Penalty function-based volumetric parameterization method for isogeometric analysis

Ye Ji, Mengyun Wang, Maodong Pan, Yi Zhang, Chungang Zhu*

Institute of Computational Science, School of Mathematical Sciences

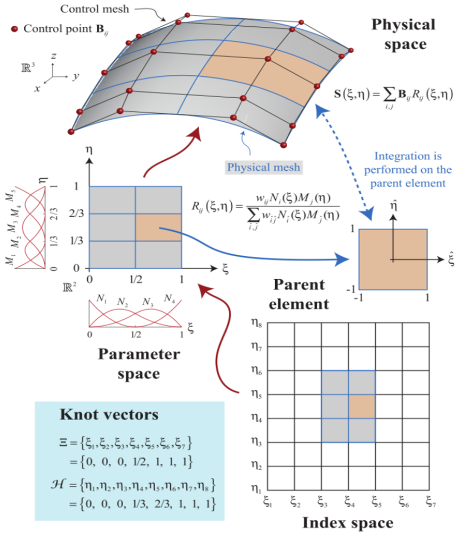
Dalian University of Technology

Dalian, China

May 11, 2022

Catalogue

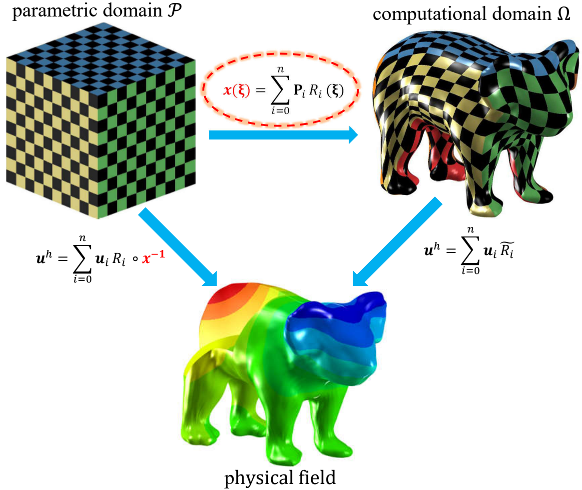
Isogeometric analysis (IGA)



Source: Figure from [Cottrell et al. 2009]

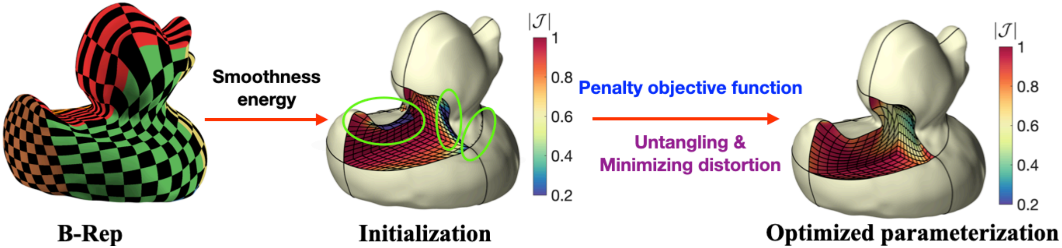
- Proposed by T.J.R. Hughes et al., 2005.
- **KEY IDEA:** approximate the physical fields with the same basis functions as that used to generate CAD models.
- Advantages:
  - Integration of design and analysis;
  - Exact and efficient geometry;
  - No data type transition and mesh generation;
  - Simplified mesh refinement;
  - High order continuous field;
  - Superior approximation properties.
- Very broad applications: such as shell analysis, fluid-structure interaction, and shape and topology optimization.

Problem statement



- However, modern CAD systems usually focus on boundary representations (B-Reps) in solid modeling.
- **Problem statement**:
  - From a given B-Rep, constructing an analysis-suitable parameterization x (a fundamental task in IGA).
  - Analysis-suitable parameterizations should
    - be **bijective**;
    - ensure as **low angle and volume distortion** as possible.

Framework overview of the proposed method



- **Robust and efficient volumetric parameterization** method based on penalty function;
- **Untangling and minimizing distortion perform simultaneously**;
- Avoids extra foldover elimination steps and is **very easy-to-implement**.

