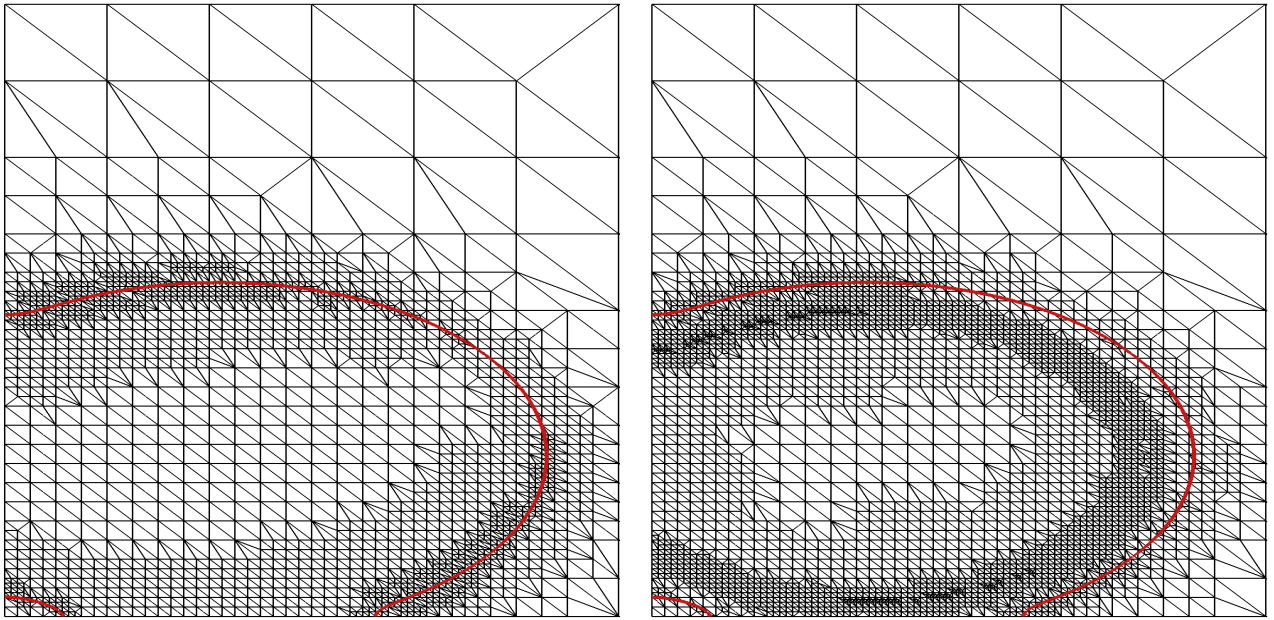
Adaptive Spectral Deferred Correction Methods for Cardiac Simulation

M. Weiser

joint work with B. Erdmann (ZIB)
S. Scacchi (U Milan)
R. Krause, D. Ruprecht (USI)
M. Minion (Stanford/Berkeley)



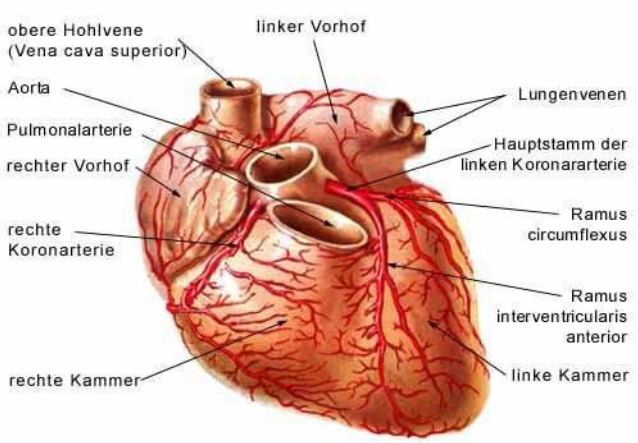
Zuse Institute Berlin

MATHEON

Contents

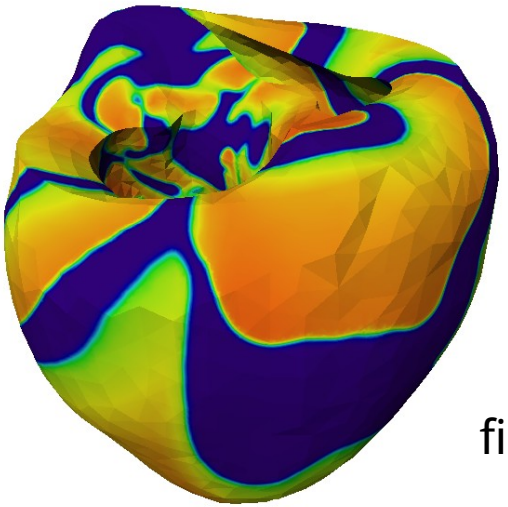
Cardiac Excitation Models

Electrocardiac Modeling



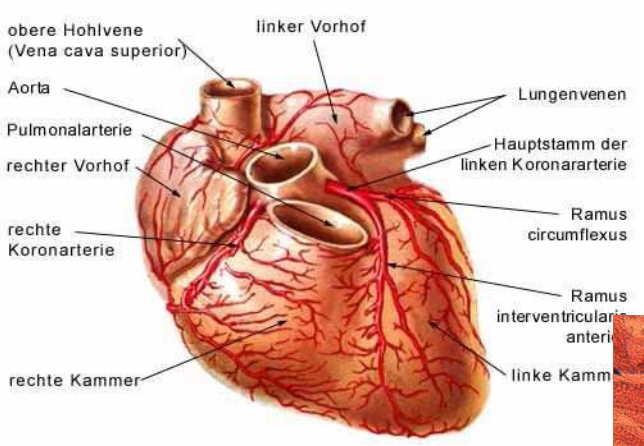
heart function:
pumping due to
coherent contraction

electrical excitation

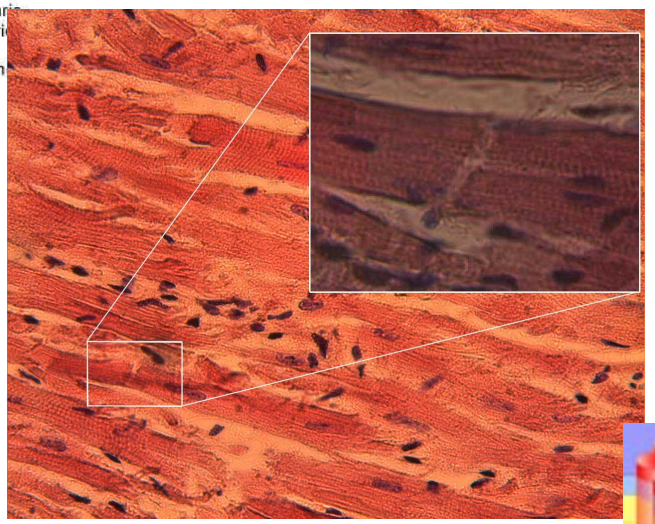


fibrillation

Electrocardiac Modeling

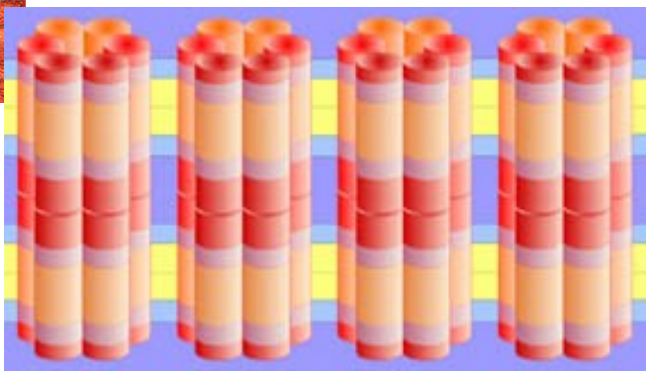


heart function:
pumping due to
coherent contraction

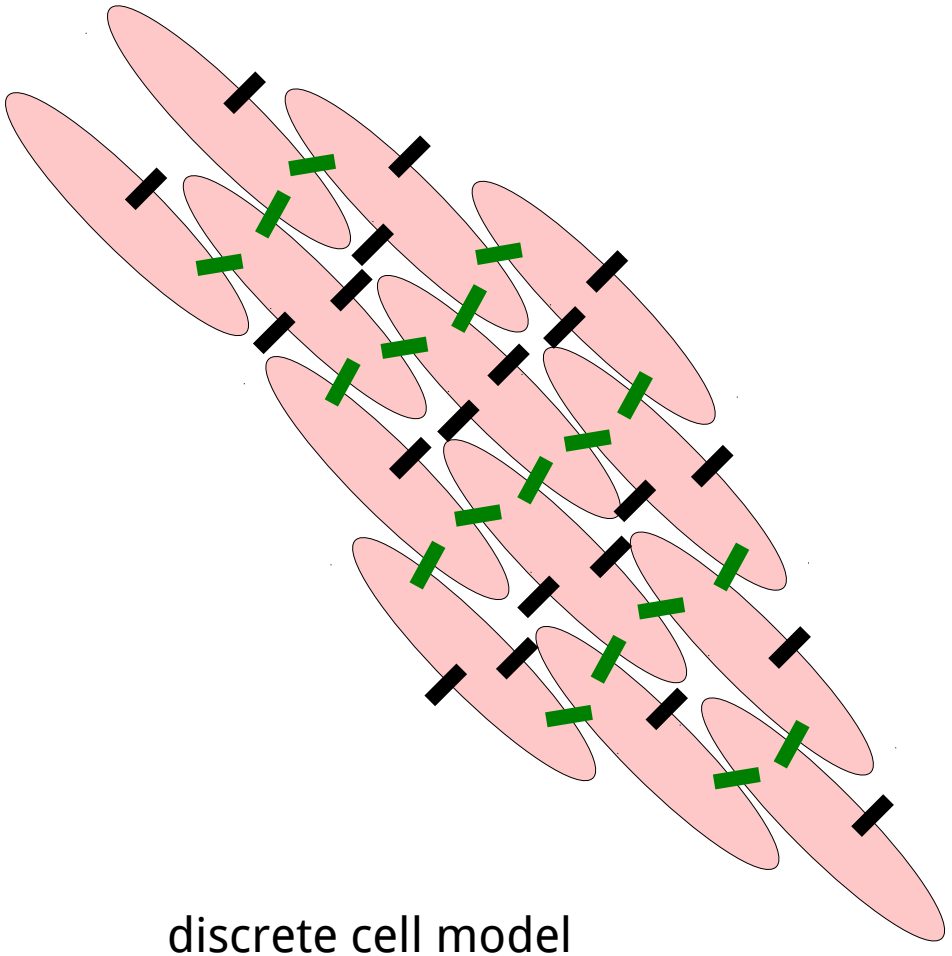


gap junctions:
ion diffusion
between cells

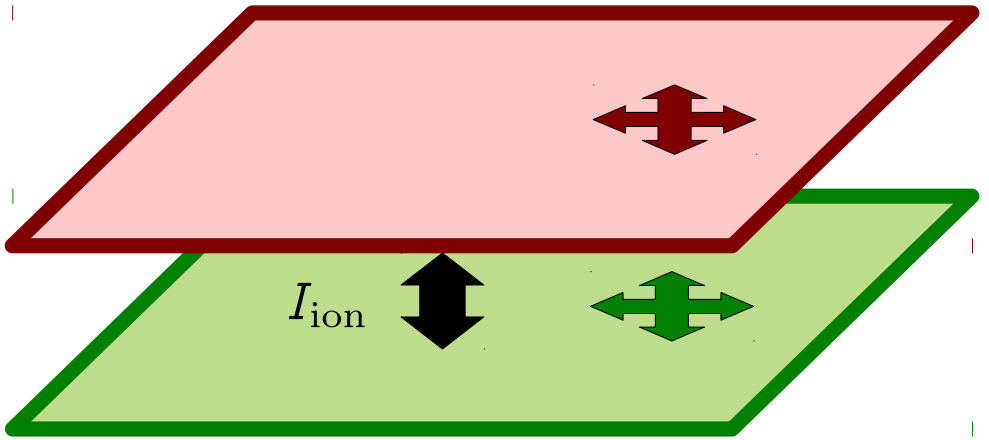
ion channels:
ion transport
across cell membrane



