catmull-Clark subdivision solids: [Burkhart et al., 2010]
- accuracy and efficiency improvement of IGA framework by reparameterization and refinement operations
    - h-refinement: knot insertion [J.A. Cottrell et al.2007]
    - p-refinement: degree elevation [T. Hughes et al., 2005]
    - k-refinement: knot insertion + degree elevation [T. Hughes et al. 2008]

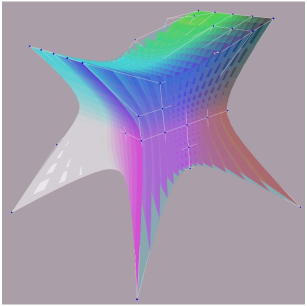
Outline

1. *Introduction of isogeometric analysis (IGA)*
2. ***Parameterization of computational domain***
3. *Linear and easy-to-check sufficient condition for injective B-spline planar parameterization*
4. *Shape optimization method for problems with exact solution*
   - Main idea and approach
   - Results and comparison
5. *Conclusion and future work*

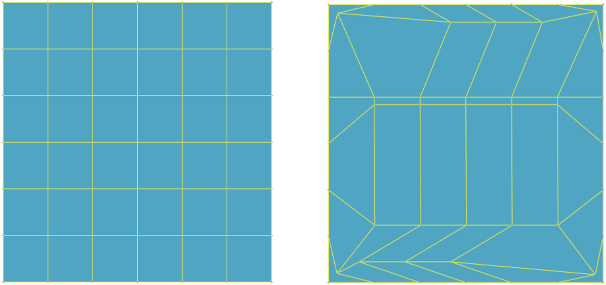
Representation in IGA

- computational domain:
  - 2D: planar B-spline surface
  - 3D: B-spline volume
- solution field :
  - 2D: B-spline surface with 3D control points
  - 3D: B-spline volume with 4D control points





- First step in IGA
- Mesh generation in FEA
- Given planar closed boundary which consists of four B-spline curves, it has various different parametrization



Test model — heat conduction problem

Given a domain Ω with $\Gamma = \partial\Omega_D \cup \partial\Omega_N$,

$$\begin{aligned} &\nabla(\kappa(x)\nabla T(x)) = f(x) \quad \text{in } \Omega \\ &T(x) = T_0(x) \text{ on } \partial\Omega_D \\ &\kappa(x)\frac{\partial T}{\partial n}(x) = \Phi_0(x) \text{ on } \partial\Omega_N, \end{aligned} \tag{1}$$

where x are the Cartesian coordinates, T represents the temperature field and κ the thermal conductivity. Dirichlet and Neumann boundary conditions are applied on $\partial\Omega_D$ and $\partial\Omega_N$ respectively, T_0 and Φ_0 being the imposed temperature and thermal flux (n unit vector normal to the boundary). f is a user-defined function that allows to generate problems with an analytical solution, by adding a source term to the classical heat conduction equation.

Isogeometric solving

- weak form

$$\int_\Omega \nabla(\kappa(x)\nabla T(x))\,\psi(x)\,d\Omega = \int_\Omega f(x)\,\psi(x)\,d\Omega \quad \forall\psi \in H^1_{\partial\Omega_D}(\Omega),$$

- integration by parts

$$-\int_\Omega \kappa(x)\nabla T(x)\,\nabla\psi(x)\,d\Omega + \int_{\partial\Omega_N} \Phi_0(x)\,\psi(x)\,d\Gamma = \int_\Omega f(x)\,\psi(x)\,d\Omega.$$

- temperature field and test function

$$T(\xi,\eta) = \sum_{i=1}^{n_i}\sum_{j=1}^{n_j} \hat{N}_i^{p_i}(\xi)\,\hat{N}_j^{p_j}(\eta)\,T_{ij},$$

$$\psi(x) = N_{ij}(x) = \hat{N}_{ij}\circ\sigma^{-1}(x) = \hat{N}_{ij}(\xi,\eta) = \hat{N}_i^{p_i}(\xi)\,\hat{N}_j^{p_j}(\eta).$$

where σ is the map from parametric domain $\mathcal{P}$ to computational domain Ω.

