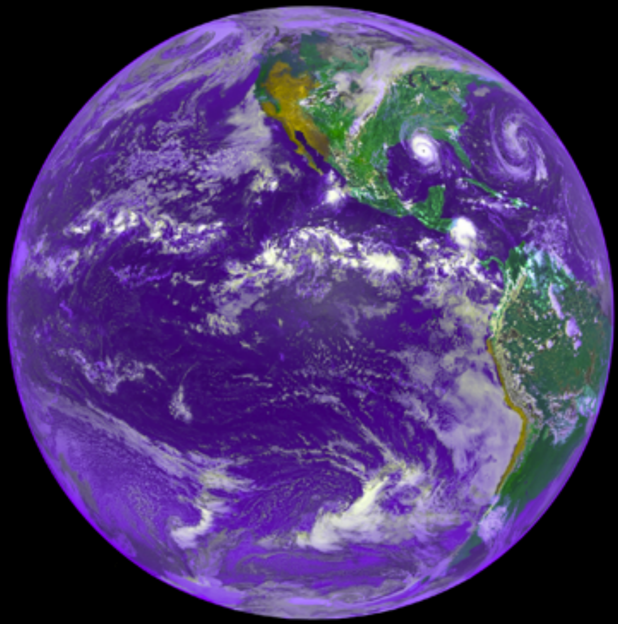
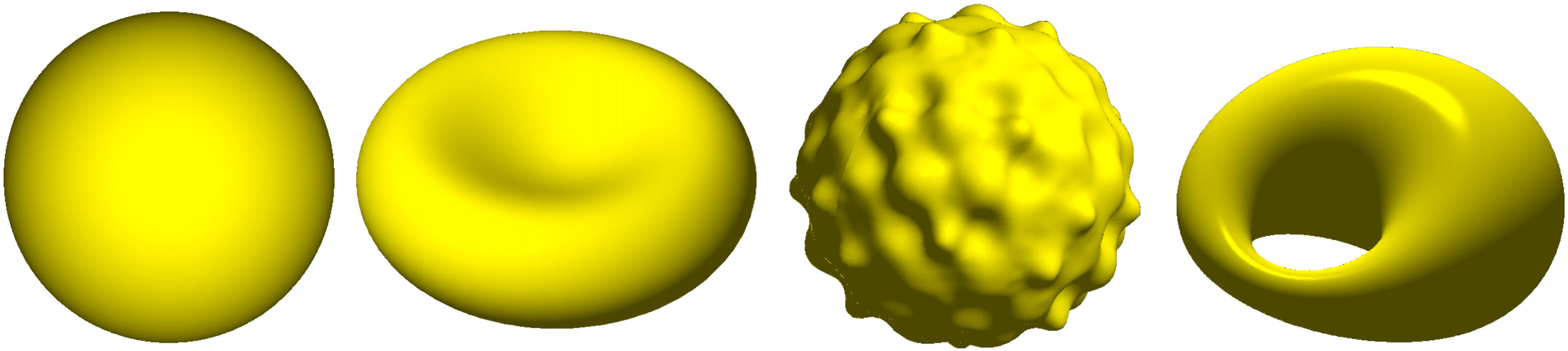
2013 Dolomites Research Week on Approximation

Lecture 7: Kernel methods for more general surfaces



Grady B. Wright
Boise State University

Overview



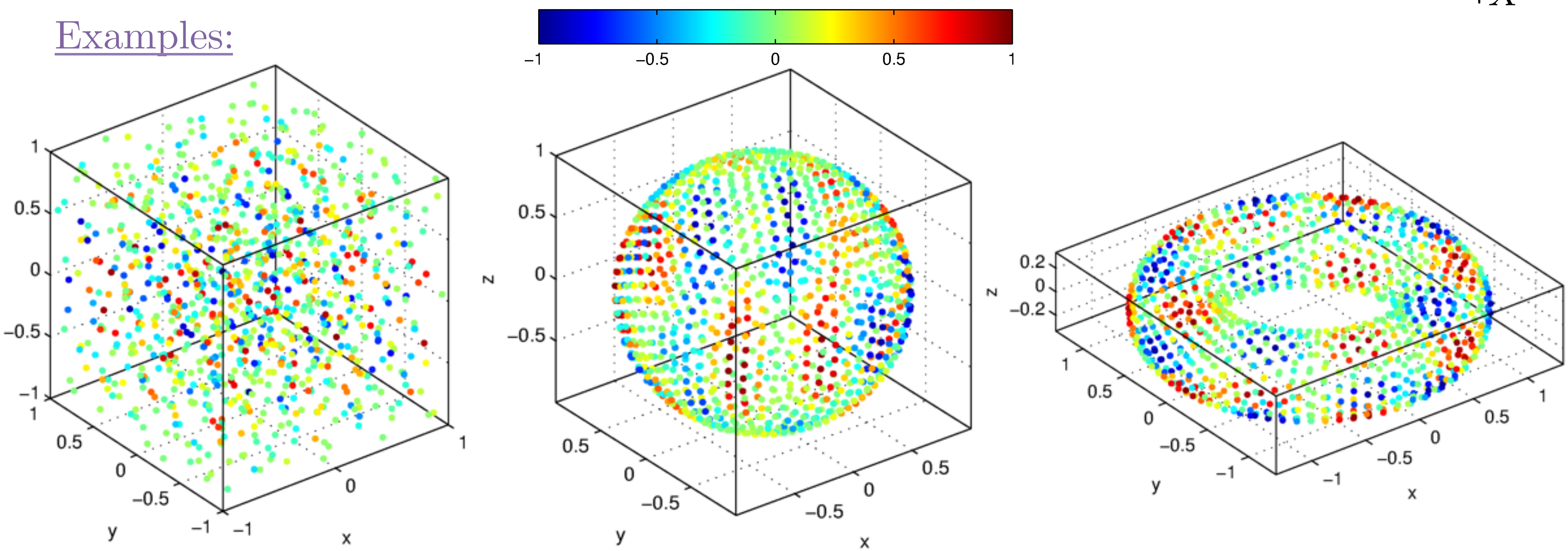
- Background
- Kernel approximation on surfaces
- Applications to numerically solving PDEs on surfaces

Interpolation with kernels

- Let $\Omega \subset \mathbb{R}^d$ and $X = \{\mathbf{x}_j\}_{j=1}^N$ a set of nodes on Ω.

- Consider a continuous target function $f : \Omega \to \mathbb{R}$ sampled at X: $f\big|_X$.

Examples:



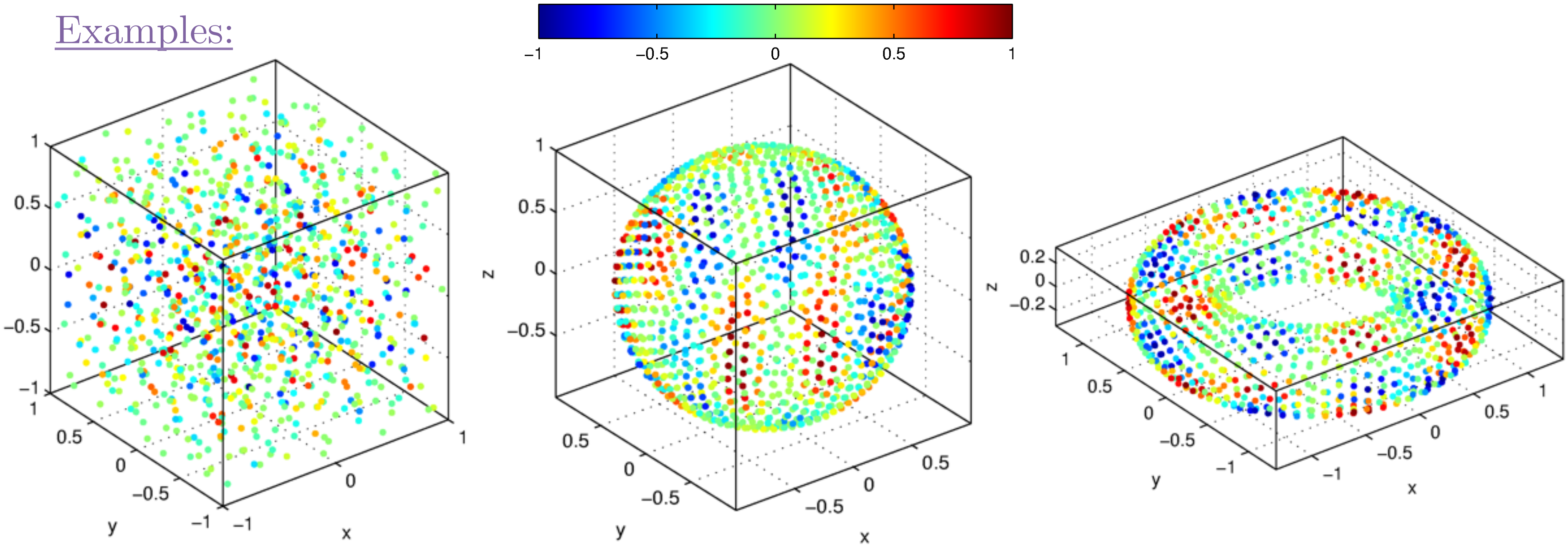
$\Omega = [-1, 1]$ $\qquad$ $\Omega = \mathbb{S}^2$ $\qquad$ $\Omega = \mathbb{T}^2$

- Kernel interpolant to $f\big|_X$: $\qquad I_X f = \sum_{j=1}^{N} c_j \phi(\cdot, \mathbf{x}_j)$

  where $\phi : \Omega \times \Omega \to \mathbb{R}$ and c_j come from requiring $I_X f\big|_X = f\big|_X$

Interpolation with kernels

Examples:



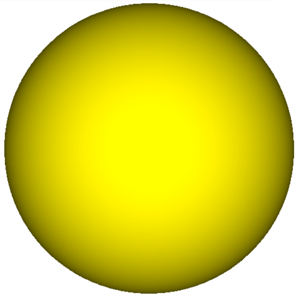
$\Omega = [-1, 1]$ $\qquad\qquad$ $\Omega = \mathbb{S}^2$ $\qquad\qquad$ $\Omega = \mathbb{T}^2$

- Kernel interpolant to $f\big|_X$: $\qquad I_X f = \sum_{j=1}^{N} c_j \phi(\cdot, \mathbf{x}_j)$

- We call ϕ a positive definite kernel if $A = \{\phi(\mathbf{x}_i, \mathbf{x}_j)\}$ is positive definite for any $X = \{\mathbf{x}_j\}_{j=1}^{N} \subset \Omega$.

