IsoGlib: an Isogeometric Analysis library for the solution of high-order Partial Differential Equations on surfaces

A. Bartezzaghi, L. Dedè, A. Quarteroni

Chair of Modelling and Scientific Computing
Mathematics Institute of Computational Science and Engineering
Ecole Polytechnique Fédérale de Lausanne,
Switzerland

PDESoft 2014

Heidelberg, July 14–16, 2014

Outline

- Introduction
- A model problem: Laplace–Beltrami (second–order) PDE
- High–order PDEs
- Overview of IsoGLib
- Numerical results: convergence rates and phase–field models
- Conclusions

Introduction

- In several instances Partial Differential Equations (PDEs) are defined on lower dimensional manifolds with respect to the hosting space (e.g. surfaces in 3D and curves in 2D/3D).

- Applications are e.g. in fluid dynamics (thin films), structural problems (beams, shells), biology (biomembranes), image processing, and electromagnetism.

- The numerical approximation of the PDEs generally requires the generation of an approximated geometry compatible with the analysis, leading to geometrical error and inconsistency. Specifically, the approximation of the curvature may significantly affect the total numerical error.

Introduction

- Several geometries of practical interest are exactly represented by B–splines or NURBS; e.g. conic sections (sphere, cylindrical shell,...).
- IGA is an approximation method for PDEs based on the isoparametric concept for which the same basis functions used for the geometrical representation are then also used for the numerical approximations of the PDEs.

  [Hughes, Cottrell, Bazilevs, 2005]
- We consider the numerical approximation of PDEs on lower dimensional manifolds (surfaces) by means of Isogeometric Analysis (IGA). Specifically, we consider NURBS–based IGA.

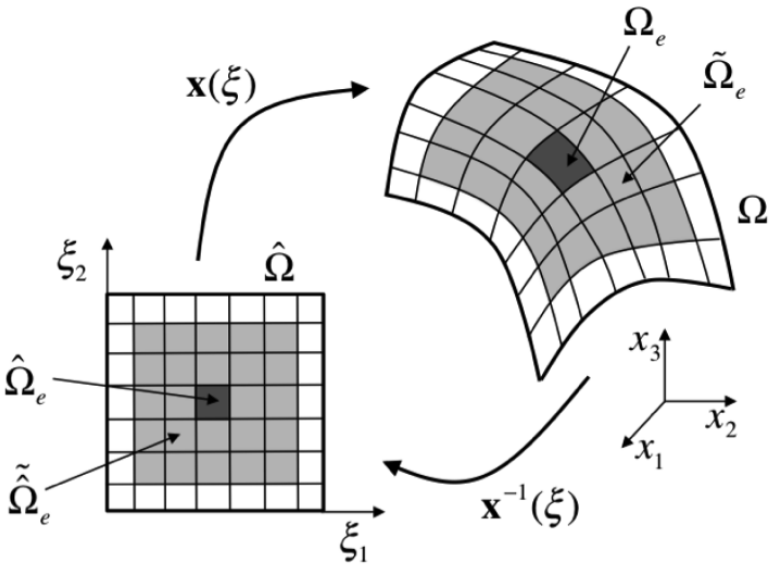
Surfaces represented by NURBS: geometrical mapping

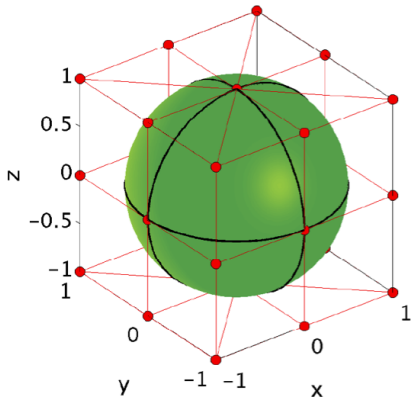
We consider a surface $\Omega \subset \mathbb{R}^d$ represented by a geometrical mapping from the parameter space $\mathbb{R}^\kappa$ into the physical space $\mathbb{R}^d$, with $d > \kappa \geq 1$, as:

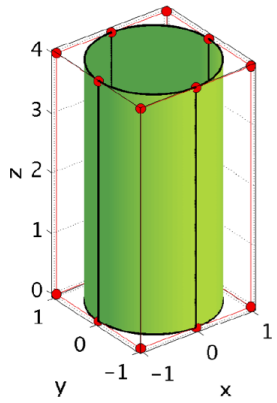
$$\boxed{\mathbf{x} \; : \; \widehat{\Omega} \to \mathbb{R}^d, \qquad \boldsymbol{\xi} \to \mathbf{x}(\boldsymbol{\xi}),}$$

