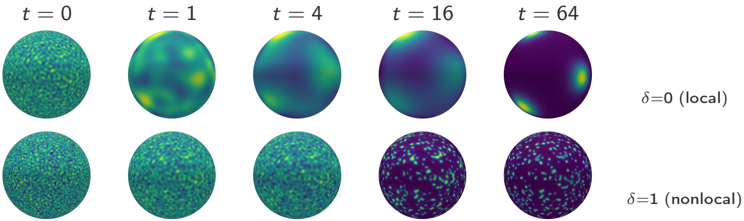
Pattern formation on the sphere

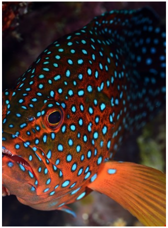
APAM Seminar — February 1, 2018

Hadrien Montanelli

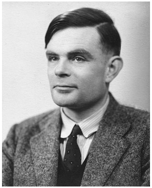
& Q. Du, Y. Nakatsukasa, M. Slevinsky, S. Shvartsman, N. Trefethen



- Nature exhibits a seemingly endless array of patterns


- Alan Turing: those patterns are somehow similar
- He came up with a simple model[1] that could give rise to all manner of patterns: reaction-diffusion equations



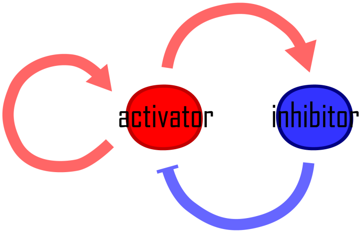
[1] Turing, *The chemical basis of morphogenesis* (1952)

- Pray-predator system

  Inside each organism there are two substances: activator & inhibitor

  Activator: stimulates production

  Inhibitor: slows production down



- Notion of spreading rate

- Mathematically: reaction-diffusion equations

$$\begin{cases} u_t = \epsilon_u^2 \Delta u + d_u F(u,v), \\ v_t = \epsilon_v^2 \Delta v + d_v G(u,v) \end{cases}$$

- Effects of curvature (plane $\to$ sphere) and nonlocality ($\Delta = \mathcal{L}_0 \to \mathcal{L}_\delta$)?

Introduction (3/3)

My contributions

- **Problem:** Computing solutions of local and nonlocal PDEs of the form

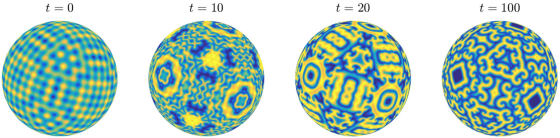
$$u_t(t,\theta,\varphi) = \epsilon^2\mathcal{L}_\delta u + \mathcal{N}(u), \quad (\theta,\varphi) \in \times[0,\pi] \times [-\pi,\pi]$$

- **Method (local):** 2D Fourier series & Implicit-explicit schemes

  My contribution: a new spectral method

- **Method (nonlocal):** Spherical harmonics & Exponential integrators

  My contribution: nonlocal reaction-diffusion equations & a spectral method



δ=0 (local)

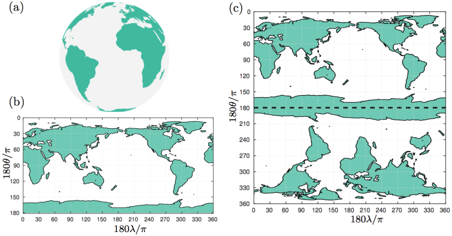
Solving local PDEs (1/6)

Discretizing space with 2D Fourier series

- Local PDEs:

$$u_t = \epsilon^2 \Delta u + \mathcal{N}(u), \quad \Delta u = u_{\theta\theta} + \frac{\cos\theta}{\sin\theta} u_\theta + \frac{1}{\sin^2\theta} u_{\varphi\varphi}$$

- Double Fourier Sphere method (Meriless, Orszag, Townsend et al.[2])



- Approximations by 2D Fourier series with $N = n^2$ coeffs:

$$u(t,\theta,\varphi) \approx \sum_{\ell=-n/2}^{n/2-1} \sum_{m=-n/2}^{n/2-1} \hat{u}_{\ell m}(t) e^{i(\ell\theta + m\varphi)}$$

[2] Townsend, Wilber & Wright, *Computing with functions in spherical and polar geometries* (2016)

Solving local PDEs (2/6)