- In this case c_j are uniquely determined.

Kernel interpolation on surfaces



- Kernels on the sphere:
  - Schoenberg (1942)
  - See Lecture 1 slides for more...

- Kernels on specific manifolds ($SO(3)$, motion group, projective spaces):
  - Erb, Filber, Hangelbroek, Schmid, zu Castel,...

- Kernels on arbitrary Riemannian manifolds:
  - Narcowich (1995)
  - Dyn, Hangelbroek, Levesley, Ragozin, Schaback, Ward, Wendland.

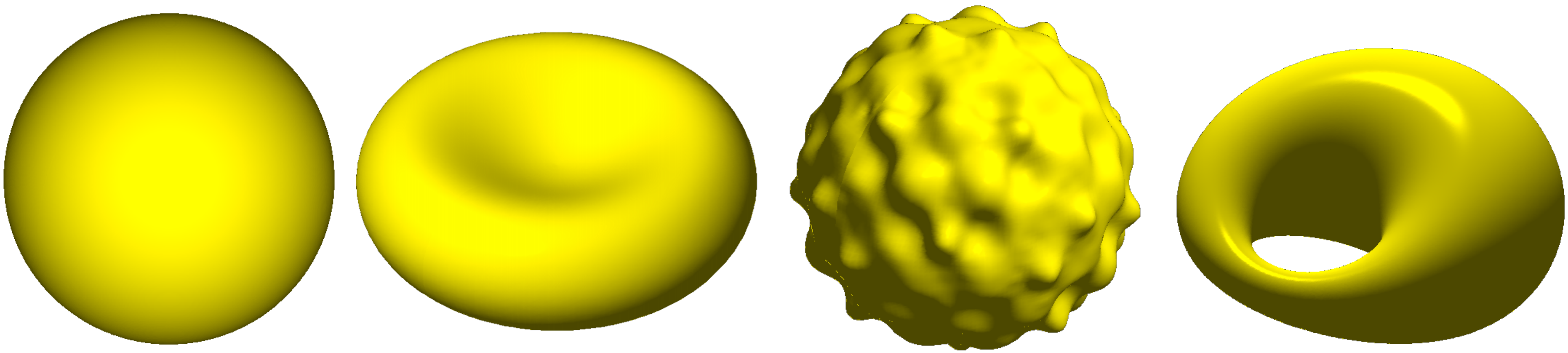
- In these studies the kernels used are highly dependent on the manifold.
  - Inherent benefits to this.
  - However, for arbitrary manifolds it is difficult (or impossible) to compute these kernel.

Kernel interpolation on surfaces

- Types of surfaces: $\mathbb{M}$

  Compact, smooth embedded submanifolds of $\mathbb{R}^d$ without a boundary.

- Examples:



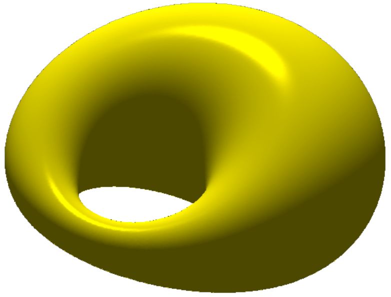
- Applications:
  - geophysics
  - atmospheric sciences
  - biology
  - chemistry
  - computer graphics

Kernel interpolation on surfaces

- One approach for kernels on general surfaces:
  Use a restricted positive definite kernel from $\mathbb{R}^d$

- Let ϕ be a positive definite kernel on $\mathbb{R}^d$, $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\Big|_{\mathbb{M},\mathbb{M}}$:

$$I_X f = \sum_{j=1}^{N} c_j \psi(\cdot, \mathbf{x}_j)$$



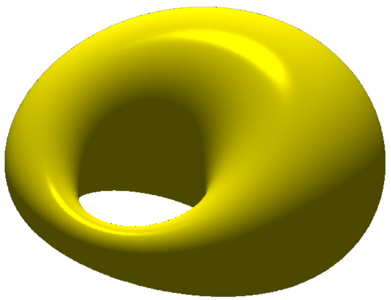
- Such ϕ are easy to come, e.g.
  - Let ϕ be a positive definite radial kernel (RBFs):
    $\phi(\mathbf{x},\mathbf{y}) = \phi(\|\mathbf{x}-\mathbf{y}\|_2) = \phi(r)$

Kernel interpolation on surfaces

- One approach for kernels on general surfaces:
  Use a restricted positive definite kernel from $\mathbb{R}^d$

- Let ϕ be a positive definite kernel on $\mathbb{R}^d$, $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\Big|_{\mathbb{M},\mathbb{M}}$:

$$I_X f = \sum_{j=1}^{N} c_j \psi(\cdot, \mathbf{x}_j)$$



- Such ϕ are easy to come, e.g.
  - Let ϕ be a positive definite radial kernel (RBFs):
    $\phi(\mathbf{x},\mathbf{y}) = \phi(\|\mathbf{x}-\mathbf{y}\|_2) = \phi(r)$

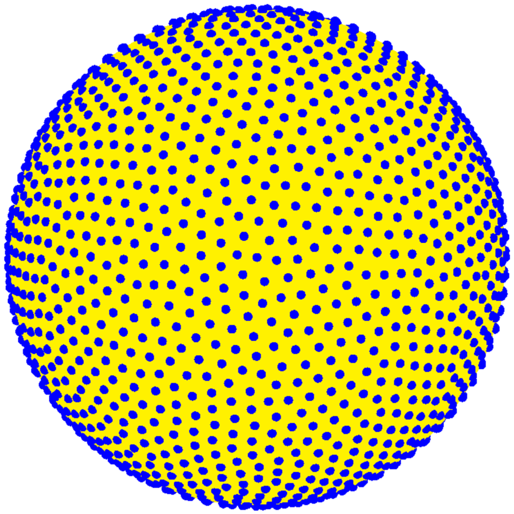
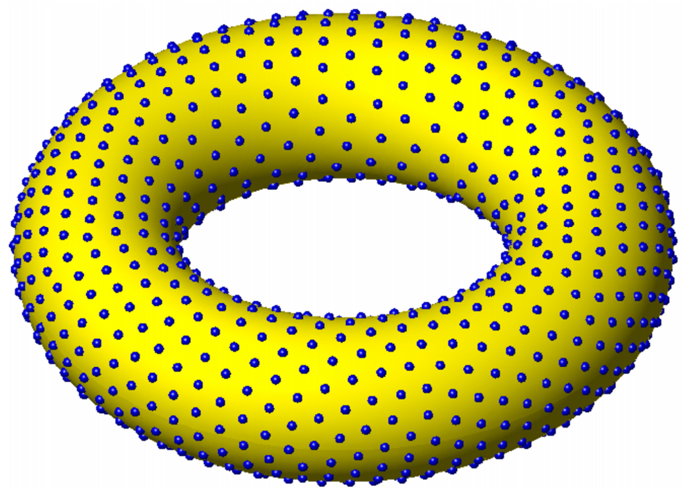
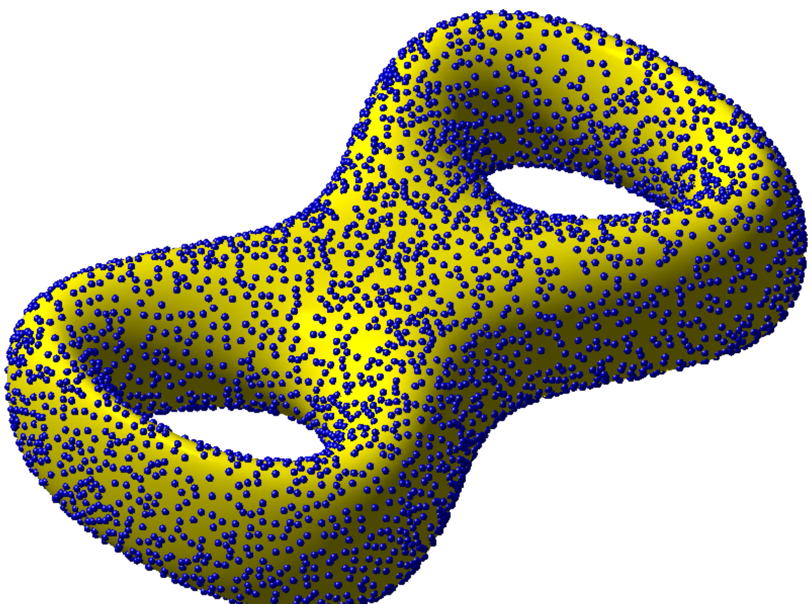
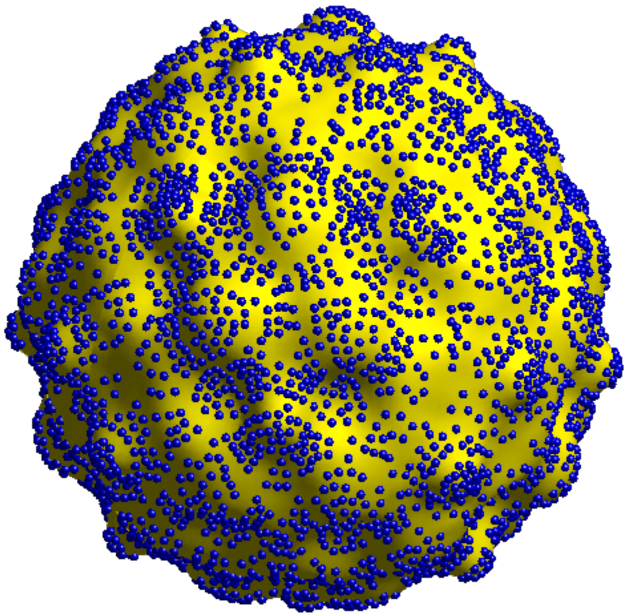
- For $\mathbb{M} = \mathbb{S}^2$, this approach has been thoroughly studied.