Catalogue

Related work - planar parameterization

- Crucial influence of parameterization quality on subsequent analysis: Cohen+2010, Xu+2013a, Pilgerstorfer+2014.
- Planar domain parameterization:
  - Single-patch:
    - Algebraic methods: discrete Coons method [Farin and Hansford 1999], linear methods [Gravesen+2012];
    - Constrained optimization methods: Xu+2011, Gravesen+2014, Ugalde+2018;
    - Variation harmonic mapping [Xu+2013b], PDE-based method [Hinz+2018], Teichmüller mapping [Nian and Chen 2016], low-rank quasi-conformal method [Pan+2018], large elastic deformation method [Shamanskiy+2020];
    - Barrier function method [Ji+2021];
    - Jacobian regularization technique [Garanzha+1999 2021, Wang and Ma 2021].
  - Multi-patch: Xu+2015, Buchegger+2018, Xu+2018, Xiao+2018, Kapl+2017a 2017b 2018 2019, Blidia+2020, Bastl and Slabá 2021, Wang+2022.

Related work - volumetric parameterization

- Compared with the planar problem, constructing analysis-suitable volumetric parameterizations is more challenging both geometrically and computationally.
- **Single-block**:
  - Constrained optimization methods: Xu+2013c 2017, Wang and Qian 2014
    Suffer from computing huge amounts of constraints (impractical for large-scale problems);
  - Spline fitting methods: Martin+2009, Lin+2015, Liu+2020, Yuan+2021
    Need mesh generation of the discretized computational domains;
  - Barrier function methods: Pan and Chen 2019, Pan+2020
    Need an already bijective initialization which is usually difficult to obtain.
- Multi-block: Xu+2013 2017, Lin+2018, Chen+2019 2022, Haberleitner+2019.
- Non-standard B-splines or NURBS: such as C^1 Powell-Sabin splines, toric patches, THB-splines, T-splines, PHT-splines, and Catmull-Clark volumetric subdivision.

Catalogue

Problem restatement

- A NURBS parameterization $\boldsymbol{x}$ from the parametric domain $\mathcal{P} = [0,1]^3$ to computational domain Ω is of the following form

$$\boldsymbol{x}(\boldsymbol{\xi}) = \mathbf{R}^{\mathrm{T}}\mathbf{P} = \underbrace{\sum_{i\in\mathcal{I}_I} \mathbf{P}_i R_i(\boldsymbol{\xi})}_{\text{unknown}} + \underbrace{\sum_{j\in\mathcal{I}_B} \mathbf{P}_j R_j(\boldsymbol{\xi})}_{\text{known}}, \quad (1)$$

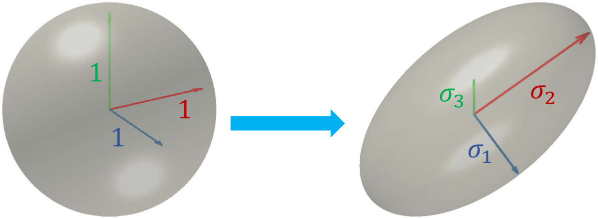
  where $\mathbf{P}_i$ are unknown inner control points and $\mathbf{P}_j$ are the given boundary control points.
- **GOAL:** To construct the unknown inner control points $\mathbf{P}_i$ such that the resulting parameterization $\boldsymbol{x}$ is bijective and has the lowest possible angle and volume distortion.

Objective function: Angle distortion

- 3D Most-Isometric ParameterizationS (MIPS) energy [Fu+2015]

$$E_{\text{mips}} = \frac{1}{8}\left(\frac{\sigma_1}{\sigma_2}+\frac{\sigma_2}{\sigma_1}\right)\left(\frac{\sigma_2}{\sigma_3}+\frac{\sigma_3}{\sigma_2}\right)\left(\frac{\sigma_1}{\sigma_3}+\frac{\sigma_3}{\sigma_1}\right) = \frac{1}{8}\left(\kappa_F^2(\boldsymbol{\mathcal{J}})-1\right); \quad (2)$$

- When $\sigma_1 = \sigma_2 = \sigma_3$, the parameterization $\boldsymbol{x}$ has the lowest angle distortion.