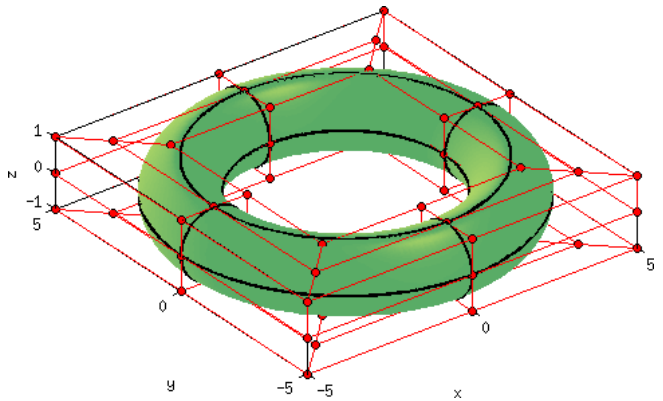
being $\widehat{\Omega} \subset \mathbb{R}^\kappa$ the parameter domain.

Examples of NURBS mappings:









Surfaces represented by NURBS: geometrical mapping

The geometrical mapping represented in terms of NURBS reads:

$$\boxed{\mathbf{x}(\boldsymbol{\xi}) = \sum_{i=1}^{n_{bf}} \widehat{R}_i(\boldsymbol{\xi})\, \mathbf{P}_i,}$$

where:

- $\widehat{R}_i(\boldsymbol{\xi})$ are the NURBS basis functions defined in the parameter domain $\widehat{\Omega}$;
- $\mathbf{P}_i \in \mathbb{R}^d$ are the control points in the physical space for $i = 1, \ldots, n_{bf}$.

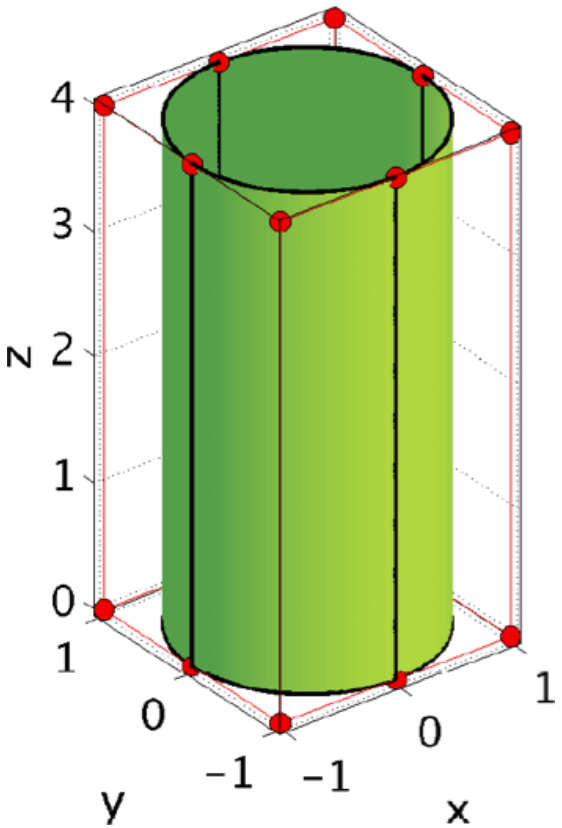
The NURBS basis functions $\widehat{R}_i(\boldsymbol{\xi})$ are defined from B–splines basis functions $\widehat{N}_i(\boldsymbol{\xi})$ and weights $w_i \in \mathbb{R}$ as:

$$\widehat{R}_i(\boldsymbol{\xi}) := \frac{w_i}{\displaystyle\sum_{i'=1}^{n_{bf}} w_{i'}\, \widehat{N}_{i'}(\boldsymbol{\xi})}\, \widehat{N}_i(\boldsymbol{\xi}) \qquad \text{for } i = 1, \ldots, n_{bf}.$$

Conic sections can be represented by NURBS basis functions with appropriate weights.

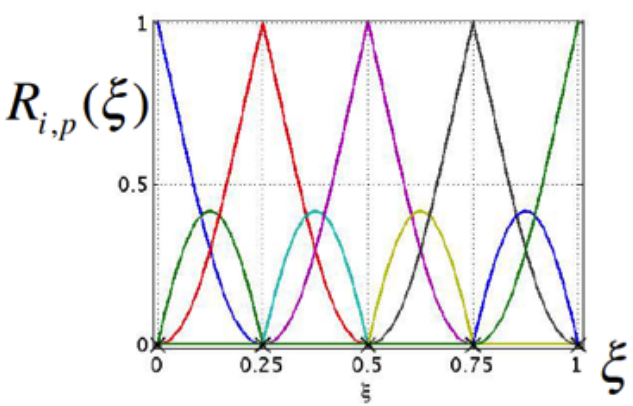
Surfaces represented by NURBS: conic sections