- Surprisingly, for general surfaces, virtually nothing had been done:
  - Powell (2001) DAMTP Technical Report.
  - Fasshauer (2007), p. 83

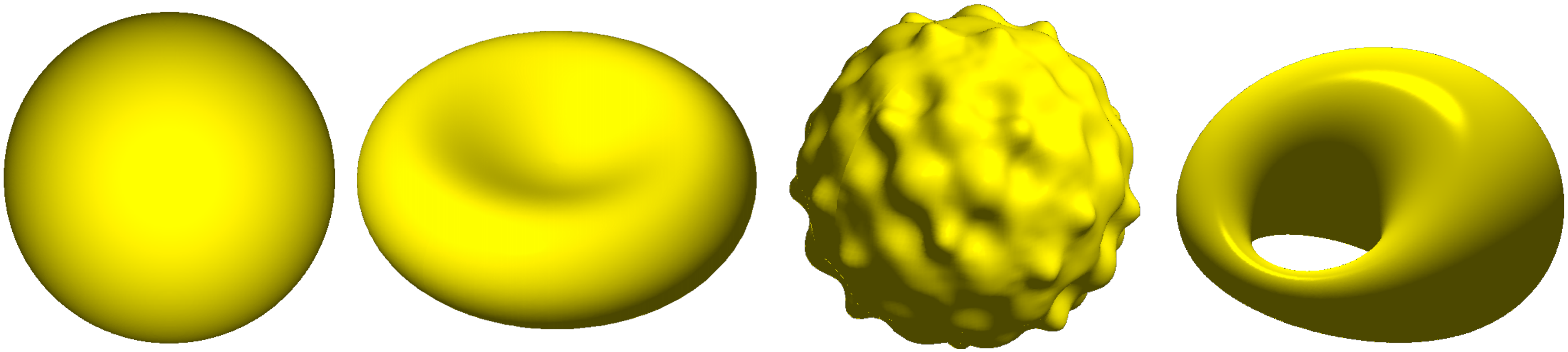
Nodes on surfaces

- Kernel methods do not require a mesh, just a set of nodes.

- Some ideas:

Motivation: Reaction diffusion equations on surfaces



- Prototypical model: 2 interacting species

$$\frac{\partial u}{\partial t} = \delta_u \Delta_{\mathbb{M}} u + f_u(t,u,v)$$
$$\frac{\partial v}{\partial t} = \delta_v \Delta_{\mathbb{M}} v + f_v(t,u,v)$$

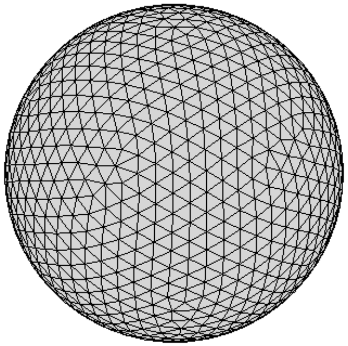
$\Delta_{\mathbb{M}}$ is the Laplace-Beltrami operator for the surface

- Applications
  - Biology: diffusive transport on a membrane, pattern formation on animal coats, and tumor growth.
  - Chemistry: waves in excitable media (cardiac arrhythmia, electrical signals in the brain).
  - Computer graphics: texture mapping and synthesis and image processing.

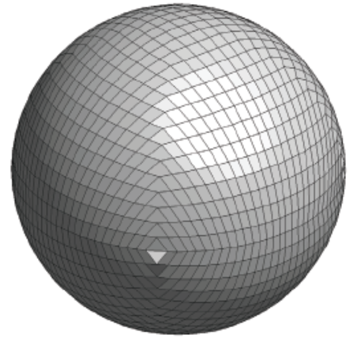
Current methods and kernel-based approach

- Current numerical method can be split into 2 categories:

1. Surface-based: approximate the PDE *on the surface* using *intrinsic* coordinates.

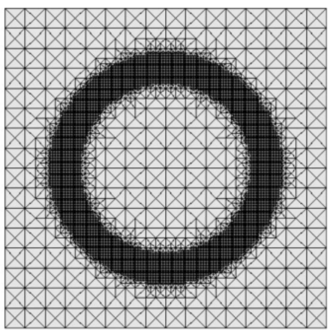


<u>Triangulated Mesh</u>
Dziuk (1988)
Stam (2003)
Xu (2004)
Dziuk & Elliot (2007)

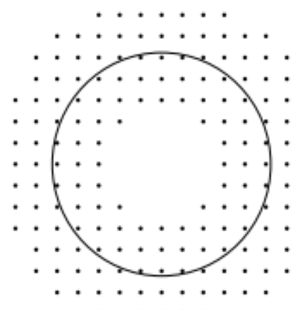


<u>Logically rectangular grid</u>
Calhoun and Helzel (2009)

2. Embedded: approximate the PDE in the *embedding space*, restrict solution to surface.

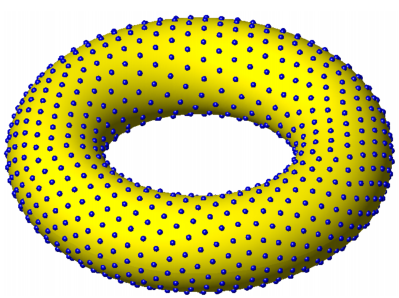


<u>Level Set</u>
Bertalmio *et al.* (2001)
Schwartz *et al.* (2005)
Greer (2006)
Sbalzarini *et al.* (2006)
Dziuk & Elliot (2010)



<u>Closest point:</u>
Ruuth & Merriman (2008)
MacDonald & Ruuth (2008)
MacDonald & Ruuth (2009)

- Kernel-based method: Fuselier & W (2013)



- Similarity to 1: approximate the PDE *on the surface*.
- Similarity to 2: use *extrinsic* coordinates.
- Differences: method is mesh-free;
  computations done in same dimension as the surface.

Interpolation with restricted PD kernels

- Let ϕ be a positive definite kernel on $\mathbb{R}^d$, $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$, and $k = \dim(\mathbb{M})$.

- Kernel interpolant: $I_X f = \sum_{j=1}^{N} c_j \psi(\cdot, \mathbf{x}_j)$, where $X = \{\mathbf{x}_j\}_{j=1}^{N} \subset \mathbb{M}$

- Approximation classes can be found from the native space of ψ: $\mathcal{N}_\psi$

  - $F_\psi = \left\{ f = \sum_j c_j \psi(\cdot, \mathbf{x}_j) \;\middle|\; c_j \in \mathbb{R},\ \mathbf{x}_j \in \mathbb{M} \right\}$
  - $\|f\|^2_{\mathcal{N}_\psi} = \sum_j \sum_k c_j c_k \psi(\mathbf{x}_j, \mathbf{x}_k)$, $f \in F_\psi$
  - $\mathcal{N}_\psi = \overline{F_\psi}$

- What is $\mathcal{N}_\psi$?

Interpolation with restricted PD kernels

- Let ϕ be a positive definite kernel on $\mathbb{R}^d$, $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$, and $k = \dim(\mathbb{M})$.

- Kernel interpolant: $I_X f = \displaystyle\sum_{j=1}^{N} c_j \psi(\cdot, \mathbf{x}_j)$, where $X = \{\mathbf{x}_j\}_{j=1}^{N} \subset \mathbb{M}$

- What is $\mathcal{N}_\psi$?

- Suppose the Fourier transform of ϕ on $\mathbb{R}^d$ satisfies $\hat{\phi}(\boldsymbol{\xi}) \sim (1+\|\boldsymbol{\xi}\|_2^2)^{-\tau}$ then $\mathcal{N}_\phi = H^\tau(\mathbb{R}^d)$

- Theorem (Fuselier,W 2012): If ϕ satisfies $\hat{\phi}(\boldsymbol{\xi}) \sim (1+\|\boldsymbol{\xi}\|_2^2)^{-\tau}$ with $\tau > d/2$, then $\mathcal{N}_\psi = H^{\tau-(d-k)/2}(\mathbb{M})$ with equivalent norms.