Objective function: Volume distortion

- Volume distortion energy term:

$$E_{\text{vol}} = \frac{vol(\Omega)}{|\boldsymbol{\mathcal{J}}|} + \frac{|\boldsymbol{\mathcal{J}}|}{vol(\Omega)}, \tag{3}$$

  where $vol(\Omega)$ denotes the volume of the computational domain Ω;

- *How to calculate $vol(\Omega)$?* Divergence Theorem!

$$vol(\Omega) = \iiint\limits_{\Omega} 1 d\Omega == \frac{1}{3} \iint\limits_{\partial\Omega} \left(x_1 \ \mathrm{d}x_2 \mathrm{d}x_3 + x_2 \ \mathrm{d}x_3 \mathrm{d}x_1 + x_3 \ \mathrm{d}x_1 \mathrm{d}x_2\right); \tag{4}$$

- **Only the given B-Rep is required** in (4), and **reduces computational costs**.

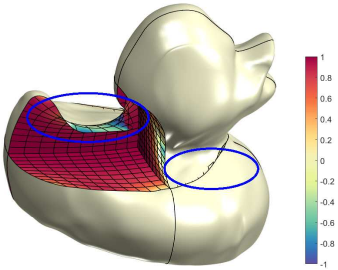
Problem with the basic objective function

- Recall the MIPS energy

$$E_{\text{mips}} = \frac{1}{8}\left(\frac{\sigma_1}{\sigma_2}+\frac{\sigma_2}{\sigma_1}\right)\left(\frac{\sigma_2}{\sigma_3}+\frac{\sigma_3}{\sigma_2}\right)\left(\frac{\sigma_1}{\sigma_3}+\frac{\sigma_3}{\sigma_1}\right) = \frac{1}{8}\left(\frac{\left(\sigma_1^2+\sigma_2^2+\sigma_3^2\right)\left(\sigma_2^2\sigma_3^2+\sigma_1^2\sigma_3^2+\sigma_1^2\sigma_2^2\right)}{|\mathcal{J}|^2}-1\right); \tag{5}$$

- The Jacobian determinant $|\mathcal{J}|$ appears in the denominator, which forms a barrier and suppresses foldovers;
- However, the **prerequisite is to find an already bijective initialization**, which is difficult to obtain efficiently for complex computational domains;
- Several previous works [Pan+2020, Ji+2021] try to handle this issue by **"extra" foldovers elimination steps** (usually complicated and time-consuming).

Initialization



Initial parameterization.

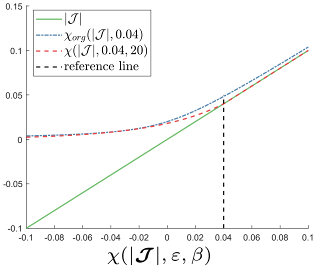
- The Initialization is obtained by minimizing the smoothness energy (**often NOT bijective**)

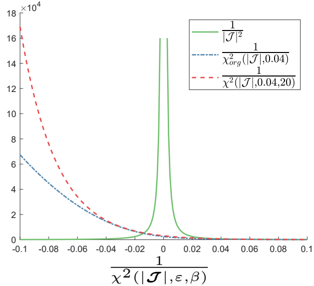
$$\iiint_{\mathcal{P}} \|\Delta \mathbf{x}\|^2 \ \mathrm{d}\mathcal{P}, \tag{6}$$

where $\Delta = \frac{\partial^2}{\partial \xi_1^2} + \frac{\partial^2}{\partial \xi_2^2} + \frac{\partial^2}{\partial \xi_3^2}$;

- With many foldovers.
- Next step is untangling and minimizing distortion.

Basic idea: Penalty function and Jacobian regularization





- **Penalty function:**

$$\chi(|\mathcal{J}|, \varepsilon, \beta) = \begin{cases} \varepsilon \cdot e^{\beta(|\mathcal{J}|-\varepsilon)} & \text{if } |\mathcal{J}| \leq \varepsilon \\ |\mathcal{J}| & \text{if } |\mathcal{J}| > \varepsilon \end{cases}, \quad (7)$$

where ε is a small positive number and β is a penalty coefficient;

- $\chi(|\mathcal{J}|, \varepsilon, \beta)$ equals a small positive number if $|\mathcal{J}| < \varepsilon$, and strictly equals the Jacobian determinant $|\mathcal{J}|$ if $|\mathcal{J}| \geq \varepsilon$;

- Consequently, $\frac{1}{\chi^2(|\mathcal{J}|,\varepsilon,\beta)}$ have **very large values to penalize the negative Jacobians and small values to accept positive Jacobians**.