Isogeometric solving—continued

$$\sum_{k=1}^{n_k}\sum_{l=1}^{n_l} T_{kl}\int \kappa(x)\nabla N_{kl}(x)\nabla N_{ij}(x)d\Omega = \int \Phi_0(x)N_{ij}(x)d\Gamma - \int f(x)N_{ij}(x)d\Omega.$$

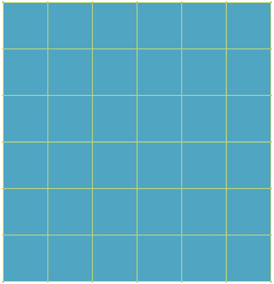
- rewritten weak formulation
- stiffness matrix and right-hand term

$$\begin{aligned} M_{ij,kl} &= \int_\Omega \kappa(x)\nabla N_{kl}(x)\ \nabla N_{ij}(x)\ d\Omega \\ &= \int_\mathcal{P} \kappa(T(u))\nabla_u \tilde{N}_{kl}(u)B(u)^T B(u)\ \nabla_u \tilde{N}_{kl}(u)J(u)\ d\mathcal{P} \\ S_{ij} &= \int_{\partial\mathcal{P}_N} \Phi_0(T(u))\ \tilde{N}_{kl}(u)J(u)\ d\tilde{\Gamma} - \int_\mathcal{P} f(T(u))\ \tilde{N}_{kl}(u)J(u)\ d\mathcal{P}. \end{aligned}$$

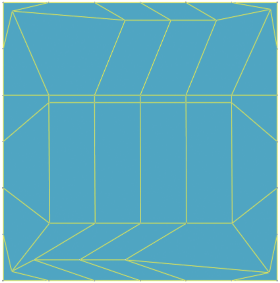
where J is the Jacobian of the transformation, B^K is the transposed of the inverse of the Jacobian matrix.

Given ....

- source term : $\boldsymbol{f}(x, y) = -\frac{4\pi^2}{9}\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- boundary condition : $\boldsymbol{T}_0(\boldsymbol{x}) = 0$ and $\Phi_0(\boldsymbol{x}) = 0$
- exact solution : $\boldsymbol{T}(x, y) = 2\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- computational domain : $\Omega(x, y) = [0, 6] \times [0, 6]$

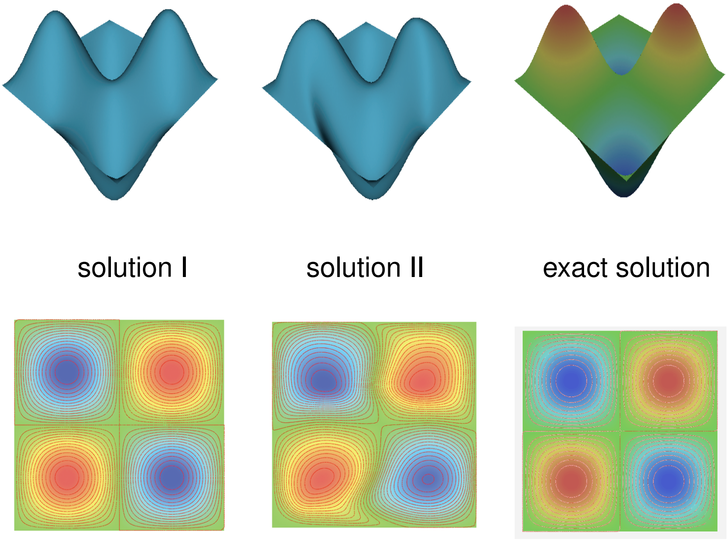


control point placement I



control point placement II

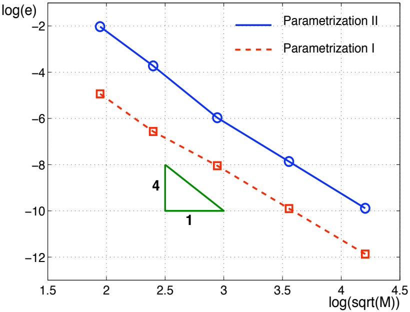
Solution comparison



Error analysis

- relative L_2 norm:

$$e = \sqrt{\frac{\int_\Omega (\boldsymbol{T} - \tilde{\boldsymbol{T}})^T (\boldsymbol{T} - \tilde{\boldsymbol{T}}) d\Omega}{\int_\Omega \boldsymbol{T}^T \boldsymbol{T} d\Omega}}$$