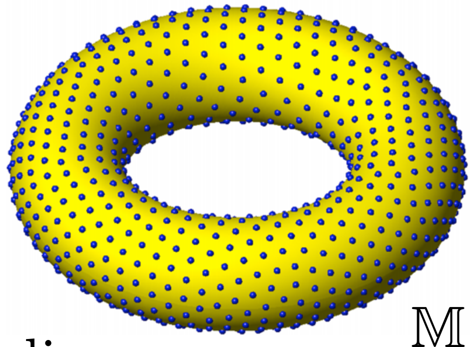
  Main idea: Trace theorem and restriction and extension operators on the native space from Schaback (1999).

Interpolation error estimates

- Specific error estimate results from Fuselier & W (2012).
  - More general results are given in the paper.

Notation:

- $\mathbb{M} \subset \mathbb{R}^3$, $\dim(\mathbb{M}) = 2$.
- $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$
- $\hat{\phi}(\boldsymbol{\xi}) \sim (1 + \|\boldsymbol{\xi}\|_2^2)^{-\tau}$, $\tau > 3/2$
- $s = \tau - 1/2$
- $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$
- h_X = mesh-norm
- q_X = separation radius
- $\rho_X = h_X/q_X$, mesh ratio



Theorem: target functions in the native space

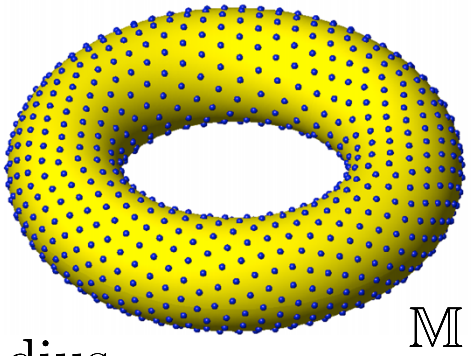
$$\text{If } f \in H^s(\mathbb{M}) \text{ then } \|f - I_X f\|_{L_2(\mathbb{M})} = \mathcal{O}(h_X^s)$$

Interpolation error estimates

- Specific error estimate results from Fuselier & W (2012).
  - More general results are given in the paper.

Notation:

- $\mathbb{M} \subset \mathbb{R}^3$, $\dim(\mathbb{M}) = 2$.
- $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$
- $\hat{\phi}(\boldsymbol{\xi}) \sim (1 + \|\boldsymbol{\xi}\|_2^2)^{-\tau}$, $\tau > 3/2$
- $s = \tau - 1/2$
- $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$
- h_X = mesh-norm
- q_X = separation radius
- $\rho_X = h_X/q_X$, mesh ratio



Corollary: target functions approx. twice as smooth as the native space

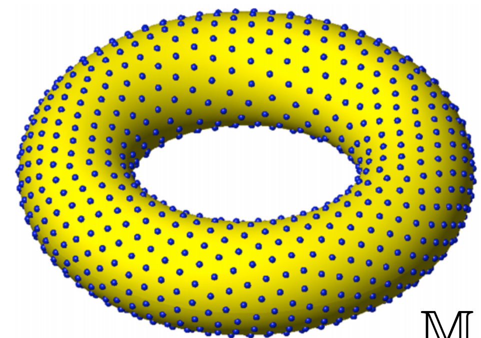
$$\text{If } f \in H^s(\mathbb{M}) \text{ and } T^{-1}f \in L_2(\mathbb{M}) \text{ then } \|f - I_X f\|_{L_2(\mathbb{M})} = \mathcal{O}(h_X^{2s})$$

Interpolation error estimates

- Specific error estimate results from Fuselier & W (2012).
  - More general results are given in the paper.

Notation:

- $\mathbb{M} \subset \mathbb{R}^3$, $\dim(\mathbb{M}) = 2$.
- $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$
- $\hat{\phi}(\boldsymbol{\xi}) \sim (1 + \|\boldsymbol{\xi}\|_2^2)^{-\tau}$, $\tau > 3/2$
- $s = \tau - 1/2$
- $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$
- h_X = mesh-norm
- q_X = separation radius
- $\rho_X = h_X/q_X$, mesh ratio



Theorem: target functions rougher than the native space

If $f \in H^\beta(\mathbb{M})$ with $s > \beta > 1$ then $\|f - I_X f\|_{L_2(\mathbb{M})} = \mathcal{O}(h_X^\beta \rho_X^{s-\beta})$

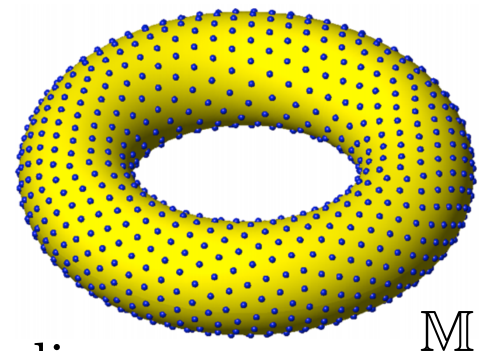
Proof required results Narcowich, Ward, & Wendland (2005; 2006) on $\mathbb{R}^d$

Interpolation error estimates

- Specific error estimate results from Fuselier & W (2012).
  - More general results are given in the paper.

Notation:

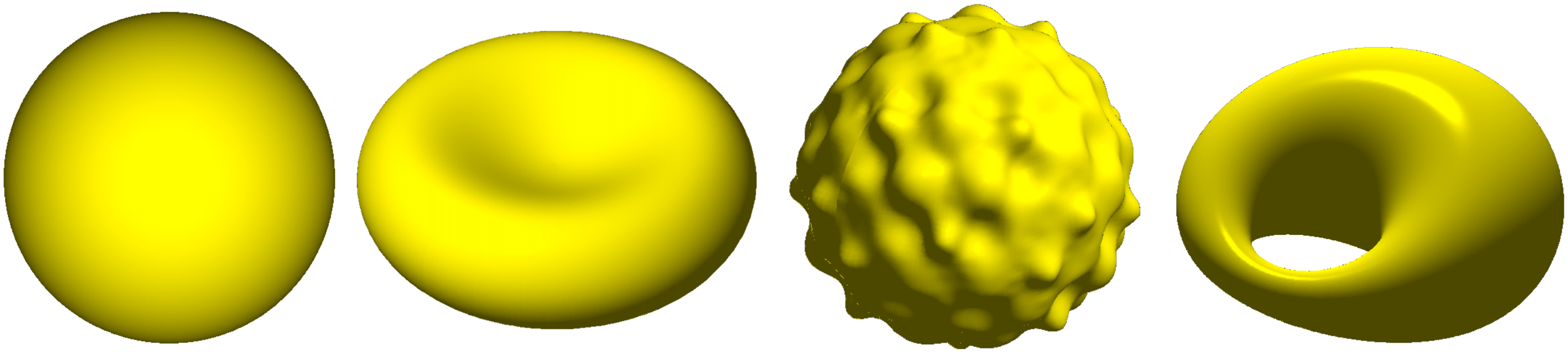
- $\mathbb{M} \subset \mathbb{R}^3$, $\dim(\mathbb{M}) = 2$.
- $\psi(\cdot,\cdot) = \phi(\cdot,\cdot)\big|_{\mathbb{M},\mathbb{M}}$
- $\hat{\phi}(\boldsymbol{\xi}) \sim (1 + \|\boldsymbol{\xi}\|_2^2)^{-\tau}$, $\tau > 3/2$
- $s = \tau - 1/2$
- $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$
- h_X = mesh-norm
- q_X = separation radius
- $\rho_X = h_X/q_X$, mesh ratio



Main point: can use simple RBFs for interpolation on surfaces:

$$I_X f = \sum_{j=1}^{N} c_j \psi(\mathbf{x}, \mathbf{x}_j) = \sum_{j=1}^{N} c_j \phi(\|\mathbf{x} - \mathbf{x}_j\|_2)$$

Return: Reaction diffusion equations on surfaces



- Prototypical model: 2 interacting species

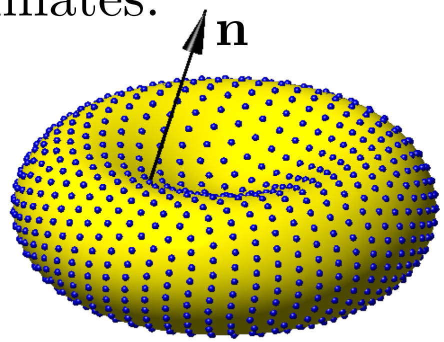
$$\frac{\partial u}{\partial t} = \delta_u \Delta_{\mathbb{M}} u + f_u(t,u,v)$$
$$\frac{\partial v}{\partial t} = \delta_v \Delta_{\mathbb{M}} v + f_v(t,u,v)$$

$\Delta_{\mathbb{M}}$ is the Laplace-Beltrami operator for the surface

- Applications
  - Biology: diffusive transport on a membrane, pattern formation on animal coats, and tumor growth.
  - Chemistry: waves in excitable media (cardiac arrhythmia, electrical signals in the brain).
  - Computer graphics: texture mapping and synthesis and image processing.