Homogenization



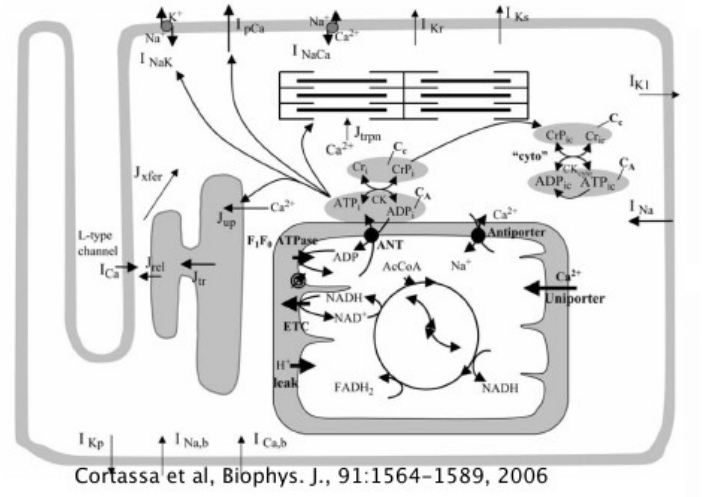
discrete cell model



continuous model

[Pennacchio, Savaré, Colli Franzone 2006]

Membrane Models



Cortassa et al, Biophys. J., 91:1564–1589, 2006

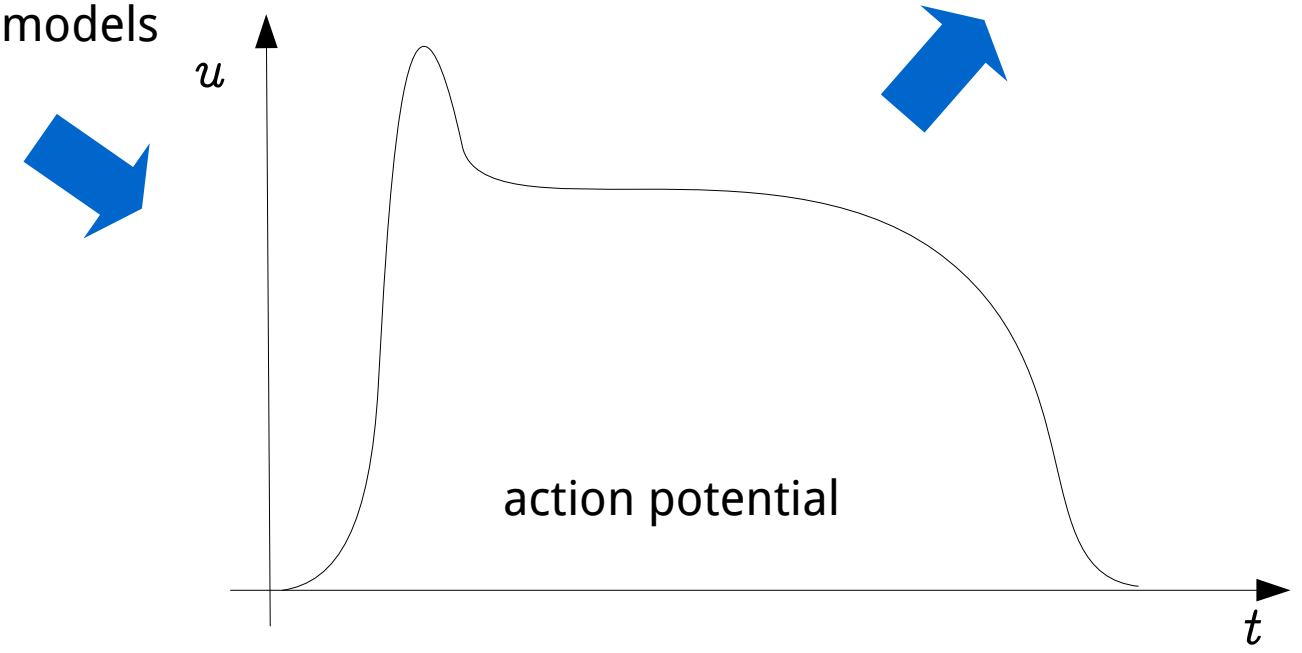
physiological models
(8 – 70 ODEs)

$$u = u_i - u_e$$

$$I_{\text{ion}}(u, w) = \alpha u(1-u)(u - (\bar{u} + w))$$

$$\dot{w} = u - w$$

phenomenological models
(1 – 4 ODEs)



Continuous Model: Bidomain Equations

Bidomain modell [Tung 1978]

intra cellular potential (ion concentration)
$$\frac{\partial}{\partial t}(u_i - u_e) = \mathrm{div}(D_i \nabla u_i) + I_{\mathrm{ion}}(u_i - u_e, w)$$

extra cellular potential (ion concentration)
$$\frac{\partial}{\partial t}(u_i - u_e) = -\mathrm{div}(D_e \nabla u_e) + I_{\mathrm{ion}}(u_i - u_e, w)$$

gating variables
$$\dot{w} = f(u_i - u_e, w)$$

Elliptic constraint reformulation

transmembrane voltage
$$u = u_i - u_e$$
$$2\dot{u} = \mathrm{div}(D_i \nabla u) + \mathrm{div}((D_i - D_e)\nabla u_e) + 2I_{\mathrm{ion}}(u, w)$$

extracellular potential
$$0 = \mathrm{div}((D_i + D_e)\nabla u_e) + \mathrm{div}(D_i \nabla u)$$
only for bidomain

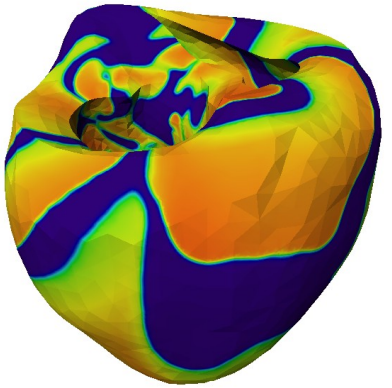
gating variables
$$\dot{w} = f(u, w)$$

existence & uniqueness: [Bourgault, Coudiere, Pierre 2009]

Computational Challenges

Monodomain equations

$$\dot{u} = \mathrm{div}(D\nabla u) + I_{\mathrm{ion}}(u, w)$$
$$\dot{w} = f(u, w)$$



Complexity

simulated time: 2 s
time step: 0.1 ms (explicit)
0.6 ms (implicit)

spatial domain: 100 mm
uniform grid size: 0.2 mm (linear FE)

equations: 2 – 50

→ 10^{12} – 10^{14} unknowns

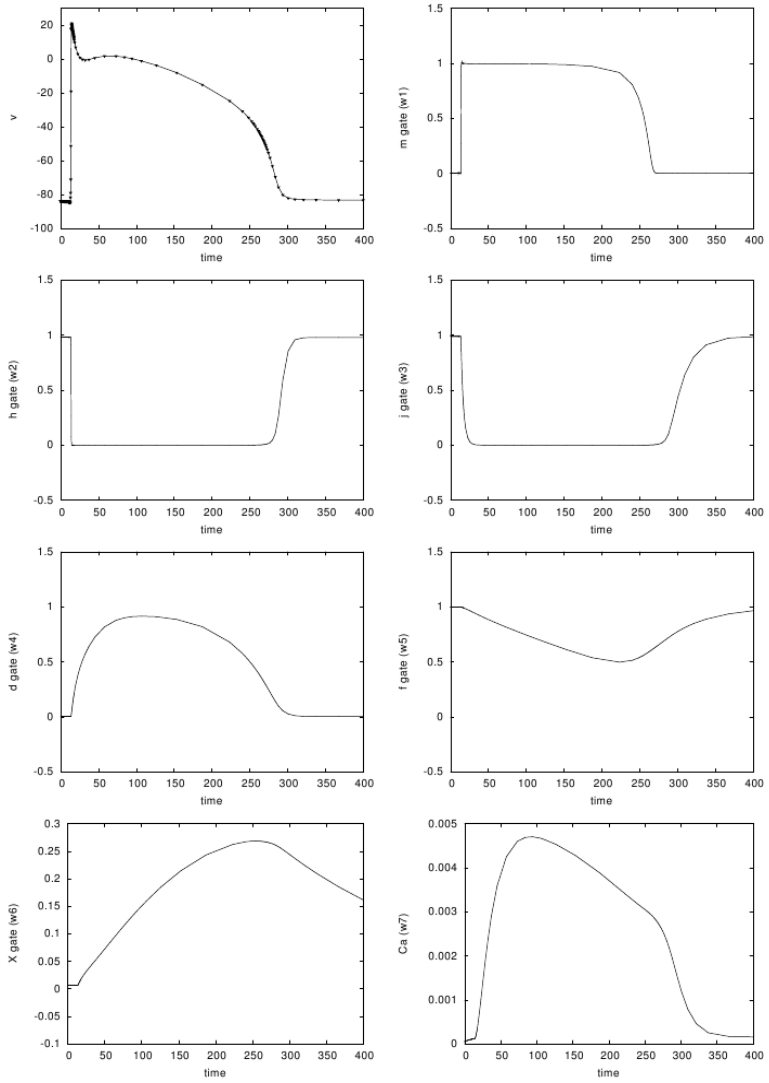
Approaches

- brute silicon force — massive parallelization (HPC, cluster, GPUs)
- adaptivity — exploiting solution structure
  - temporal locality of dynamics
  - spatial locality of features
  - low dimensionality of spatiotemporal features

computatioanl techniques survey: [Linge, Sundnes, Hanslien, Lines, Tveito 2009]

Standard Adaptivity

Time Adaptivity



Luo-Rudy model

Evolution problem

$$\dot{u} = F(u)$$

Time stepping

(e.g. Runge-Kutta)

$$\phi_i = u_k + \tau \sum_j a_{ij} F(\phi_j)$$

$$u_{k+1} = u_k + \tau \sum_i b_i F(\phi_i)$$

Embedded error estimator

$$\epsilon_t \approx \tau \sum_i b_i^e F(\phi_i)$$

Time step selection

$$\tau_{\text{opt}} = \sqrt[p+1]{\frac{\text{TOL}_t}{\epsilon_t}}\,\tau$$

Spatial Adaptivity

Stationary problem

$$(I - \tau \nabla \cdot D \nabla) u_{k+1} = u_k + \tau I_{\text{ion}}(u_k)$$

FE ansatz

$$\int_\Omega (v u^h_{k+1} + \tau \nabla v \cdot D \nabla u^h_{k+1})\, dx$$

$$= \int_\Omega (u_k + \tau I_{\text{ion}}(u_k)) v \, dx \quad \forall v \in V^h$$

Hierarchic error estimator

extended ansatz space V^e
(e.g. quadratic bubbles)

$$\int_\Omega (v u^e_{k+1} + \tau \nabla v \cdot D \nabla u^e_{k+1})\, dx$$

$$= \int_\Omega \left((u_k - u^h_{k+1} + \tau I_{\text{ion}}(u_k)) v - \tau \nabla v \cdot D \nabla u^h_{k+1} \right) dx \quad \forall v \in V^e$$

$$u_{k+1} - u^h_{k+1} \approx u^e_{k+1}$$

[Colli Franzone, Deuflhard, Erdmann, Lang, Pavarino 2006, Deuflhard, Erdmann, Roitzsch, Lines 2007, see also Belhamadia 2008, Belhamadia, Fortin, Bourgault 2009, Whiteley 2007]