How to find good parametrization for IGA

- Injective mapping

$$M_{ij,kl} = \int_\Omega \kappa(x)\nabla N_{kl}(x)\ \nabla N_{ij}(x)\ d\Omega$$
$$= \int_\mathcal{P} \kappa(T(u))\nabla_u \tilde{N}_{kl}(u)B(u)^T B(u)\ \nabla_u \tilde{N}_{kl}(u)J(u)\ d\mathcal{P}$$

B^T is the transposed of the inverse of the Jacobian matrix

- Minimal cost function (error value)

Outline

1. *Introduction of isogeometric analysis (IGA)*
2. *Parameterization of computational domain*
3. ***Linear and easy-to-check sufficient condition for injective B-spline planar parameterization***
4. *Shape optimization method for problems with exact solution*
   - Main idea and approach
   - Results and comparison
5. *Conclusion and future work*

A sufficient condition for injective mapping

Proposition [Kestelman,1971]

Suppose that σ is a C^1 parameterization from a compact domain $\mathcal{P} \subset \mathbb{R}^n$ with a connected boundary to a geometry $\Omega \subset \mathbb{R}^n$. If σ is injective on the boundary $\partial\mathcal{P}$ of $\mathcal{P}$ and its Jacobian J_σ does not vanish on $\mathcal{P}$, then σ is injective.

Proposition

Let σ be a C^1 parameterization from $[a,b] \times [c,d]$ to $\Omega \subset \mathbb{R}^2$ which defines a regular boundary. If its Jacobian J_σ does not vanish on $[a,b] \times [c,d]$, then σ is injective.

Injectivity cones

- Planar B-spline parameterization:

$$\sigma : u \in \mathcal{P} := [a,b] \times [c,d] \mapsto \sigma(u) := \sum_{0 \le i \le l_1, 0 \le j \le l_2} c_{i,j} N_{i,j}(u),$$

- Derivatives:

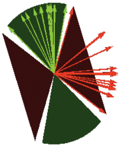
$$\partial_{u_1}\sigma(u) := \sum_{0 \le i \le l_1 - 1, 0 \le j \le l_2} \omega^1_{i,j} \Delta^1_{i,j} N^1_{i,j}(u),$$

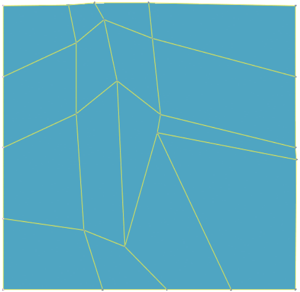
$$\partial_{u_2}\sigma(u) := \sum_{0 \le i \le l_1 - 1, 0 \le j \le l_2 - 1} \omega^2_{i,j} \Delta^2_{i,j} N^2_{i,j}(u),$$

where $\Delta^1_{i,j} := c_{i+1,j} - c_{i,j}$, $\Delta^2_{i,j} := c_{i,j+1} - c_{i,j}$

Injectivity cones — continued

- $C_1(c)$: the convex cone of $\mathbb{R}^2$ generated by the half rays $\mathbb{R}_+ \cdot \Delta^1_{i,j}$
- $C_2(c)$: the convex cone of $\mathbb{R}^2$ generated by the half rays $\mathbb{R}_+ \cdot \Delta^2_{i,j}$.



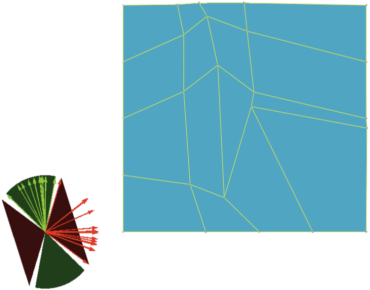


Easy-to-test injectivity condition

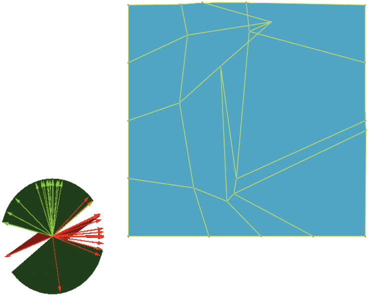
Proposition

Let σ be a B-spline parametrisation, which is at least C^1 from $\mathcal{P} := [a,b] \times [c,d]$ to $\Omega \subset \mathbb{R}^2$ given by the control points c. If the boundary curves do not intersect and have no self-intersection point and the cones $C_1(c), C_2(c)$ are transverse, then σ is injective on $\mathcal{P}$.