The local Laplace–Beltrami matrix

- Operator Δ discretized with a $N \times N$ matrix $\mathbf{L}$ acting on Fourier coefficients
- PDE $u_t = \epsilon^2 \Delta u + \mathcal{N}(u) \Longrightarrow$ system of ODEs $\hat{u}'(t) = \epsilon^2 \mathbf{L} \hat{u} + \mathbf{N}(\hat{u})$
- Method for constructing $\mathbf{L}$:[3]

  standard Fourier matrices

  projection matrices to filter out the sawtooth mode

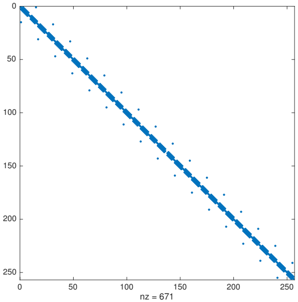
- $\mathbf{L}$ has good numerical properties:

  preserves doubled-up symmetry

  preserves smoothness at the poles

  has real and negative eigenvalues

  can be inverted in $\mathcal{O}(N)$ operations



[3] M. & Nakatsukasa, *Fourth-order time-stepping for PDEs on the sphere* (2017)

Solving local PDEs (3/6)

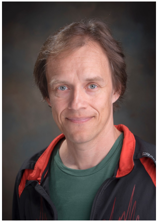
Time-stepping with implicit-explicit schemes

- PDE $u_t = \epsilon^2 \Delta u + \mathcal{N}(u) \implies$ system of ODEs $\hat{u}'(t) = \epsilon^2 \mathbf{L}\hat{u} + \mathbf{N}(\hat{u})$
- Discretization with time-step h: $\hat{u}^k = \hat{u}(t_k)$, $t_k = kh$
- Implicit-explicit (Ascher et al.[4]): implicit formula for $\mathbf{L}$, explicit for $\mathbf{N}$

$$(3\mathbf{I} - 2h\epsilon^2\mathbf{L})\hat{u}^{k+1} = 4\hat{u}^k - \hat{u}^{k-1} + 4h\mathbf{N}(\hat{u}^k) - 2h\mathbf{N}(\hat{u}^{k-1})$$

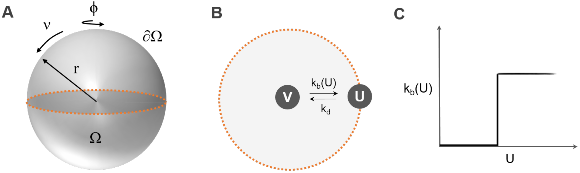
Nonlinear term in physical space: $\mathbf{N}(\hat{u}^k) = \mathbf{F}\mathcal{N}(\mathbf{F}^{-1}\hat{u}^k)$ *with FFT matrix* $\mathbf{F}$

Cost per time-step: $\mathcal{O}(N \log N)$

[4] Ascher, Ruuth & Wetton, *Implicit-explicit methods for time-dependent PDEs* (1995)

- Embryogenesis model:[5]



Equations:

$$\begin{cases} U_t = \dfrac{\epsilon_u^2}{r^2}\Delta U + k_b H(U-B)V - k_d U \quad \text{on } \partial\Omega, \\ V_t = \epsilon_v^2 \Delta V \quad \text{in } \Omega, \\ \epsilon_v^2 \nabla V \cdot n = -k_b H(U-B)V + k_d U \end{cases}$$

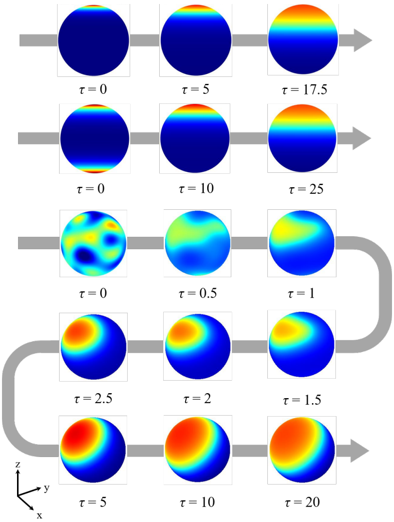
Nondimensionalizing:

$$u = k_d U / k_b V_{max}, \ \tau = k_d t,$$
$$\epsilon^2 = \epsilon_u^2 / k_d r^2, \ \alpha = 3k_b/2k_d r,$$
$$\beta = k_d B / k_b V_{max}$$

$$u_\tau = \epsilon^2 \Delta u + \left(1 - \frac{\alpha}{2\pi}\bar{u}\right) H(u-\beta) - u$$

[5] Diegmiller, M., Muratov & Shvartsman, *Spherical caps in cell polarization* (2018)