Spatial Adaptivity

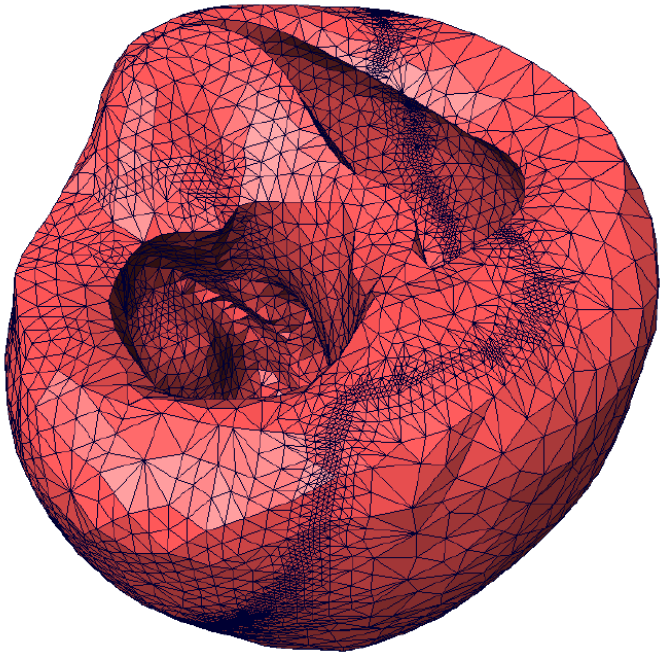
Stationary problem

$$(I - \tau \nabla \cdot D \nabla) u_{k+1} = u_k + \tau I_{\text{ion}}(u_k)$$

FE ansatz

$$\int_\Omega (v u^h_{k+1} + \tau \nabla v \cdot D \nabla u^h_{k+1})\, dx$$

$$= \int_\Omega (u_k + \tau I_{\text{ion}}(u_k)) v \, dx \quad \forall v \in V^h$$



Hierarchic error estimator

extended ansatz space V^e
(e.g. quadratic bubbles)

$$\int_\Omega (v u^e_{k+1} + \tau \nabla v \cdot D \nabla u^e_{k+1})\, dx$$

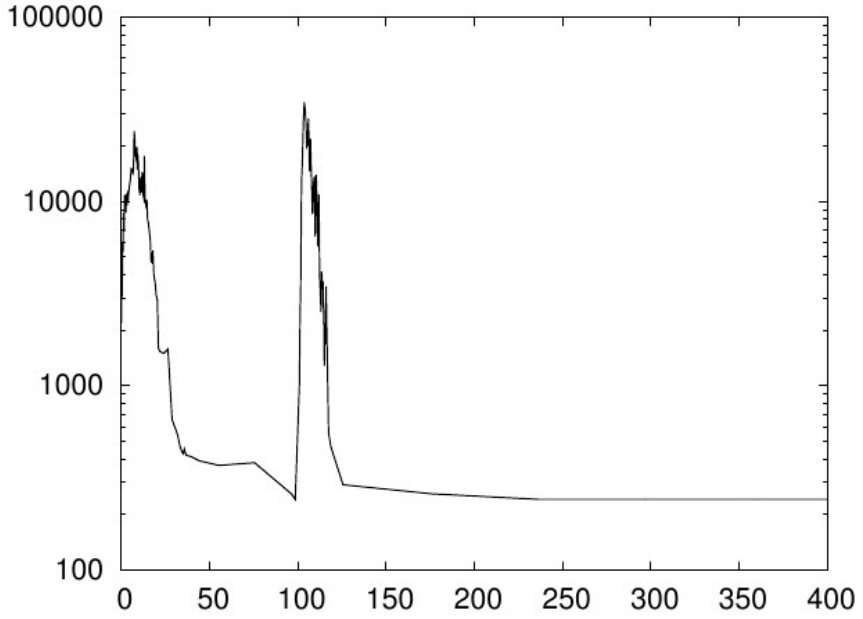
$$= \int_\Omega \left((u_k - u^h_{k+1} + \tau I_{\text{ion}}(u_k)) v - \tau \nabla v \cdot D \nabla u^h_{k+1} \right) dx \quad \forall v \in V^e$$

$$u_{k+1} - u^h_{k+1} \approx u^e_{k+1}$$

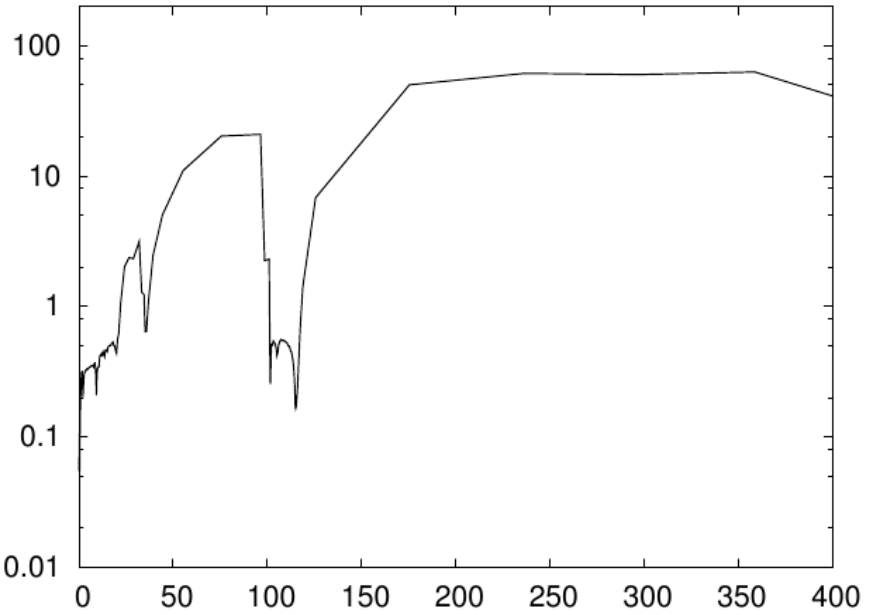
[Colli Franzone, Deuflhard, Erdmann, Lang, Pavarino 2006, Deuflhard, Erdmann, Roitzsch, Lines 2007, see also Belhamadia 2008, Belhamadia, Fortin, Bourgault 2009, Whiteley 2007]

Impact of Adaptivity

spatial adaptivity: mesh nodes



time adaptivity: time step size

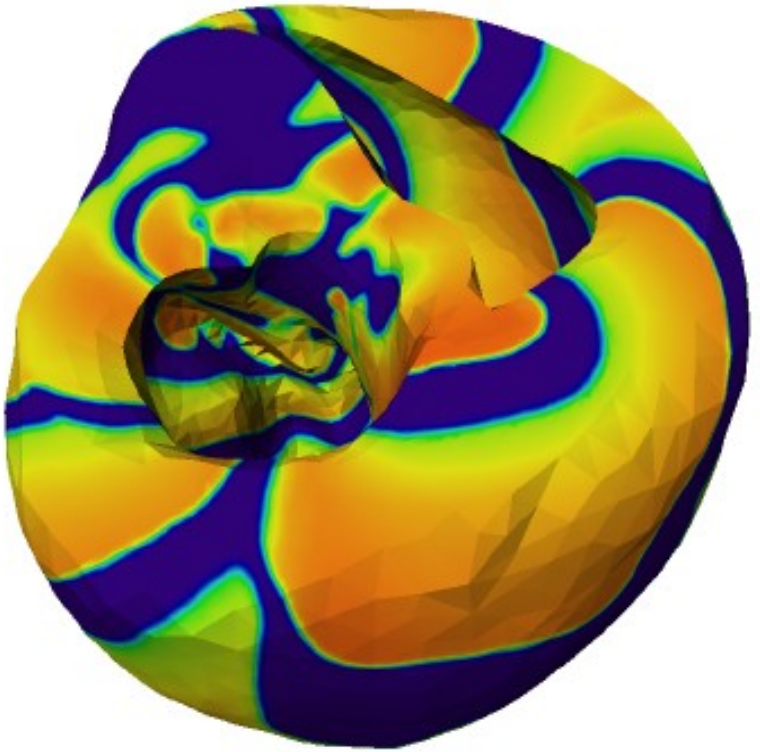


Cost/time ratio varies by 4 orders of magnitude!

Numerical Example: Fibrillation

Sequential general purpose KARDOS code

- 6 weeks computing time for 2 seconds simulated time

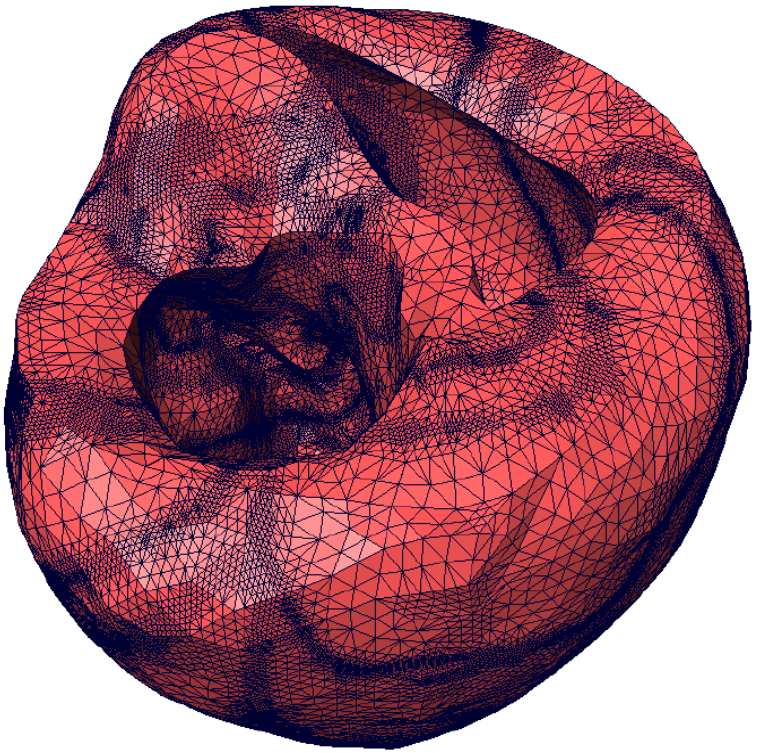


580ms

Why?

Sequential general purpose KARDOS code

- 6 weeks computing time for 2 seconds simulated time



Pitfalls

- at each time point, depolarization front is in the computational domain
- many, long depolarization fronts
- solution of linear equation systems
- overhead of adaptivity:
  - mesh modifications
  - repeated assembly of stiffness/mass matrix
  - hierarchical error estimator

max nodes: 2.1e6
uniform: 3.7e8

Time Integrator Wishlist

re-use of coarse grid computations

exploit information gathered on coarser grids for improved accuracy on finer grids: iterative improvement

inexact solves in implicit schemes

high accuracy despite inexact solves: iterative improvement

longer time steps

amortize overhead of adaptivity over longer time steps using one fixed mesh: many stages (caveat: larger meshes, see lateron)

higher order with splitting

higher convergence order than 2 despite pointwise propagation of ODEs: iterative coupling

local time stepping

allow different time steps in different equations or parts of the domain (important for systems with different time scales or uniform meshes)

re-use of previous computation on time step reduction

(important for time adaptivity)

Spectral Deferred Correction Methods

Spectral Deferred Corrections (SDC)

$$\dot{u} = f(u)$$

time grid $0 = \tau_0 < \tau_1 < \ldots, < \tau_n = \tau$

one time step

approximate solution $u^k \in \mathbb{P}_n,\ u_i := u(\tau_i),\ \dot{u}_i = \dot{u}(\tau_i)$

defect

$$\begin{aligned} \delta u := u^* - u,\ \delta\dot{u} &= \dot{u}^* - \dot{u} \\ &= f(u^*) - f(u) + f(u) - \dot{u} \\ &\approx f'(u)\delta u + f(u) - \dot{u} \end{aligned}$$

SDC method

- integrate defect ODE with simple method (e.g., Euler) on $\tau_0, \ldots, \tau_n$

$$(I - (\tau_{i+1} - \tau_i) f'(u_{i+1}))\delta u_{i+1} = \delta u_i + (\tau_{i+1} - \tau_i)(f(u_{i+1}) - \dot{u}_{i+1})$$