transverse cones



non-transverse cones



Outline

1. Introduction of isogeometric analysis (IGA)
2. Parameterization of computational domain
3. Linear and easy-to-check sufficient condition for injective B-spline planar parameterization
4. **Shape optimization method for problems with exact solution**
   - Main idea and approach
   - Results and comparison
5. Conclusion and future work

Outline

1. *Introduction of isogeometric analysis (IGA)*
2. *Parameterization of computational domain*
3. *Linear and easy-to-check sufficient condition for injective B-spline planar parameterization*
4. ***Shape optimization method for problems with exact solution***
   - Main idea and approach
   - Results and comparison
5. *Conclusion and future work*

Problem statement

Problem statement

given four coplanar boundary B-spline curves, find the inner control points such that the parameterization of a computational domain is optimal for an IGA problem with known exact solution

- Remark: on-going work for isogeometric problems without known exact solution

Main idea

- Inspired from shape optimization
- Shape optimization: optimize the boundary model to minimize the error function (cost function) of simulation

Main idea

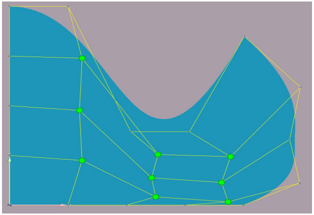
Let the inner control points, rather than boundary control points, be the design variables for the shape optimization, and find the best placement of inner control points to make the value of a cost function as small as possible.

Initial construction of inner control points

- Discrete Coons method [G.Farin, 1999]

  Given the boundary control points $\boldsymbol{P}_{0,j}, \boldsymbol{P}_{n,j}, \boldsymbol{P}_{i,0}, \boldsymbol{P}_{i,m}$, the inner control points $\boldsymbol{P}_{i,j}$ can be constructed as follows:

$$\begin{aligned}\boldsymbol{P}_{i,j} &= (1-\frac{i}{n})\boldsymbol{P}_{0,j} + \frac{i}{n}\boldsymbol{P}_{n,j} + (1-\frac{j}{m})\boldsymbol{P}_{i,0} + \frac{j}{m}\boldsymbol{P}_{i,m} \\ &\quad -[1-\frac{i}{n} \quad \frac{i}{n}]\begin{pmatrix} \boldsymbol{P}_{0,0} & \boldsymbol{P}_{0,m} \\ \boldsymbol{P}_{n,0} & \boldsymbol{P}_{n,m} \end{pmatrix}\begin{pmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{pmatrix}\end{aligned}$$



Main approach

- optimization variables: the coordinates of the inner control points
- cost function: error of the IGA solution
- optimization algorithm: steepest-descent method in conjunction with a back-tracking line-search
  1. Evaluation of perturbed points $x_k + \epsilon e_k$
  2. Estimation of the gradient $\nabla f(x_k)$ by finite-difference
  3. Define search direction $d_k = -\nabla f(x_k)$
  4. Line search : find ρ such as $f(x_k + \rho d_k) < f(x_k)$

Outline

1 *Introduction of isogeometric analysis (IGA)*

2 *Parameterization of computational domain*

3 *Linear and easy-to-check sufficient condition for injective B-spline planar parameterization*

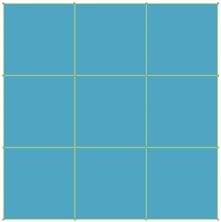
4 *Shape optimization method for problems with exact solution*
- Main idea and approach
- Results and comparison

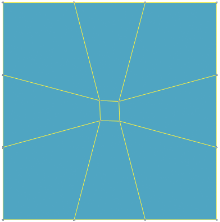
5 *Conclusion and future work*

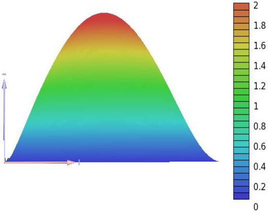
Example 1: given...

