Recent Challenges and Discoveries in Modeling the Seismic Cycle

Brittany A. Erickson
Department of Computer Science, University of Oregon

with Zac Cross, Tamanna Saini, Tobias Harvey, Alexandre Chen



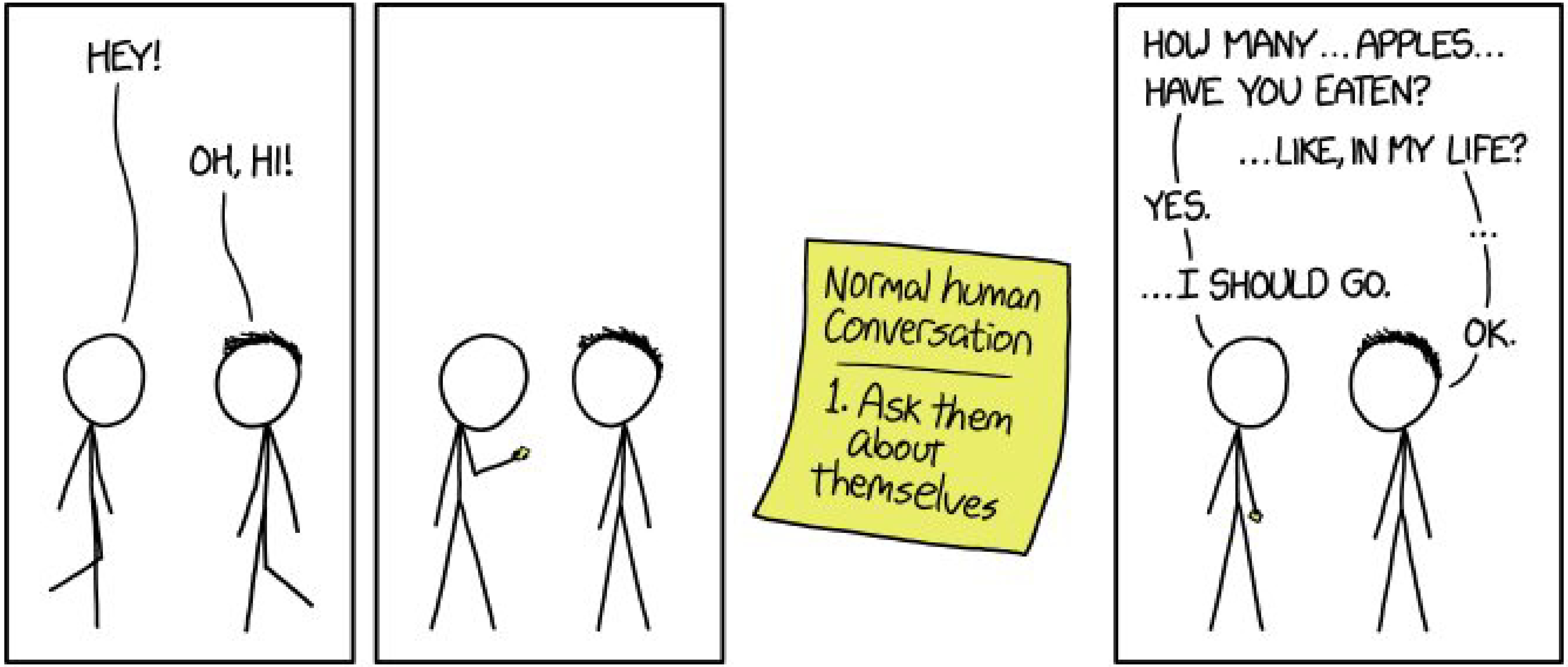


CRESCENT
CASCADIA REGION EARTHQUAKE SCIENCE CENTER

SCEC
AN NSF+USGS CENTER

Talk overview

- Tidbits of (awkward?) conversation while talking about science
- Science drivers
- Overwhelming challenges - focus is on **volume-based SEAS methods**
- Next steps…



Some random things people have said to me (or I've overheard) over the years:

To Steve Day: "Your dynamic rupture simulations are so cool, *but why haven't you done earthquake cycles*?" — unnamed student at SCEC annual meeting.

"In the time it takes to compute a single time step in an earthquake cycle model, I can simulate an *entire* dynamic rupture" — **Steve**

Some random things people have said to me (or I've overheard) over the years:

To Steve Day: "Your dynamic rupture simulations are so cool, *but why haven't you done earthquake cycles*?" — unnamed student at SCEC annual meeting.

"In the time it takes to compute a single time step in an earthquake cycle model, I can simulate an *entire* dynamic rupture" — **Steve**

To me the other day: "I estimate it will take 27 hours just to form the matrix, let alone use it" — one of my current grad students.

"Sounds fun…" — **me.**

Some random things people have said to me (or I've overheard) over the years:

To Steve Day: "Your dynamic rupture simulations are so cool, *but why haven't you done earthquake cycles*?" — unnamed student at SCEC annual meeting.

"In the time it takes to compute a single time step in an earthquake cycle model, I can simulate an *entire* dynamic rupture" — **Steve**

To me the other day: "I estimate it will take 27 hours just to form the matrix, let alone use it" — one of my current grad students.

"Sounds fun…" — **me.**

"I didn't believe you when you said people can make a career out of solving $Ax = b$…. but… that's literally the only thing our company does" — one of my former grad students.

Some random things people have said to me (or I've overheard) over the years:

To Steve Day: "Your dynamic rupture simulations are so cool, *but why haven't you done earthquake cycles*?" — unnamed student at SCEC annual meeting.

"In the time it takes to compute a single time step in an earthquake cycle model, I can simulate an *entire* dynamic rupture" — **Steve**

To me the other day: "I estimate it will take 27 hours just to form the matrix, let alone use it" — one of my current grad students.

"Sounds fun…" — **me.**

"I didn't believe you when you said people can make a career out of solving $Ax = b$…. but… that's literally the only thing our company does" — one of my former grad students.

A talk given at a math conference earlier this month "For my last slide, I'll include some potential applications, even though I don't care about applications of my work…"


3D physically robust, computationally tractable, verified and validated earthquake cycles models
How do we bridge this gap?
Earthquake Science Questions
Fault locking and rupture potential?
Recurrence times of great earthquakes?
How do fault ruptures start and stop?
How do tectonics drive earthquake hazards?
How is strain accumulated and released?
Forces that distribute load onto a fault network?
Constitutive relations?
Coupling mechanisms?
Subsequent failures?

A (generalized) comparison:

Dynamic Rupture:	**Earthquake Cycle:**
• Simulates a single earthquake • Ad-hoc initial conditions • Main computational cost *at each time step* (assuming explicit method) is the SpMV: $A \cdot x$, where A is an $n \times n$ matrix and x is an $n \times 1$ vector	

A (generalized) comparison:

Dynamic Rupture:

- Simulates a single earthquake
- Ad-hoc initial conditions
- Main computational cost *at each time step* (assuming explicit method) is the SpMV: $A \cdot x$, where A is an $n \times n$ matrix and x is an $n \times 1$ vector

The flops necessary to compute an SpMV is proportional to the $nnz(A)$, e.g. $3n$.

In 3D, with grid resolution of ~10 m and a computational domain of (~100 km), $n \sim 3 \cdot 10^{12}$.

With wave speeds ~ 3km/s, CFL dictates a stable time step of about 1 ms. Need to simulate ~ minutes -> ~1 trillion time steps (or SpMVs).

Exascale can perform 10^{18} FLOPS, but very hard to achieve.

Theoretically, should be able to compute simulations in ~10-100hrs [e.g. Roten et al (2016), Uphoff et al. (2017)]

Earthquake Cycle:

A (generalized) comparison:

Dynamic Rupture:

- Simulates a single earthquake
- Ad-hoc initial conditions
- Main computational cost *at each time step* (assuming explicit method) is the SpMV: $A \cdot x$, where A is an $n \times n$ matrix and x is an $n \times 1$ vector

The flops necessary to compute an SpMV is proportional to the $nnz(A)$, e.g. $3n$.

In 3D, with grid resolution of ~10 m and a computational domain of (~100 km), $n \sim 3 \cdot 10^{12}$.

With wave speeds ~ 3km/s, CFL dictates a stable time step of about 1 ms. Need to simulate ~ minutes -> ~1 trillion time steps (or SpMVs).

Exascale can perform 10^{18} FLOPS, but very hard to achieve.

Theoretically, should be able to compute simulations in ~10-100hrs [e.g. Roten et al (2016), Uphoff et al. (2017)]

Earthquake Cycle:

- Simulates thousands of years of seismic activity
- Self-consistent initial conditions
- Main computational cost (for *some versions* SEAS models) *at each time step* is the linear solve: $Ax = b$
- Where A is a sparse $n \times n$ matrix and x, b are $n \times 1$ vectors

A (generalized) comparison:

Dynamic Rupture:

- Simulates a single earthquake
- Ad-hoc initial conditions
- Main computational cost *at each time step* (assuming explicit method) is the SpMV: $A \cdot x$, where A is an $n \times n$ matrix and x is an $n \times 1$ vector

The flops necessary to compute an SpMV is proportional to the $nnz(A)$, e.g. $3n$.

In 3D, with grid resolution of ~10 m and a computational domain of (~100 km), $n \sim 3 \cdot 10^{12}$.

With wave speeds ~ 3km/s, CFL dictates a stable time step of about 1 ms. Need to simulate ~ minutes -> ~1 trillion time steps (or SpMVs).

Exascale can perform 10^{18} FLOPS, but very hard to achieve.

Theoretically, should be able to compute simulations in ~10-100hrs [e.g. Roten et al (2016), Uphoff et al. (2017)]

Earthquake Cycle:

- Simulates thousands of years of seismic activity
- Self-consistent initial conditions
- Main computational cost (for *some versions* SEAS models) *at each time step* is the linear solve: $Ax = b$
- Where A is a sparse $n \times n$ matrix and x, b are $n \times 1$ vectors

Volume-based methods: In 3D, $n \sim 3 \cdot 10^{12}$.

WORST CASE: flops necessary to solve the sparse linear system is $O(n^2)$. The linear solve (i.e. one time-step) will take ~3 months*

OPTIMAL: flops necessary to solve the sparse linear system is $O(n)$. The linear solve (i.e. one time-step) will take ~1 ms*

REASONABLE GOAL: flops necessary to solve the sparse linear system is $O(n) - O(n^2)$. Will still require making some sacrifices.

* if achieving peak performance on an exascale machine.

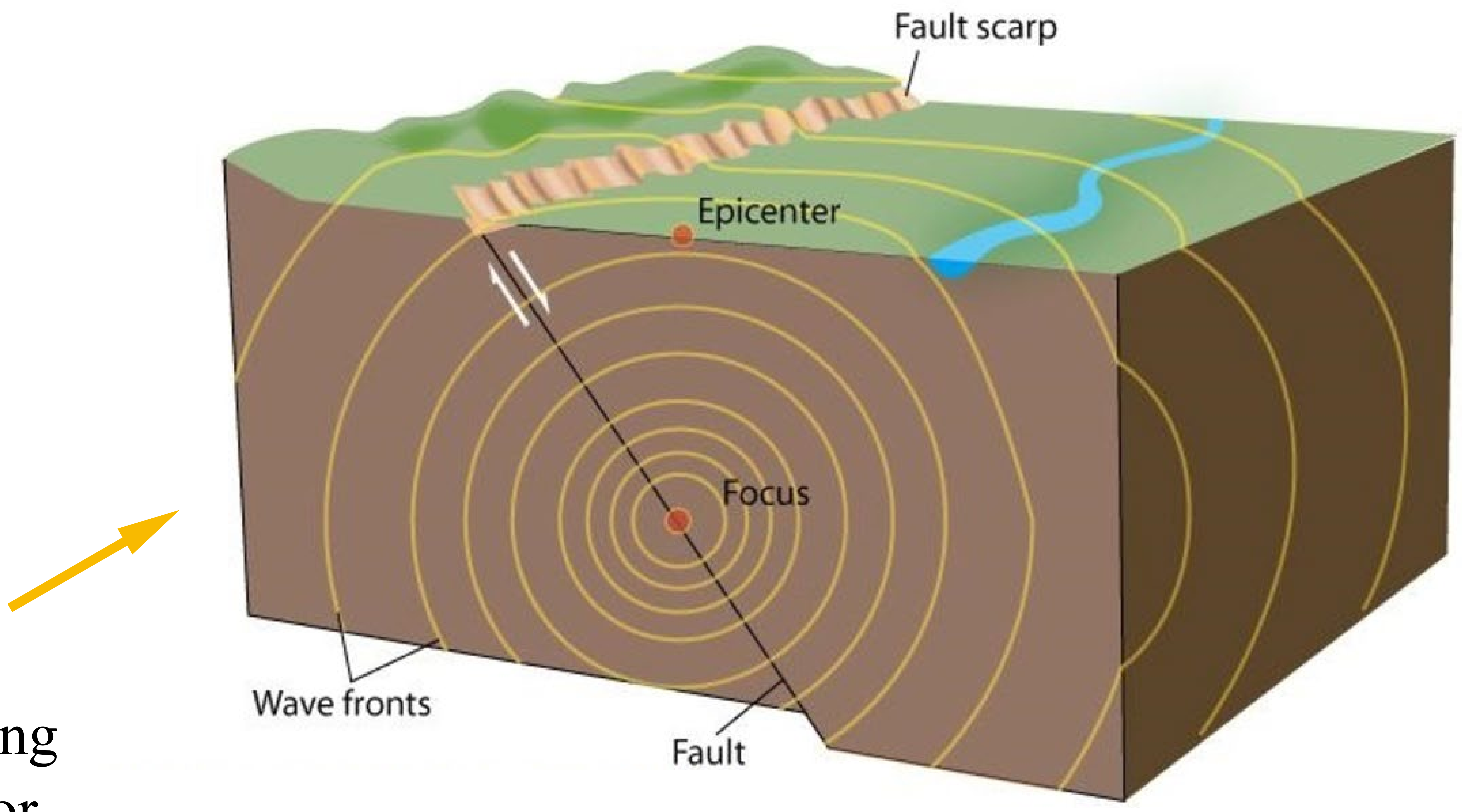
What's going on here?

- Wave-speeds in the Earth: $a = 5$ km/s

- Courant-Friederichs-Lewy (CFL, 1928) condition:

$$a\frac{\Delta t}{\Delta x} \leq 1$$

~ 5 km/s

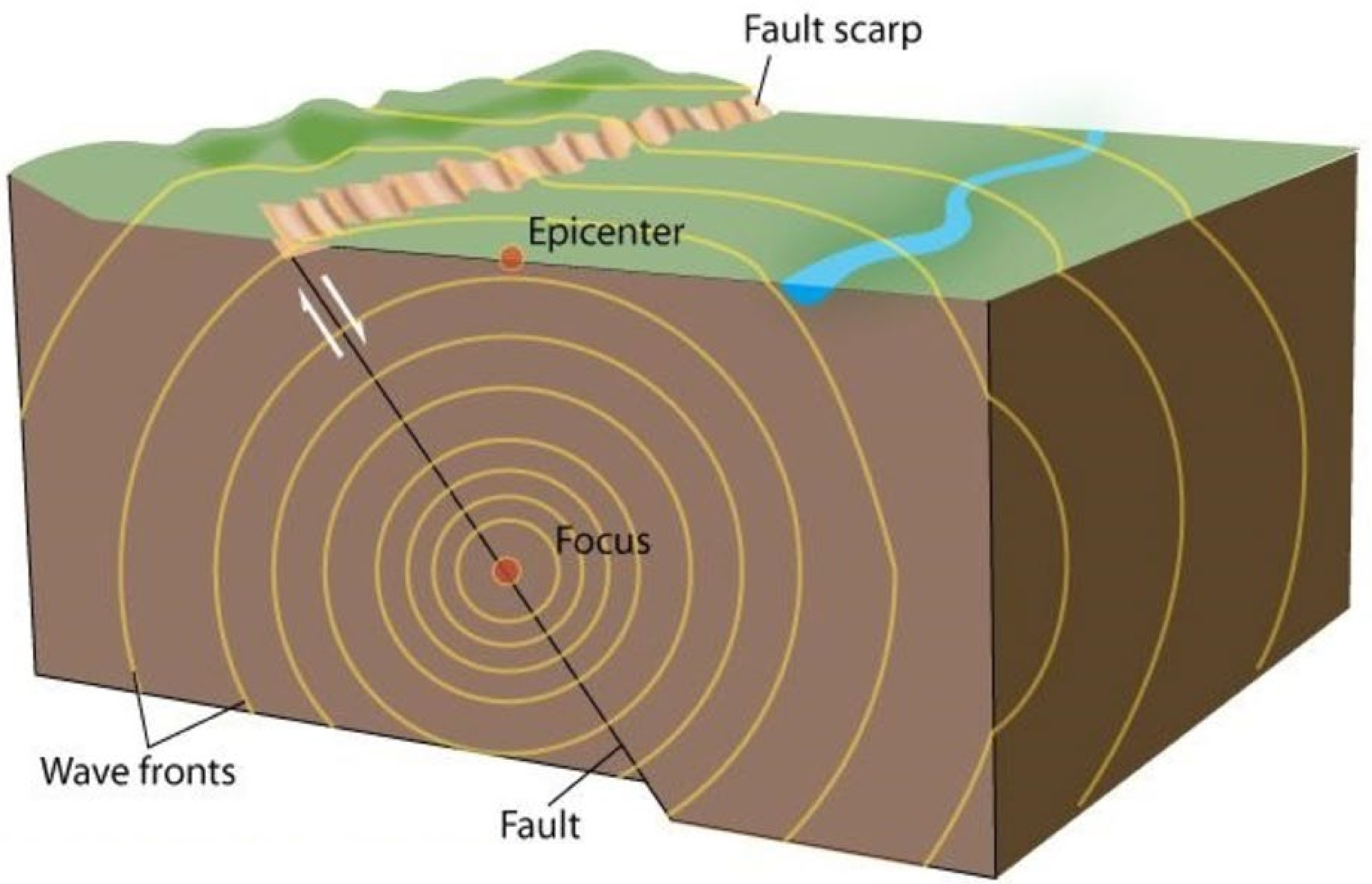


- Typical grid resolution is $\Delta x \approx 10$ m, requiring $\Delta t \approx 10^{-6}$ seconds for numerical stability (for explicit methods). Makes it hard to adapt methods for dynamic rupture for the whole earthquake cycle.