- update $u_i \leftarrow u_i + \delta u_i$

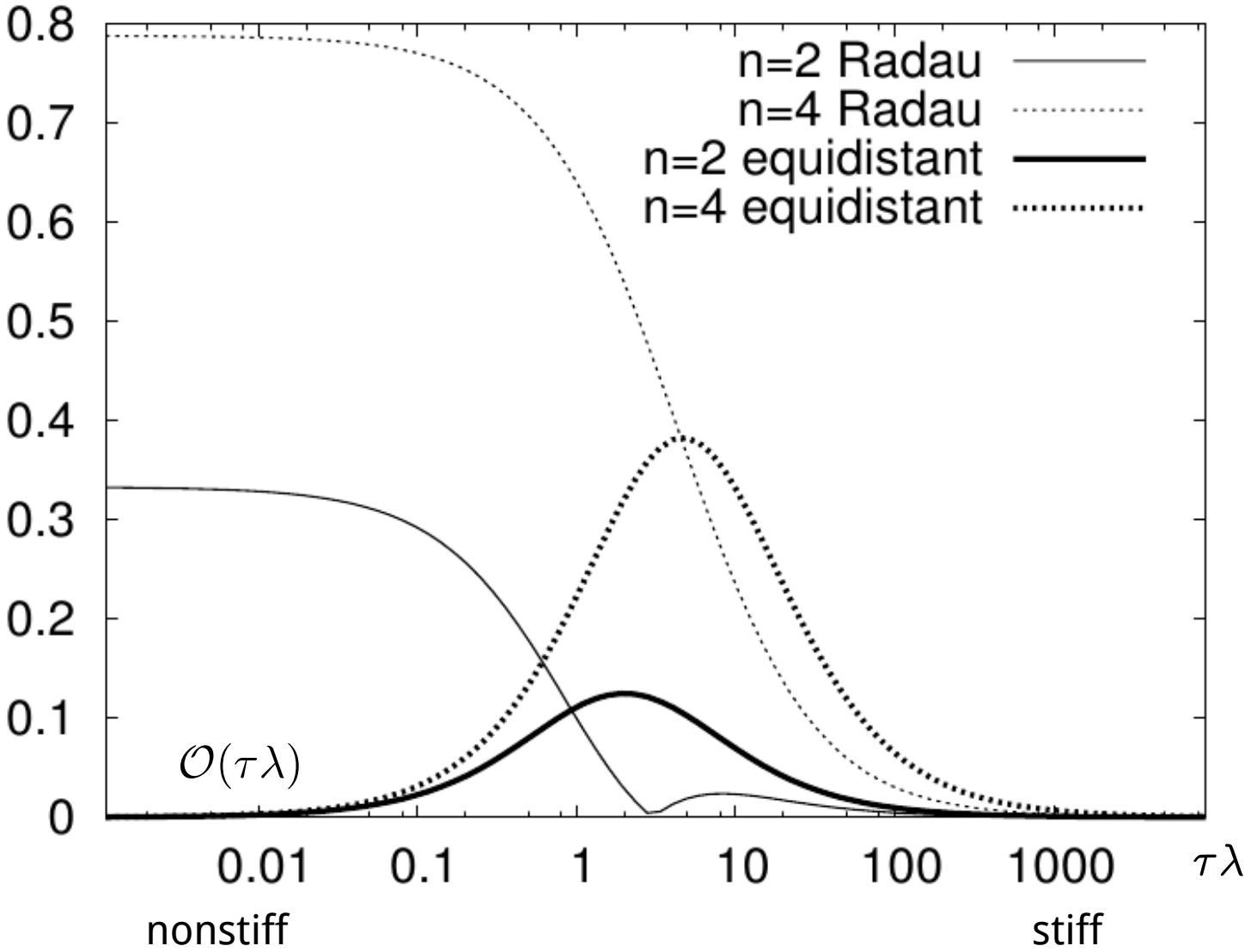
differentiation: DSDC

fixed point: collocation conditions $\dot{u}_i = f(u_i),\ i = 1, \ldots, n$

[Zadunaisky '76, Frank/Überhuber '77, Auzinger et al. '03, ...]

DSDC

contraction rate (spectral radius) of DSDC iterations on $\dot{u} = -\lambda u$



- good convergence rate on equidistant grids (one order per iteration)
- bad convergence rate on nonuniform grids (no order at all)
- good contraction for stiff components

Spectral Deferred Corrections

$$u(t) = \int_0^t f(u(s))\,ds$$

time grid $0 = \tau_0 < \tau_1 < \ldots, < \tau_n = \tau$

approximate rhs $\tilde{u} \in \mathbb{P}_n,\ u_i = u(\tau_i)$

defect $\delta u := u^* - u, \quad \delta u(t) = \int_0^t (f(u^*) - \dot{u})\,ds$

$$= \int_0^t (f(u^*) - f(u) + f(u) - \dot{u})\,ds$$

$$\approx \int_0^t (f'(u)\delta u + f(u) - \dot{u})\,ds$$

SDC method

- integrate Picard equation with simple rule (e.g., rectangular) on $\tau_0, \ldots, \tau_n$

$$(I - (\tau_{i+1} - \tau_i) f'(u_{i+1}))\delta u_{i+1} = \delta u_i + (\tau_{i+1} - \tau_i)\Big(\sum_j S_{ij} f(u_j) - u_{i+1} - u_i\Big)$$

- update $u_i \leftarrow u_i + \delta u_i$

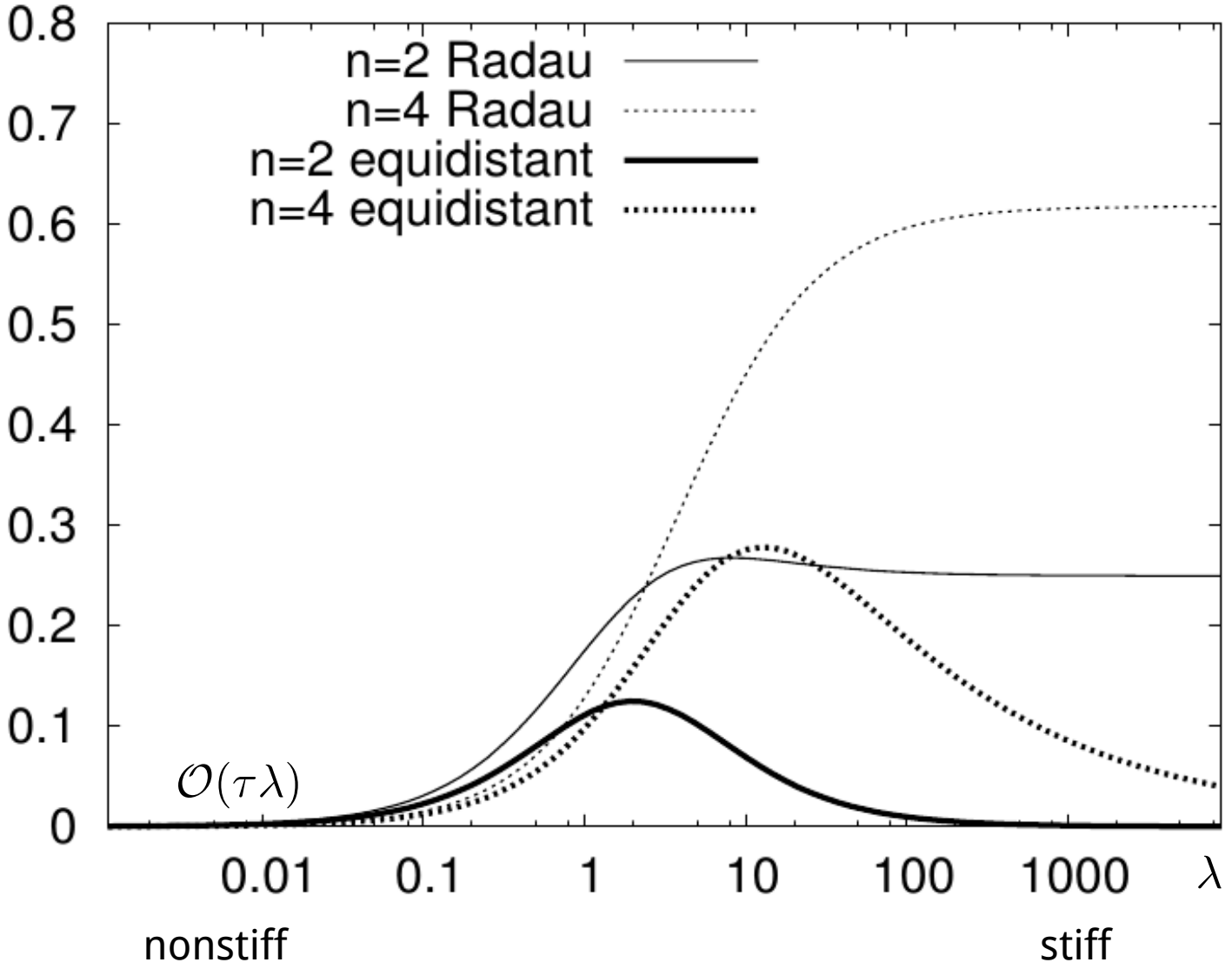
fixed point: collocation conditions $\dot{u}_i = f(u_i),\ i = 1, \ldots, n$

quadrature: QSDC

[Dutt/Greengard/Rokhlin '00, Auzinger et al. '03,Layton/Minion '07, ...]

QSDC

contraction rate (spectral radius) of DSDC iterations on $\dot{u} = -\lambda u$



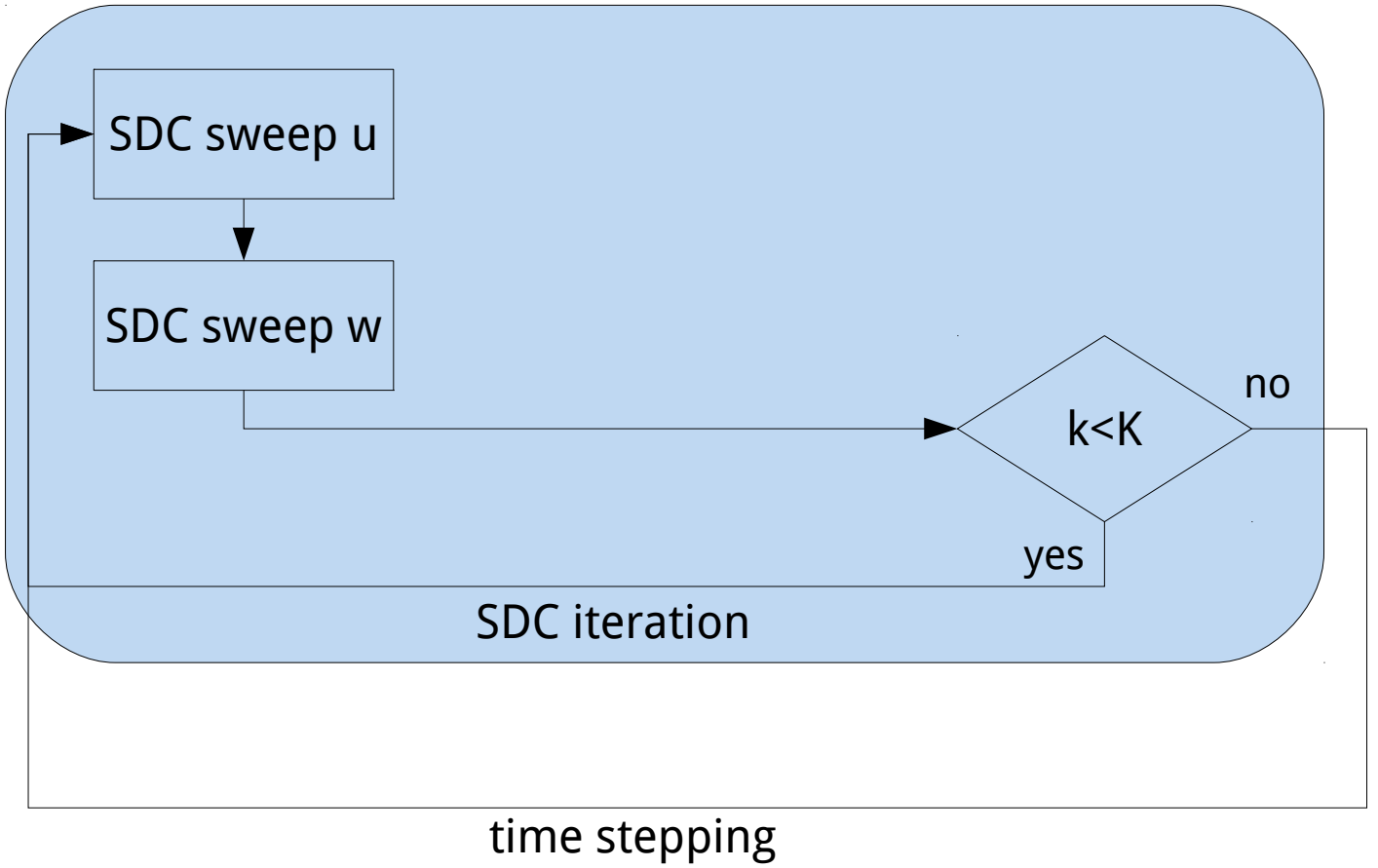
- good convergence rate on equidistant grids (one order per iteration)
- bad convergence rate on nonuniform grids for stiff components (DAEs)
- good contraction for non-stiff components (one order per iteration)