- source term : $\boldsymbol{f}(x,y) = -\frac{4\pi^2}{9}\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- boundary condition : $\boldsymbol{T}_0(\boldsymbol{x}) = 0$ and $\Phi_0(\boldsymbol{x}) = 0$
- exact solution : $\boldsymbol{T}(x,y) = 2\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- computational domain : $\Omega(x,y) = [0,3]\times[0,3]$

Example 1: results



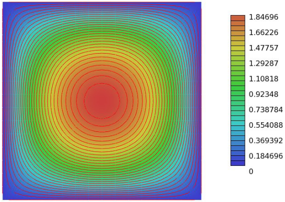


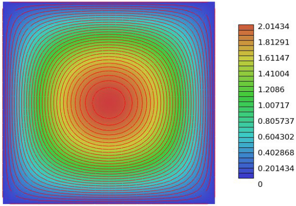


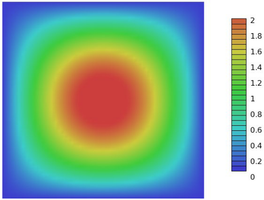
initial solution

final solution

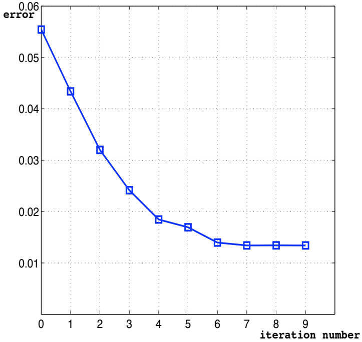
exact solution

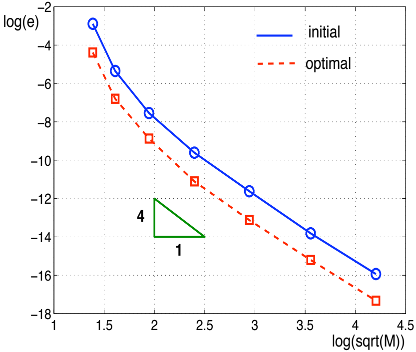






Example 1: error analysis





error history of optimization(24.52%) error history of h-refinement

Example 2: given...

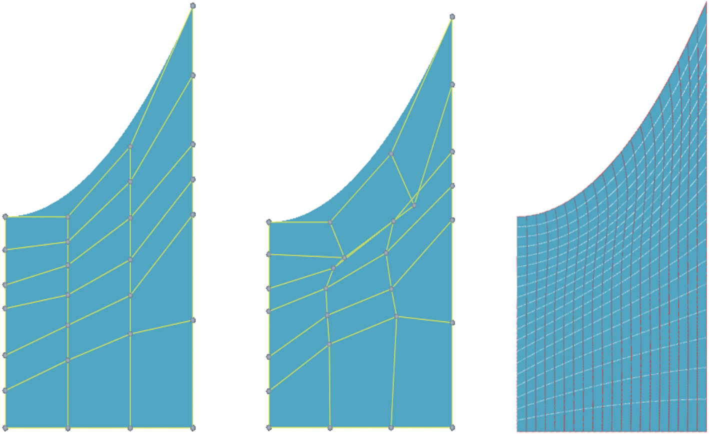
- boundary condition : $\boldsymbol{T}_0(\boldsymbol{x}) = 0$ and $\Phi_0(\boldsymbol{x}) = 0$
- exact solution :

$$\boldsymbol{T}(x, y) = \sin(\pi(y - x^2)) \sin(\pi x) \sin(\pi y)$$

- computational domain :