Corrected objective function

- With this basic idea, finally, we solve the following optimization problem:

$$\begin{aligned}\underset{\mathbf{P}_i,\ i\in\mathcal{I}_I}{\arg\min} E^c &= \iiint\limits_{\mathcal{P}} \left(\lambda_1 E^c_{\text{mips}} + \lambda_2 E^c_{\text{vol}}\right)\, \mathrm{d}\mathcal{P} \\ &= \iiint\limits_{\mathcal{P}} \left(\frac{\lambda_1}{8}(\kappa_F^2(\mathcal{J}) \cdot \frac{|\mathcal{J}|^2}{\chi^2(|\mathcal{J}|,\varepsilon,\beta)} - 1) + \lambda_2\left(\frac{vol(\Omega)}{\chi(|\mathcal{J}|,\varepsilon,\beta)} + \frac{\chi(|\mathcal{J}|,\varepsilon,\beta)}{vol(\Omega)}\right)\right)\, \mathrm{d}\mathcal{P},\end{aligned} \tag{8}$$

where $\mathbf{P}_i,\ i \in \mathcal{I}_I$ are the unknown inner control points.

Analytical gradient computation

- During the gradient-based optimization process, an analytical gradient calculation is very important for efficiency and stability;
- Through the chain rule, we have

$$\partial_p \kappa_F^2(\mathcal{J}) = 2\ Tr((\|\mathcal{J}^{-1}\|_F^2 \mathcal{J}^{\mathrm{T}} - \|\mathcal{J}\|_F^2 (\mathcal{J}\mathcal{J}^{\mathrm{T}}\mathcal{J})^{-1})\partial_p \mathcal{J}). \tag{9}$$

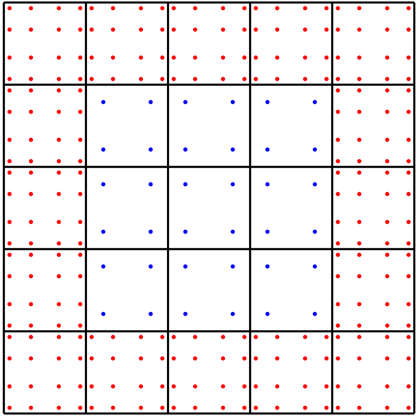
and

$$\partial_p \kappa_{F,\varepsilon}^2(\mathcal{J}) = \frac{\partial_p \kappa_F^2(\mathcal{J})|\mathcal{J}|^2 + 2\kappa_F^2(\mathcal{J})|\mathcal{J}|\partial_p|\mathcal{J}|}{\chi^2} - 2\kappa_{F,\varepsilon}^2(\mathcal{J})\frac{\partial \chi}{\partial|\mathcal{J}|}\frac{\partial_p|\mathcal{J}|}{\chi}; \tag{10}$$

- Eventually, we obtain the partial derivatives of the corrected objective function

$$\partial_p \kappa_{F,\varepsilon}^2(\mathcal{J}) = \frac{\partial_p \kappa_F^2(\mathcal{J})|\mathcal{J}|^2 + 2\kappa_F^2(\mathcal{J})|\mathcal{J}|\partial_p|\mathcal{J}|}{\chi^2} - 2\kappa_{F,\varepsilon}^2(\mathcal{J})\frac{\partial \chi}{\partial|\mathcal{J}|}\frac{\partial_p|\mathcal{J}|}{\chi}. \tag{11}$$

Reduced numerical integration scheme

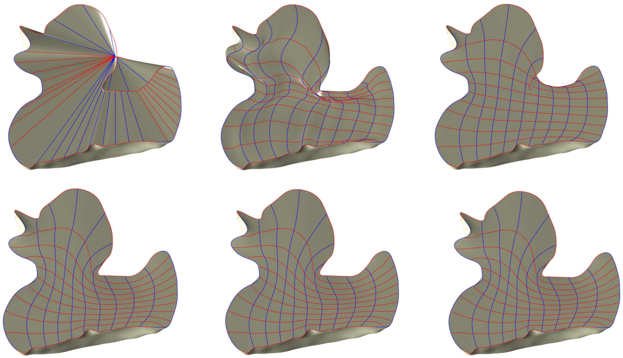


Bi-cubic NURBS parameterization: 4×4 Gaussian integration points for the layer elements and 2×2 points for the inner elements.