SDC vs. Extrapolation and Rosenbrock

	SDC	Rosenbrock	Extrapolation
order per stage	-	+	o
reuse on time step reduction	+	-	o
reuse on mesh refinement	+	o	-
achievable order	+	-	+
flexibility	+	-	-

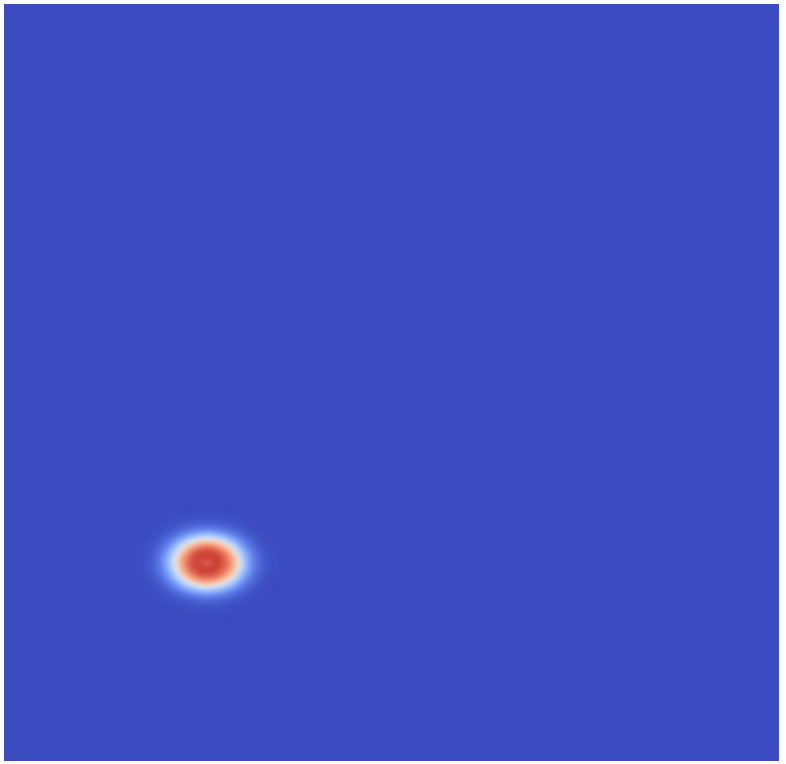
Open questions:
- Can qualitative advantages (flexibility) be turned into superior efficiency?
- How to exploit the flexibility for maximum efficiency?

SDC for Monodomain

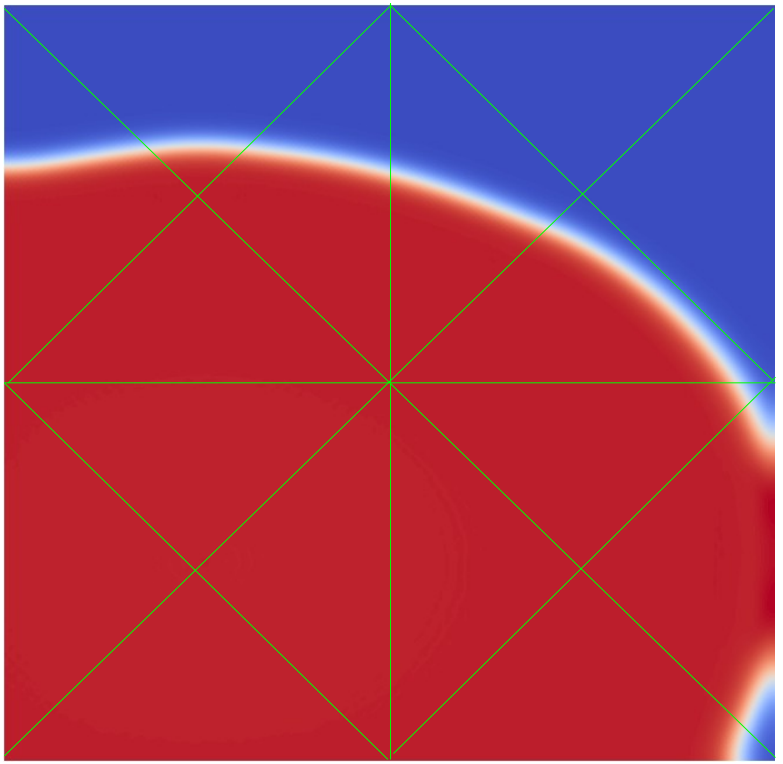


Monodomain Test Example

$\Omega =]0, 2[^2$ Aliev-Panfilov model, cubic FE, 33,000 nodes, 2 x 296,000 dofs
(6 uniform refinements)