Main computational challenges

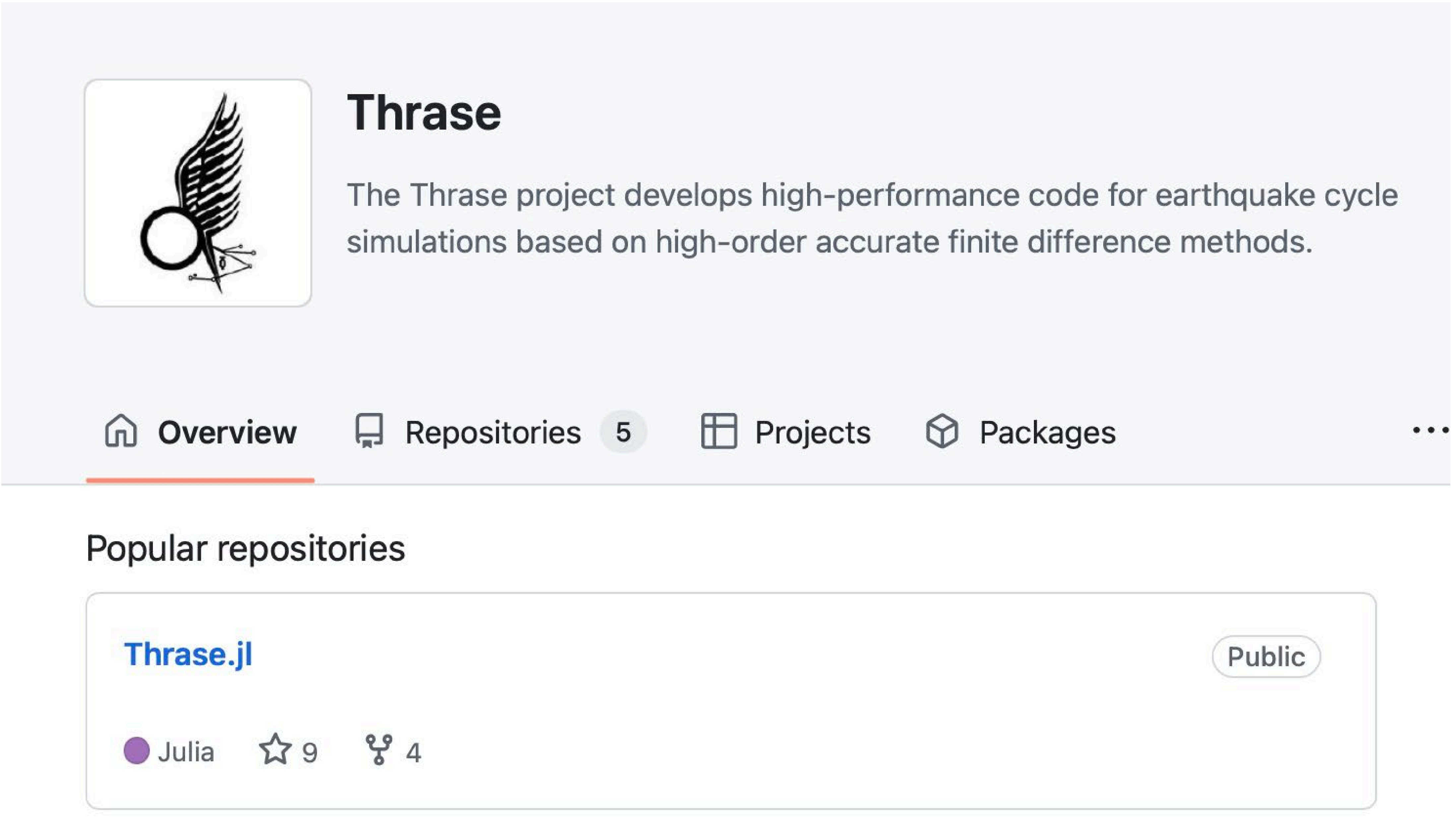
- Multiple scales in space/time -> huge problems in linear algebra
- Nonlinearities from friction, rheology etc. can lead to numerical stiffness and more -> need novel time-stepping and optimization algorithms
- How to load faults?
- Boundary/interface conditions must be implemented in a stable way
- Complex fault geometries, material heterogeneities demand unstructured meshes, with hp-refinement

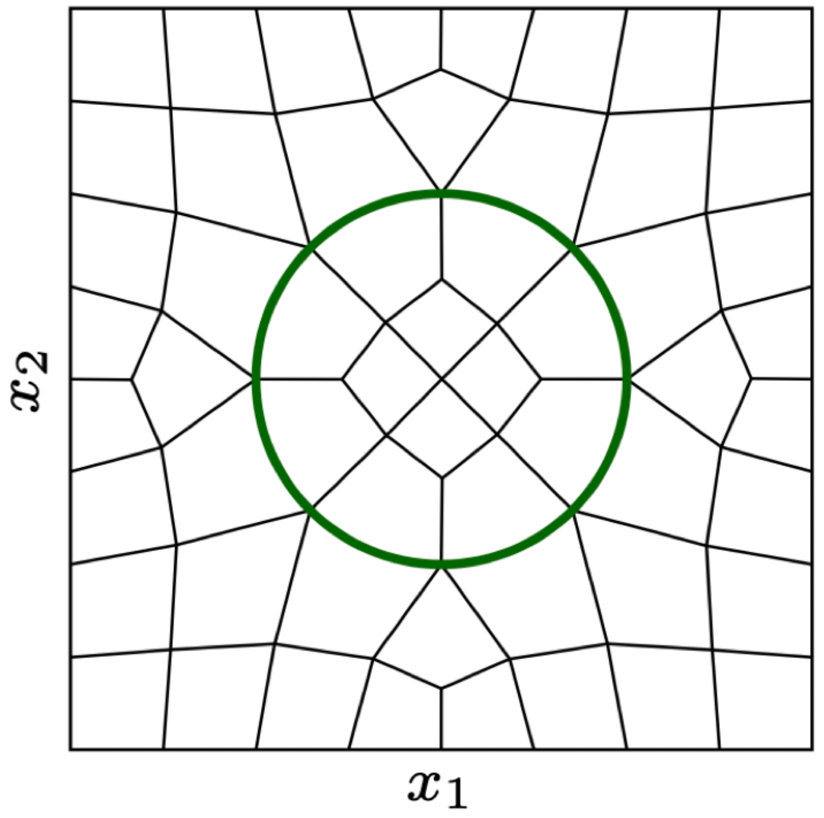
~ 5 km/s

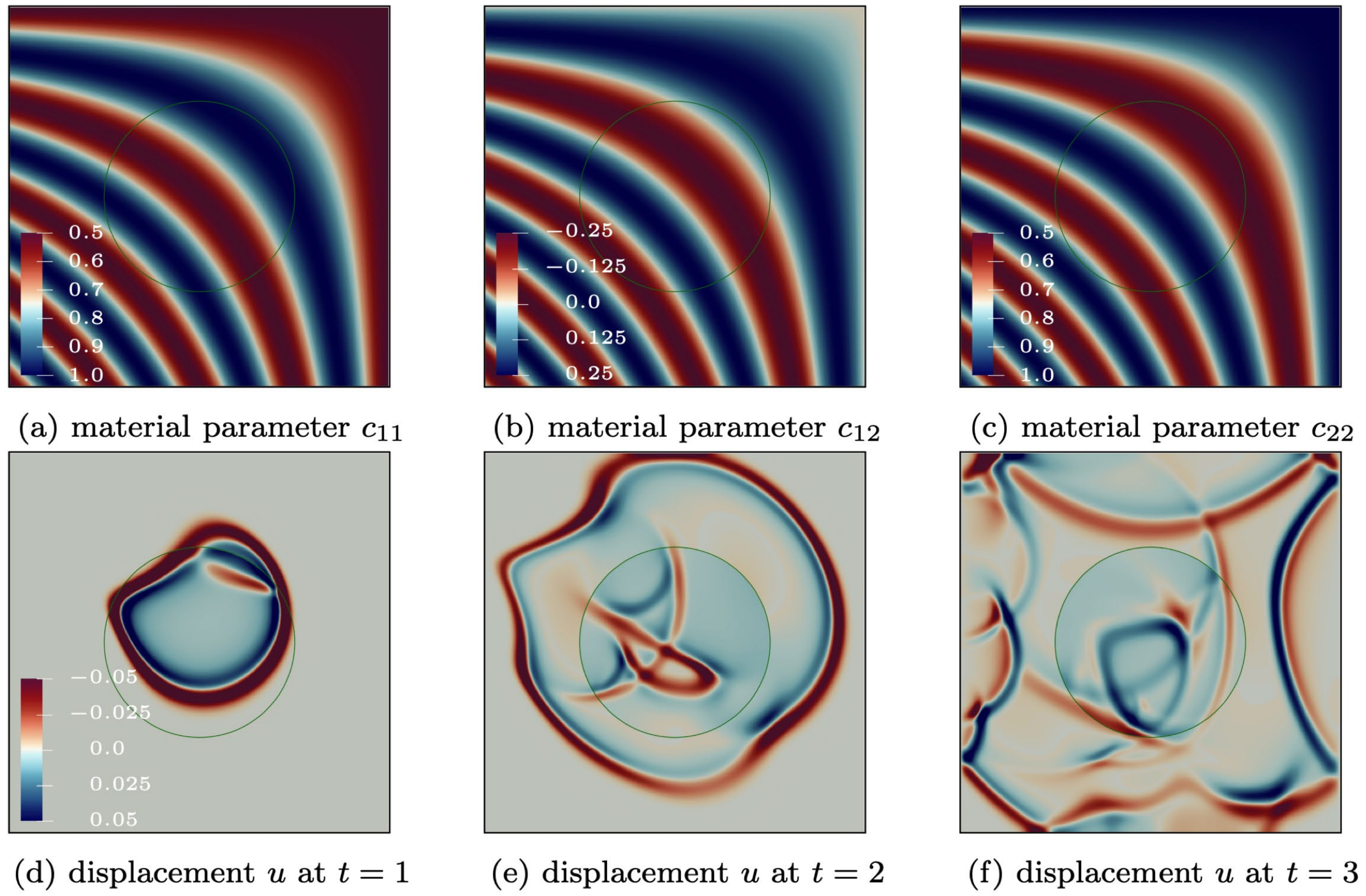


Some highlights on our group's contributions

- Thrase is based on high-order accurate SBP-SAT finite difference methods (similar to certain types of FEM)
  - Can account for complex geometries, unstructured meshes, material/frictional heterogeneities, bulk inelasticity, etc.
- We targeted the earthquake cycle with full dynamics
- Developed robust numerical methods for dynamic rupture simulations, connect to interseismic solver & *switch*.
  - Non-stiff with rate-and-state friction (Erickson et al., 2022)
  - Explore material heterogeneities (sedimentary basins) with full dynamics (Harvey et al., 2022)

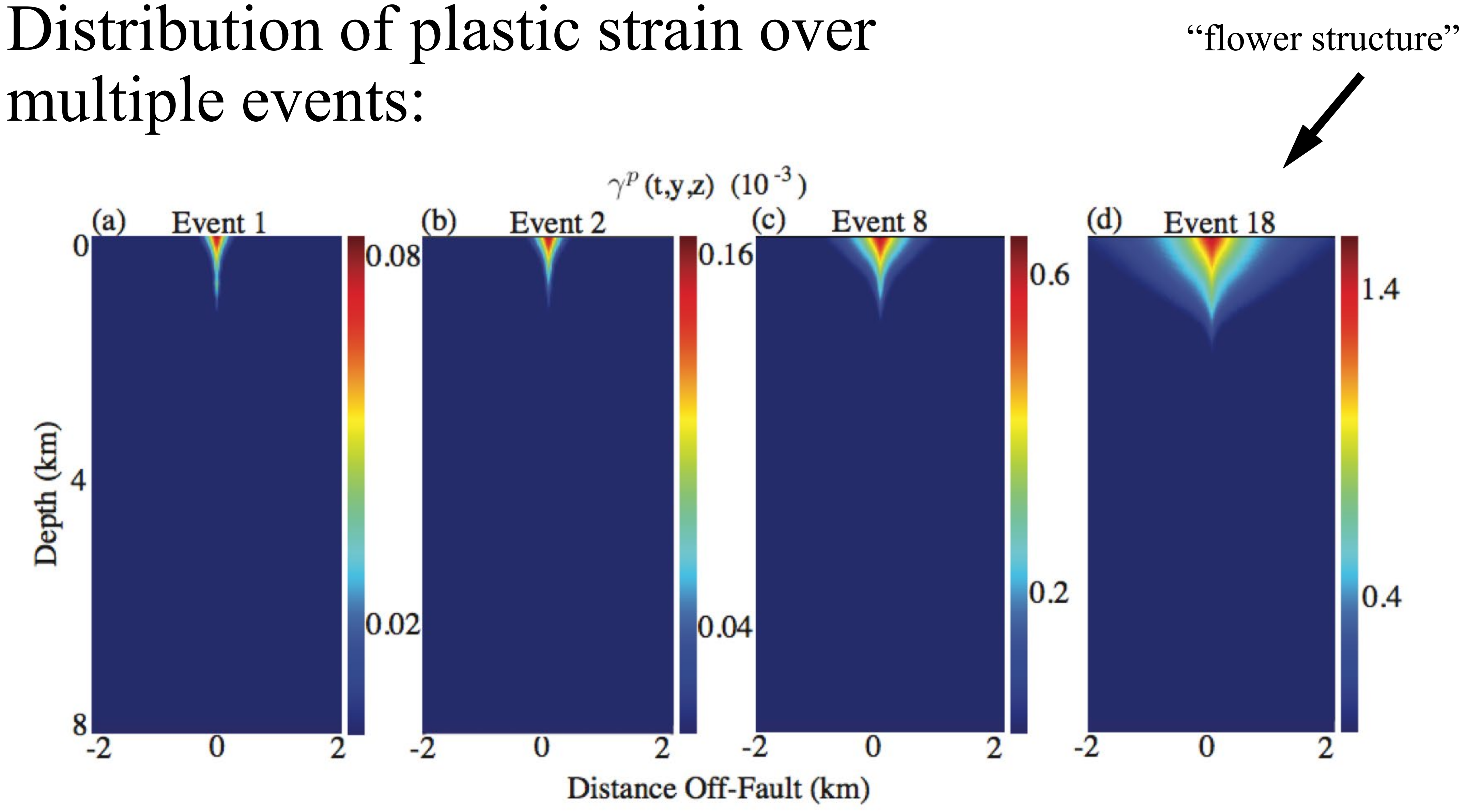






(Erickson et al., JSC, 2022)

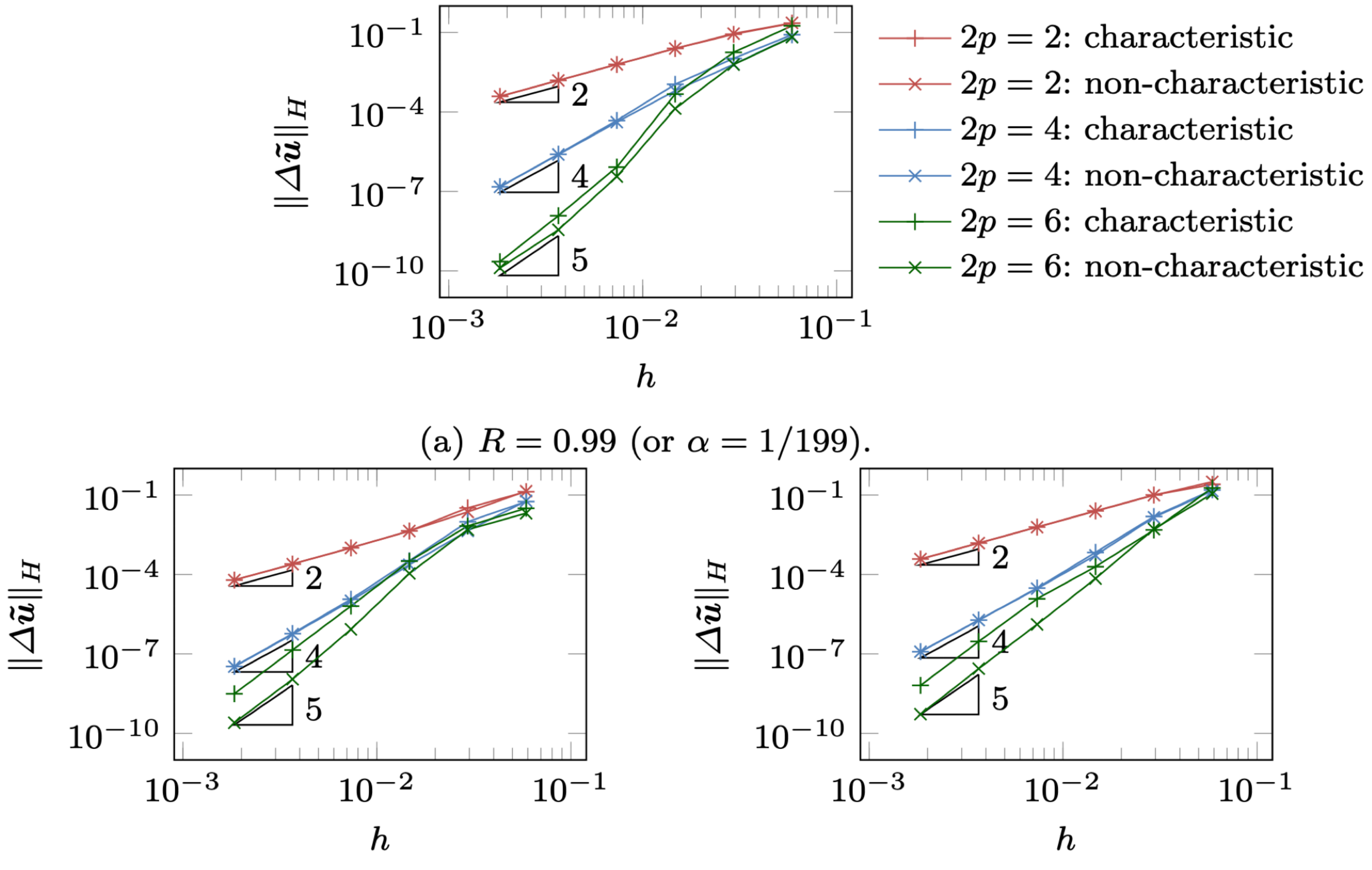
Distribution of plastic strain over multiple events:



Erickson et al. (2017)

Thrase is built on FD methods that impose BC in a characteristic manner:

- Removal of stiffness enables any explicit time-stepping scheme; otherwise need to develop (specialized) implicit schemes
- Numerical stability and convergence
- We opt to use a low-storage, 4th order RK scheme during coseismic phases

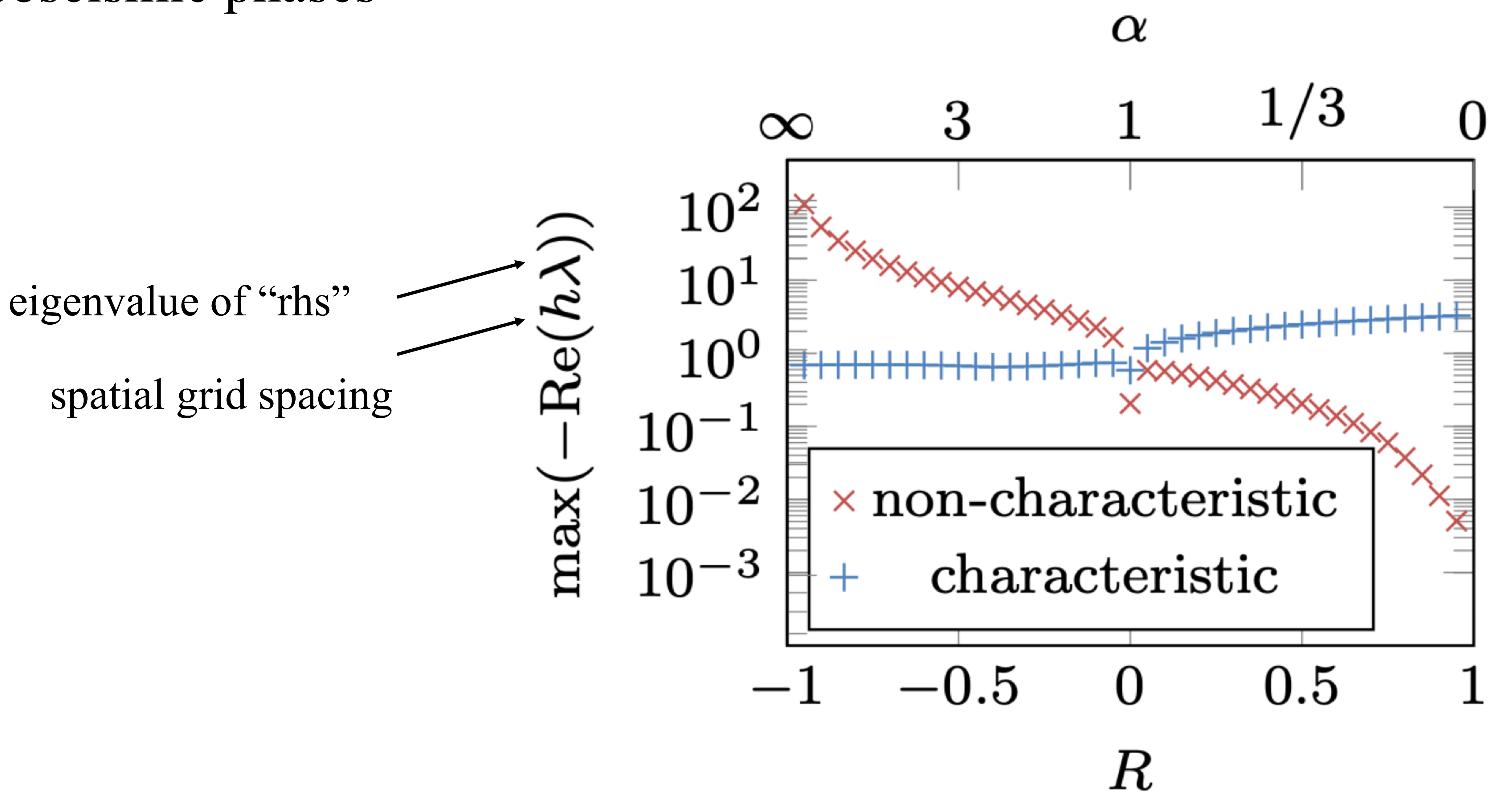


(a) $R = 0.99$ (or $\alpha = 1/199$).

(b) $R = 0$ (or $\alpha = 1$).

(c) $R = -0.99$ (or $\alpha = 199$).

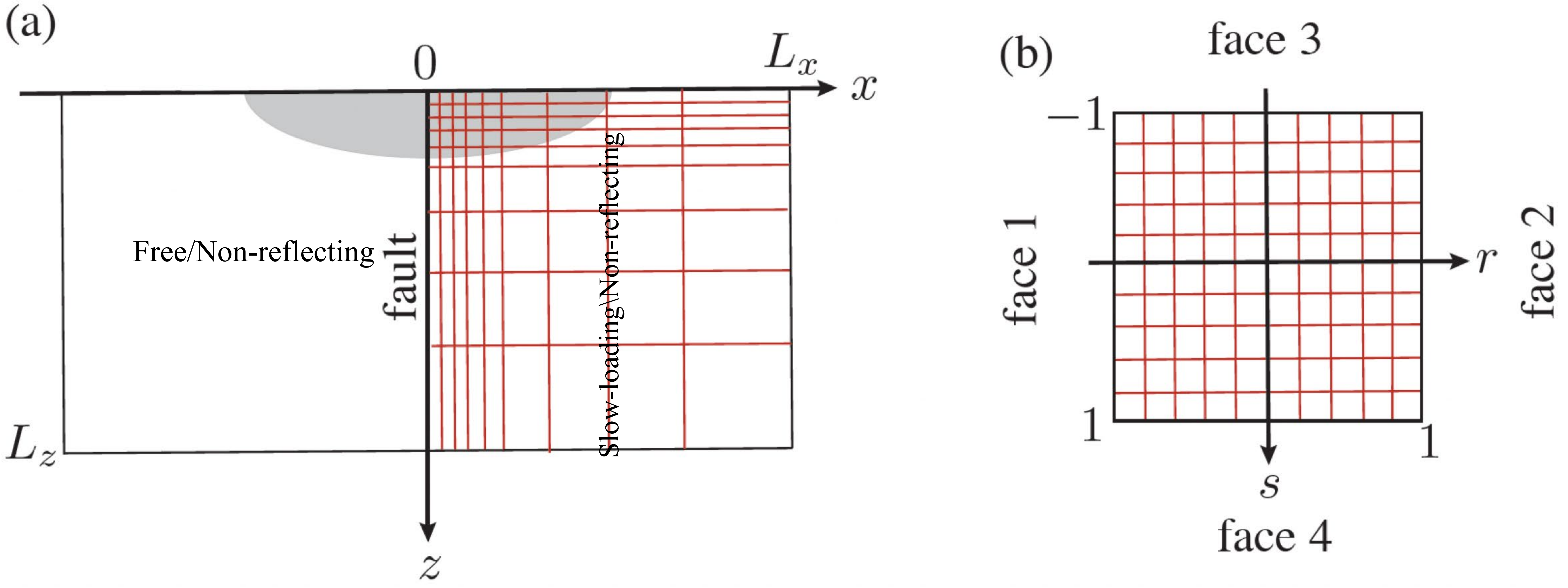
Up to 5th order spatial accuracy achieved for characteristic boundary treatment, over a range of α, where $F(V) = \alpha V$ is linearized friction



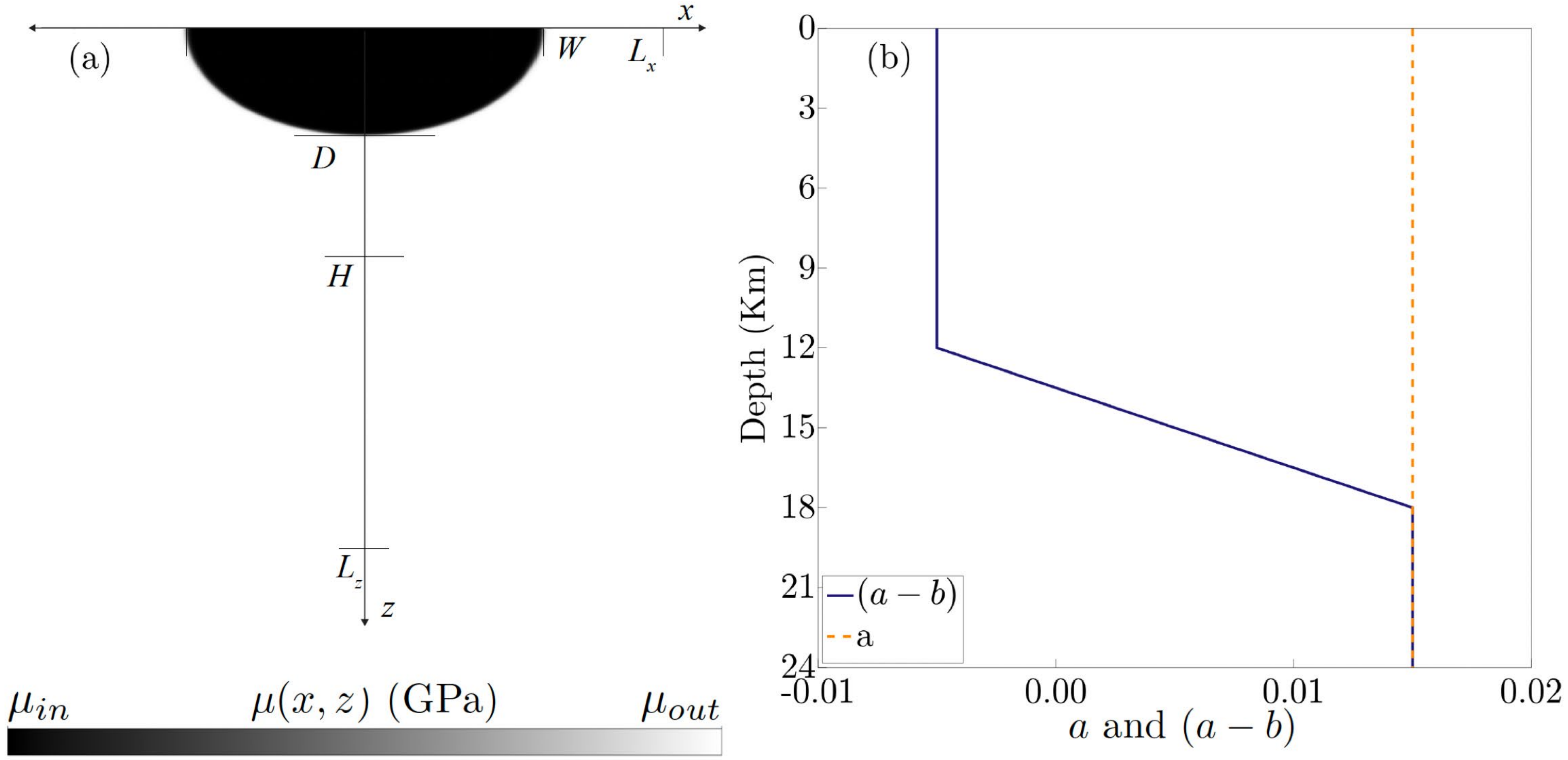
Generalized boundary conditions specified by reflection coefficient R. Non-characteristic method becomes stiff as $\alpha \to \infty$, where $F(V) = \alpha V$ is linearized friction

Effects of sediments

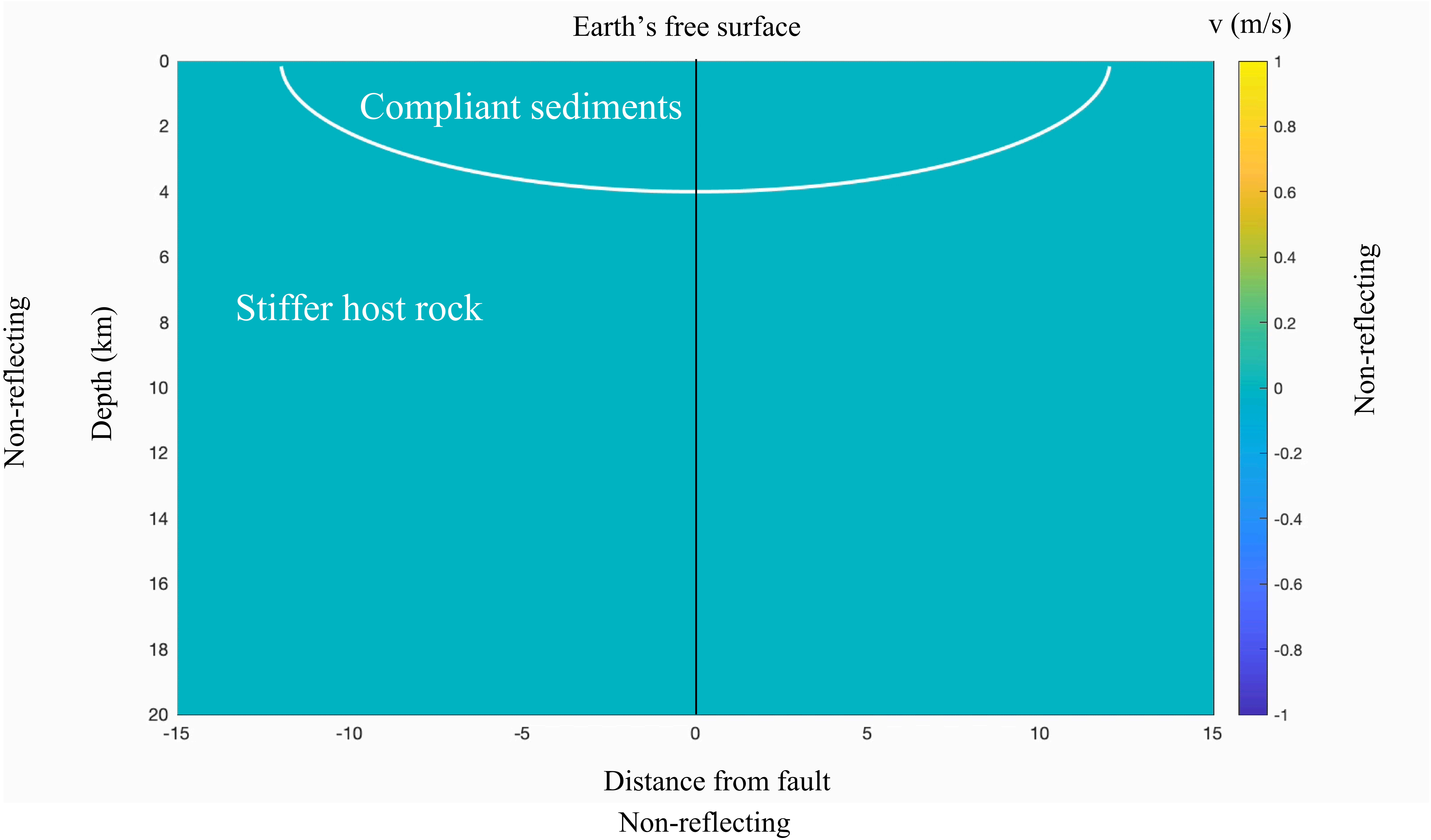
- Revisit the quasi-dynamic simulations of Erickson and Dunham (2014)
- 2D antiplane, vertical strike-slip fault
- Grid stretching - physical domain (a) mapped to logical space (b)



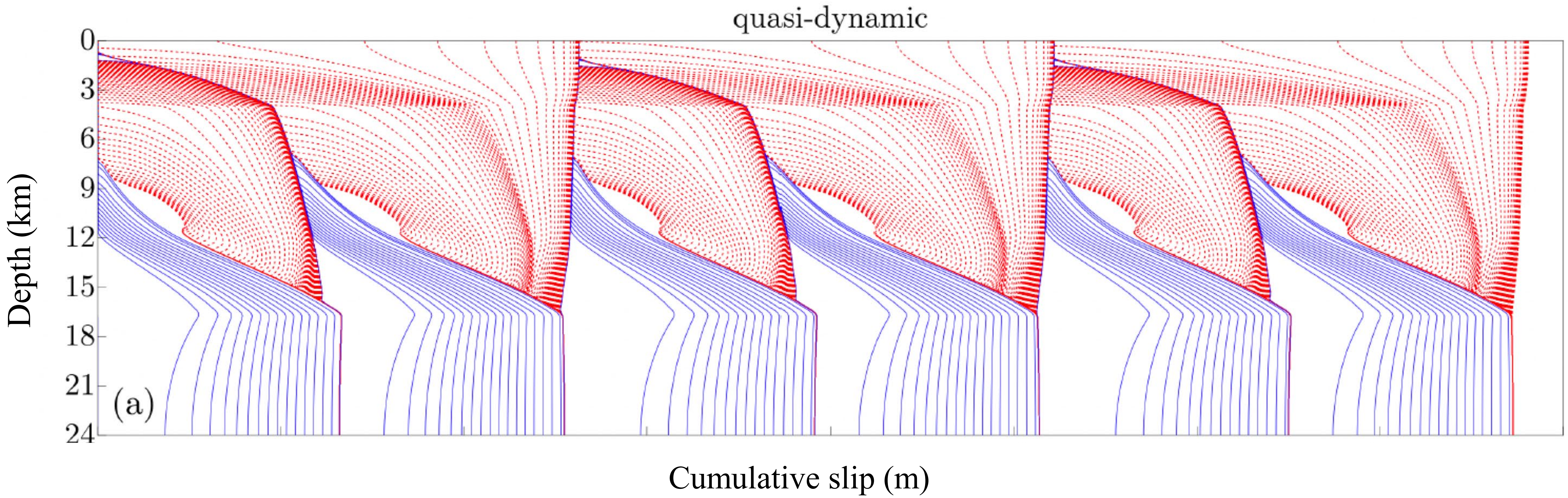
- (a) sedimentary basin of depth D (4 km here), decreased shear modulus within basin
- (b) Rate-and-state fault with depth-variable parameters, aging law
- Incorporate a switch to a fully-dynamic solver once an event nucleates