- **OBSERVATION**: the Jacobians vary greatly near the boundary, but are often relatively flat inside.
- More integration points for the layer elements, and fewer integration points for the inner elements.
- In addition, we precompute the basis functions before iteration to further improve the computational efficiency.

Catalogue

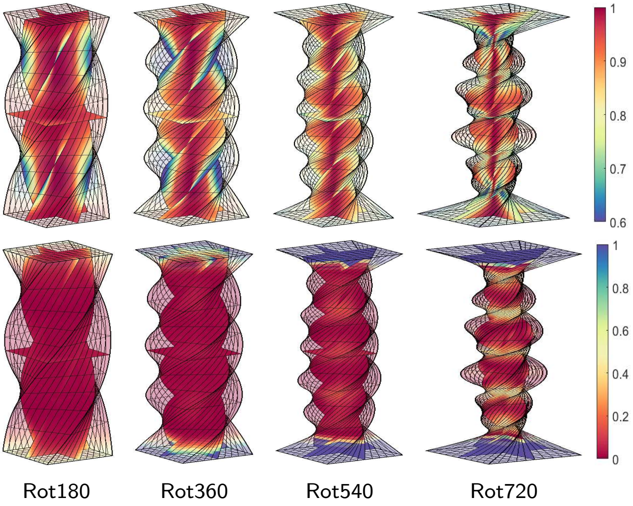
Parameterization results from different initialization methods



Same point   Discrete Coons   Smoothness energy

- The resulting parameterizations are almost the same from different initializations.
- It means our method converges to the same minimum and is insensitive to different initializations.

Robustness test



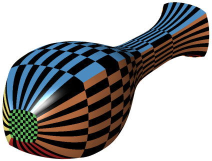
- Rotated cuboids parameterized by tri-cubic NURBS solids.
- Quality metrics:
  - **Scaled Jacobian** (optimal value 1):

$$m_{SJ} = \frac{|\mathcal{J}|}{\|\mathbf{x}_{,\xi_1}\| \cdot \|\mathbf{x}_{,\xi_2}\| \cdot \|\mathbf{x}_{,\xi_3}\|}.$$

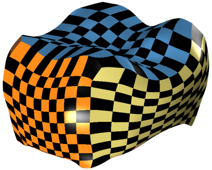
  - **Uniformity metric** (optimal value 0):