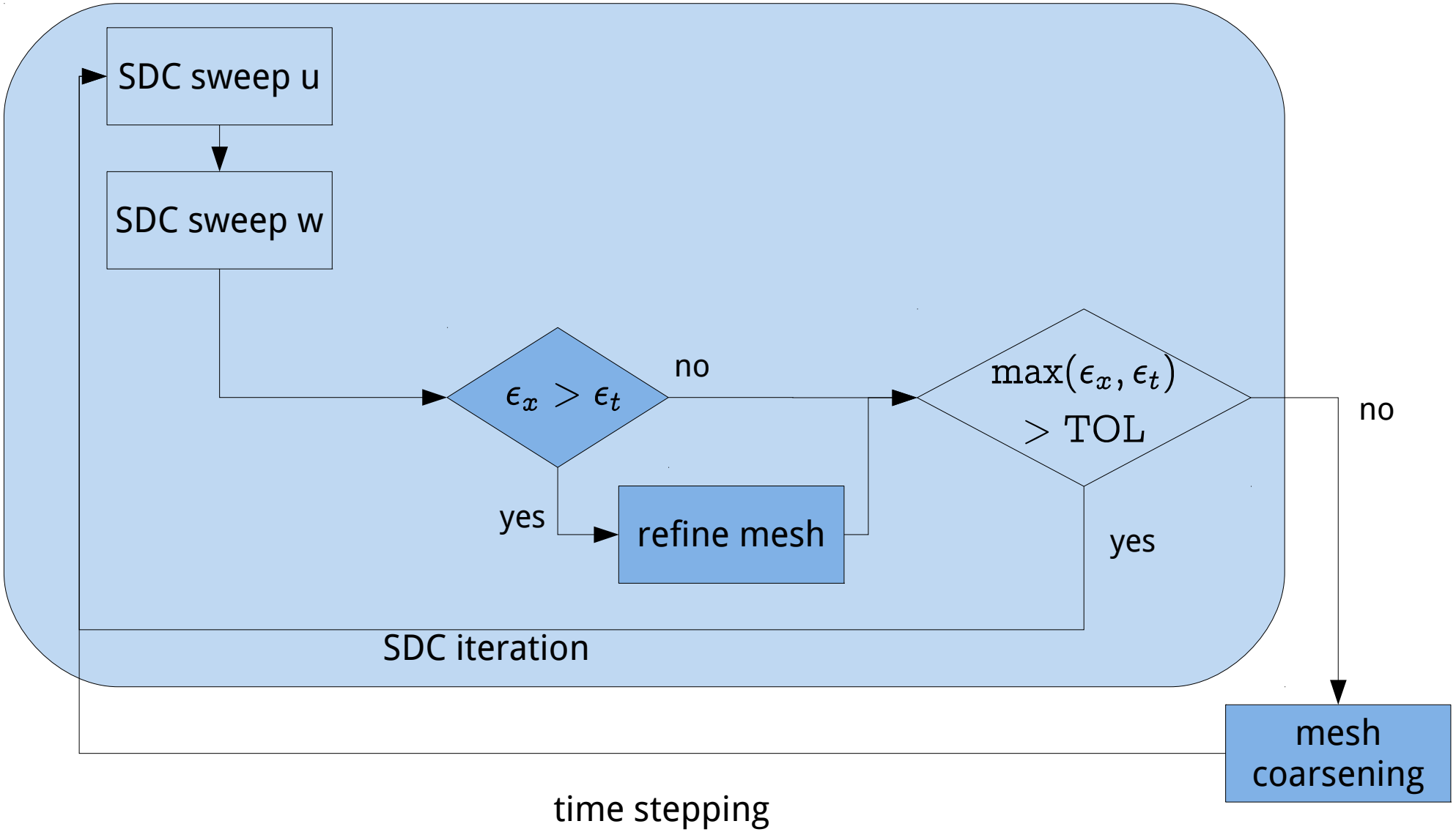
excitation after 1ms



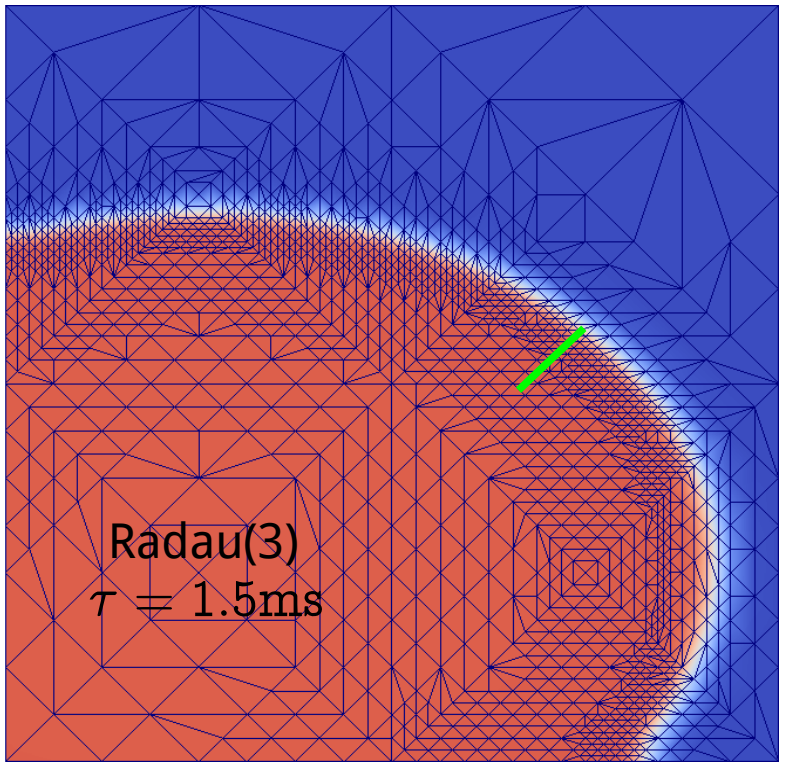
final time T=24ms

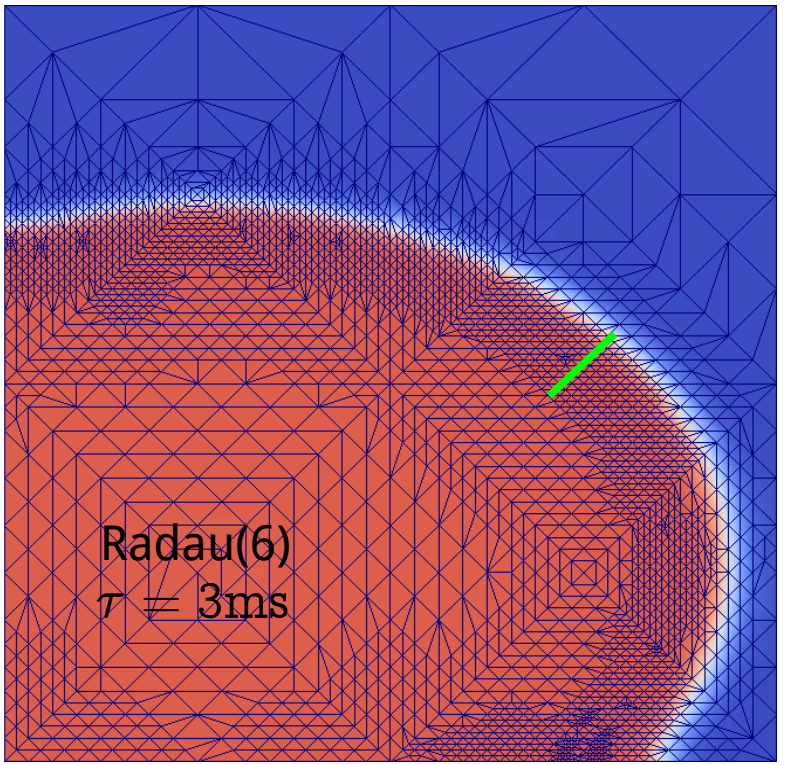
SDC, Radau(3) $\tau = 1\text{ms}$

Interleaved Mesh Refinement



Adaptive Computation



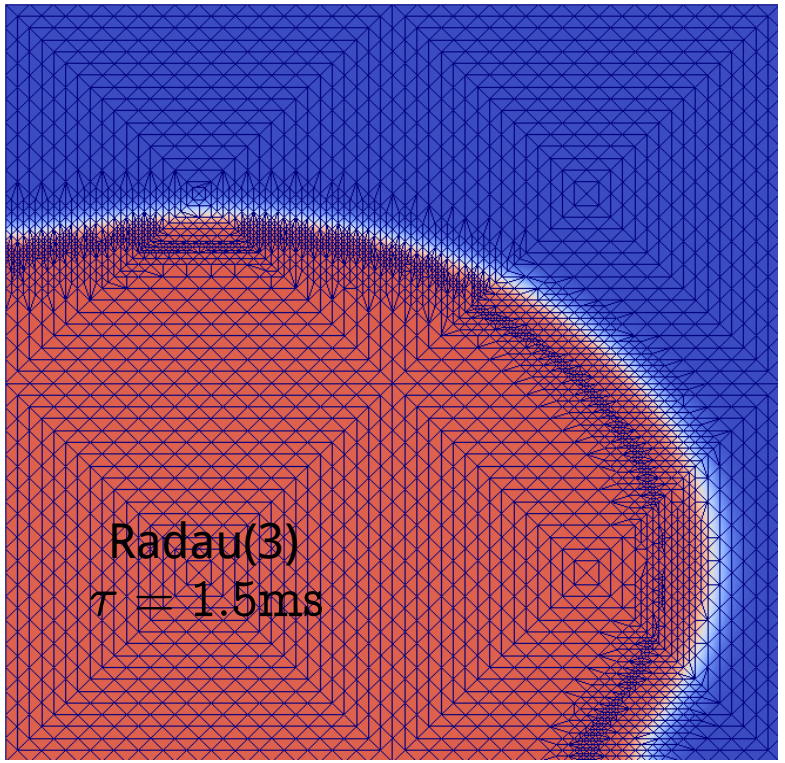


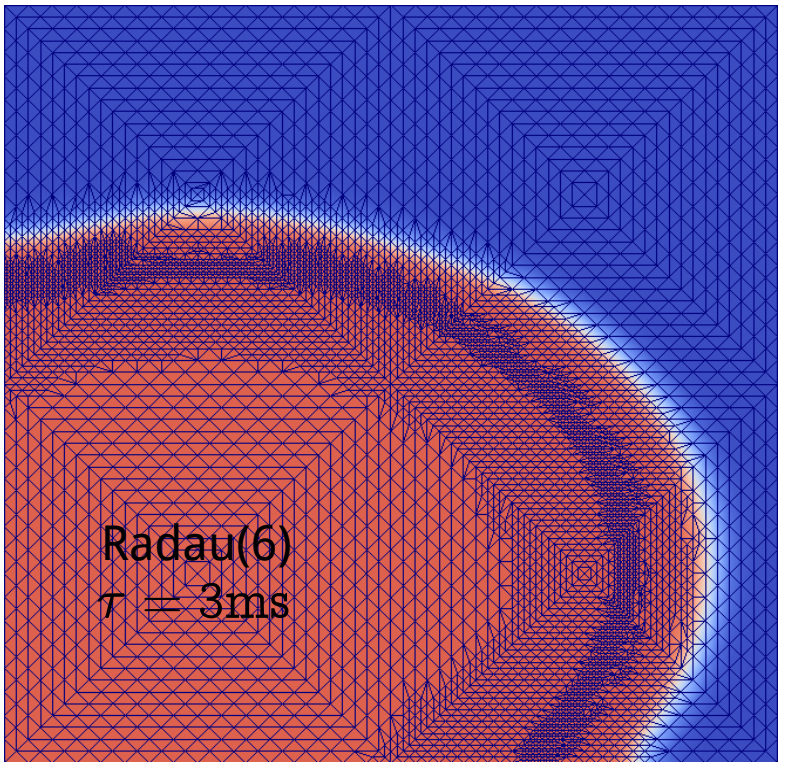
$t = 18\text{ms}$

max. refinement: 6
6 sweeps

	Radau(3)	Radau(6)	uniform
avg. dof	10,325	15,096	295,681
overhead	18s	15s	5s
integration	10s	13s	499s

Adaptive Computation





$t = 18\text{ms}$

max. refinement: 6
min. refinement: 4
6 sweeps

	Radau(3)	Radau(6)	uniform
avg. dof	30,536	38,199	295,681
overhead	39s	30s	5s
integration	43s	45s	499s

Spatial Adaptivity

Rosenbrock / Runge-Kutta

fixed order scheme

$$k_i = \tau \left(f(u_n + \sum_{j=1}^{i-1} \alpha_{ij} k_j) + f'(u_n) \sum_{j=1}^{i} \gamma_{ij} k_j \right)$$

$$u_{n+1} = u_n + \sum_{j=1}^{s} \beta_j k_j$$ ⬅ all stages enter directly: compute on the same mesh!

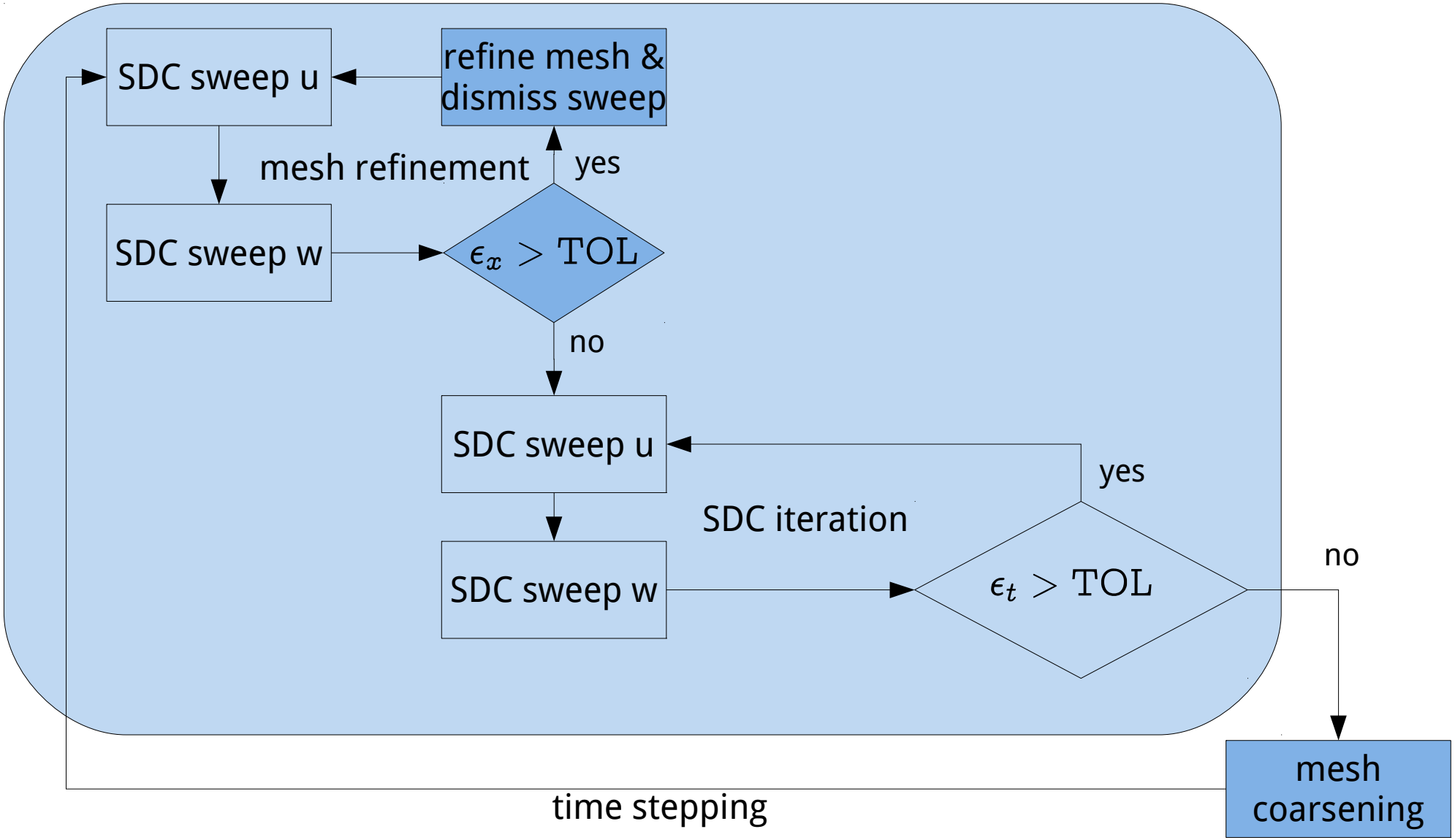
conservative strategy: error estimation & refinement for final result u_{n+1}
drawback: recompute all stages from scratch

optimistic strategy: error estimation & refinement for first stage k_1 (Euler step)
drawback: mesh not adapted to u_{n+1}

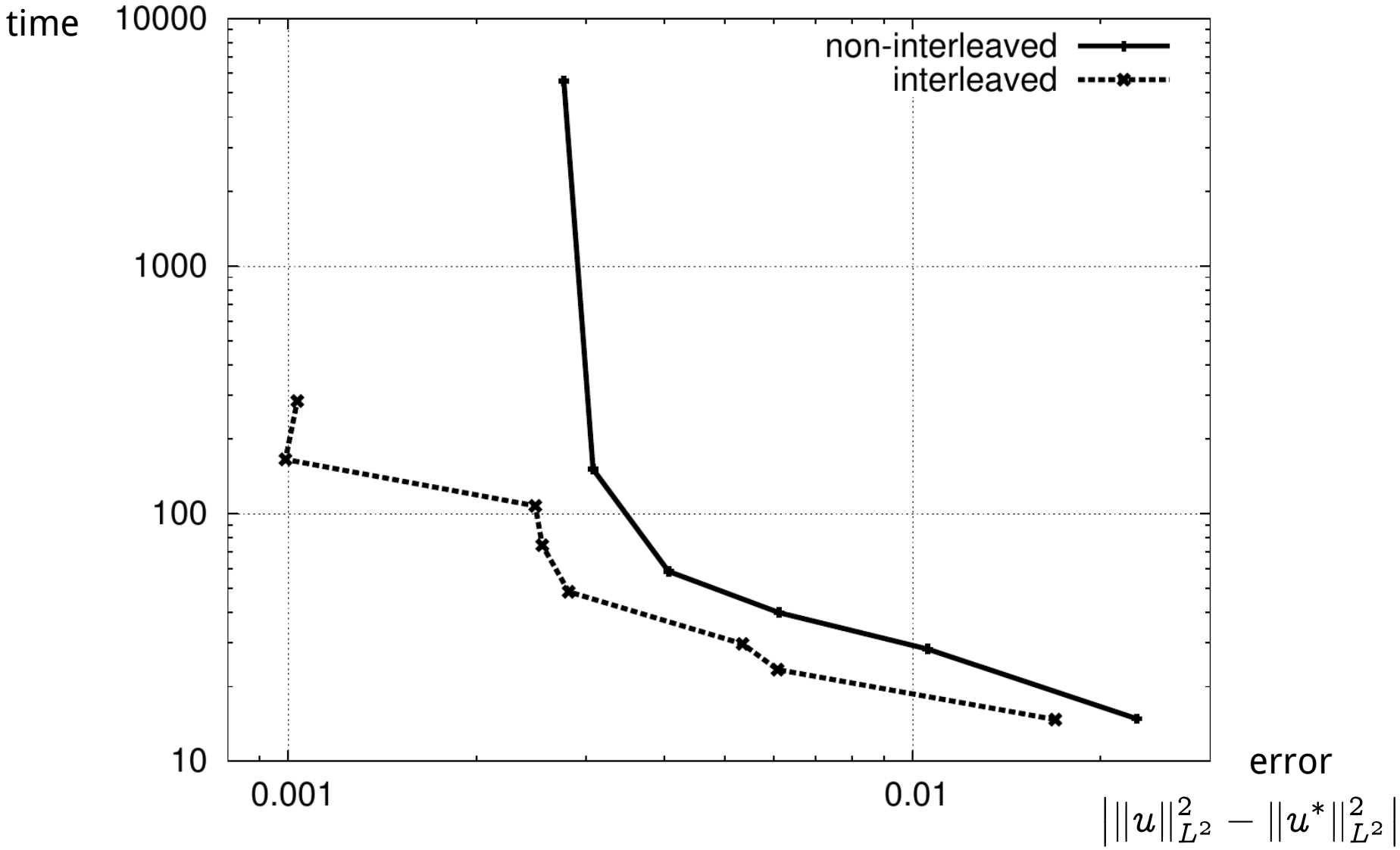
SDC

iterative scheme: correct spatial discretization errors in next sweep
➡ interleave mesh refinement and SDC sweeps