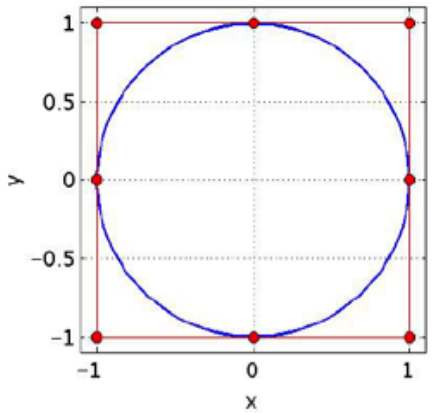
Example: cylinder surface represented by NURBS basis functions.



$$\Xi = \left\{ \{0\}^3, \left\{\frac{1}{4}\right\}^2, \left\{\frac{2}{4}\right\}^2, \left\{\frac{3}{4}\right\}^2, \{1\}^3 \right\} \qquad p = 2, \quad n = 9$$

$$w_i = 1, \quad i = 1,3,5,7,9 \qquad w_i = \frac{1}{\sqrt{2}}, \quad i = 2,4,6,8$$





Surfaces represented by NURBS: geometrical mapping

Properties of the geometrical mapping $\mathbf{x} : \widehat{\Omega} \to \mathbb{R}^d$:

$$\widehat{\mathrm{F}} : \widehat{\Omega} \to \mathbb{R}^{d\times\kappa}, \qquad \boldsymbol{\xi} \to \widehat{\mathrm{F}}(\boldsymbol{\xi}), \quad \widehat{\mathrm{F}}_{i,\alpha}(\boldsymbol{\xi}) := \frac{\partial \mathrm{x}_i}{\partial \xi_\alpha}(\boldsymbol{\xi}),$$

$$\widehat{\mathrm{G}} : \widehat{\Omega} \to \mathbb{R}^{\kappa\times\kappa}, \qquad \boldsymbol{\xi} \to \widehat{\mathrm{G}}(\boldsymbol{\xi}), \qquad \widehat{\mathrm{G}}(\boldsymbol{\xi}) := \left(\widehat{\mathrm{F}}(\boldsymbol{\xi})\right)^T \widehat{\mathrm{F}}(\boldsymbol{\xi}),$$

$$\widehat{g} : \widehat{\Omega} \to \mathbb{R}, \qquad \boldsymbol{\xi} \to \widehat{g}(\boldsymbol{\xi}), \qquad \widehat{g}(\boldsymbol{\xi}) := \sqrt{\det\left(\widehat{\mathrm{G}}(\boldsymbol{\xi})\right)}.$$

We assume that the geometrical mapping $\mathbf{x} : \widehat{\Omega} \to \mathbb{R}^d$ is "sufficiently" smooth, e.g. $C^1(\widehat{\Omega})$, and invertible $a.e.$ in $\widehat{\Omega}$ ($\widehat{g}(\boldsymbol{\xi}) > 0$ $a.e.$ in $\widehat{\Omega}$).
Therefore:

$$\boxed{\phi(\mathbf{x}) = \widehat{\phi}(\boldsymbol{\xi}) \circ \mathbf{x}^{-1}(\boldsymbol{\xi}),}$$

where $\widehat{\phi}(\boldsymbol{\xi}) := \phi\left(\mathbf{x}(\boldsymbol{\xi})\right)$.

Surfaces represented by NURBS: functions and operators

By using the geometrical mapping, the gradient on the manifold can be written as:

$$\boxed{\nabla_\Omega \phi(\mathbf{x}) = \left[\widehat{\mathrm{F}}(\boldsymbol{\xi})\, \widehat{\mathrm{G}}^{-1}(\boldsymbol{\xi}) \widehat{\nabla}\widehat{\phi}(\boldsymbol{\xi})\right] \circ \mathbf{x}^{-1}(\boldsymbol{\xi}),}$$

where $\widehat{\nabla}\widehat{\phi} \ : \ \widehat{\Omega} \to \mathbb{R}^\kappa$ is the gradient operator in the parameter space.

We introduce the Laplace–Beltrami operator $\Delta_\Omega \phi(\mathbf{x}) := \nabla_\Omega \cdot (\nabla_\Omega \phi(\mathbf{x}))$ associated to the manifold Ω for a function $\phi \in C^2(\Omega)$ and write it using the geometrical mapping as:

$$\boxed{\Delta_\Omega \phi(\mathbf{x}) = \left[\frac{1}{\widehat{g}(\boldsymbol{\xi})}\, \widehat{\nabla} \cdot \left(\widehat{g}(\boldsymbol{\xi})\, \widehat{\mathrm{G}}^{-1}(\boldsymbol{\xi})\, \widehat{\nabla}\widehat{\phi}(\boldsymbol{\xi})\right)\right] \circ \mathbf{x}^{-1}(\boldsymbol{\xi}).}$$

Finally, the differential $d\mathbf{x}$ $(d\Omega)$ reads $d\mathbf{x} = \widehat{g}(\boldsymbol{\xi})\, d\boldsymbol{\xi}$ $(d\Omega = \widehat{g}(\boldsymbol{\xi})\, d\widehat{\Omega})$.