$$m_{unif.} = \left(\frac{|\mathcal{J}|}{vol(\Omega)} - 1\right)^2.$$

Six more complicated models

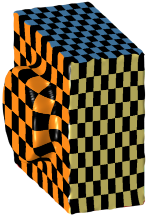


Vase

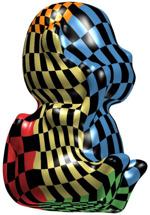


Tooth

Duck



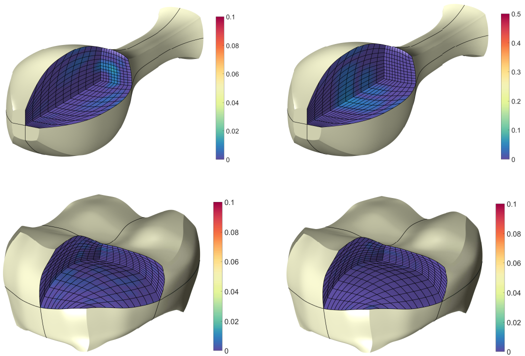
Component



Monkey

Koala

Comparison: Reduced numerical integration vs. high precision integration

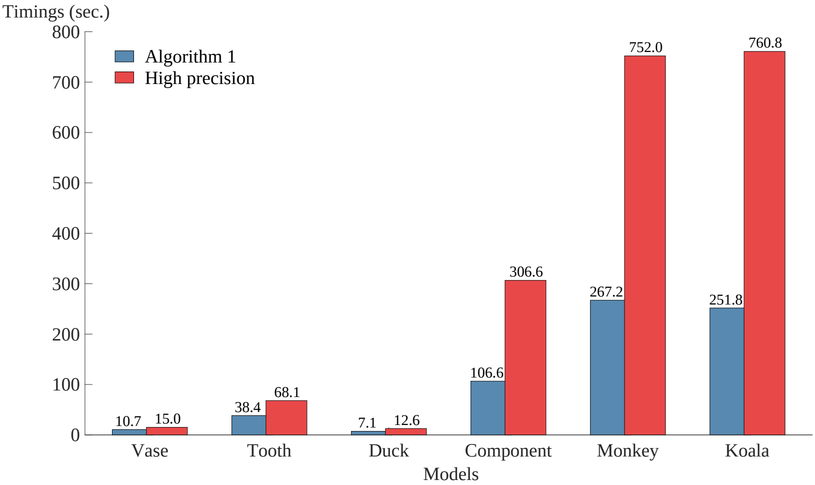


$\left| m_{SJ}^{\text{HP}} - m_{SJ}^{\text{Algo. 1}} \right|$ $\left| m_{unif.}^{\text{HP}} - m_{unif.}^{\text{Algo. 1}} \right|$

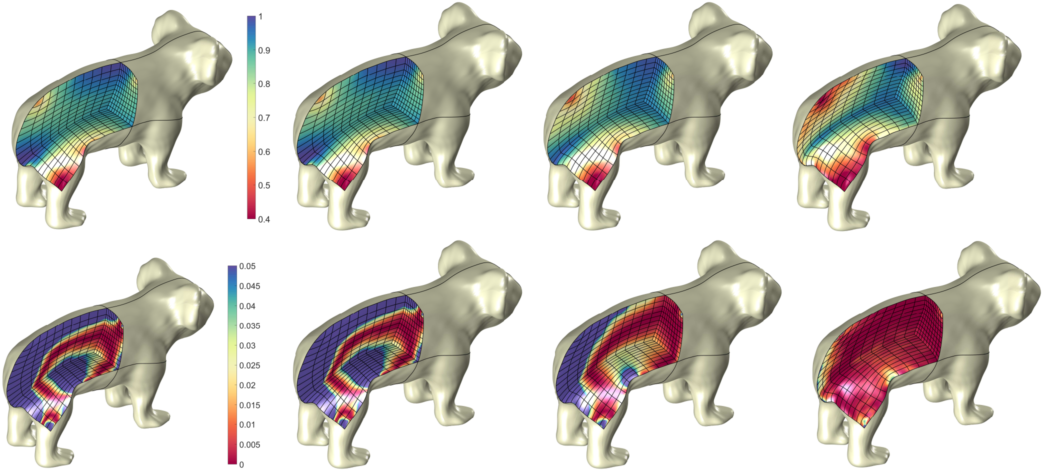
- Reduced integration strategy is adopted to accelerate the proposed method. However, *will this cause a loss of parameterization quality?*
- **NO!** The absolute differences of quality metrics are extremely close to 0.

Comparison: Reduced numerical integration vs. high precision integration

- However, it dramatically reduces the computational costs.

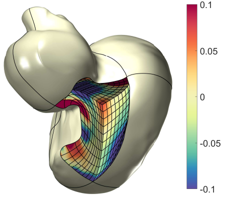


Influence of different proportions of parameters λ_1 and λ_2

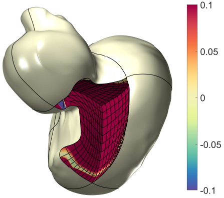


$\lambda_1 = 4,\ \lambda_2 = 0$ $\lambda_1 = 4,\ \lambda_2 = 1$ $\lambda_1 = 4,\ \lambda_2 = 10$ $\lambda_1 = 4,\ \lambda_2 = 100$

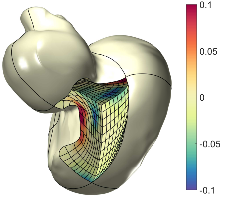
Comparison: Our method vs. current competitive approaches



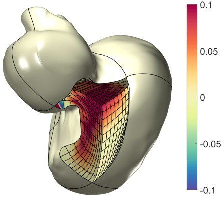
$m_{SJ}^{\text{Algo. 1}} - m_{SJ}^{\text{Pan}}$