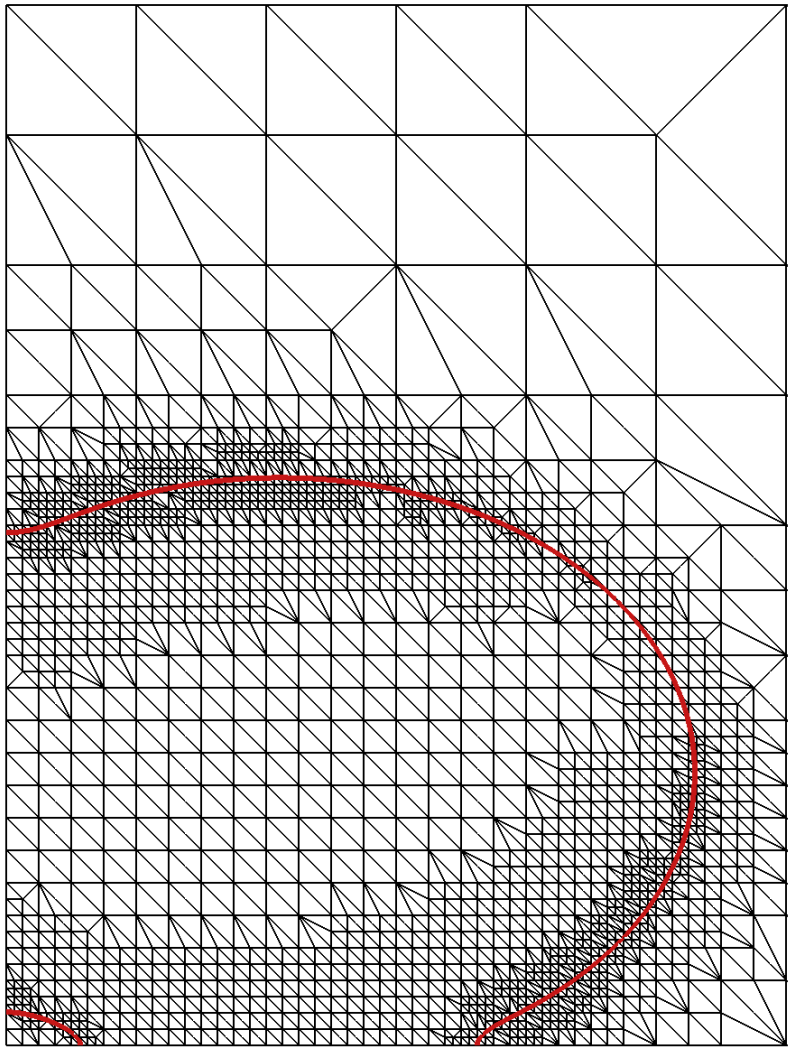
Fake Rosenbrock Refinement



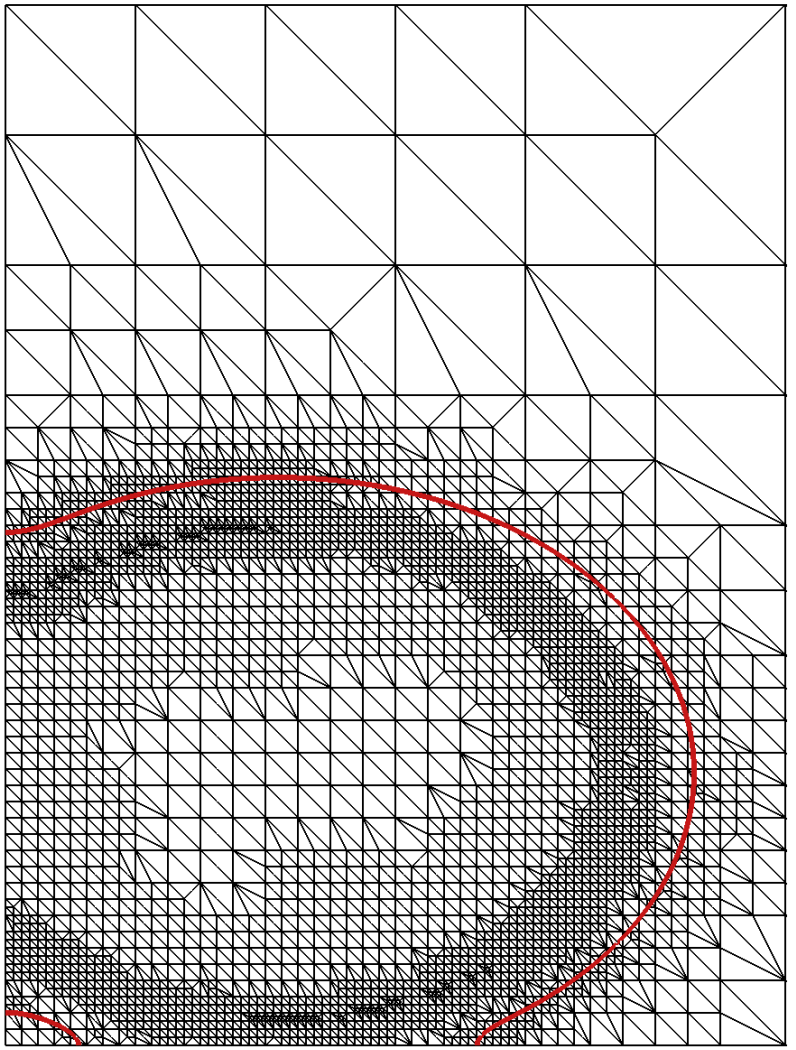
Adaptivity Modes



Adaptivity Modes: Maladaptation

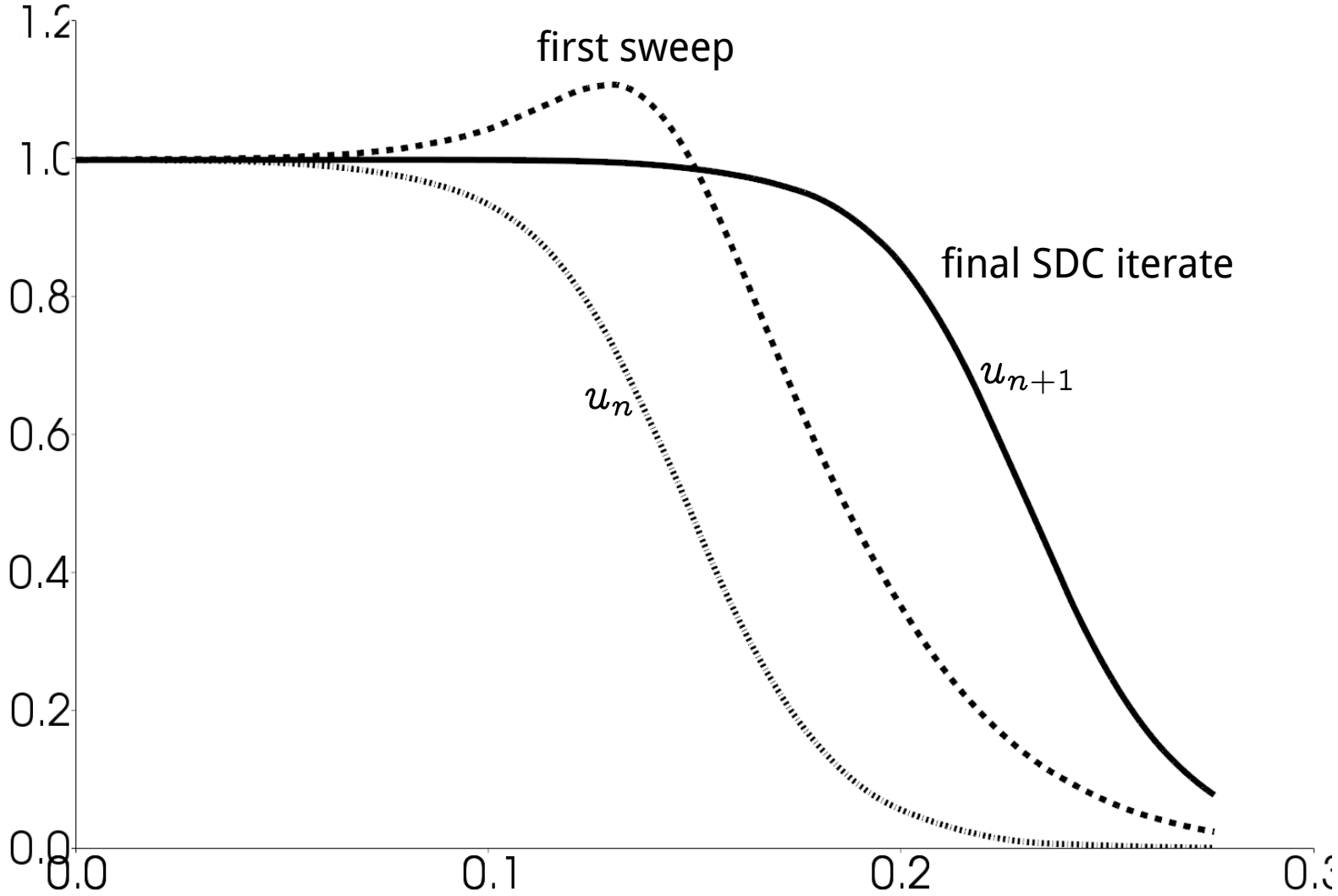


interleaved refinement

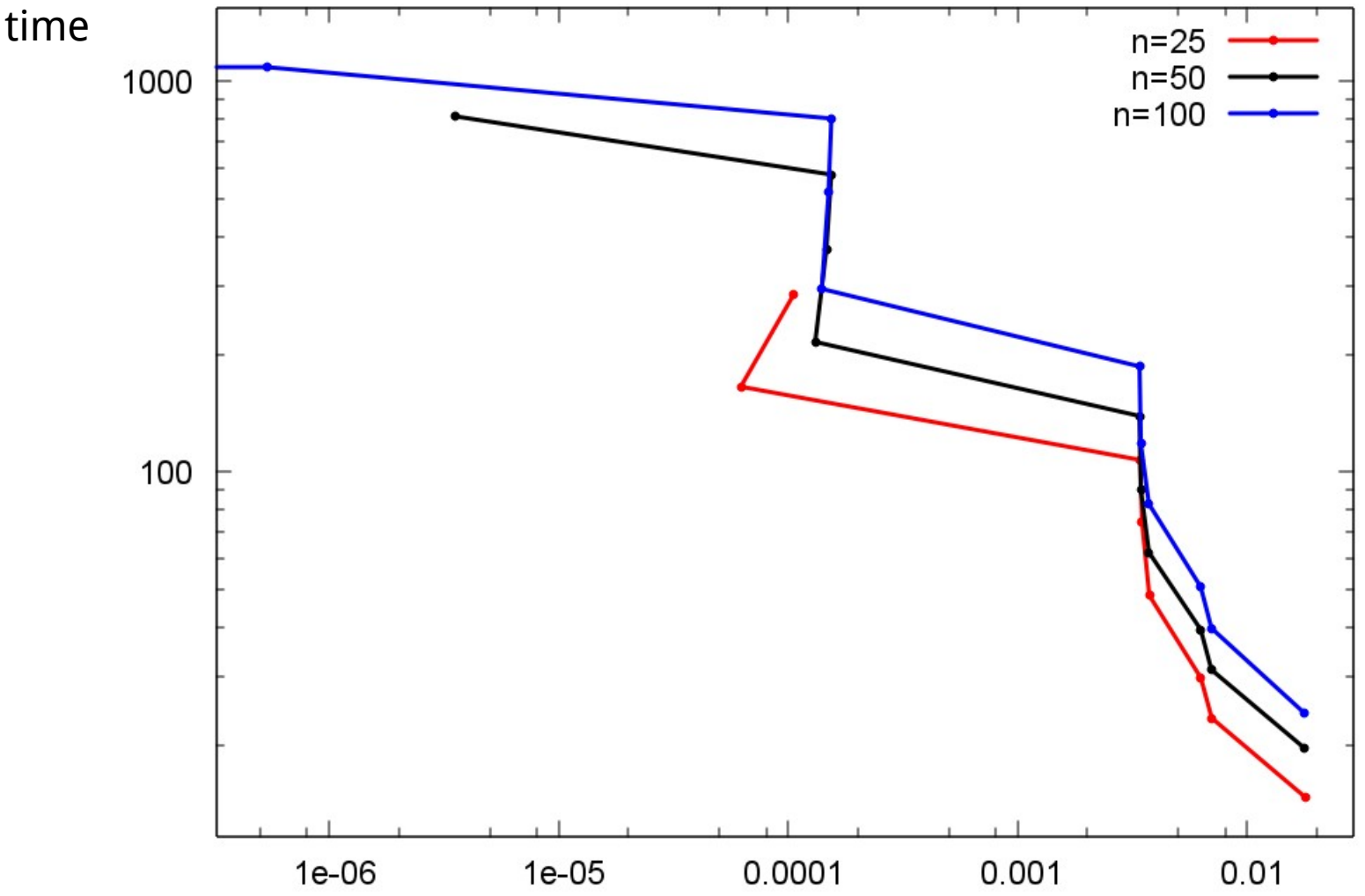


„Rosenbrock“ refinement

First Sweep (Euler)



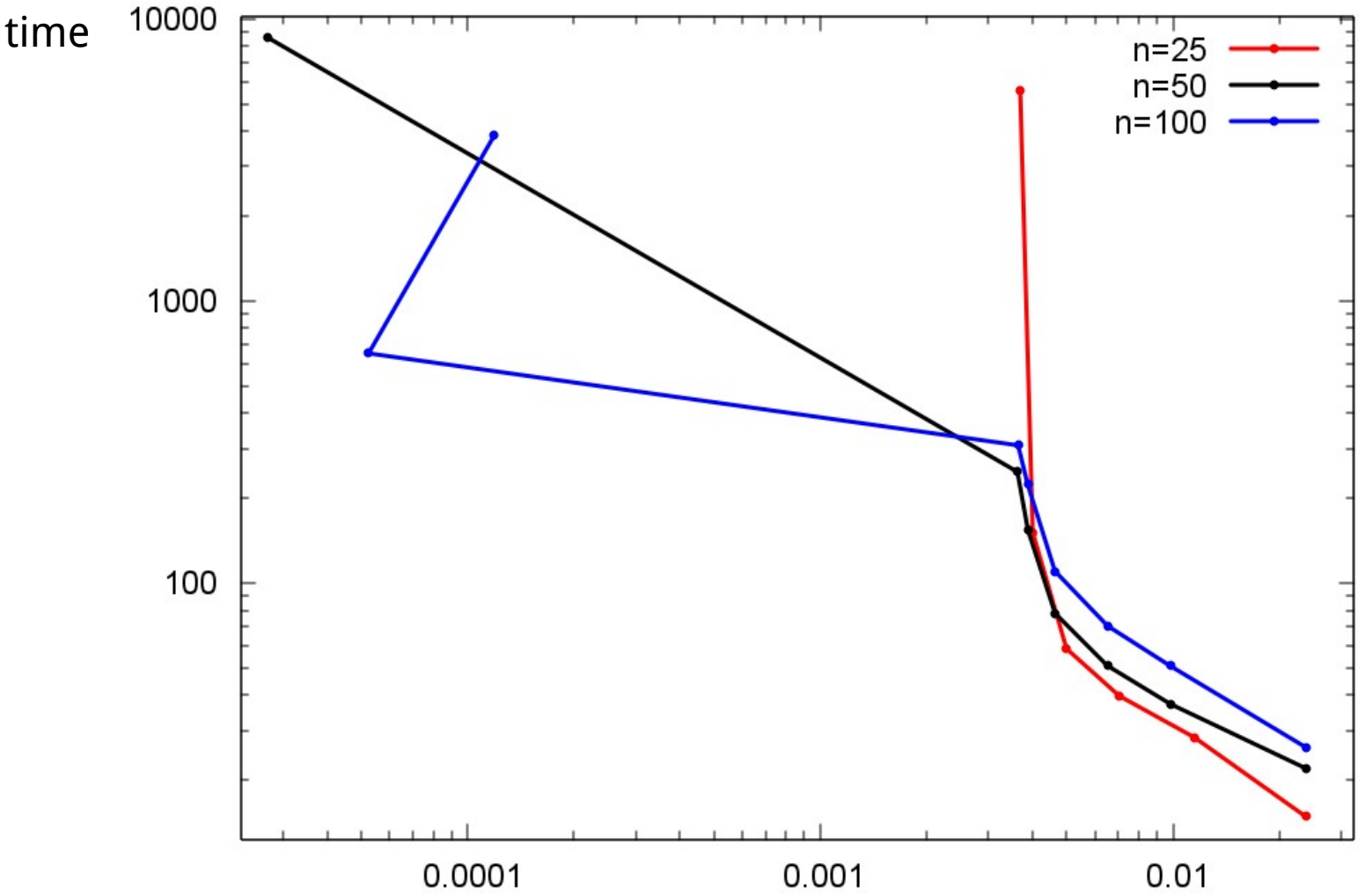
Inexact Linear System Solve



fixed number
of CG iterations
(Jacobi precond.)

interleaved refinement: inexact solve more efficient