Effects of sediments during the coseismic phase

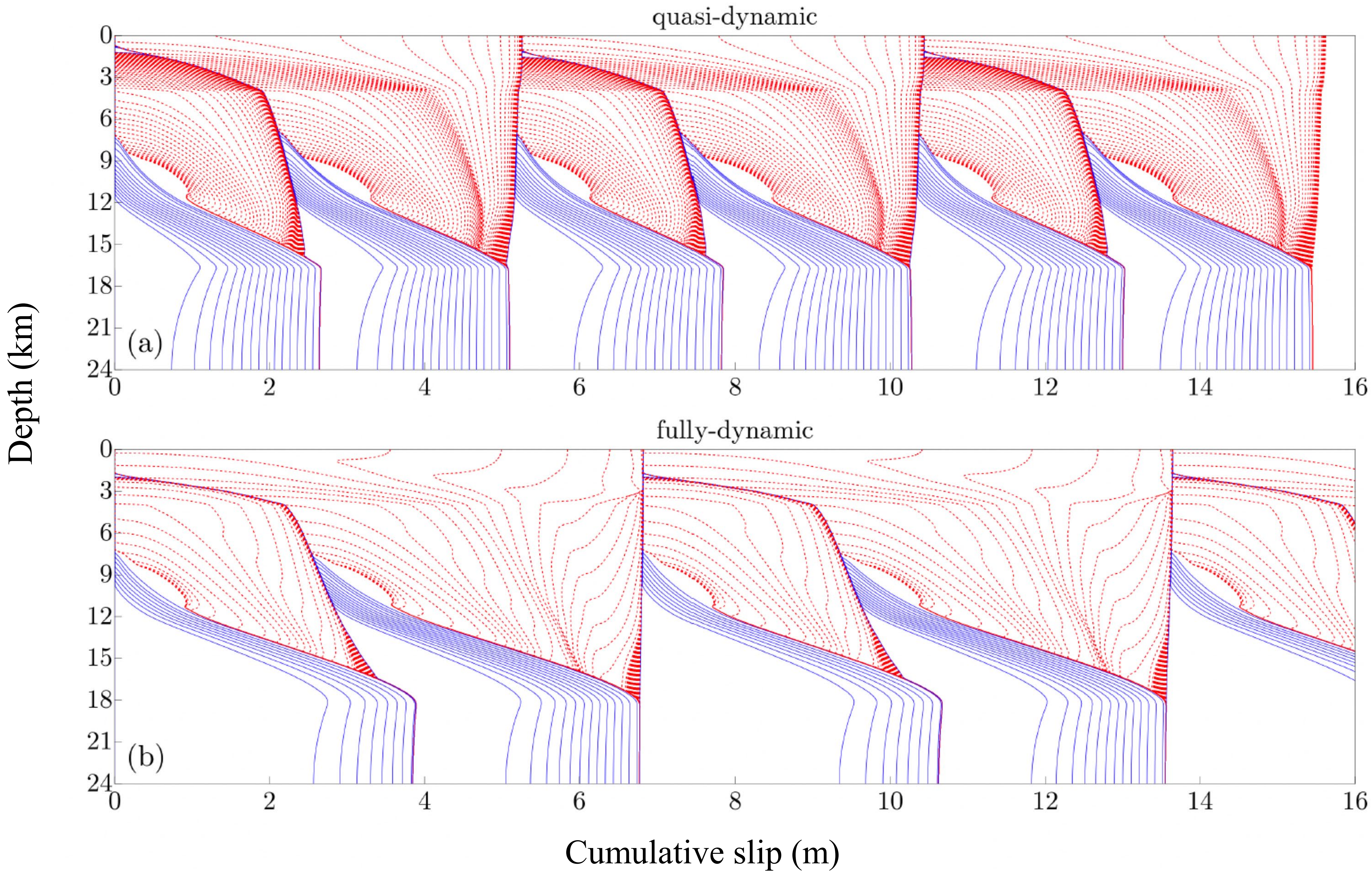


Revisit the study of Erickson and Dunham (2014) to explore effects of sediments with full dynamics (FD)



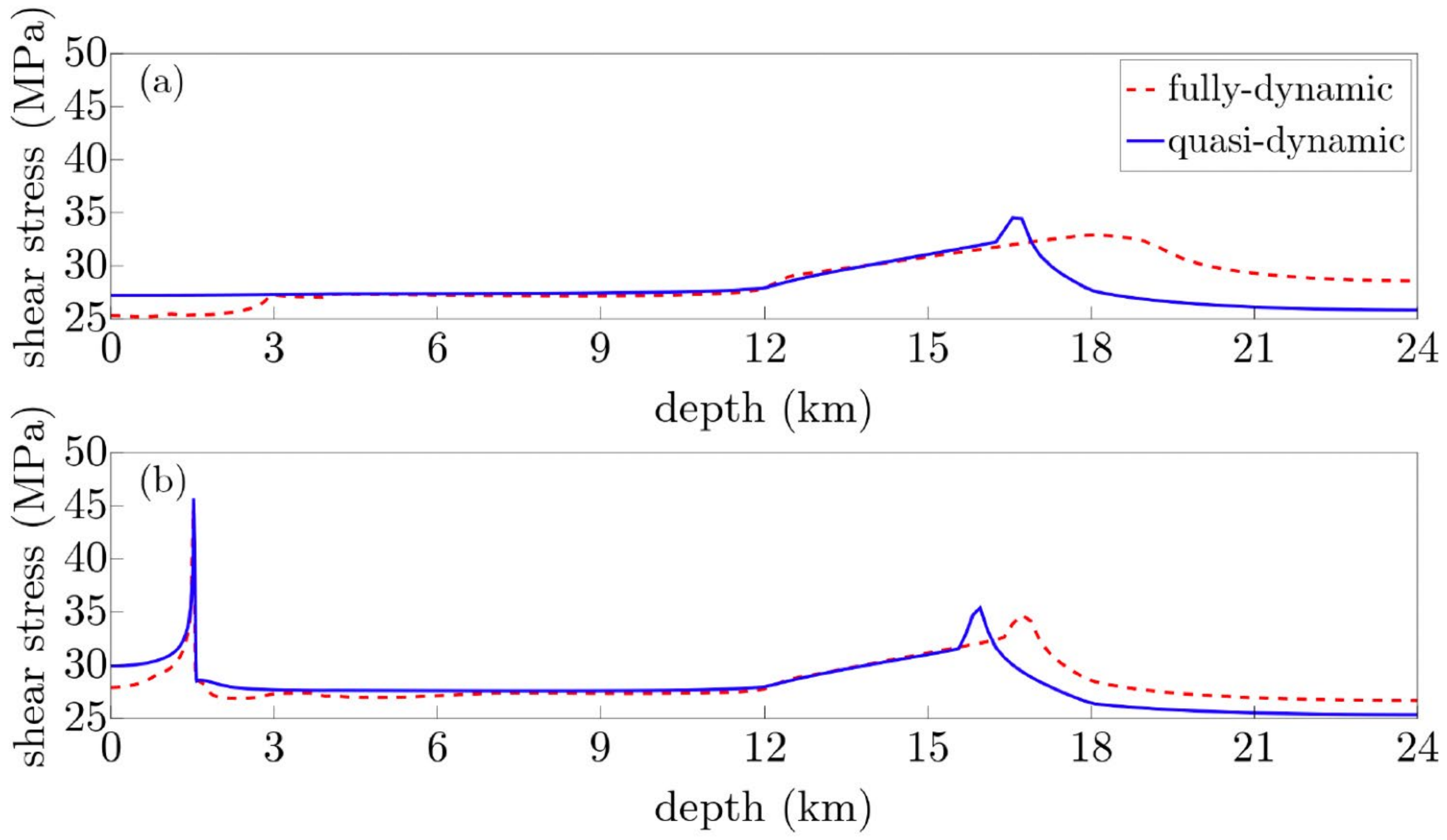
- Basin depth D = 4 km, quasi-dynamic only
- Sub-basin events emerge for sufficiently deep or compliant basins - leading to an alternating sequence with surface rupturing events
- insight into possible shallow-slip deficit - observations of sub-basin events just a precursor and shallow slip accommodated by future surface-rupturing event?
- Possibility that full dynamics would correspond entirely to surface rupturing events? (Full dynamics capable of penetrating through basin in isolated events…)

Revisit the study of Erickson and Dunham (2014) to explore effects of sediments with full dynamics (FD)



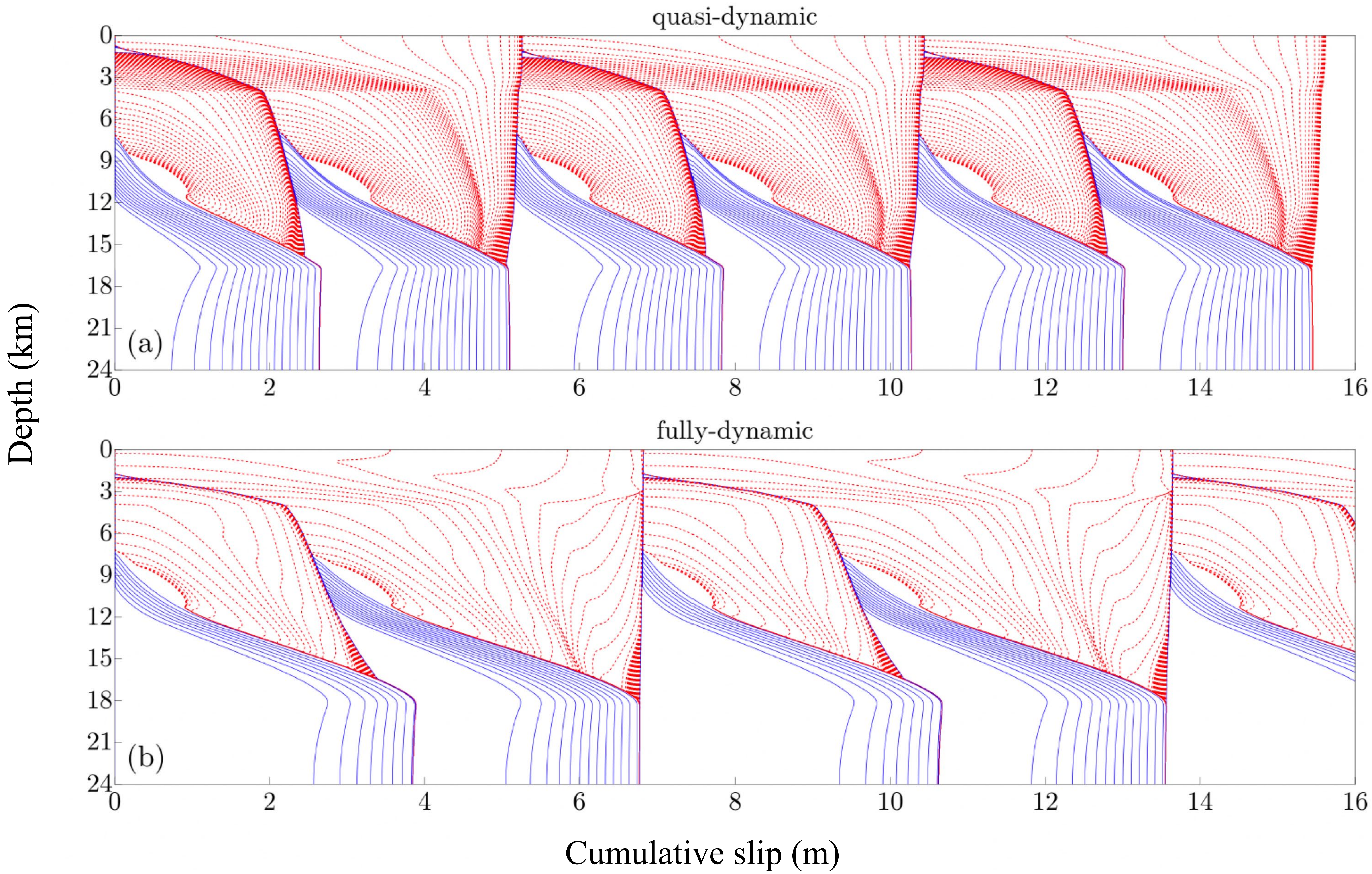
Shear stress

- following a surface rupturing event: lower in basin for FD (a, below)
- Following sub-basin rupture leaves behind a stress concentration that promotes subsequent rupture



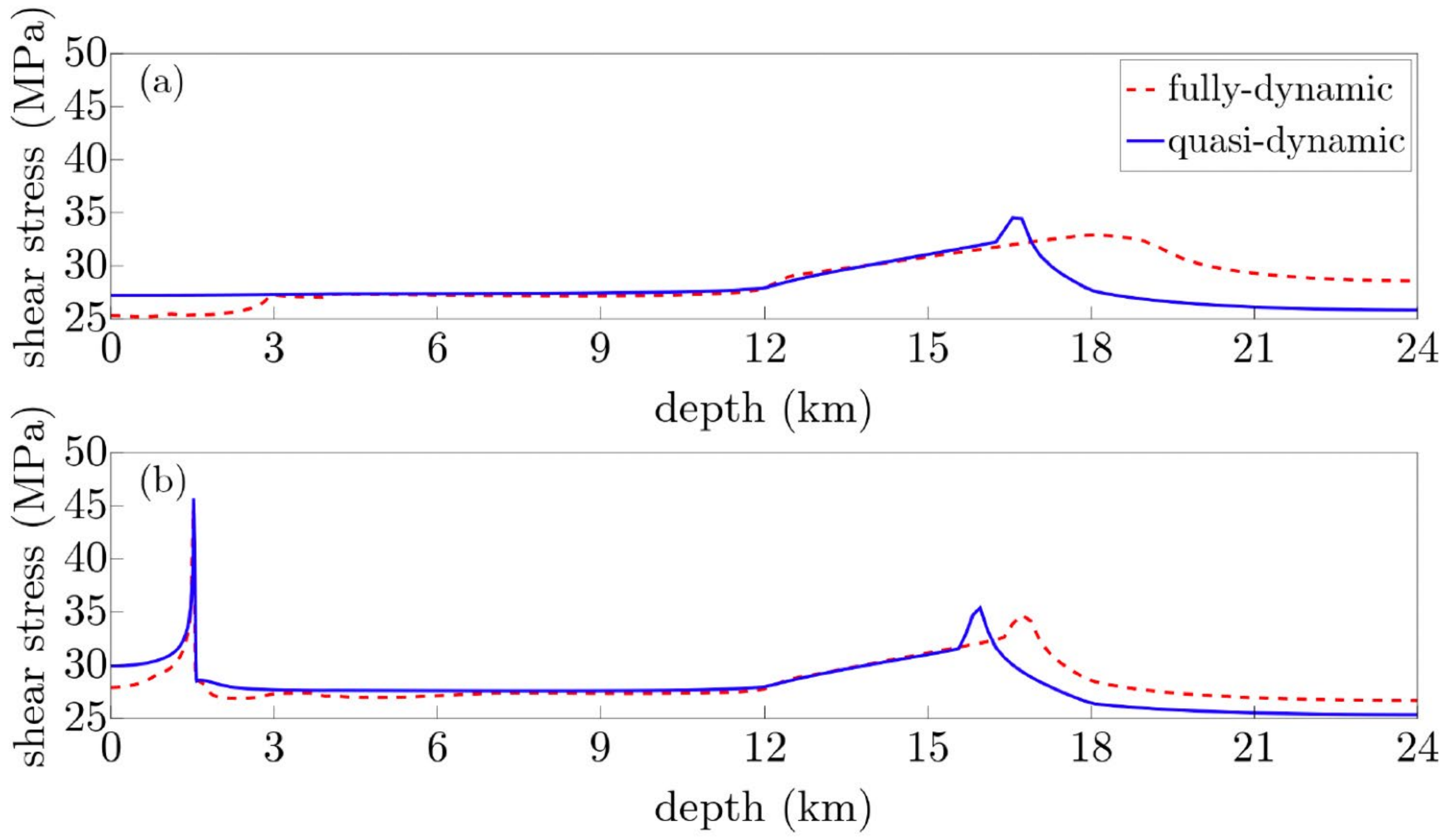
- FD sequences still host alternating sequences of sub-basin/surface-rupturing events
- FD events generate higher slip rates (more slip) and rupture speeds

Revisit the study of Erickson and Dunham (2014) to explore effects of sediments with full dynamics (FD)



Shear stress

- following a surface rupturing event: lower in basin for FD (a, below)
- Following sub-basin rupture leaves behind a stress concentration that promotes subsequent rupture