Solving local PDEs (5/6)
Application

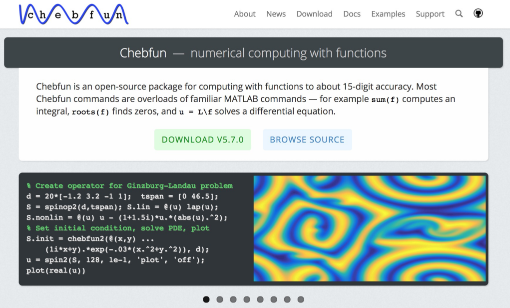


- A local perturbation will form a single spherical cap
- Multiple competing perturbations will relax to a single cap
- Random initial conditions will also relax to a single cap
- Symmetry breaking is crucial for the development of virtually all living organisms

Solving local PDEs (6/6)

Demo

- Chebfun : MATLAB package for computing to $\approx$ 15 digits of accuracy



Aurentz, Austin, Driscoll, Filip, Güttel, Hale, Hashemi, Nakatsukasa, Platte, Townsend, Trefethen, Wright

Solving nonlocal PDEs (1/6)

Discretizing space with spherical harmonics

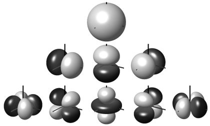
- Nonlocal PDEs:

$$u_t = \epsilon^2 \mathcal{L}_\delta u + \mathcal{N}(u), \quad \mathcal{L}_\delta u(\mathbf{x}) = \int_{\mathbb{S}^2} \rho_\delta(|\mathbf{x}-\mathbf{y}|)\,[u(\mathbf{y}) - u(\mathbf{x})]\,d\Omega(\mathbf{y})$$

- Operator $\mathcal{L}_\delta$ decouples the spherical harmonic modes ($\ell \geq 0$, $-\ell \leq m \leq \ell$):

$$\mathcal{L}_\delta Y_\ell^m(\theta,\varphi) = \underbrace{\left(2\pi \int_{-1}^{1} (P_\ell(t) - 1)\rho_\delta(\sqrt{2(1-t)})\,dt\right)}_{\lambda_\delta(\ell)} Y_\ell^m(\theta,\varphi),$$

$$Y_\ell^m(\theta,\varphi) = e^{im\varphi} P_\ell^m(\cos\theta)$$



- Approximations by spherical harmonic series with $N = (n+1)(2n+1)$ coeffs:

$$u(t,\theta,\varphi) \approx \sum_{\ell=0}^{n} \sum_{m=-\ell}^{+\ell} \tilde{u}_{\ell m}(t) Y_\ell^m(\theta,\varphi)$$

- $\mathcal{L}_\delta$ discretized with a matrix $\mathbf{L}_\delta$ acting on spherical harmonics coefficients[6]

- PDE $u_t = \epsilon^2 \mathcal{L}_\delta u + \mathcal{N}(u) \Longrightarrow$ system of ODEs $\tilde{u}'(t) = \epsilon^2 \mathbf{L}_\delta \tilde{u} + \mathbf{N}(\tilde{u})$

- $\mathbf{L}_\delta$ is diagonal with entries $\lambda_\delta(\ell)$:

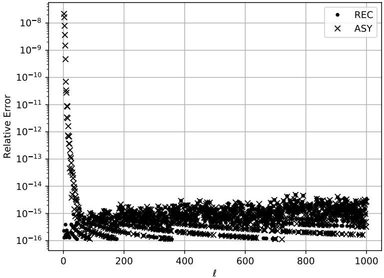
$$\lambda_\delta(\ell) = 2\pi \int_{-1}^{1} (P_\ell(t) - 1)\rho_\delta\left(\sqrt{2(1-t)}\right) dt$$

$$\rho_\delta\left(\sqrt{2(1-t)}\right) = \frac{(1+\alpha)2^{1+\alpha}}{\pi\delta^{2+2\alpha}(1-t)^{1-\alpha}}\chi_{[0,\delta]}$$

- Computation of $\lambda_\delta(\ell)$:

  Clenshaw–Curtis quadrature[7]

  recurrence & asymptotic formulas



[6] Slevinsky, M. & Du, *A spectral method for nonlocal diffusion operators on the sphere* (2018)
[7] Trefethen, *Is Gauss quadrature better than Clenshaw-Curtis?* (2008)

Solving nonlocal PDEs (3/6)