A Model Problem: Laplace–Beltrami

We consider the second order Laplace–Beltrami problem defined on the surface Ω:

$$\boxed{\begin{aligned} -\mu\Delta_\Omega u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \Gamma_D, \\ \mu\nabla_\Omega u \cdot \mathbf{n}_\Gamma &= 0 && \text{on } \Gamma_N, \end{aligned}}$$

where $\mu > 0$, $f \in L^2(\Omega)$, and $\Gamma \equiv \partial\Omega$ is the boundary of Ω (if $\text{meas}(\Gamma) > 0$) with $\overline{\Gamma_D \cup \Gamma_N} \equiv \Gamma$ and $\overset{\circ}{\Gamma}_D \cap \overset{\circ}{\Gamma}_N = \emptyset$, and $\mathbf{n}_\Gamma$ its unit normal vector.

The Laplace–Beltrami problem in weak form reads:

$$\boxed{\text{find } u \in \mathcal{V} \ : \ a(v,u) = q(v) \qquad \forall v \in \mathcal{V},}$$

where: $a(v,w) := \displaystyle\int_\Omega \mu\, \nabla_\Omega v \cdot \nabla_\Omega w \, d\Omega$, $q(v) := \displaystyle\int_\Omega v\, f \, d\Omega$, and $\mathcal{V} := \left\{ v \in H^1(\Omega) \ : \ v|_{\Gamma_D} = 0 \right\}$.

A Model Problem: Laplace–Beltrami

We recast the problem in the parameter domain $\widehat{\Omega}$ with the geometrical mapping ("pull–back") and solve it by means of Galerkin NURBS–based IGA:

$$\boxed{\text{find } \widehat{u}_h \in \widehat{\mathcal{V}}_h \ : \ \widehat{a}(\widehat{v}_h, \widehat{u}_h) = \widehat{q}(\widehat{v}_h) \qquad \forall \widehat{v}_h \in \widehat{\mathcal{V}}_h,}$$

where $\widehat{\mathcal{V}}_h := \widehat{\mathcal{V}} \cap \widehat{\mathcal{N}}_h$, with the NURBS space $\widehat{\mathcal{N}}_h := \text{span}\left\{\widehat{R}_i(\boldsymbol{\xi})\right\}_{i=1}^{n_{bf}}$, $\widehat{\mathcal{V}} := \left\{\widehat{v} \in H^1(\widehat{\Omega}) \ : \ \widehat{v}|_{\widehat{\Gamma}_D} = 0\right\}$ and

$$\widehat{a}(\widehat{v}, \widehat{w}) := \int_{\widehat{\Omega}} \mu \, \widehat{\nabla}\widehat{v} \cdot \left(\widehat{\mathrm{G}}^{-1} \, \widehat{\nabla}\widehat{w}\right) \, \widehat{g} \, d\widehat{\Omega},$$

$$\widehat{q}(\widehat{v}) := \int_{\widehat{\Omega}} \widehat{v} \, \widehat{f} \, \widehat{g} \, d\widehat{\Omega},$$

High–order Laplace–Beltrami problems

We consider the biharmonic problem defined on the surface Ω:

$$\begin{aligned} \mu\Delta_\Omega^2 u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega, \\ \mu\nabla_\Omega u \cdot \mathbf{n}_\Gamma &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where $\Delta_\Omega^2 := \Delta_\Omega \Delta_\Omega$ is the fourth–order bilaplacian operator and the triharmonic problem defined on the surface Ω:

$$\begin{aligned} -\mu\Delta_\Omega^3 u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega, \\ \mu\nabla_\Omega u \cdot \mathbf{n}_\Gamma &= 0 && \text{on } \partial\Omega, \\ \Delta_\Omega u &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where $\Delta_\Omega^3 := \Delta_\Omega \Delta_\Omega \Delta_\Omega$ is the sixth–order trilaplacian operator.

A priori Error Estimation: uniform h–refinement

Let us consider a well-posed scalar elliptic PDE of order $2m$, $m \geq 1$, in a domain $\Omega \in \mathbb{R}^d$ described by NURBS, in weak form:

$$\text{find } u \in V \quad : \quad a(u,v) = F(v) \quad \forall v \in V,$$

where $V \subseteq H^m(\Omega)$ is such that its functions satisfy the homogeneous counterpart of the essential boundary conditions.

Theorem

For an elliptic PDE of order $2m$, $m \geq 1$, let $\sigma \in \mathbb{N}$ s.t. $0 \leq \sigma \leq m$ and $F \in H^{-\sigma}(\Omega)$. Let $u \in H^r(\Omega)$, $r \geq m$, be the exact solution and u_h the approximate solution obtained with NURBS-based IGA. Then:

$$\|u - u_h\|_{H^\sigma(\Omega)} \leq C_{shape} h^\beta \|u\|_{H^r(\Omega)},$$

where $\beta := \min\{\delta - \sigma, 2(\delta - m)\}$, with $\delta := \min\{r, p+1\}$.