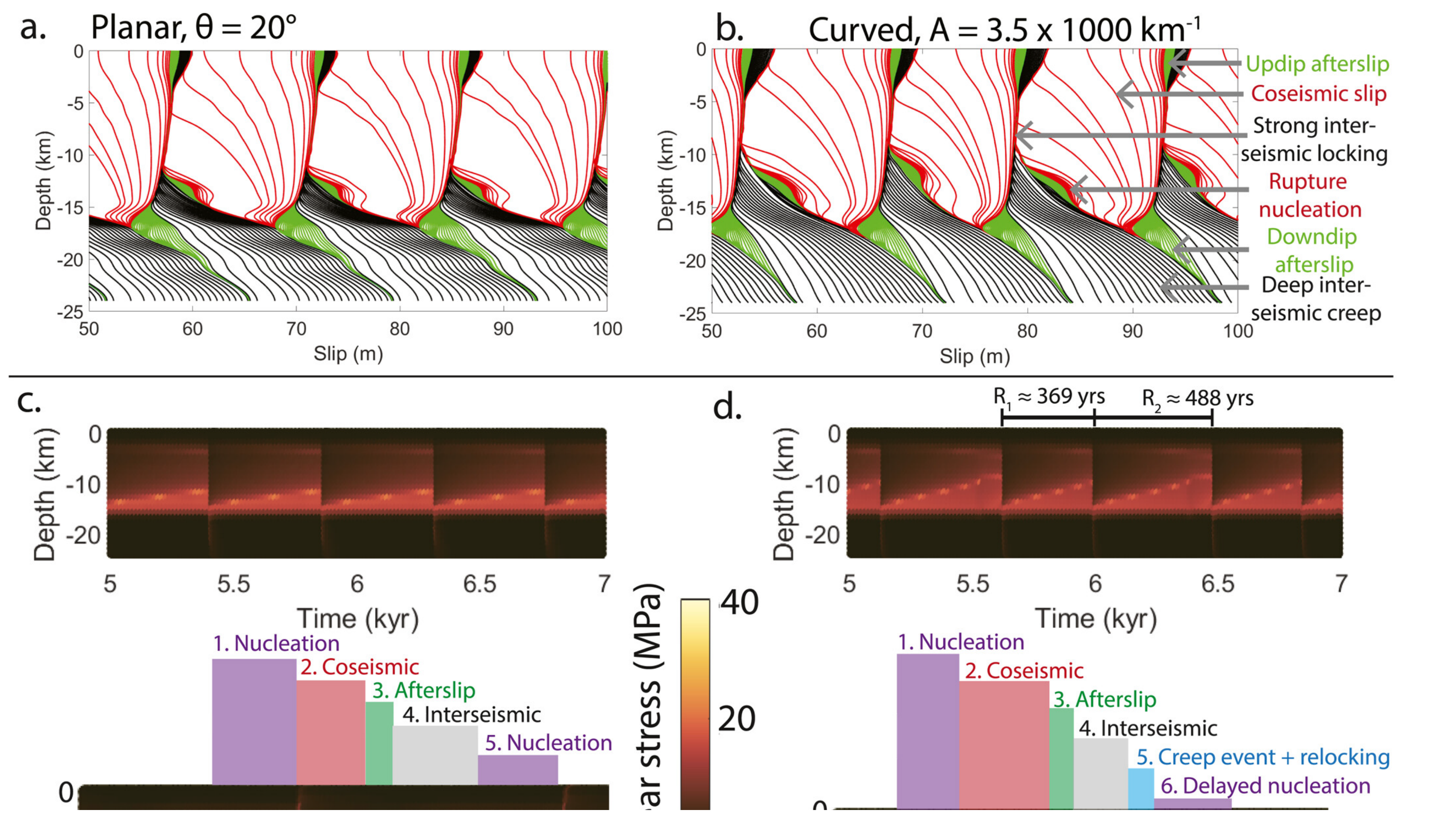


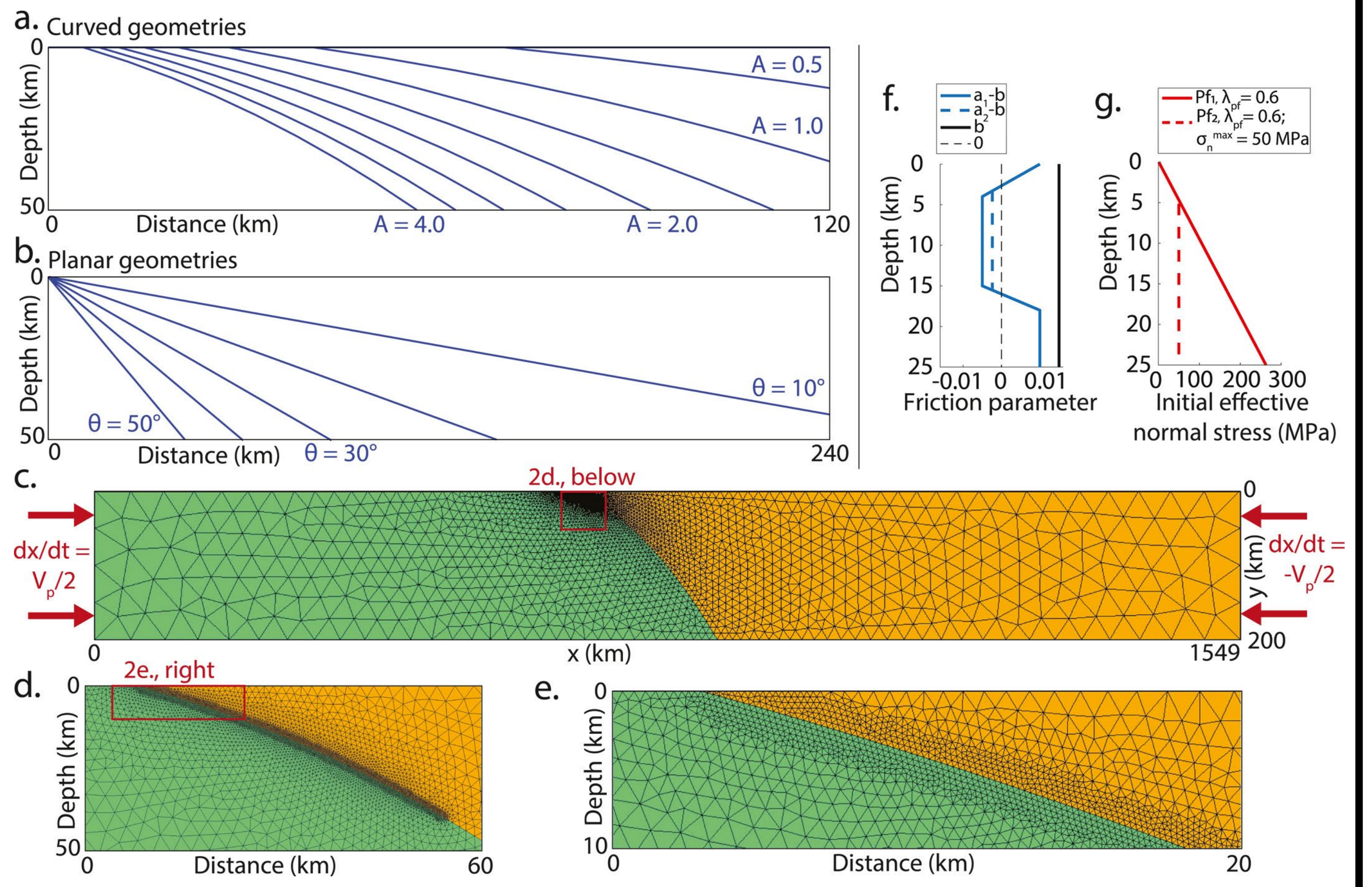
- FD sequences still host alternating sequences of sub-basin/surface-rupturing events
- FD events generate higher slip rates (more slip) and rupture speeds

Why are these periodic??
How to adapt for Cascadia?

Other Volume Based Method Targeting (Subduction Zone) Cycles:

2D subduction zone cycles using Tandem (Uphoff et al., 2023)

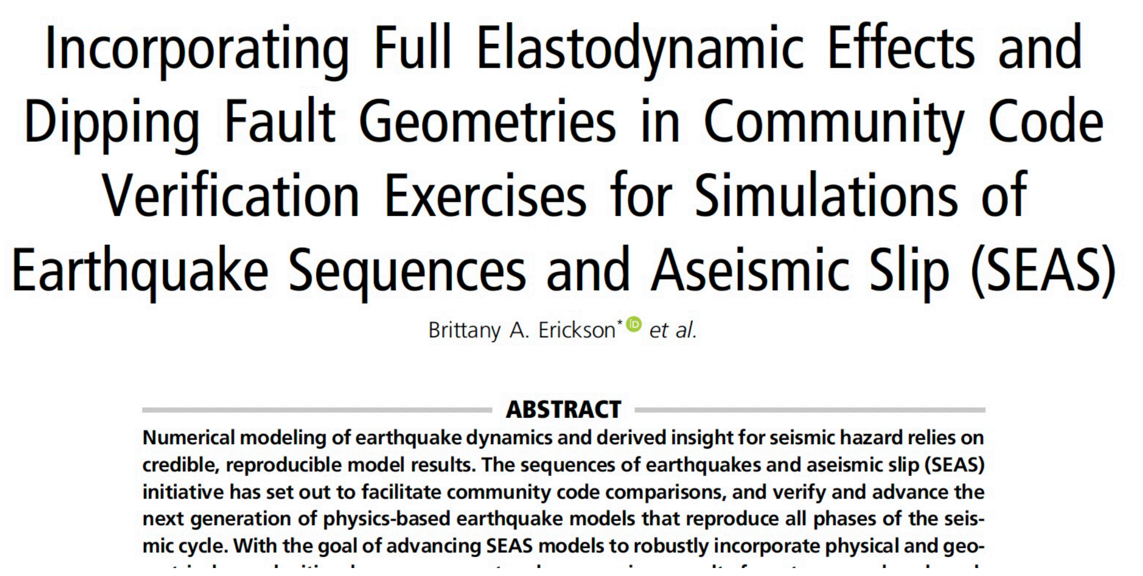




Magnitudes may be primarily controlled by the dip and dimensions of the seismogenic zone, while second-order effects from heterogeneity modulate the recurrence frequency and timing.

Biemiller et al. (2024)

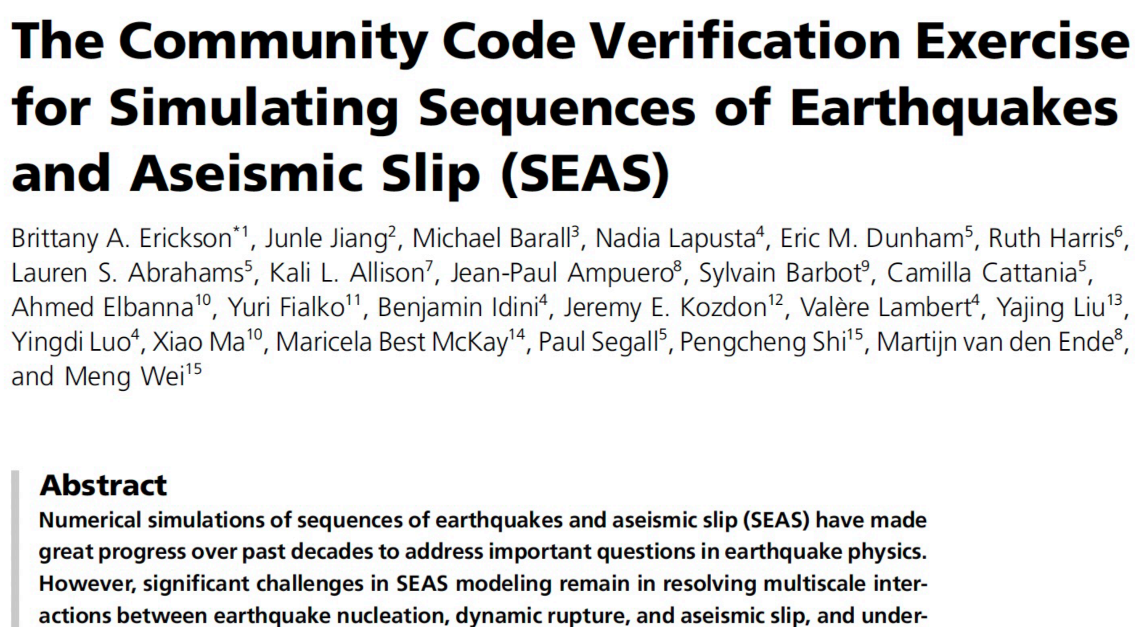
SEAS Community Benchmark Verification Exercises:


Incorporating Full Elastodynamic Effects and Dipping Fault Geometries in Community Code Verification Exercises for Simulations of Earthquake Sequences and Aseismic Slip (SEAS)

Brittany A. Erickson* et al.

ABSTRACT

Numerical modeling of earthquake dynamics and derived insight for seismic hazard relies on credible, reproducible model results. The sequences of earthquakes and aseismic slip (SEAS) initiative has set out to facilitate community code comparisons, and verify and advance the next generation of physics-based earthquake models that reproduce all phases of the seismic cycle. With the goal of advancing SEAS models to robustly incorporate physical and geo-

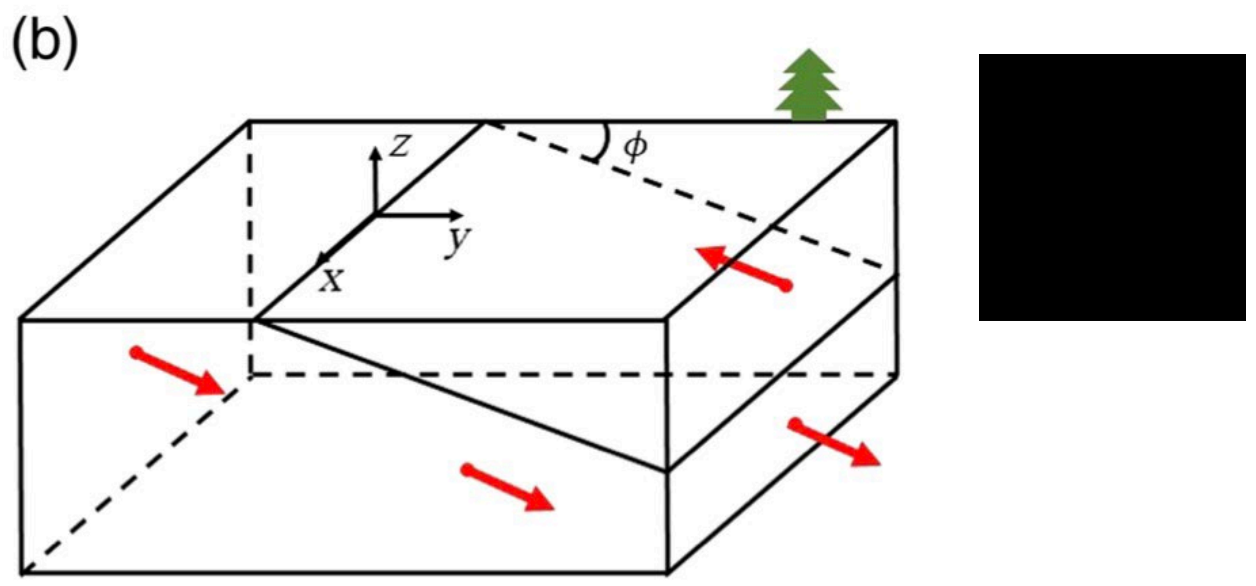

The Community Code Verification Exercise for Simulating Sequences of Earthquakes and Aseismic Slip (SEAS)

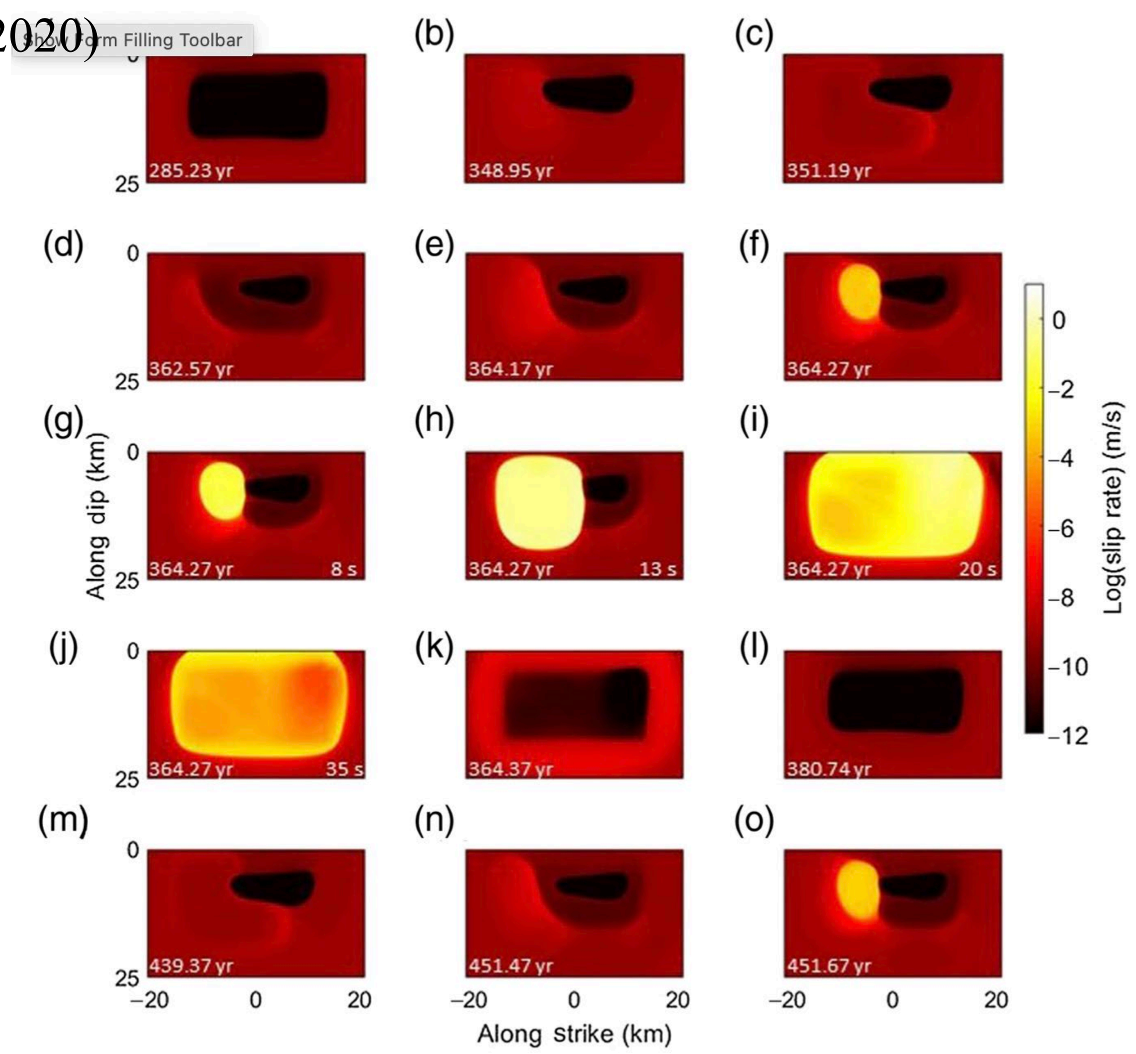
Brittany A. Erickson*[1], Junle Jiang[2], Michael Barall[3], Nadia Lapusta[4], Eric M. Dunham[5], Ruth Harris[6], Lauren S. Abrahams[5], Kali L. Allison[7], Jean-Paul Ampuero[8], Sylvain Barbot[9], Camilla Cattania[5], Ahmed Elbanna[10], Yuri Fialko[11], Benjamin Idini[4], Jeremy E. Kozdon[12], Valère Lambert[4], Yajing Liu[13], Yingdi Luo[4], Xiao Ma[10], Maricela Best McKay[14], Paul Segall[5], Pengcheng Shi[15], Martijn van den Ende[8], and Meng Wei[15]