$m_{unif.}^{\text{Pan}} - m_{unif.}^{\text{Algo. 1}}$



$m_{SJ}^{\text{Algo. 1}} - m_{SJ}^{\text{Liu}}$

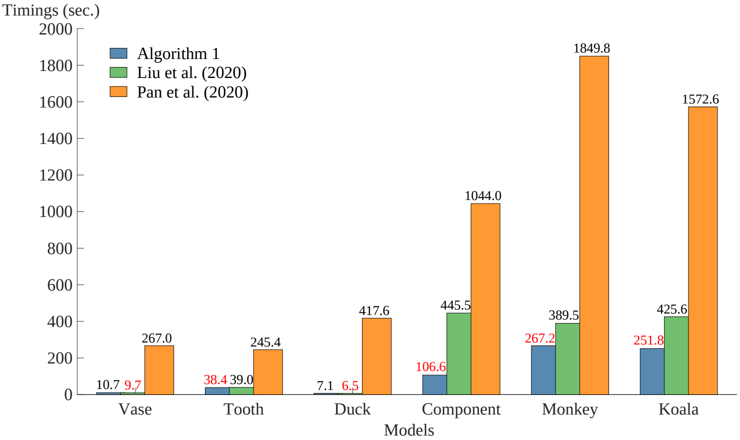


$m_{unif.}^{\text{Liu}} - m_{unif.}^{\text{Algo. 1}}$

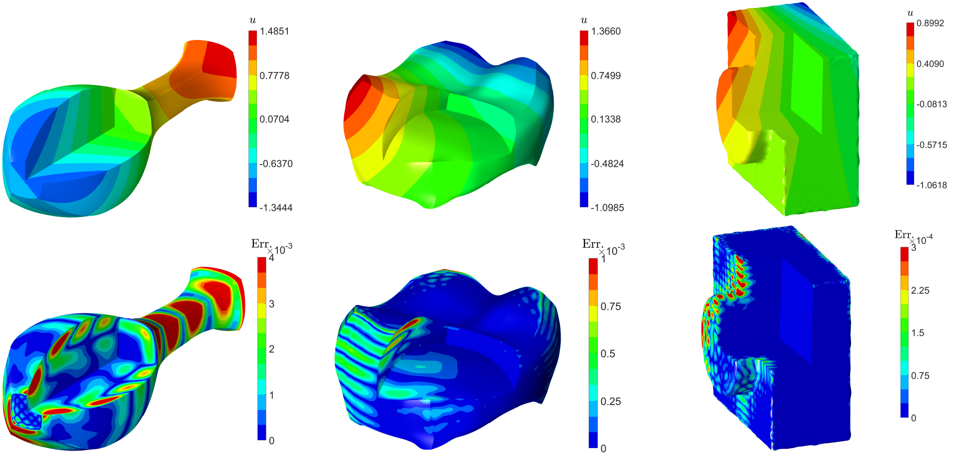
- We compare our method with two current competitors, i.e., Pan et al. 2020 and Liu et al. 2020.
- Positive values (red regions) indicate our method has lower angle distortion and/or lower volume distortion.

Efficiency: Our method vs. current competitive approaches

- Our method $\gg$ Pan et al. (2020);
- First three small-scale models, our method $\approx$ Liu et al. (2020);
- Last three large-scale models, our method $>$ Liu et al. (2020).



Application to IGA simulation: Poisson's problem



Vase, Err. $= 1.717 \times 10^{-3}$ Tooth, Err. $= 1.944 \times 10^{-4}$ Component, Err. $= 2.824 \times 10^{-5}$

Catalogue

Conclusions and future work

- Conclusions:
  - A **penalty function-based volumetric NURBS parameterization method** is proposed;
  - The volumes of computational domains are computed from the given B-Reps;
  - **Full analytical gradient** is deduced to enhance the efficiency and robustness;
  - **Reduced numerical integration strategy** is developed to enhance computational efficiency;
  - Numerical experiments demonstrate the **effectiveness and robustness** of our method.

- Future work:
  - **Role of the inner weights** on volumetric parameterization;
  - Extend our parameterization method to **high genus computational domains**;
  - In addition, we will **release all of the models and our reference implementation** in Geometry + Simulation Modules (**G+Smo**) library.

Thanks for your attention!

Q&A.

jiye@mail.dlut.edu.cn