[Tagliabue, Dedè, Quarteroni, 2014]

IsoGlib: overview

- Framework for solving problems by means of NURBS-based IGA in the framework of the Galerkin method.
  - 2D and 3D surface geometries, described by means of B-Splines and NURBS.
  - High–order surface differential operators.
  - Different time discretization methods with time–step adaptivity schemes.
  - Local transformation of basis functions, e.g. to handle periodicity and closed surfaces.
  - Support for geometric PDEs (in progress).
- Goal: maximum flexibility (applications include ADR problems, elasticity, Navier-Stokes, Cahn–Hilliard, electrophysiology simulations, etc...)

IsoGlib: overview

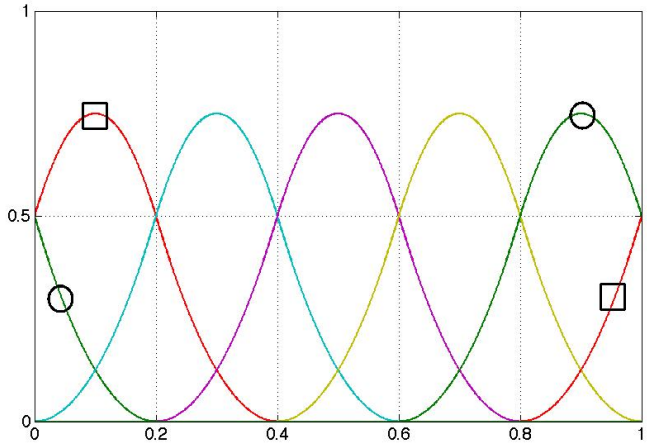
- Mesh generation and manipulation tools available from Matlab; libraries and applications for solution of the problems written in C++.
- Easy interface to simplify the integration of the library within other applications and frameworks (LifeV for example).
- Relying on external linear algebra packages to solve the systems.
  - Interface independent from the chosen package.
  - However, the only back-end currently supported is Trilinos.
- Different levels of "caching" of internal quantities to speed-up certain computations.

IsoGlib: periodicity

When coping with closed surfaces or conic sections and global C^{p-1}-continuity is desired, periodicity of the basis functions must be enforced.

IsoGlib automatically handles periodicity conditions:

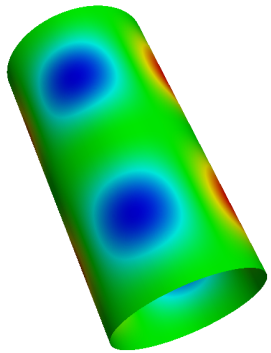
- constraints among the dofs enforced by master–slave relations across borders and internal "C^{-1} lines"
- local linear transformation of basis functions (transformations automatically calculated based on local mesh properties)



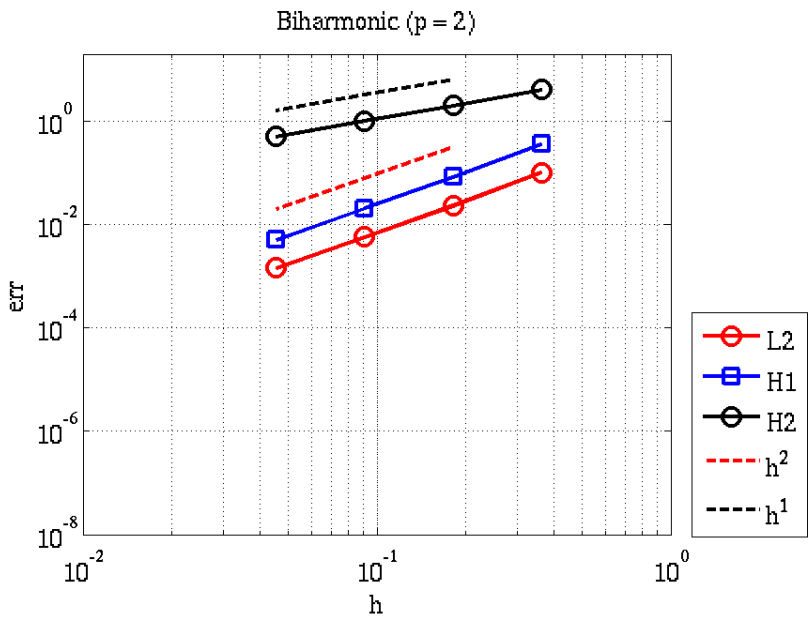
periodic basis functions, $p = 2$, C^1–continuous