Abstract

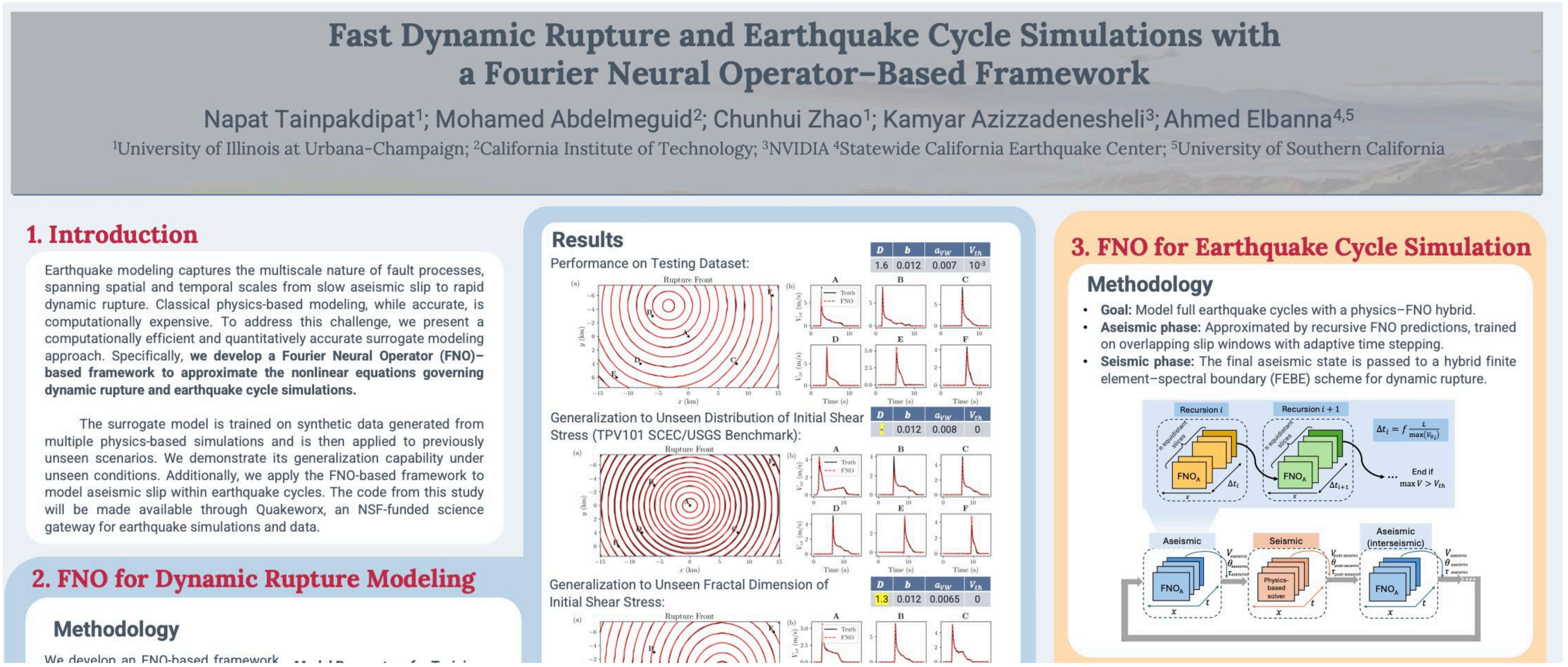
Numerical simulations of sequences of earthquakes and aseismic slip (SEAS) have made great progress over past decades to address important questions in earthquake physics. However, significant challenges in SEAS modeling remain in resolving multiscale interactions between earthquake nucleation, dynamic rupture, and aseismic slip, and under-

3D dipping thrust-fault earthquake cycles using EQsimu (Luo et al., 2020)





Shallower dipping faults -> larger coseismic slip, longer recurrence times


Fast Dynamic Rupture and Earthquake Cycle Simulations with a Fourier Neural Operator–Based Framework

Napat Tainpakdipat[1]; Mohamed Abdelmeguid[2]; Chunhui Zhao[1]; Kamyar Azizzadenesheli[3]; Ahmed Elbanna[4,5]

[1]University of Illinois at Urbana-Champaign; [2]California Institute of Technology; [3]NVIDIA [4]Statewide California Earthquake Center; [5]University of Southern California

1. Introduction

Earthquake modeling captures the multiscale nature of fault processes, spanning spatial and temporal scales from slow aseismic slip to rapid dynamic rupture. Classical physics-based modeling, while accurate, is computationally expensive. To address this challenge, we propose a computationally efficient and quantitatively accurate surrogate modeling approach. Specifically, **we develop a Fourier Neural Operator (FNO)–based framework to approximate the nonlinear equations governing dynamic rupture and earthquake cycle simulations.**

The surrogate model is trained on synthetic data generated from multiple physics-based simulations and is then applied to previously unseen scenarios. We demonstrate its generalization capability under unseen conditions. Additionally, we apply the FNO-based framework to model aseismic slip within earthquake cycles. The code from this study will be made available through Quakeworx, an NSF-funded science gateway for earthquake simulations and data.

2. FNO for Dynamic Rupture Modeling

Methodology

Results

Performance on Testing Dataset:

Generalization to Unseen Distribution of Initial Shear Stress (TPV101 SCEC/USGS Benchmark):

Generalization to Unseen Fractal Dimension of Initial Shear Stress:

3. FNO for Earthquake Cycle Simulation

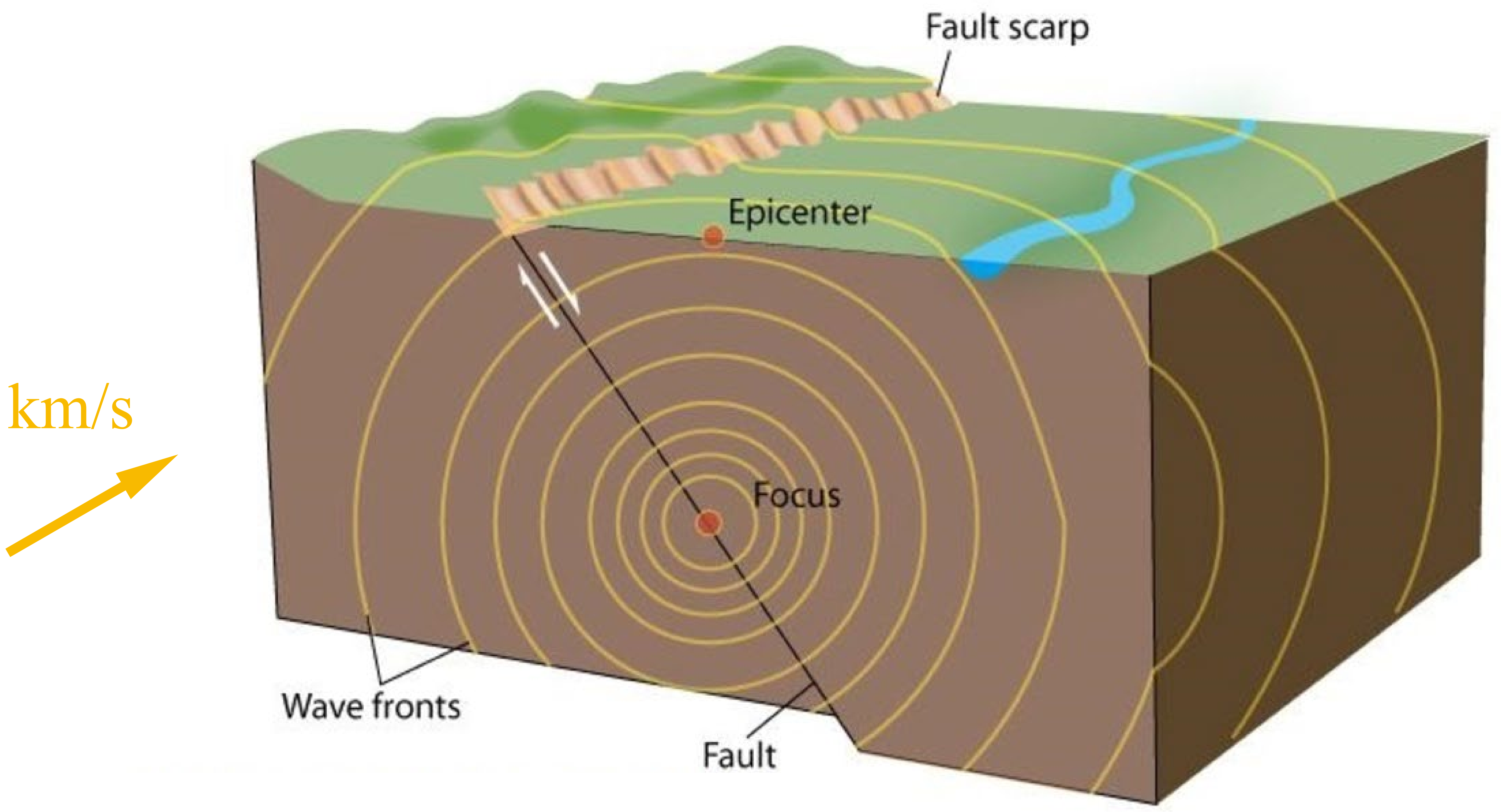
Methodology

- **Goal:** Model full earthquake cycles with a physics–FNO hybrid.
- **Aseismic phase:** Approximated by recursive FNO predictions, trained on overlapping slip windows with adaptive time stepping.
- **Seismic phase:** The final aseismic state is passed to a hybrid finite element–spectral boundary (FEBE) scheme for dynamic rupture.

Main computational challenges

- Multiple scales in space/time -> huge problems in linear algebra
- Nonlinearities from friction, rheology etc. can lead to numerical stiffness and more -> need novel time-stepping and optimization algorithms
- Boundary and interface conditions must be implemented in a stable way
- Complex fault geometries, material heterogeneities demand unstructured meshes, with hp-refinement

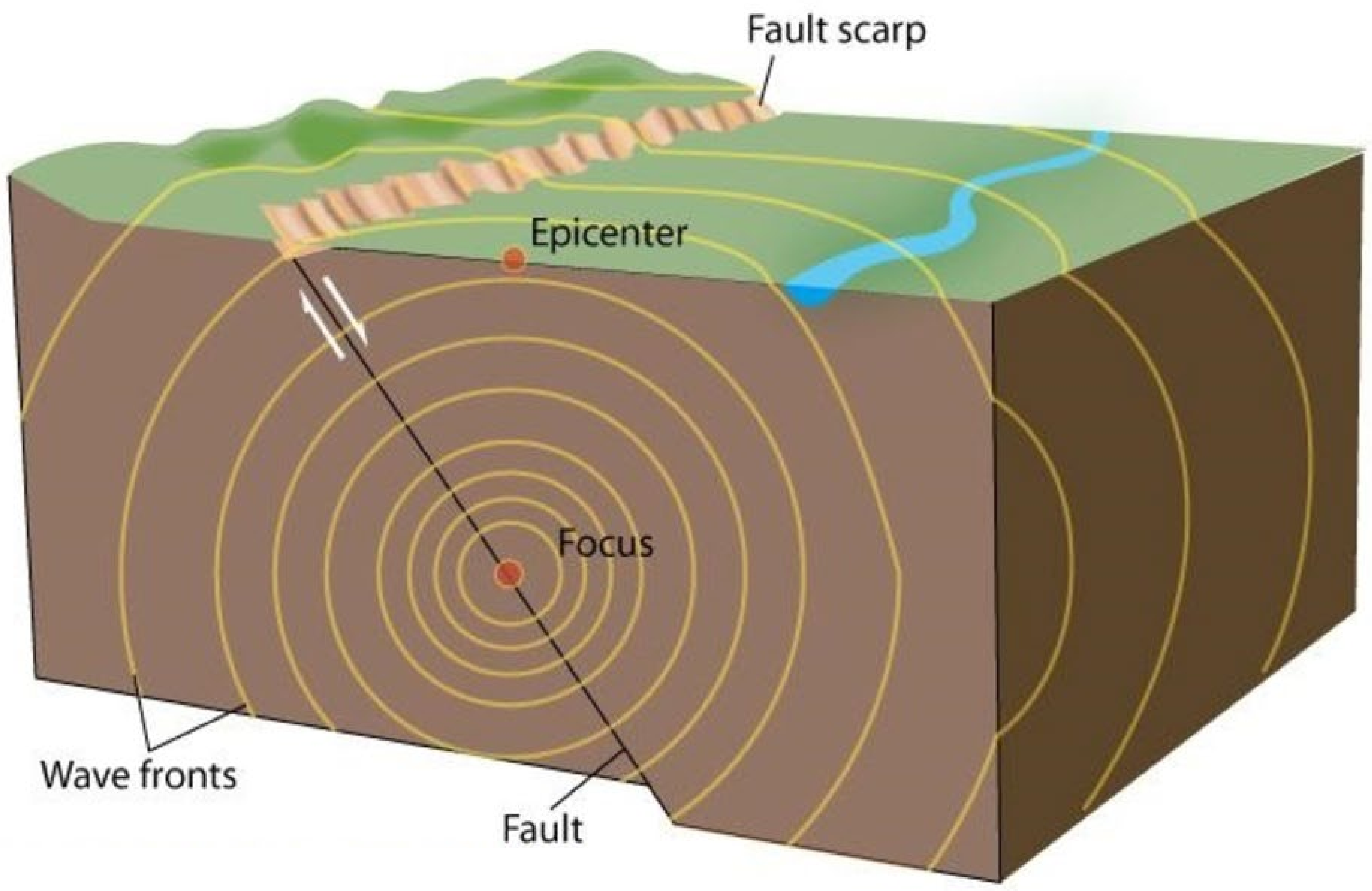
~ 5 km/s



Main computational challenges

- Multiple scales in space/time -> huge problems in linear algebra
- Nonlinearities from friction, rheology etc. can lead to numerical stiffness and more -> need novel time-stepping and optimization algorithms
- Boundary and interface conditions must be implemented in a stable way
- Complex fault geometries, material heterogeneities demand unstructured meshes, with hp-refinement
- **All of these issues pale in comparison to $Ax = b$.**

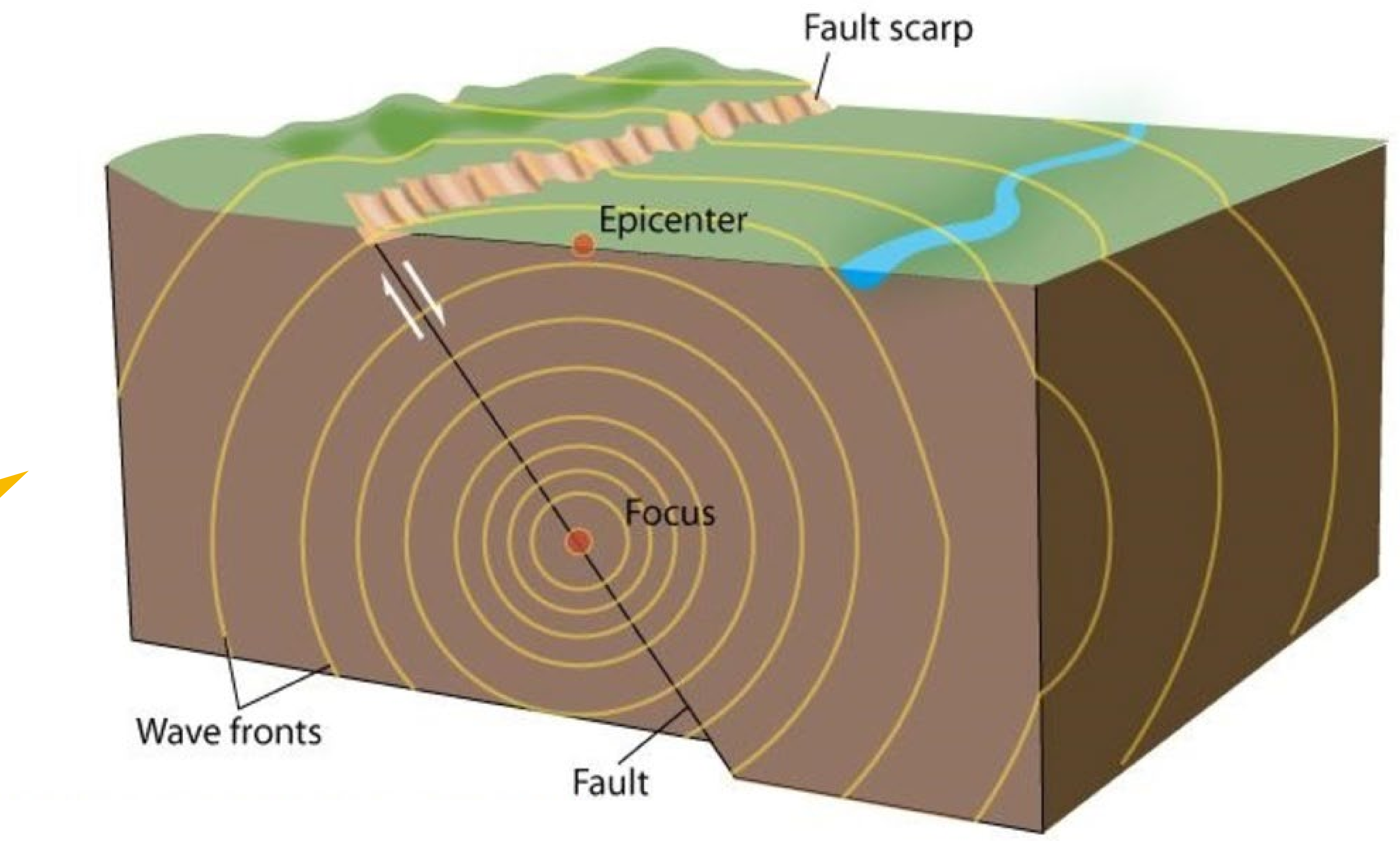
~ 5 km/s



Main computational challenges

- Multiple scales in space/time -> huge problems in linear algebra
- Nonlinearities from friction, rheology etc. can lead to numerical stiffness and more -> need novel time-stepping and optimization algorithms
- Boundary and interface conditions must be implemented in a stable way
- Complex fault geometries, material heterogeneities demand unstructured meshes, with hp-refinement
- **All of these issues pale in comparison to $Ax = b$.**

~ 5 km/s



Most of the work so far on earthquake cycle simulations has focused on smaller, 2D problems, with increasing physical complexity. *The real challenge is to return to 3D and address the linear solve.*

Geophysicists and HPC

"It is inspiring to recognize **the enduring influence of physicists in driving technological innovations** and ensuring the future progress of computational science.