Time-stepping with exponential integrators

- PDE $u_t = \epsilon^2 \mathcal{L}_\delta u + \mathcal{N}(u) \Longrightarrow$ system of ODEs $\tilde{u}'(t) = \epsilon^2 \mathbf{L}_\delta \tilde{u} + \mathbf{N}(\tilde{u})$
- Discretization with time-step h: $\tilde{u}^k = \tilde{u}(t_k)$, $t_k = kh$
- Exponential integrators (Cox & Matthews, Hochbruck & Ostermann[8]): integrate $\mathbf{L}_\delta$ exactly with matrix exponential, numerical scheme for $\mathbf{N}$

$$\tilde{u}^{k+1} = e^{h\epsilon^2 \mathbf{L}_\delta} \tilde{u}^k + (\epsilon^2 \mathbf{L}_\delta)^{-1}(e^{h\epsilon^2 \mathbf{L}_\delta} - \mathbf{I})\mathbf{N}(\tilde{u}^k)$$

Nonlinear term in physical space: $\mathbf{N}(\tilde{u}^k) = \mathbf{G}\mathcal{N}(\mathbf{G}^{-1}\tilde{u}^k)$ with FST matrix $\mathbf{G}$

Cost per time-step: $\mathcal{O}(N \log^2 N)$

[8] Hochbruck & Ostermann, *Exponential integrators* (2010)

- Brusselator model:[9]

$$A \to U$$
$$2U + V \to 3U$$
$$B + U \to V + C$$
$$U \to D$$

Equations:

$$\begin{cases} U_T = \epsilon_u^2 \mathcal{L}_\delta U + A - (B+1)U + U^2 V, \\ V_T = \epsilon_v^2 \mathcal{L}_\delta V + BU - U^2 V \end{cases}$$

Scaling:

$$u = \epsilon_u U / \sqrt{(B+1)\epsilon_v^2}, \; f = B/(B+1)$$
$$v = \sqrt{(B+1)\epsilon_v^2} V / B\epsilon_u, \; t = (B+1)T$$
$$\epsilon = \epsilon_u / \sqrt{B+1}, \; \tau = (B+1)/\epsilon_v^2$$

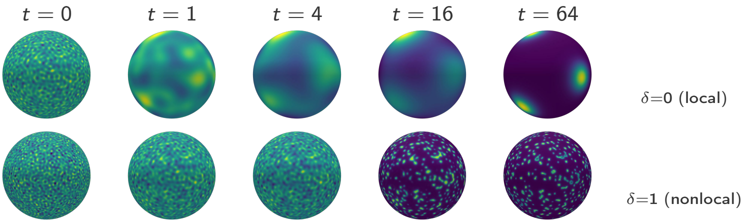
$$u_t = \epsilon^2 \mathcal{L}_\delta u + \epsilon^2 E - u + f u^2 v, \quad \tau v_t = \mathcal{L}_\delta v + \epsilon^{-2}(u - u^2 v)$$

[9] Prigogine & Lefever, *Symmetry breaking instabilities in dissipative systems* (1968)

Solving nonlocal PDEs (5/6)

Application

- Local case: perturbations from the steady state relax to spot patterns[10]
- Nonlocal case: speckled solution
- Nonlocal diffusion operators introduce new qualitative behaviors



[10] Trinh & Ward, *The dynamics of spot patterns for reaction-diffusion systems on the sphere* (2016)

Solving nonlocal PDEs (6/6)

Demo

- Julia: `FastTransforms.jl` and `SpectralTimeStepping.jl` packages

julia

home source downloads docs packages blog community ecosystems learning teaching publications GSoC juliacon

Julia is a high-level, high-performance dynamic programming language for numerical computing. It provides a sophisticated compiler, distributed parallel execution, numerical accuracy, and an extensive mathematical function library. Julia's Base library, largely written in Julia itself, also integrates mature, best-of-breed open source C and Fortran libraries for linear algebra, random number generation, signal processing, and string processing. In addition, the Julia developer community is contributing a number of external packages through Julia's built-in package manager at a rapid pace. IJulia, a collaboration between the Jupyter and Julia communities, provides a powerful browser-based graphical notebook interface to Julia.

Julia programs are organized around multiple dispatch; by defining functions and overloading them for different combinations of argument types, which can also be user-defined. For a more in-depth discussion of the rationale and advantages of Julia over other systems, see the following highlights or read the introduction in the online manual.