$$\Omega(x, y) = \{(x, y) | -1 \leq y \leq x^2, 0 \leq x \leq 1\}$$

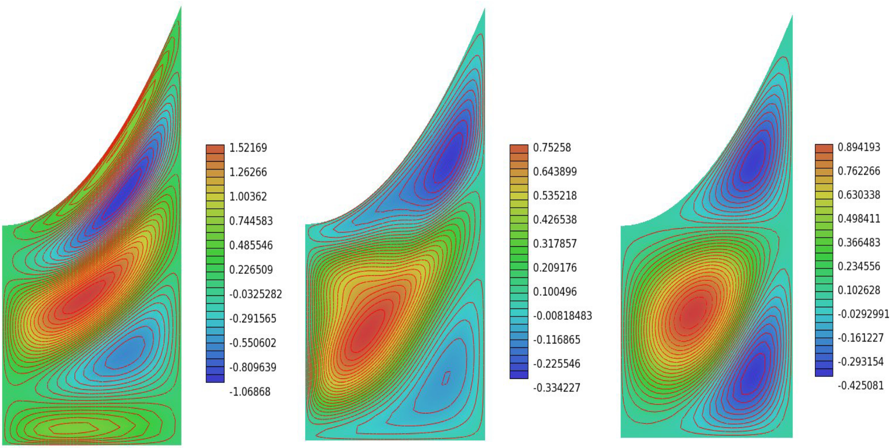
Example 2: results



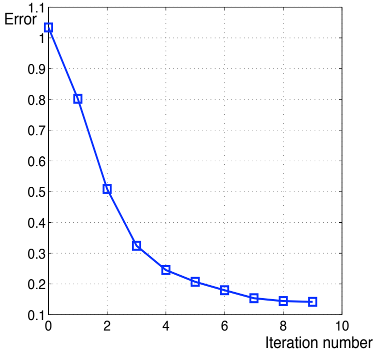
initial domain　　final domain　　final isoparametric net

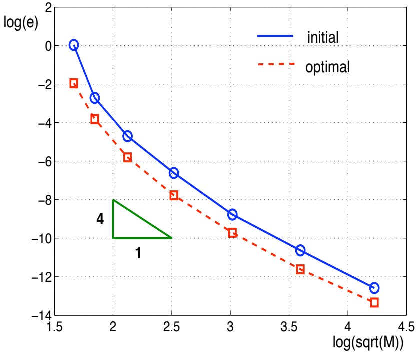
Example 2: simulation results

initial solution | final solution | exact solution



Example 2: error analysis





error history of optimization(14.65%)

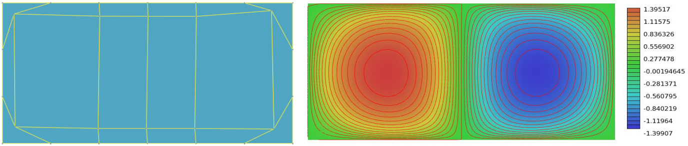
error history of h-refinement

Example 3: given...

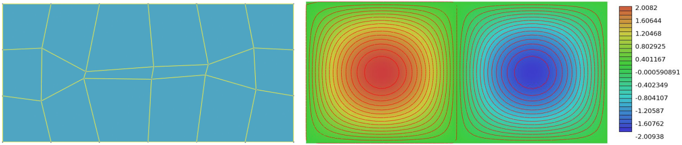
- source term : $\boldsymbol{f}(x,y) = -\frac{4\pi^2}{9}\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- boundary condition : $\boldsymbol{T}_0(\boldsymbol{x}) = 0$ and $\Phi_0(\boldsymbol{x}) = 0$
- exact solution : $\boldsymbol{T}(x,y) = 2\sin(\frac{\pi x}{3})\sin(\frac{\pi y}{3})$.
- computational domain : $\Omega(x,y) = [0,6] \times [0,3]$

Example 3: results

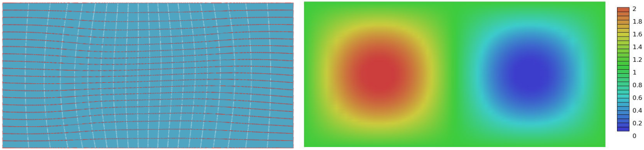
initial:



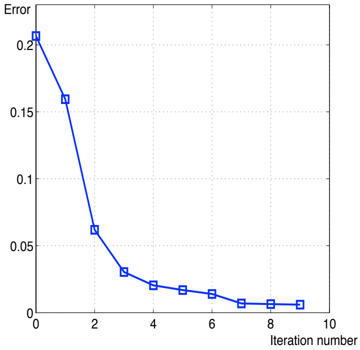
final:

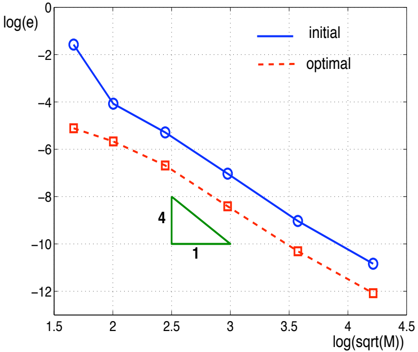


exact:



Example 3: error analysis





error history of optimization(3.31%)   error history of h-refinement

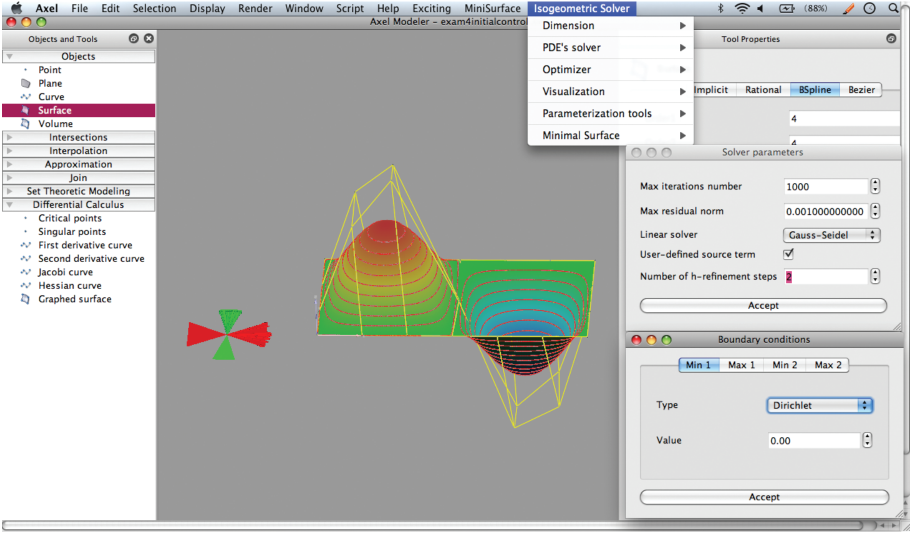
Outline

1. *Introduction of isogeometric analysis (IGA)*
2. *Parameterization of computational domain*
3. *Linear and easy-to-check sufficient condition for injective B-spline planar parameterization*
4. *Shape optimization method for problems with exact solution*
   - Main idea and approach
   - Results and comparison
5. ***Conclusion and future work***

Conclusion

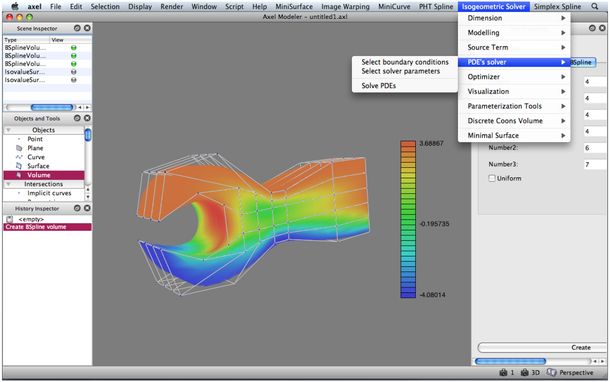
- Parametrization of computational domain has impact on accuracy and efficiency.
- Linear and easy-to-check sufficient condition for injectivity
- r-refinement method in isogeometric version: minimizing the cost function while keeping number of elements as constant
- Implementation in algebraic-geometric modeler AXEL http://axel.inria.fr/

Interface for isogeometric solver in AXEL



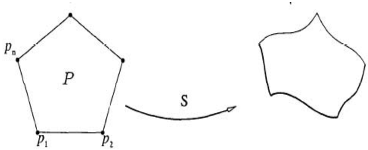
Ongoing work

- r-refinement for general problems without known exact solution: error estimation
- r-h-refinement for isogeometric analysis
- 3D cases:

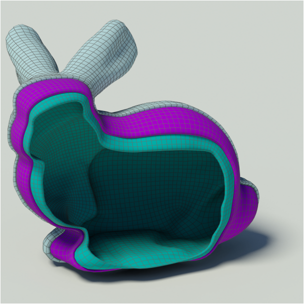


Future work— more complex geometry

- 2D problem: n-side boundary condition ($n \geq 3$)



- 3D problem: volume parametrization by using trivariate B-splines



from Dr. Ying He.

Thanks for your attention!

Question?