Numerical Results: 4^{th} order Laplace–Beltrami Problem

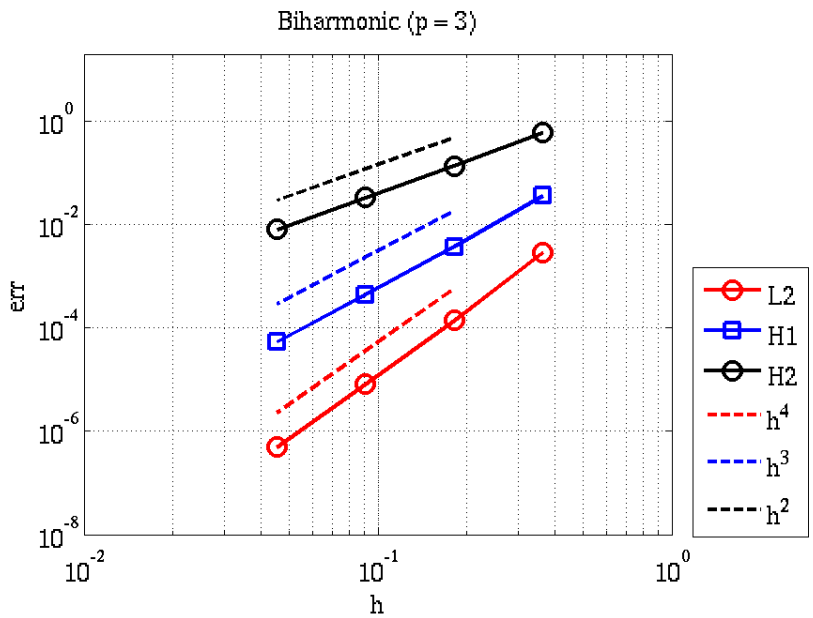
Numerical approximation of the biharmonic PDE on a (closed) cylinder:



Errors in norms H^2, H^1, and L^2 vs h:



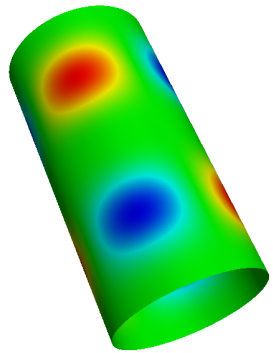
$p = 2$, C^1–continuous



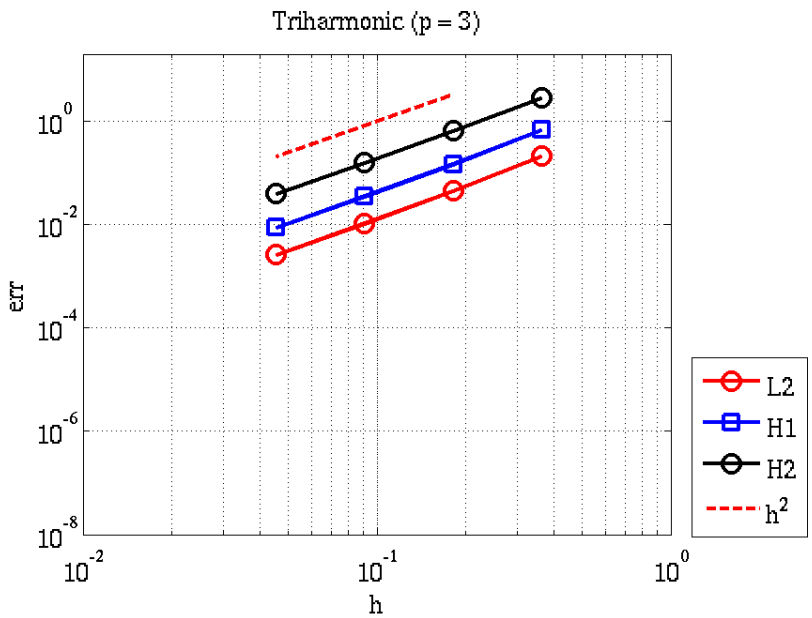
$p = 3$, C^2–continuous

Numerical Results: 6^{th} order Laplace–Beltrami Problem

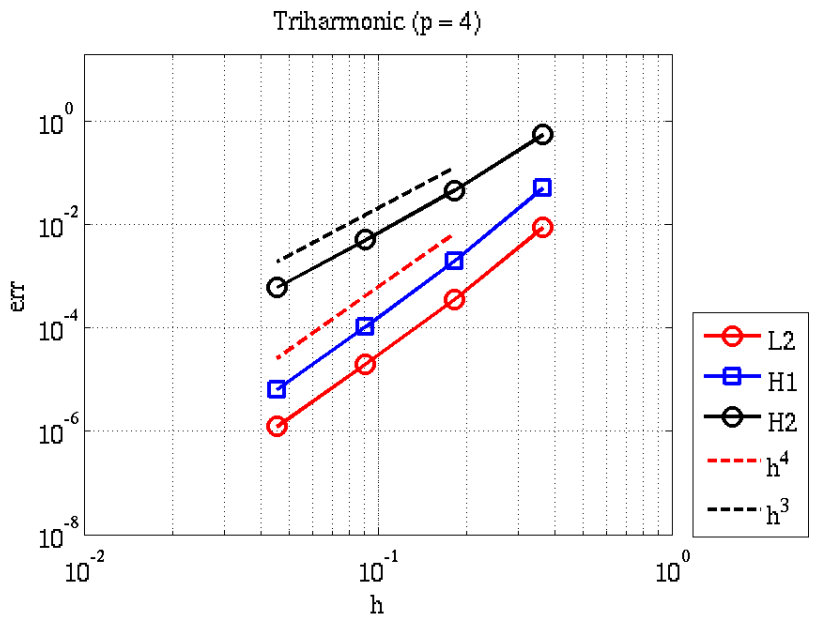
Numerical approximation of the triharmonic PDE on a (closed) cylinder:



Errors in norms H^2, H^1, and L^2 vs h:



$p = 3$, C^2–continuous



$p = 4$, C^2–continuous

Cahn–Hilliard Equation

Let Ω be a surface in $\mathbb{R}^3$. A binary mixture is contained in Ω and $u : \Omega \times [0, T) \to \mathbb{R}$ denotes the concentration of one of its components. The Cahn–Hilliard equation on a closed surface reads:

$$\begin{aligned} &\frac{\partial u}{\partial t}(t) = \nabla_\Omega \cdot \left(m_c \, \nabla_\Omega \left(\frac{d\widetilde{\Psi}}{du}(u(t)) \right) \right) && \text{in } \Omega \times (0, T), \\ &u(0) = u_{in} && \text{in } \Omega \times \{0\}. \end{aligned}$$

The two phases evolve in time, yielding the minimization of the free energy functional $\widetilde{\Psi}(t)$ and conservation of mass $M(t) = \int_\Omega u(t)\, d\Omega = \int_\Omega u_{in}\, d\Omega$:

$$\widetilde{\Psi}(t) = \int_\Omega \left(\Psi_c(u(t)) + \Psi_s(u(t)) \right) d\Omega, \qquad \frac{d\widetilde{\Psi}}{du}(u) = \Psi_{c,u}(u) - \lambda \, \Delta_\Omega u,$$

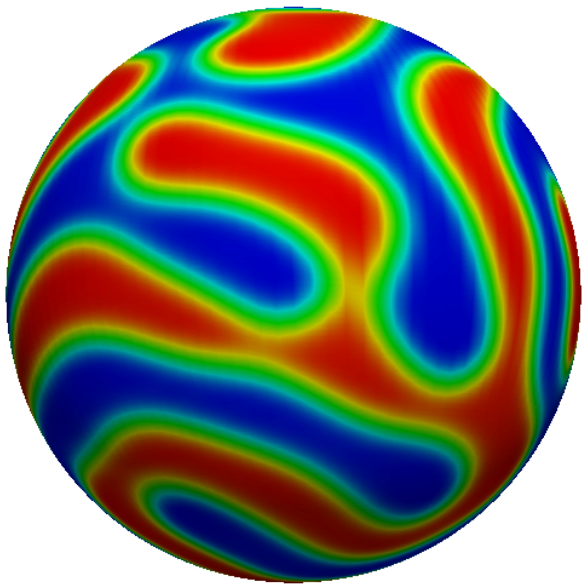
Chemical energy	Surface energy	Mobility
$\Psi_c(u) = u^2(1-u)^2$	$\Psi_s(u) = \frac{1}{2}\lambda \lvert\nabla_\Omega u(t)\rvert^2$	$m_c = m_0 u(1-u)$

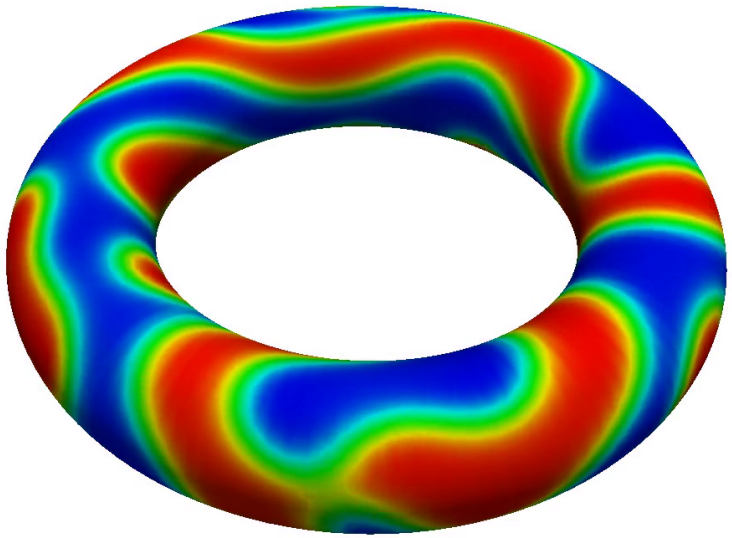
and we have that $d\widetilde{\Psi}(t)/dt \leq 0$, $dM(t)/dt = 0$.