- Also available in MATLAB, `mex`-ing from Julia

Other projects (1/3)

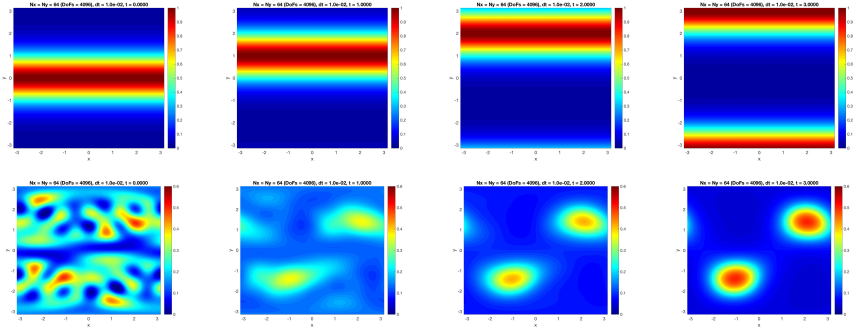
Advection equations on the sphere

- Toy problem:

$$u_t + u_\theta = 0, \quad \theta \in [0, \pi]$$

- Double Fourier Sphere symmetry is not preserved

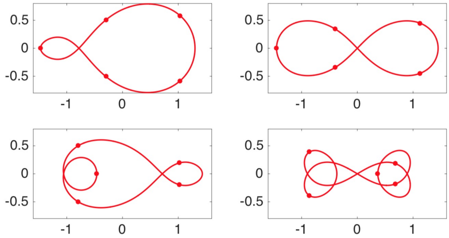
$t = 0$ $\quad$ $t = 1$ $\quad$ $t = 2$ $\quad$ $t = 3$

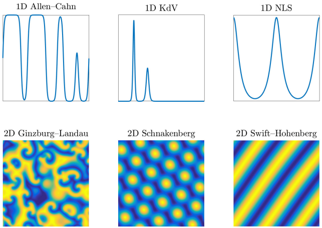


Other projects (2/3)

Chebfun-related projects

- Lead developer of the MATLAB-based Chebfun package (10k+ lines added)
- Chebfun capabilities for periodic ODEs/ODE eigenvalue problems [11]
- `cheb.choreo` code to find periodic solutions of the *n*-body problem [12] [13]
- `spin`, `spin2`, `spin3` codes to solve periodic PDEs in 1D/2D/3D [14]





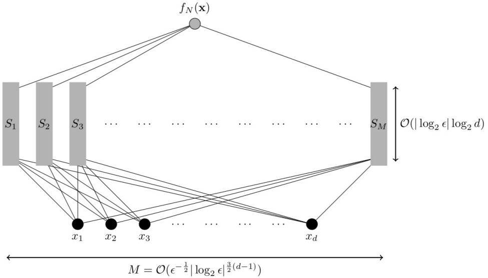
[11] Wright, Javed, M. & Trefethen, *Extension of Chebfun to periodic functions* (2015)
[12] M. & Gushterov, *Computing planar and spherical choreographies* (2015)
[13] M., *Computing hyperbolic choreographies* (2016)
[14] M. & Bootland, *Solving periodic stiff PDEs in 1D, 2D and 3D with exponential integrators* (2016)

- New error estimates for the approximation of functions by deep networks:[15]

$$f(\mathbf{x}) = \sum_{\mathbf{l}} \sum_{\mathbf{i}\in\mathbf{I_l}} v_{\mathbf{l},\mathbf{i}}\phi_{\mathbf{l},\mathbf{i}}(\mathbf{x}), \quad f_N(\mathbf{x}) = \sum_{|\mathbf{l}|_1 \leq m+d-1} \sum_{\mathbf{i}\in\mathbf{I_l}} v_{\mathbf{l},\mathbf{i}}\widetilde{\phi}_{\mathbf{l},\mathbf{i}}(\mathbf{x})$$

$$\|f - f_N\| \leq \epsilon \quad \text{whenever} \quad N = \mathcal{O}(\epsilon^{-\frac{1}{2}}|\log_2 \epsilon|^{\frac{3}{2}(d-1)+1}\log_2 d)$$

- The curse of dimensionality is significantly lessened



[15] M. & Du, *Deep ReLU networks lessen the curse of dimensionality* (2018)

Discussion

- Fast algorithms for solving both local and nonlocal PDEs on the sphere
- Nonlocal diffusion operators introduce new qualitative behaviors
- Algorithms available online and have led to collaborations across fields
- Possible extensions:

  Local and nonlocal advection equations

  Nonlocal phase-field crystal models

  PDEs on spheroids