This Perspective highlights roles physicists have in driving advances in high-performance computing (HPC), lest they be forgotten amid our celebrations of exascale and generative artificial intelligence…

As HPC hits walls of energy and storage, **will physicists in pursuit of their own applications again come to the fore with generalizable solutions**?"

nature reviews physics
https://doi.org/10.1038/s42254-024-00750-z

Perspective
Check for updates

The co-evolution of computational physics and high-performance computing

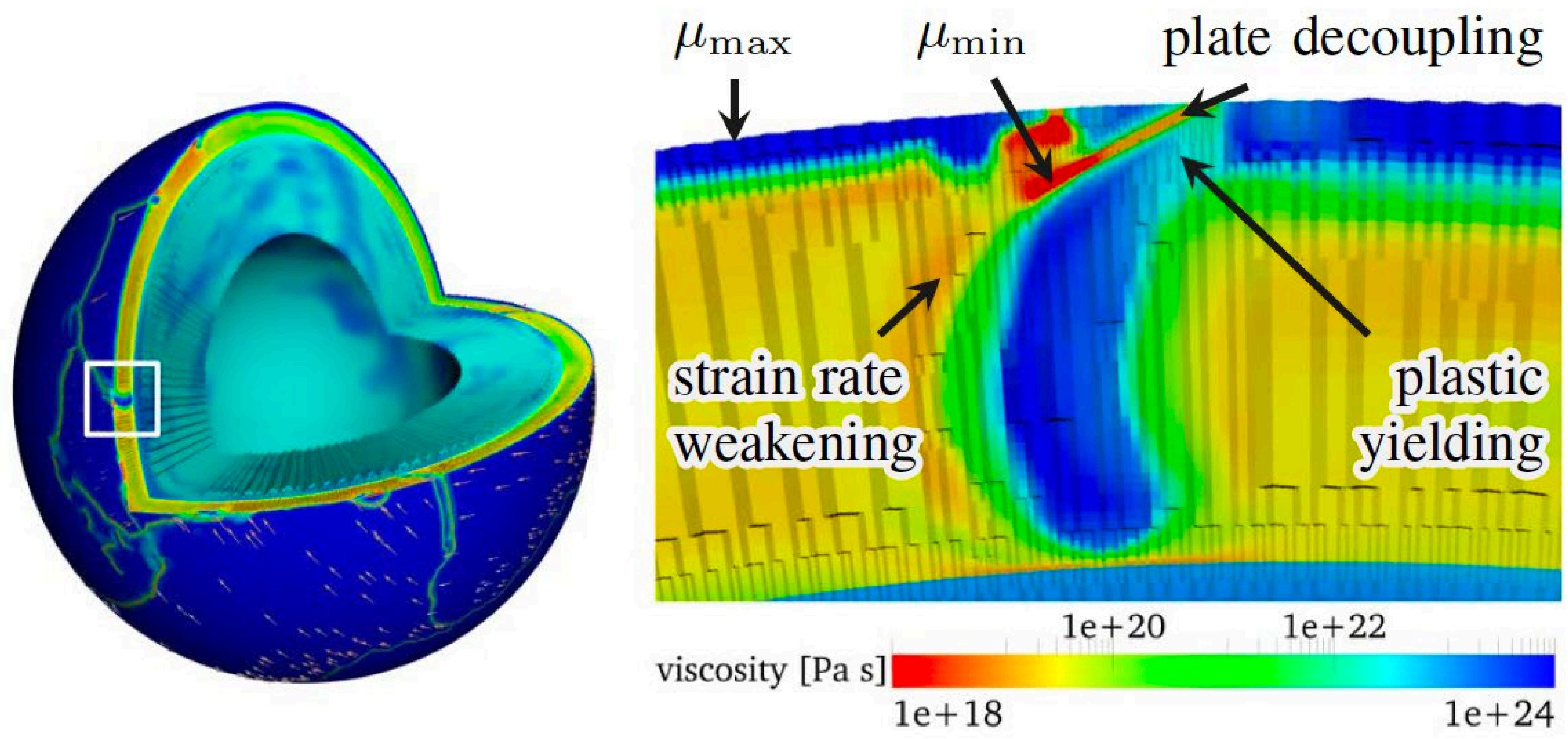
Jack Dongarra [1,2,3] & David Keyes [4,5]

Abstract

High-performance computational physics has been instrumental in advancing scientific research by regularly providing breakthroughs in speed, accuracy and modelling fidelity. This Perspective highlights the contributions of physicists to the development of high-performance computing infrastructure, algorithms and applications from the early days of computing to the exascale era. We recall the pioneering work of

Sections

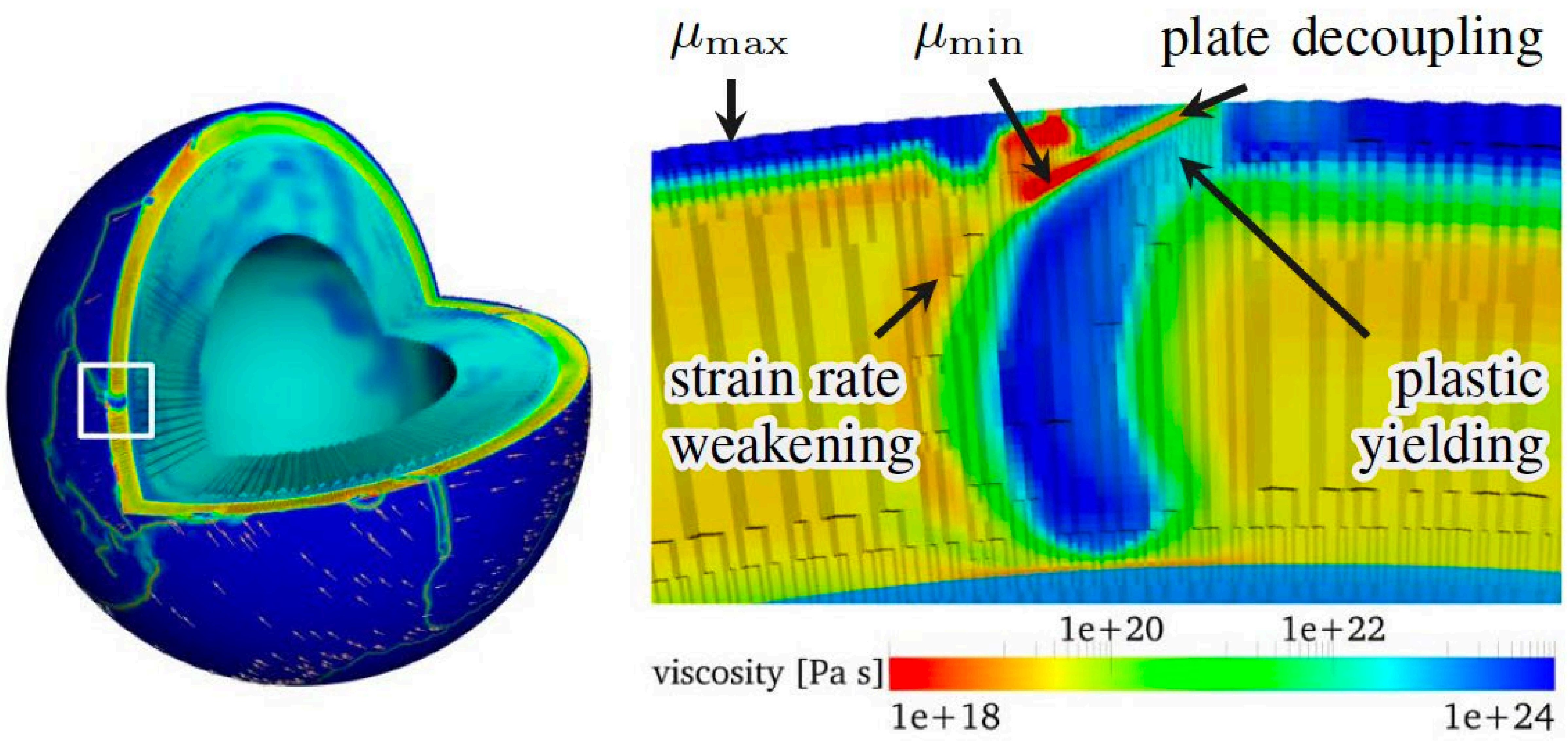
Gordon Bell Prize: pushing the boundaries of what is possible with HPC systems; awarded to teams who have demonstrated exceptional performance and scalability in solving *real-world problems*.



Rudi et al. (ACM Gordon Bell Prize Winner 2015)

Gordon Bell Prize: pushing the boundaries of what is possible with HPC systems; awarded to teams who have demonstrated exceptional performance and scalability in solving *real-world problems*.

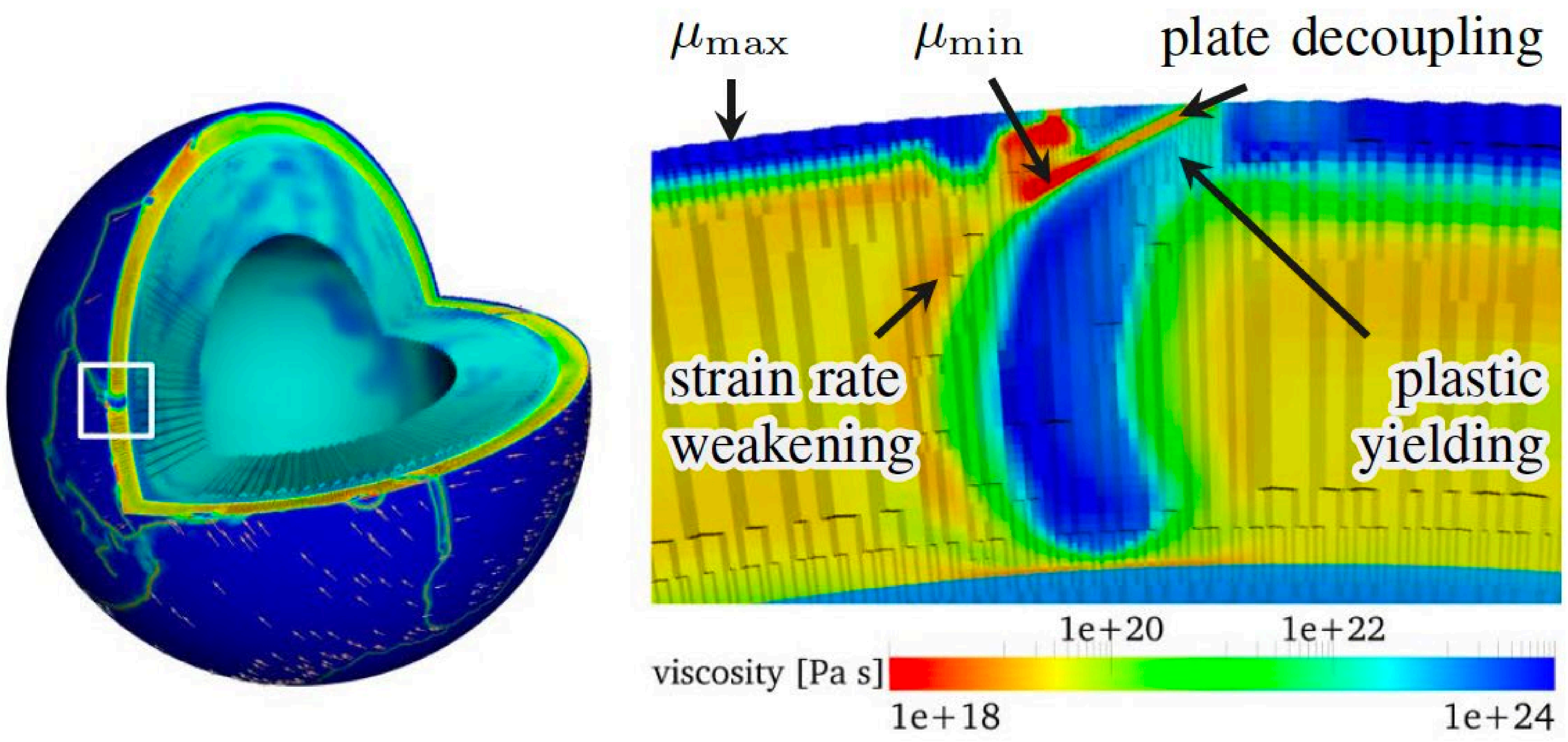
Applications matter!



Rudi et al. (ACM Gordon Bell Prize Winner 2015)

Gordon Bell Prize: pushing the boundaries of what is possible with HPC systems; awarded to teams who have demonstrated exceptional performance and scalability in solving *real-world problems*.

Applications matter!



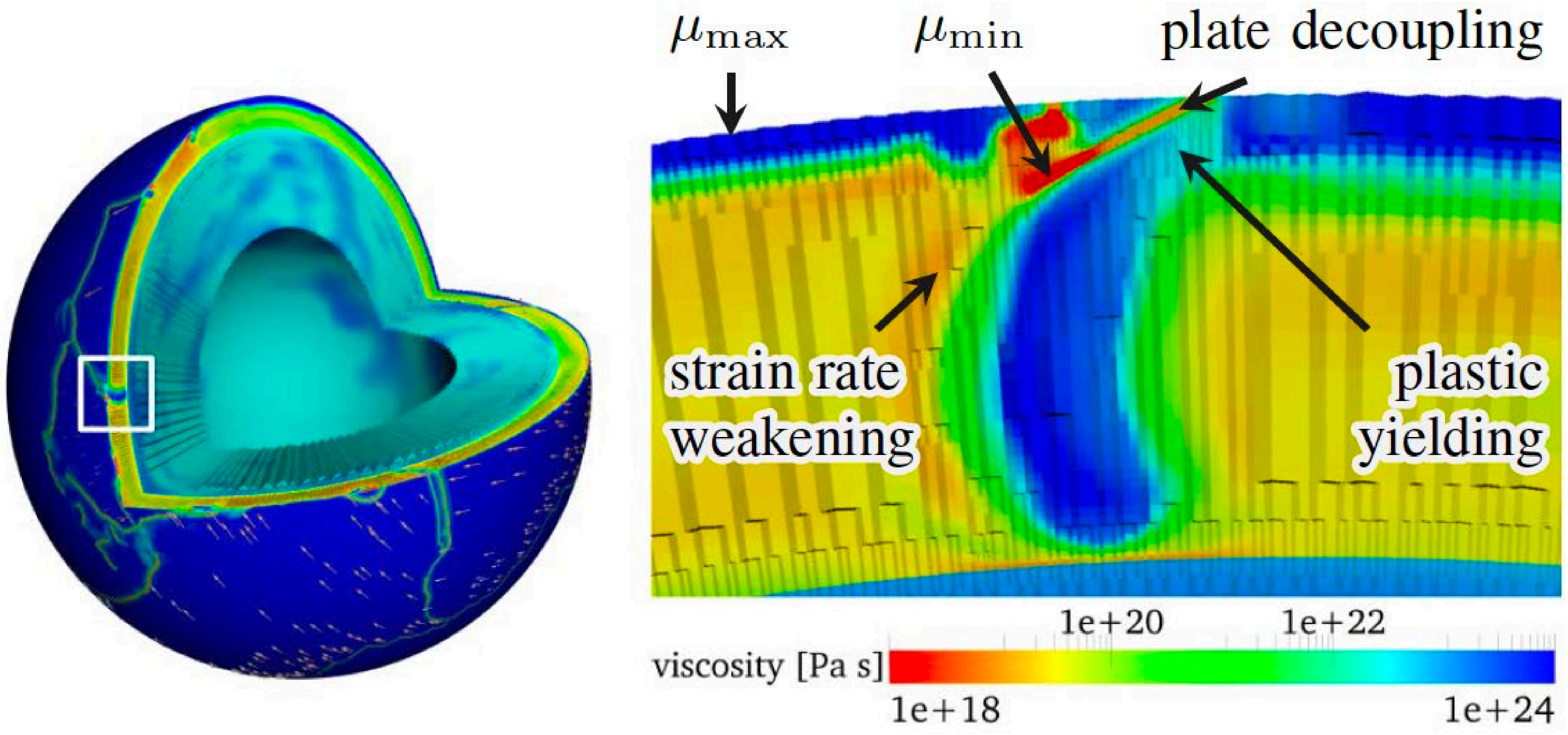
Rudi et al. (ACM Gordon Bell Prize Winner 2015)

The central computational challenge:

"to design implementations for *high-resolution realistic mantle flow* models that can handle the resulting extreme degrees of nonlinearity and ill-conditioning, the wide ranges of length scales and material properties, and the highly adapted meshes and required advanced discretizations, while also scaling to the $O(10^6)$ cores characteristic of leadership class supercomputers."

Gordon Bell Prize: pushing the boundaries of what is possible with HPC systems; awarded to teams who have demonstrated exceptional performance and scalability in solving *real-world problems*.

ALSO:



Rudi et al. (ACM Gordon Bell Prize Winner 2015)

Real-time Bayesian inference at extreme scale: A digital twin for tsunami early warning applied to the Cascadia subduction zone

Stefan Henneking
Oden Institute
The University of Texas at Austin
stefan@oden.utexas.edu

Sreeram Venkat
Oden Institute
The University of Texas at Austin
srvenkat@utexas.edu

Veselin Dobrev
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
dobrev1@llnl.gov

John Camier
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
camier1@llnl.gov

Tzanio Kolev
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
kolev1@llnl.gov

Milinda Fernando
Oden Institute
The University of Texas at Austin
milinda@oden.utexas.edu

Alice-Agnes Gabriel
Scripps Institution of Oceanography
University of California San Diego
algabriel@ucsd.edu

Omar Ghattas
Oden Institute, Walker Department of Mechanical Engineering
The University of Texas at Austin
omar@oden.utexas.edu

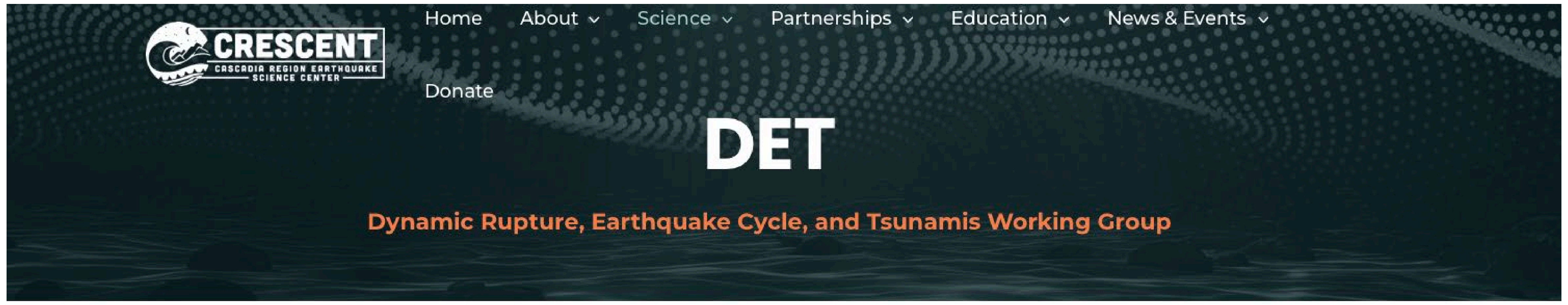
2025 ACM Gordon Bell Finalist!