Differential operators on surfaces

- Surface gradient on $\mathbb{M}$ in *extrinsic* (or Cartesian) coordinates:



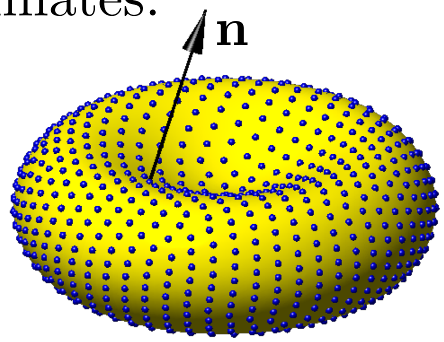
$$\nabla_{\mathbb{M}} := \mathbf{P}\nabla = (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})\,\nabla$$

- After some manipulations

$$\nabla_{\mathbb{M}} := \begin{bmatrix} (\mathbf{e}_x \cdot \mathbf{P})\,\nabla \\ (\mathbf{e}_y \cdot \mathbf{P})\,\nabla \\ (\mathbf{e}_z \cdot \mathbf{P})\,\nabla \end{bmatrix} = \begin{bmatrix} (\mathbf{e}_x - n_x\mathbf{n}) \cdot \nabla \\ (\mathbf{e}_y - n_y\mathbf{n}) \cdot \nabla \\ (\mathbf{e}_z - n_z\mathbf{n}) \cdot \nabla \end{bmatrix} = \begin{bmatrix} \mathbf{p}_x \cdot \nabla \\ \mathbf{p}_y \cdot \nabla \\ \mathbf{p}_z \cdot \nabla \end{bmatrix} = \begin{bmatrix} \mathcal{G}^x \\ \mathcal{G}^y \\ \mathcal{G}^z \end{bmatrix}$$

Differential operators on surfaces

- Surface gradient on $\mathbb{M}$ in *extrinsic* (or Cartesian) coordinates:



$$\nabla_{\mathbb{M}} := \mathbf{P}\nabla = (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})\,\nabla$$

- After some manipulations

$$\nabla_{\mathbb{M}} := \begin{bmatrix} (\mathbf{e}_x \cdot \mathbf{P})\,\nabla \\ (\mathbf{e}_y \cdot \mathbf{P})\,\nabla \\ (\mathbf{e}_z \cdot \mathbf{P})\,\nabla \end{bmatrix} = \begin{bmatrix} (\mathbf{e}_x - n_x\mathbf{n}) \cdot \nabla \\ (\mathbf{e}_y - n_y\mathbf{n}) \cdot \nabla \\ (\mathbf{e}_z - n_z\mathbf{n}) \cdot \nabla \end{bmatrix} = \begin{bmatrix} \mathbf{p}_x \cdot \nabla \\ \mathbf{p}_y \cdot \nabla \\ \mathbf{p}_z \cdot \nabla \end{bmatrix} = \begin{bmatrix} \mathcal{G}^x \\ \mathcal{G}^y \\ \mathcal{G}^z \end{bmatrix}$$

- Surface divergence of smooth vector field $\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^3$ ($\mathbf{f} = (f_x, f_y, f_z)$):

$$\nabla_{\mathbb{M}} \cdot \mathbf{f} := (\mathbf{P}\nabla) \cdot \mathbf{f} = \mathcal{G}^x f_x + \mathcal{G}^y f_y + \mathcal{G}^z f_z$$

- Laplace-Beltrami operator on $\mathbb{M}$ in *extrinsic coordinates*:

$$\Delta_{\mathbb{M}} := (\mathbf{P}\nabla)\cdot(\mathbf{P}\nabla) = \mathcal{G}^x\mathcal{G}^x + \mathcal{G}^y\mathcal{G}^y + \mathcal{G}^z\mathcal{G}^z = \mathcal{D}_{xx} + \mathcal{D}_{yy} + \mathcal{D}_{zz}$$

$\Delta_{\mathbb{M}}$ is the Laplace-Beltrami operator for the surface.

Kernel approximation of surface derivative operators

Idea from Fuselier & W (2013):

- Let $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$ and some smooth target $f : \mathbb{M} \to \mathbb{R}$.
- Interpolate $\underline{f} := f\big|_X$, using restricted (RBF) kernel interpolant:

$$I_X f = \sum_{j=1}^N c_j \psi(\mathbf{x}, \mathbf{x}_j) = \sum_{j=1}^N c_j \phi(\|\mathbf{x} - \mathbf{x}_j\|)$$

Kernel approximation of surface derivative operators

Idea from Fuselier & W (2013):

- Let $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$ and some smooth target $f : \mathbb{M} \to \mathbb{R}$.
- Interpolate $\underline{f} := f\big|_X$, using restricted (RBF) kernel interpolant:

$$I_X f = \sum_{j=1}^{N} c_j \psi(\mathbf{x}, \mathbf{x}_j) = \sum_{j=1}^{N} c_j \phi(\|\mathbf{x} - \mathbf{x}_j\|)$$

- Apply $\mathcal{G}^x$, $\mathcal{G}^y$, $\mathcal{G}^z$ to $I_X f$ and evaluate at X:

$$(\mathcal{G}^x[I_X f])\big|_X = G_x \underline{f}, \quad (\mathcal{G}^y[I_X f])\big|_X = G_y \underline{f}, \quad (\mathcal{G}^z[I_X f])\big|_X = G_z \underline{f}$$

Idea from Fuselier & W (2013):

- Let $X = \{\mathbf{x}_j\}_{j=1}^N \subset \mathbb{M}$ and some smooth target $f : \mathbb{M} \to \mathbb{R}$.
- Interpolate $\underline{f} := f\big|_X$, using restricted (RBF) kernel interpolant:

$$I_X f = \sum_{j=1}^N c_j \psi(\mathbf{x}, \mathbf{x}_j) = \sum_{j=1}^N c_j \phi(\|\mathbf{x} - \mathbf{x}_j\|)$$

- Apply $\mathcal{G}^x$, $\mathcal{G}^y$, $\mathcal{G}^z$ to $I_X f$ and evaluate at X:

$$(\mathcal{G}^x[I_X f])\big|_X = G_x \underline{f}, \quad (\mathcal{G}^y[I_X f])\big|_X = G_y \underline{f}, \quad (\mathcal{G}^z[I_X f])\big|_X = G_z \underline{f}$$

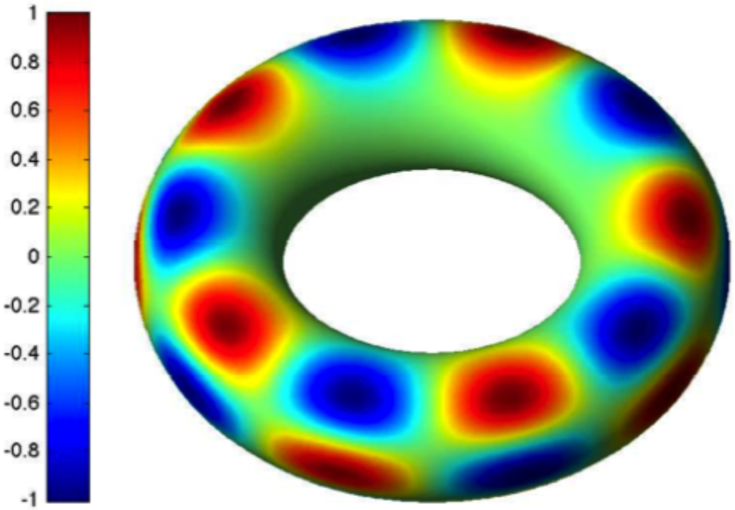
- Approximate $(\Delta_{\mathbb{M}} f)\big|_X$ using G_x, G_y, G_z:

$$(\Delta_{\mathbb{M}} f)\big|_X = ([\mathcal{G}^x\mathcal{G}^x + \mathcal{G}^y\mathcal{G}^y + \mathcal{G}^z\mathcal{G}^z] f)\big|_X \approx \underbrace{(G_xG_x + G_yG_y + G_zG_z)}_{L_{\mathbb{M}}} \underline{f}$$

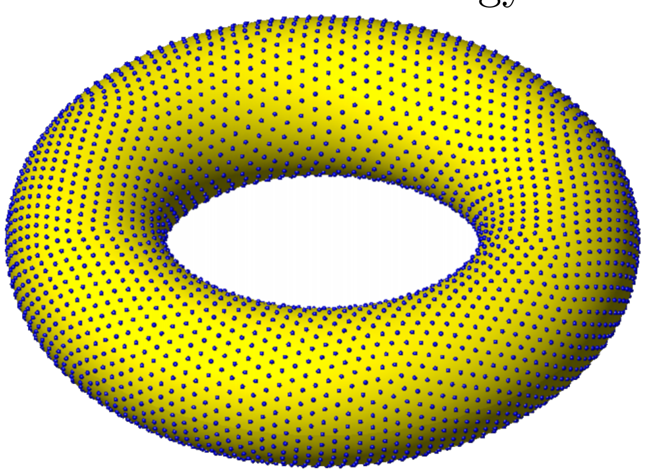
- $L_{\mathbb{M}}$ is an $N \times N$ differentiation matrix