[Liu, Dedè, Evans, Borden, Hughes, 2013]

Numerical Results: Cahn–Hilliard Equation

Phase transition on a sphere and a thorus for volume fraction $v_f = M(0)/|\Omega| = 0.5$.





Solved with NURBS–based IGA using basis functions of degree $p = 2$ and parametrically C^1–continuous.

Phase–field Crystal Equation

Nonlinear time-dependent 6^{th}–order PDE describing the crystal growth inside a two-phase liquid system at interatomic length scales and diffusive time scale, under mass conservation. The equation for the atomistic density u reads:

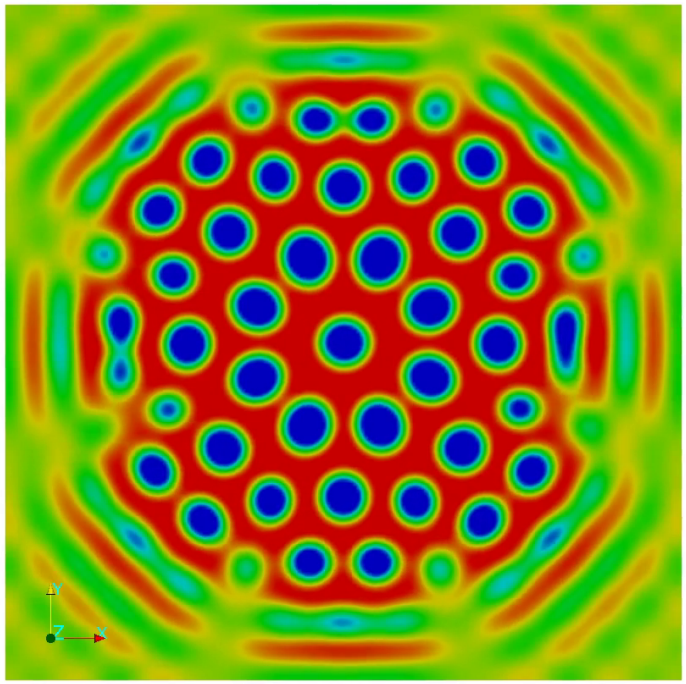
$$\begin{aligned} &\frac{\partial u}{\partial t} = \Delta_\Omega \left(\phi\left(u\right) + Dk^4 u + 2Dk^2 \Delta_\Omega u + D\Delta_\Omega^2 u \right) && \text{in } \Omega \times (0,T), \\ &u(0) = u_{in} && \text{in } \Omega \times \{0\}. \end{aligned}$$

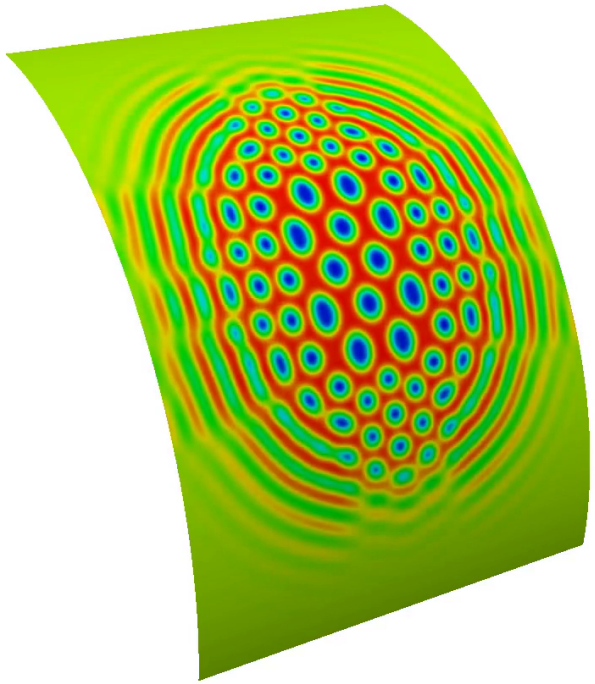
where $\phi\left(u\right) = -\frac{\epsilon}{2}u^2 - \frac{g}{3}u^3 + \frac{1}{4}u^4$, k and D are positive numbers, ϵ and g are positive physical constants.

[Gómez, Nogueira, 2012]

Numerical Results: Phase–field Crystal Equation

Numerical results on the plane and on a quarter of a cylindrical shell:





Solved with NURBS–based IGA using basis functions of degree $p = 3$ and parametrically C^2–continuous.

Conclusions

- NURBS–based IGA provides a "natural" framework for the spatial approximation of PDEs on surfaces.
- High–order PDEs (order $2k$, with $k > 2$) on surfaces can be solved with NURBS–based IGA by using smooth (C^{k-1}–continuous) and periodic basis functions.
- The current implementation in IsoGlib is a starting base:
  - High–order surface operators are needed to solve some problems of interest, like geometric PDEs with high–order flows.