The central computational challenge:

“to design implementations for *high-resolution realistic mantle flow* models that can handle the resulting extreme degrees of nonlinearity and ill-conditioning, the wide ranges of length scales and material properties, and the highly adapted meshes and required advanced discretizations, while also scaling to the $O(10^6)$ cores characteristic of leadership class supercomputers.”

To address the science questions, CRESCENT DET will need to ask:

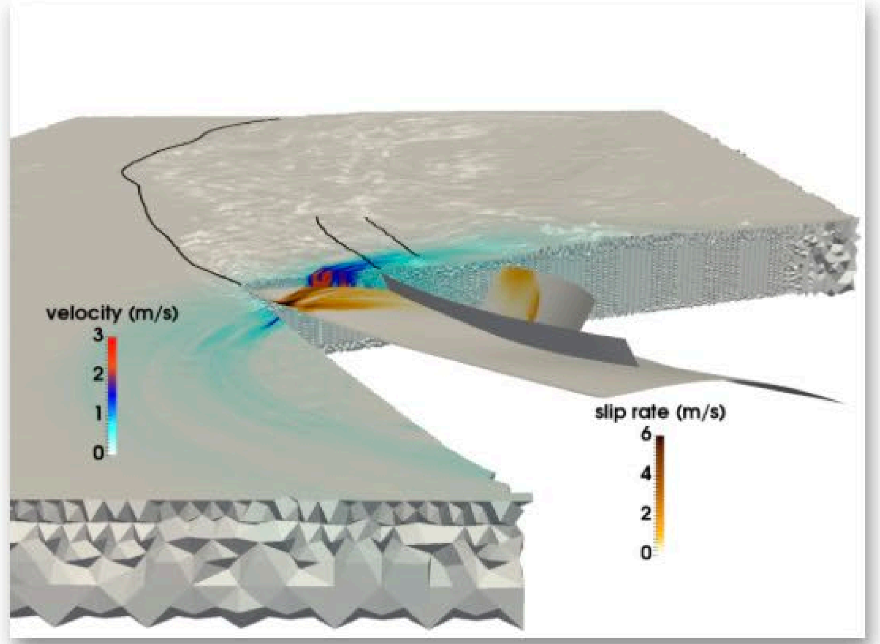
- How can we leverage the huge body of knowledge surrounding computational strategies for efficiently solving $Ax = b$?
- How can we exploit/adapt today's high-performance computers for our science?
- Could alternate modeling paradigms be useful?



Working Group

Dynamic Rupture, Earthquake Cycle, and Tsunamis

The Dynamic Rupture, Earthquake Cycle, and Tsunamis (DET) group is developing models (computer simulations) of earthquakes and tsunamis in Cascadia. The models account for tectonic loading over hundreds of years, slow slip events, and megathrust ruptures. The models, which are built on the structural models developed by the CVM and CFM groups, will be validated against geophysical and geological data from the C3S and CPAL groups. As part of this effort, DET is developing and using open-source modeling software, organizing community code comparison and benchmarking exercises for subduction zone earthquakes, and providing training for external users of the modeling software and workflows.



Dynamic rupture simulation of a subduction zone megathrust rupture, including activation of seaward and landward verging splay faults. Colors show slip rate on the faults and particle velocity (shaking) in the solid (Uphoff et al., SC'19).

Eric Dunham
DET Lead
Stanford University
edunham@stanford.edu

Alice Gabriel
University of California San Diego
algabriel@ucsd.edu

Brittany Erickson
University of Oregon
bae@uoregon.edu

Benchun Duan
Texas A&M University
bduan@geos.tamu.edu

Ruth Harris
United States Geological Survey
harris@usgs.gov

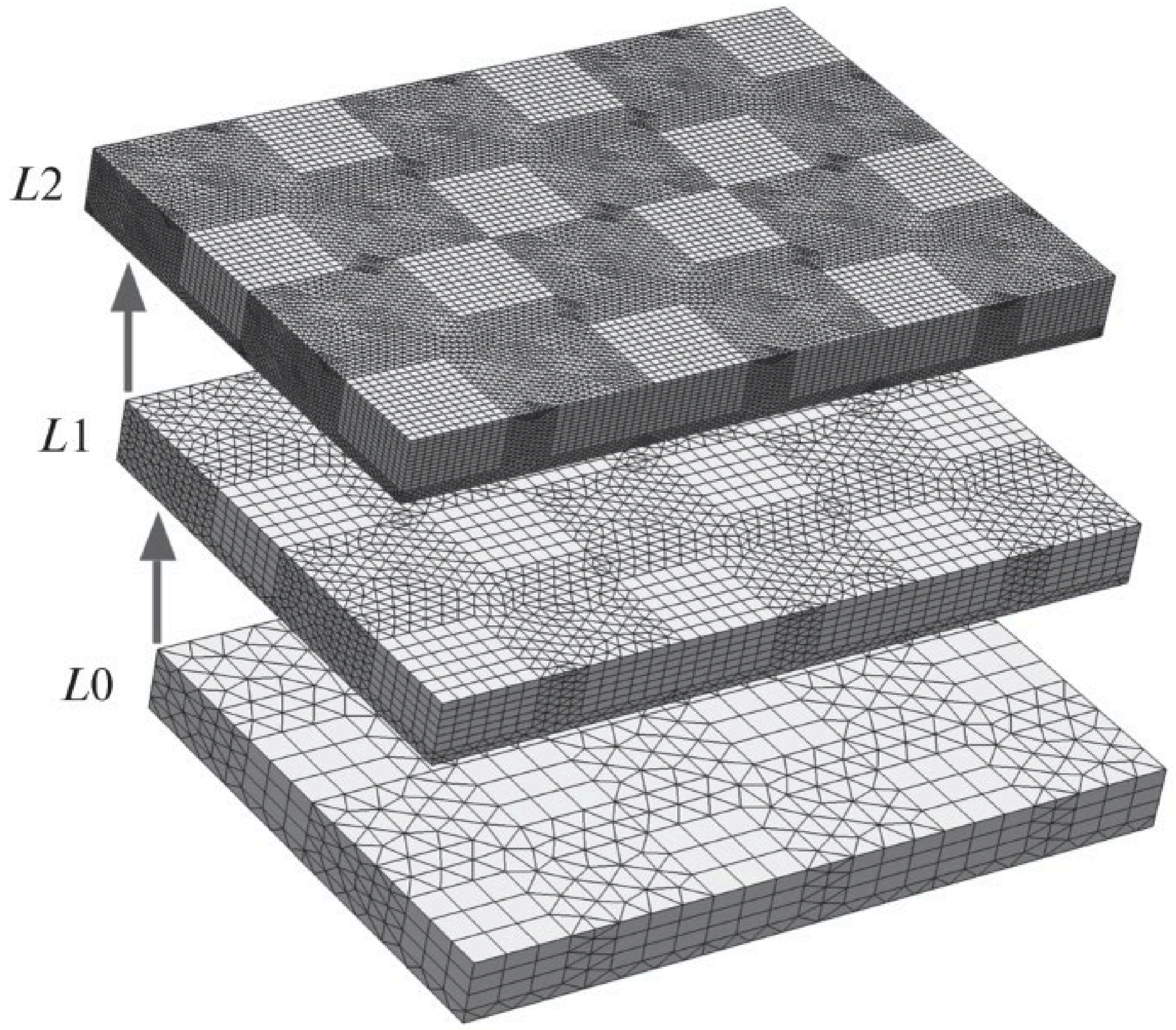
Ignacio Sepulveda
San Diego State University
isepulveda@sdsu.edu

Yihe Huang
University of Michigan
yiheh@umich.edu

Yajing Liu
McGill University
yajing.liu@mcgill.ca

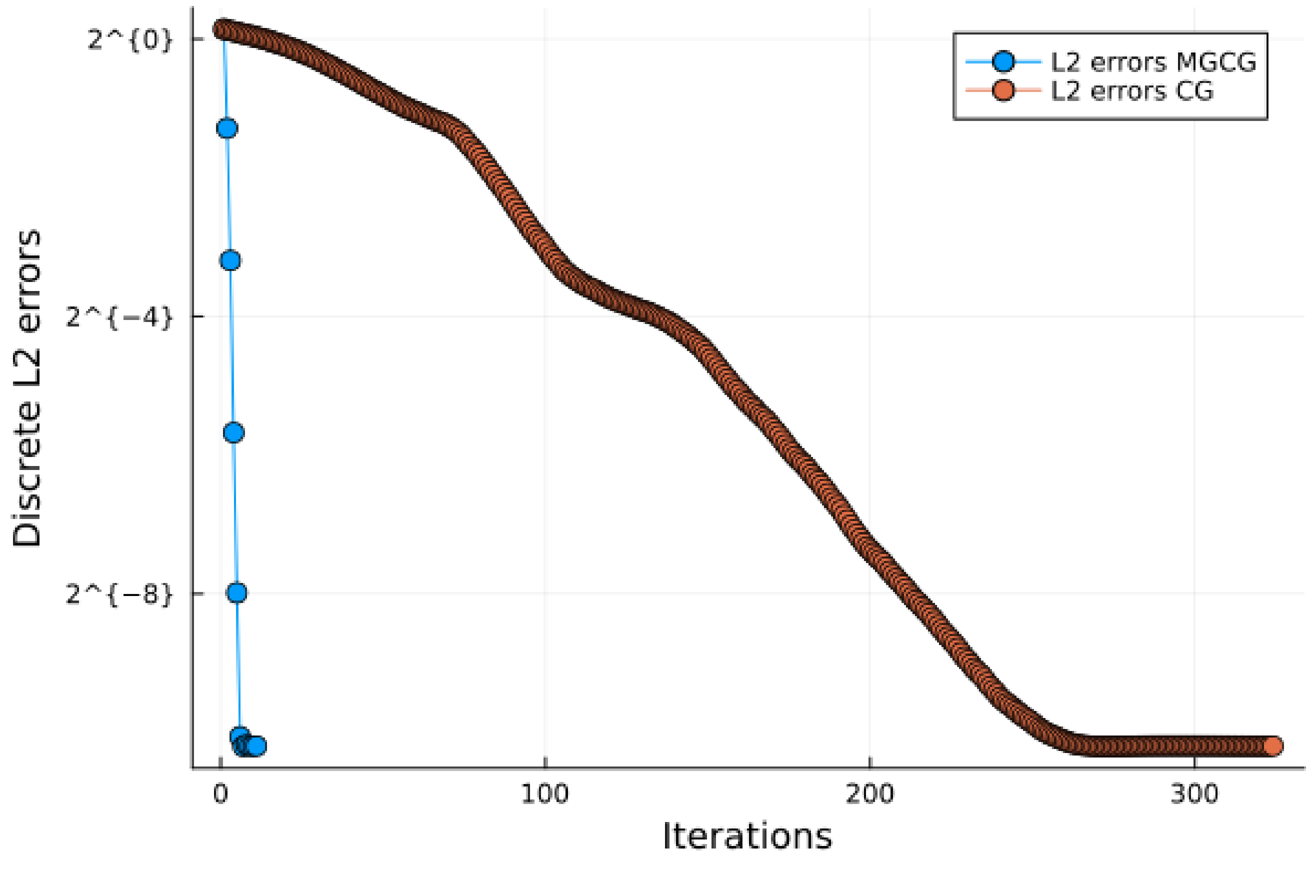
James Biemiller
United States Geological Survey
jbiemiller@usgs.gov

Towards optimal linear solvers



Multigrid (MG) methods utilize coarse meshes

MG can vastly improve convergence:



Iterations and time to converge:

PETSc

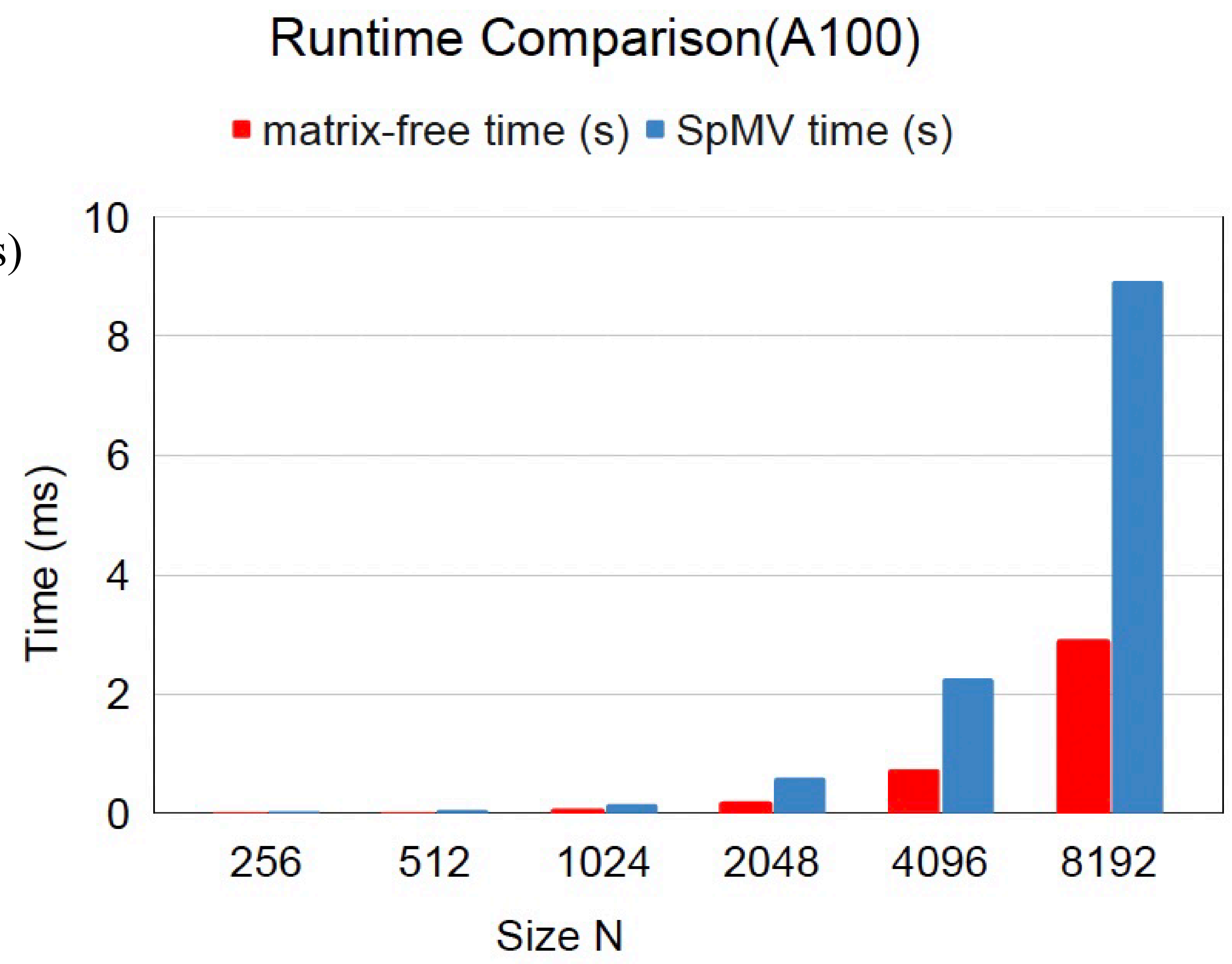
mg_levels_ksp_type	mg_levels_pc_type	iters	time
chebyshev	sor	18	4.105 s
	jacobi	22	3.382 s
	bjacobi	17	3.945 s
richardson	sor	18	3.581 s
	jacobi	49	3.729 s
	bjacobi	16	3.729 s
cg	sor	17	4.081 s
	jacobi	23	3.849 s
	bjacobi	16	3.971 s
richardson	none	11	0.086 s

Chen et al. '24

Iteration counts for custom multi-grid preconditioned conjugate-gradient (MGCG):

N	MGCG-Galerkin	MGCG	CG
64	6 / 2.33 E-9	6 / 2.25 E-09	221 / 9.98 E-9
128	6 / 8.39 E-9	6 / 2.42 E-9	431 / 9.61 E-9
256	7 / 2.44 E-9	6 / 2.16 E-9	839 / 9.79 E-9
512	7 / 2.63 E-9	6 / 1.89 E-9	1643 / 9.70 E-9
1024	7 / 2.72 E-9	6 / 1.73 E-9	3208 / 9.93 E-9

Matrix-free methods
also improve run time
(and memory requirements)



Chen et al. In Proceedings of the 38th ACM International Conference on Supercomputing (ICS '24)

What is the way forward for earthquake cycle simulations, then?

- It will be the job of geoscientists to solve the earthquake cycle problem, and we can do it/are doing it.

- Don't reinvent the wheel: what do we know about solving $Ax = b$?
  - Scalable preconditioning: $PAx = Pb$
  - Conjugate Gradient with off-the-shelf or custom preconditioning (probably based on multigrid)

- How can we exploit/adapt today's high-performance computers for our science?
  - Mixed-precision algorithms
  - Algorithms minimizing data transfer
  - Exploiting data sparcity
  - Many more avenues… e.g. auto-tuning, AI, asynchronous algorithms

- New modeling paradigms
  - Reduced order models (Magen, May & Gabriel, 2025)
  - Physic—informed Neural Networks (Erickson & Rucker, 2025)

What is the way forward for earthquake cycle simulations, then?

- Costs, energy, the environment, and J. Brown’s reasonable computing*

* https://github.com/jedbrown/talks/blob/main/20250806-ComputingAndEnvironment.ipynb

CLIMATE

Cheyenne to host massive AI data center using more electricity than all Wyoming homes combined

Computing Centers Costs: $1million per MW per year

Wyoming center would be using 1.8 GW of electricity -> Billions of dollars to run for a few years -> ~32 trillion miles on a gas-powered vehicle.

We need computing, but we can consider steps towards energy efficiency:

- verification of correctness, validation and model usefulness
- mathematically sound modeling frameworks
- energy efficient hardware
- repositories of simulation data (for reuse)
- continued conversations about this topic….

Thank you!

bae@uoregon.edu