Example: convergence of discrete surface Laplacian

Smooth target f



N near-minimal Riesz energy nodes

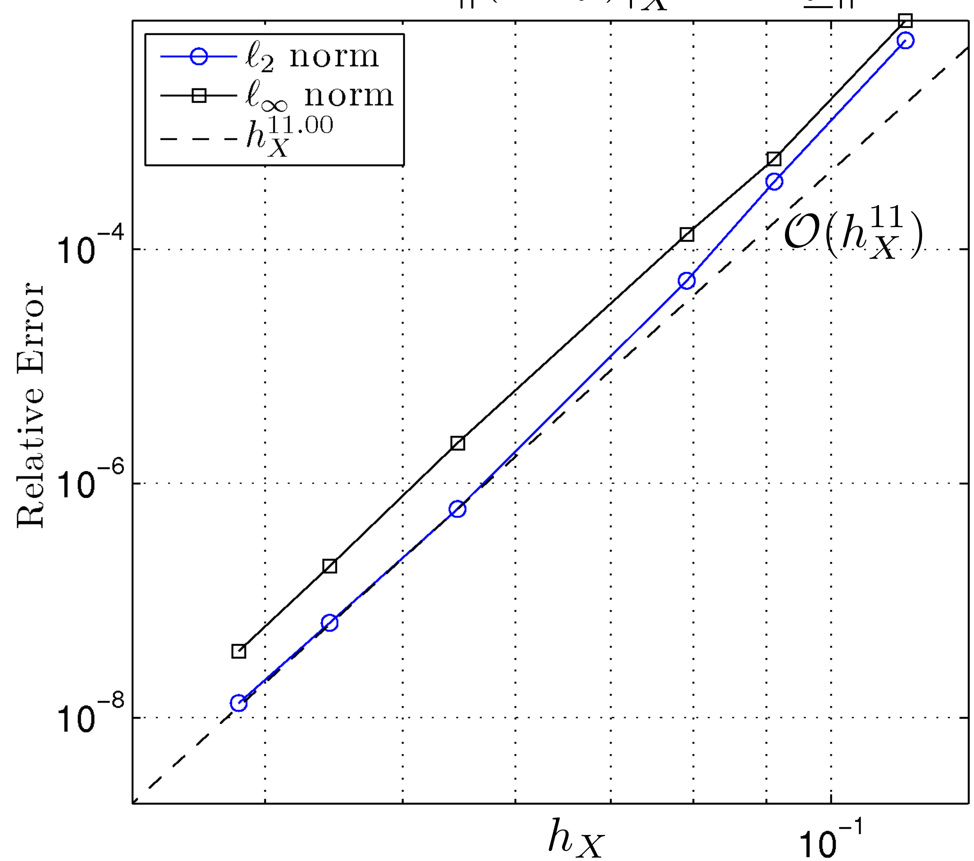


Mesh-norm: $h_X \sim 1/\sqrt{N}$

Matérn kernel: $\psi \Rightarrow \phi(r) = (\varepsilon r)^{9/2} K_{9/2}(\varepsilon r)$

$$\mathcal{N}_\psi = H^{11/2}(\mathbb{M})$$

Error: $\left\| (\Delta_{\mathbb{M}} f)\big|_X - L_{\mathbb{M}} \underline{f} \right\|$



- Error estimates given in Fuselier & W (2013)
- Observed convergence rate is 2 orders higher than theory predicts.

Applications: Turing patterns

- Pattern formation via non-linear reaction-diffusion systems; Turing (1952)

Possible mechanism for animal coat formation (and other morphogenesis phenomena)



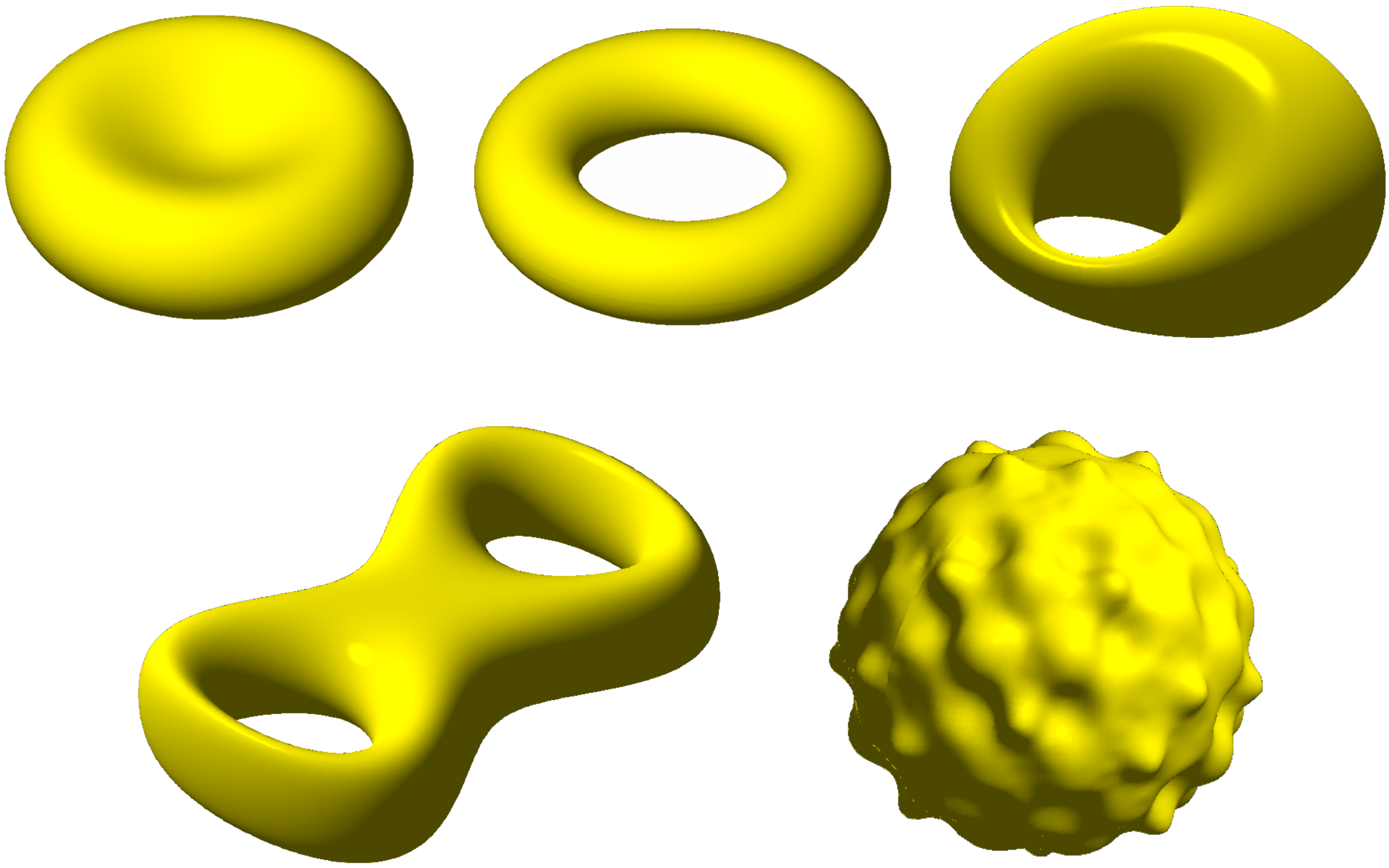
- Example system: Barrio *et al.* (1999)

$$\frac{\partial u}{\partial t} = \delta_u \Delta_{\mathbb{M}} u + \alpha u(1 - \tau_1 v^2) + v(1 - \tau_2 u)$$
$$\frac{\partial v}{\partial t} = \delta_v \Delta_{\mathbb{M}} v + \beta v \left(1 + \frac{\alpha \tau_1}{\beta} uv\right) + u(\gamma + \tau_2 v)$$

- These types of systems have been studied extensively in planar domains.
- Recent studies have focused on the sphere.
- Growing interest in studying these on more general surfaces.
- Numerical method: collocation and method-of-lines (like method from Tutorials 4-6)

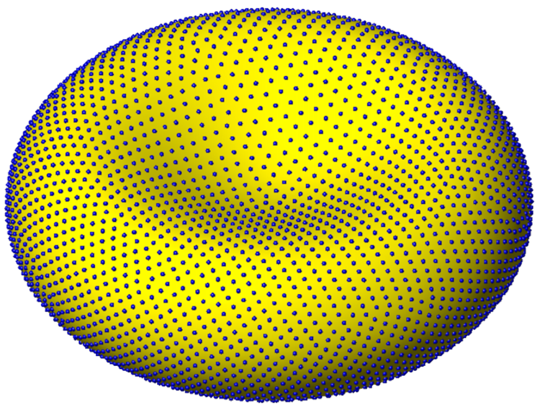
Application: Turing patterns

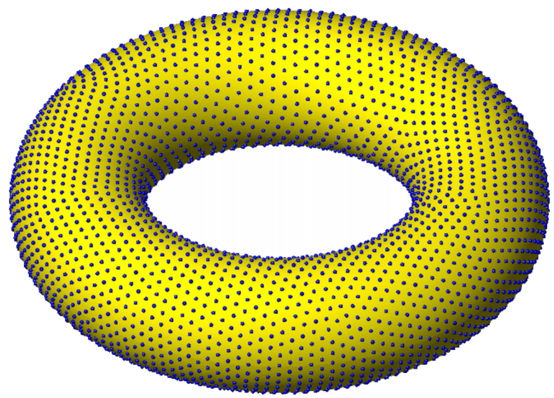
- Surfaces used in the numerical experiments:

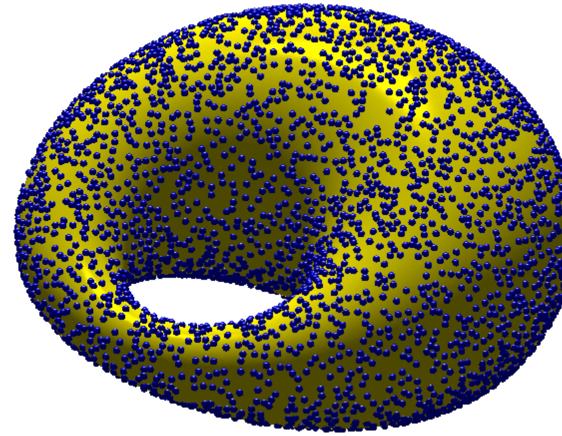


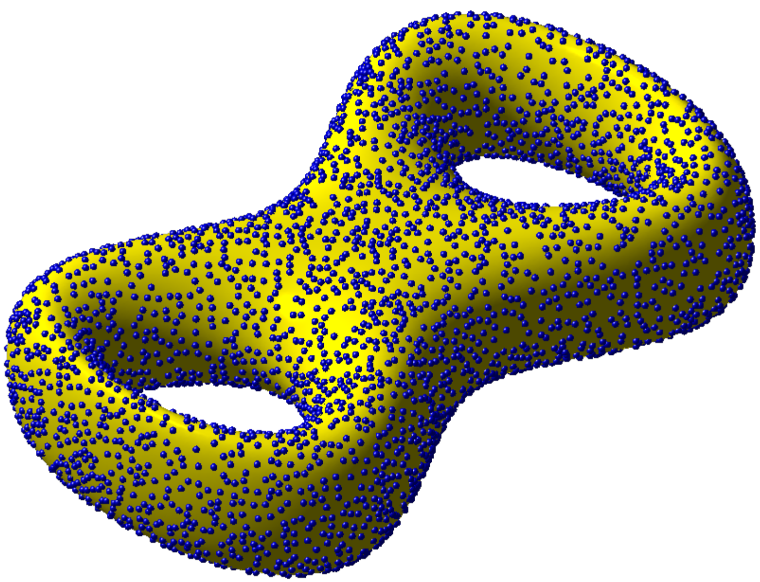
Application: Turing patterns

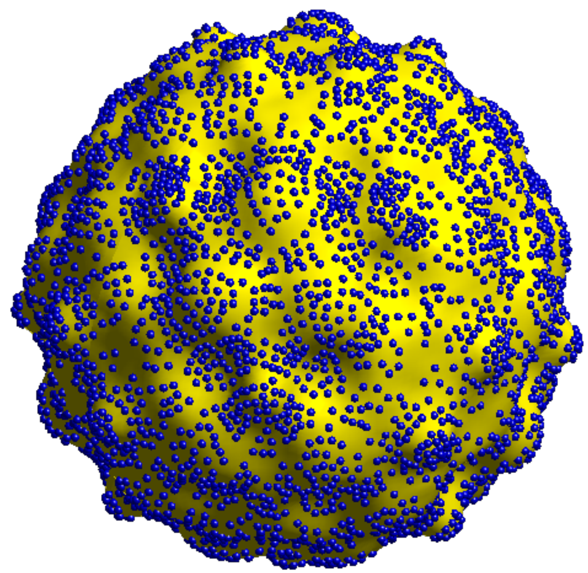
- Node sets X used in the numerical experiments:





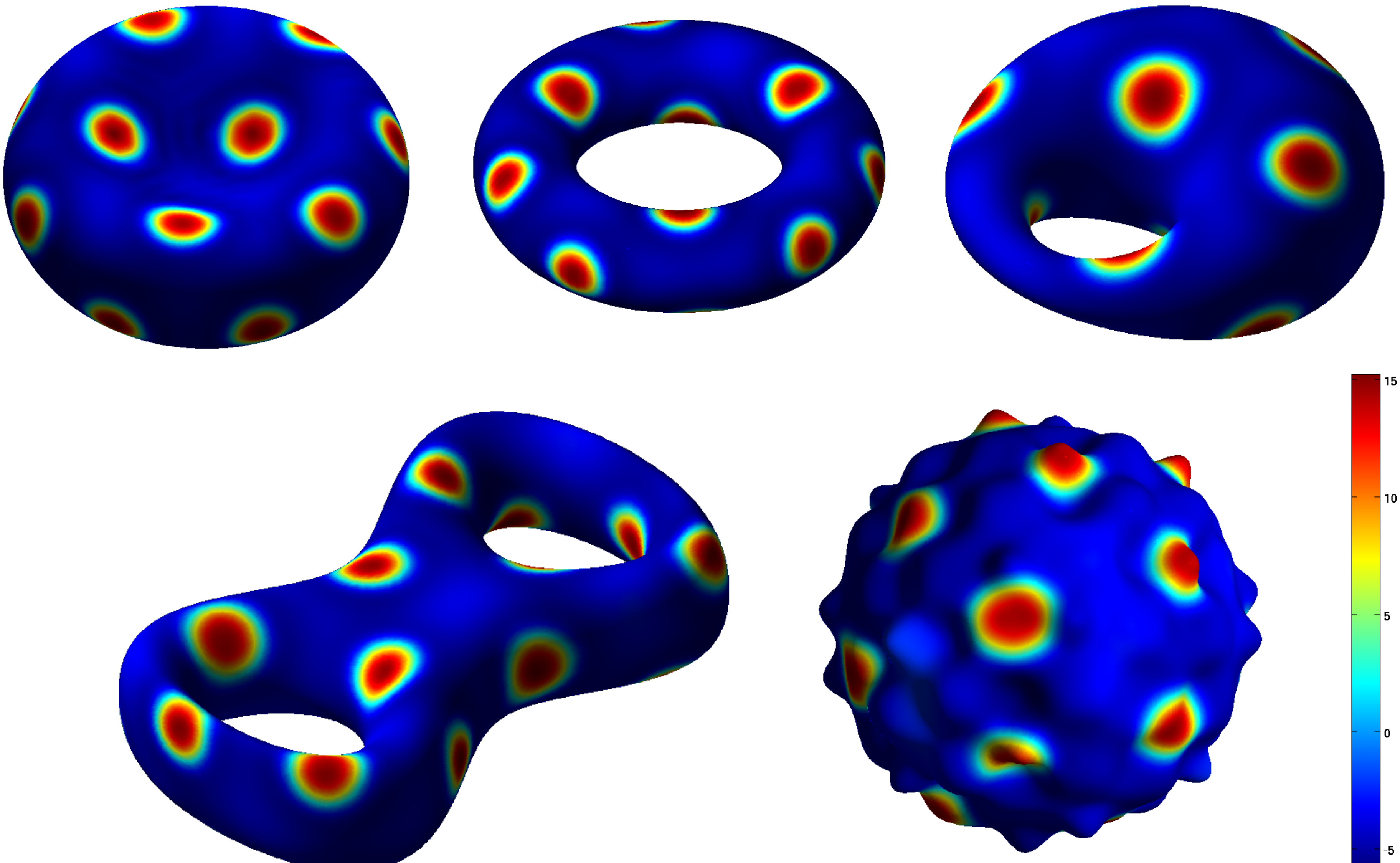






Application: Turing patterns

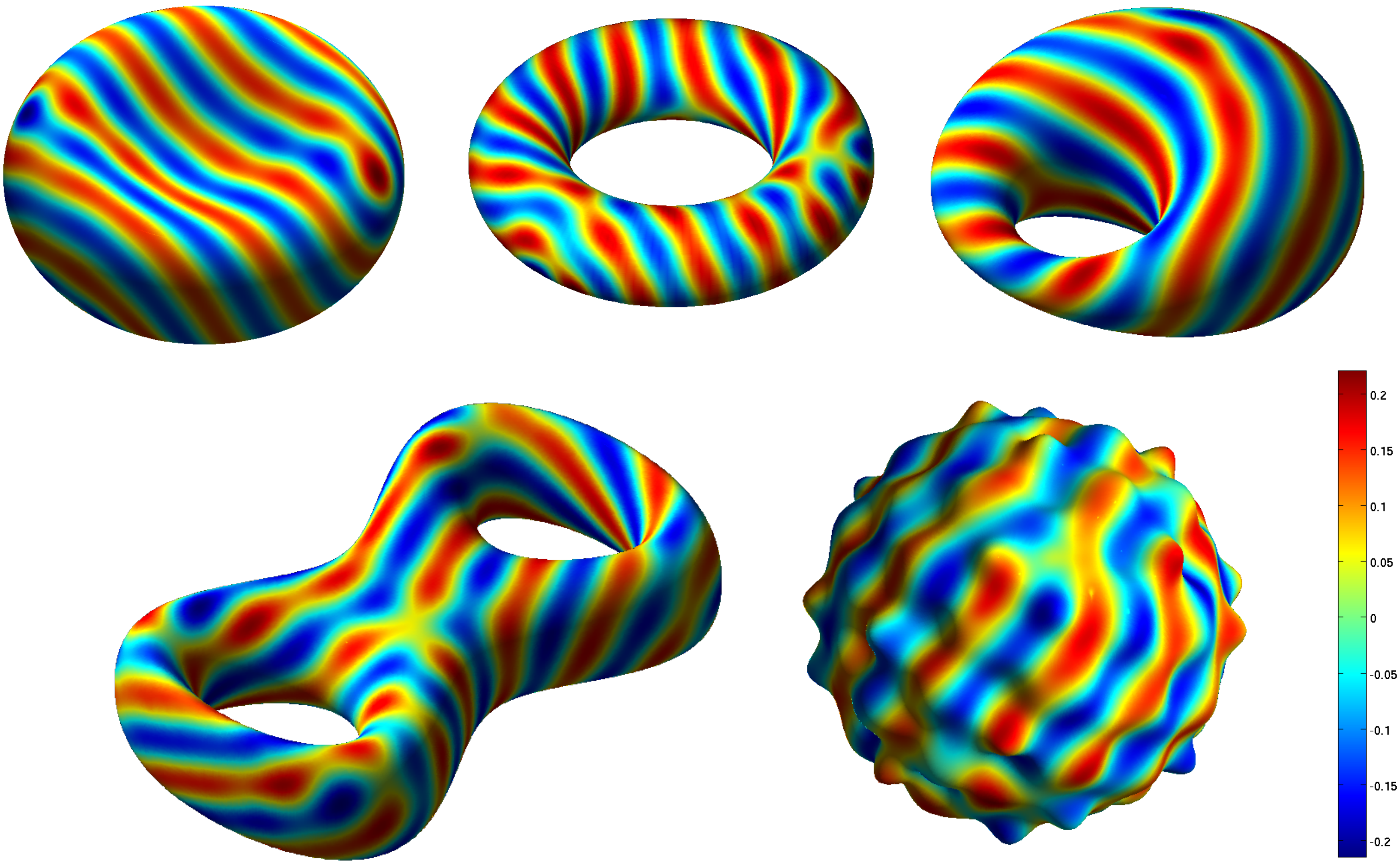
- Numerical solutions: *steady* spot patterns (visualization of u component)



Initial condition: u and v set to random values between +/- 0.5

Application: Turing patterns

- Numerical solutions: *steady* stripe patterns (visualization of u component)



Initial condition: u and v set to random values between +/- 0.5

Application: spiral waves in excitable media

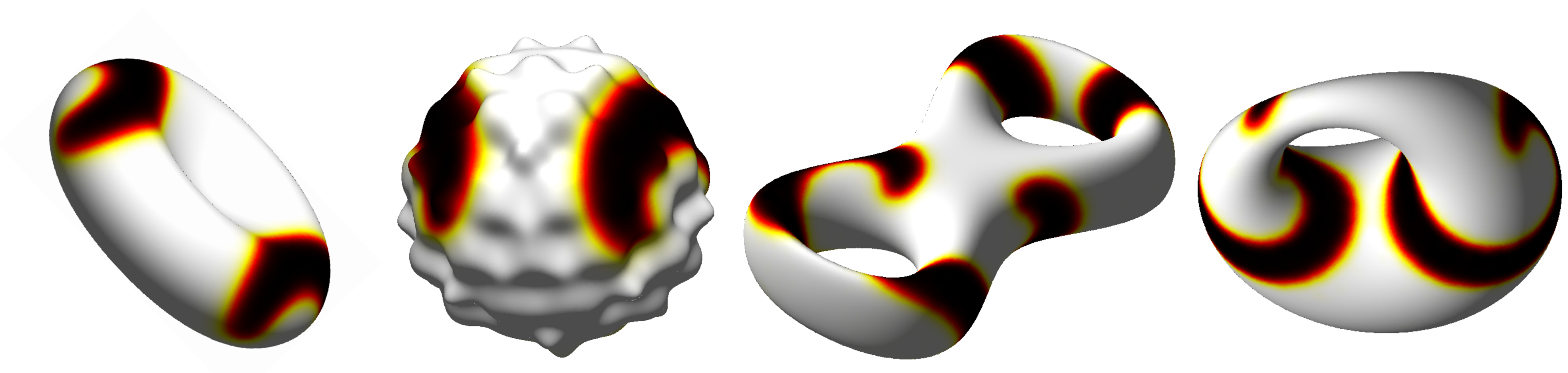
- Example system: Barkley (1991)

$$\frac{\partial u}{\partial t} = \delta_u \Delta_{\mathbb{M}} u + \frac{1}{\epsilon} u (1-u) \left(u - \frac{v+b}{a} \right)$$
$$\frac{\partial v}{\partial t} = \delta_v \Delta_{\mathbb{M}} v + u - v$$

u=activator species
v=inhibitor species

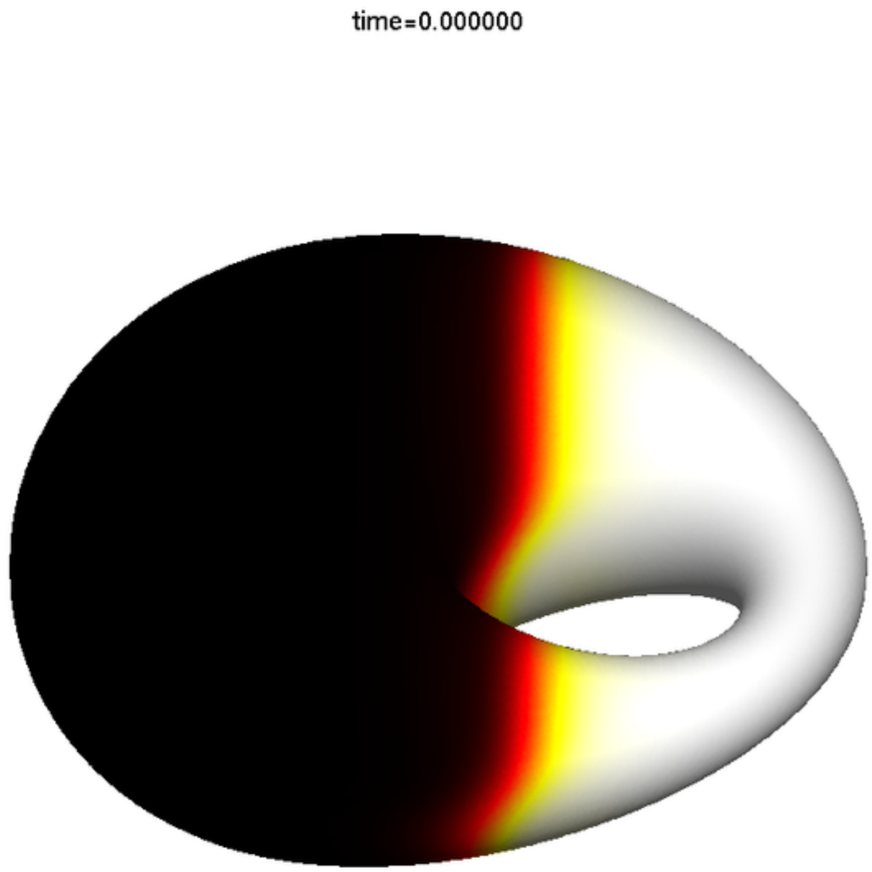
Simplification of FitzHugh-Nagumo model for a spiking neuron.

- Studied extensively on planar regions and somewhat on the sphere.
- Growing interest more physically relevant domains like surfaces.
- Snapshots from different numerical simulations with our method:



visualization of the u (activator) component

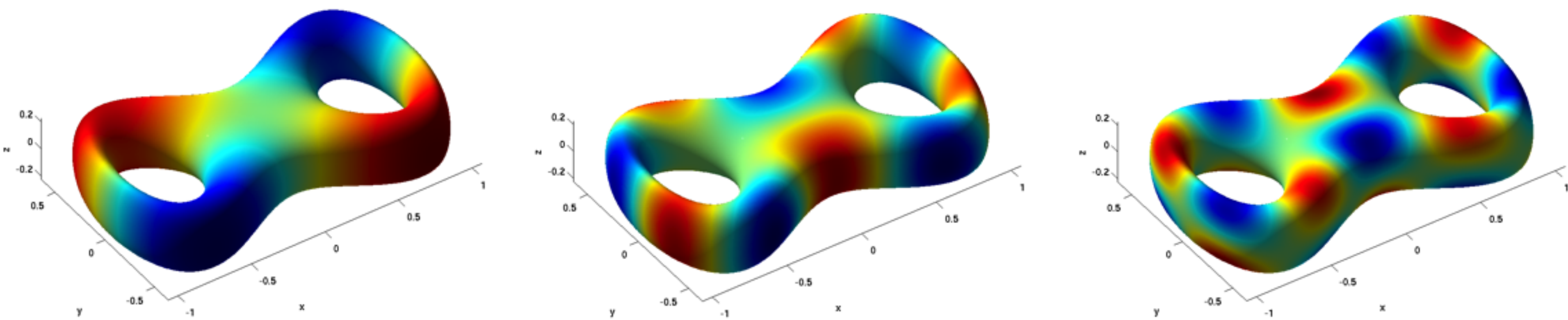
Application: spiral waves in excitable media



Other applications

- The discrete approximation to the surface Laplacian can also be used approximate the surface harmonics.

- Question: Can one hear the shape of a Bretzel?



Current work

- We are presently developing an RBF-FD approach to approximating the surface Laplacian (Joint work with PhD student Varun Shankar).
- This will reduce the computational complexity from $O(N^2)$ per-time step to $O(N)$.
- It will also allow us to go use much larger node sets, and handle more complicated surfaces.
- Below is an example of simulations of the Turing model using the RBF-FD method:

Summary and future

- Restricted kernels offer a relatively simple method for interpolation on rather general surfaces.
  - Interpolation error estimates are similar to what you expect from $\mathbb{R}^d$.
- Method can be used to approximate surface derivatives in a relatively straightforward manner.
  - These approximation can provide high rates of approximation.
  - Can be used to also solve PDEs to high accuracy.
- Future: Biological Applications
  - PDEs on moving surfaces.
  - PDEs that feed back on the shape of the object.
- Future: Improve computational cost
  - Radial basis finite difference formulas (RBF-FD)
  - Partition of unity methods
  - Localized bases

Thank you to the organizers