Inexact Linear System Solve

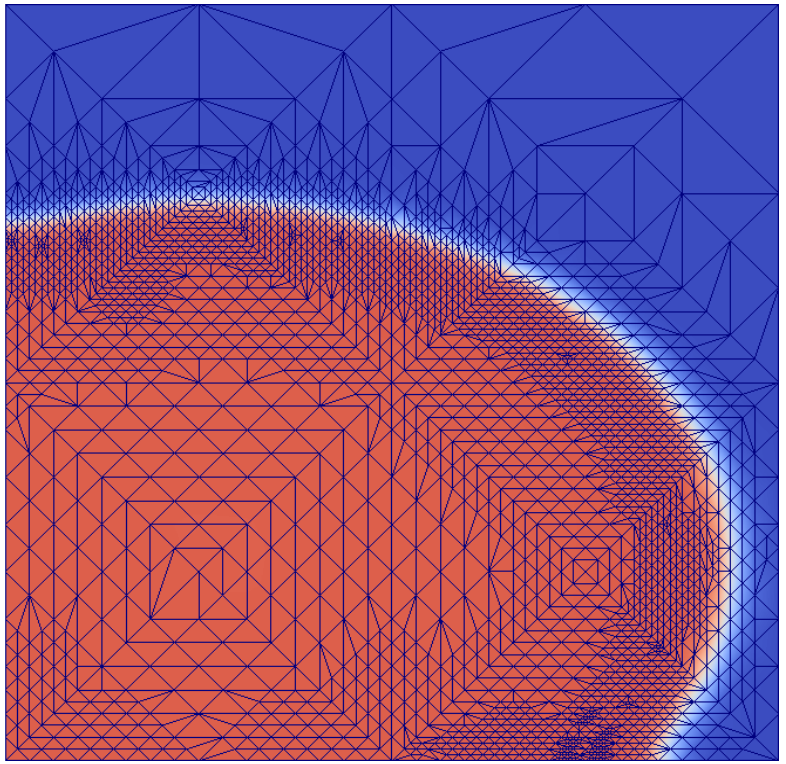


fixed number
of CG iterations
(Jacobi precond.)

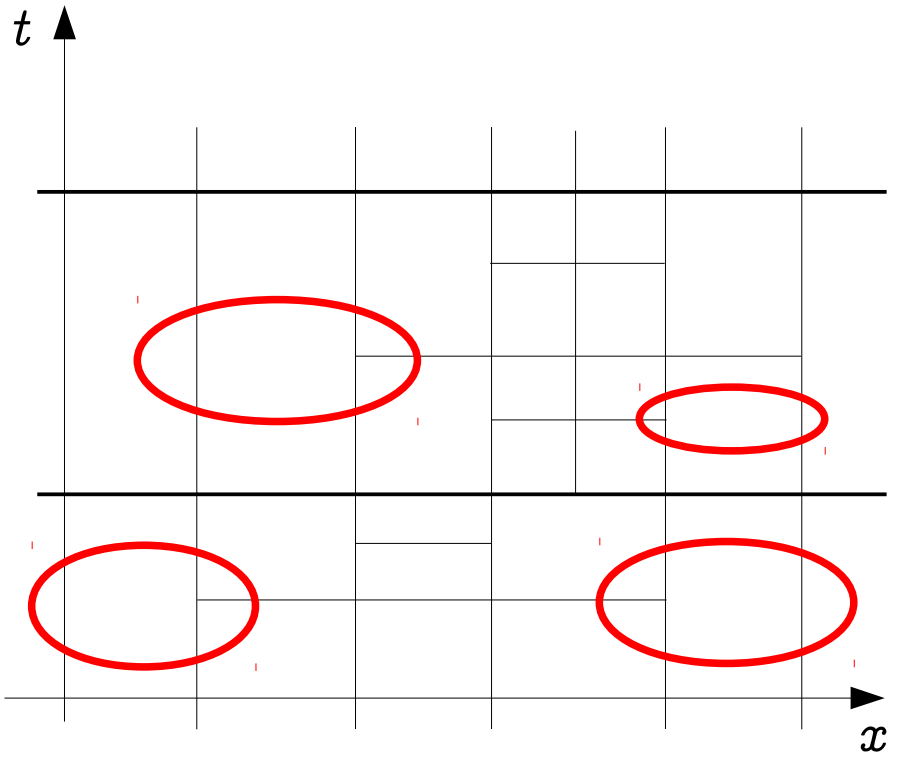
error
$\left| ||u||_{L^2}^2 - ||u^*||_{L^2}^2 \right|$

„Rosenbrock“ refinement: exact solve required for high accuracy
(improves accuracy of first sweep)

Multirate Integration I



different spatial dynamics



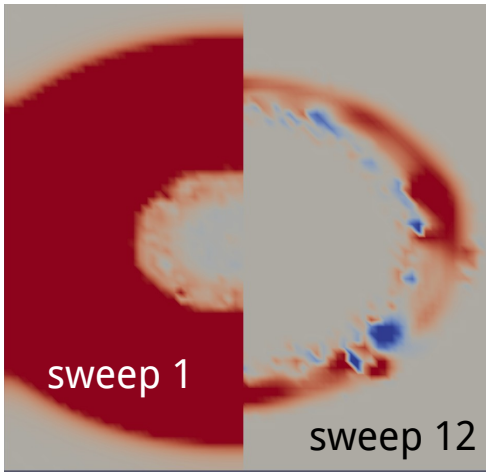
multirate integration

Multirate SDC

Spatial multirate: "algebraic adaptivity"

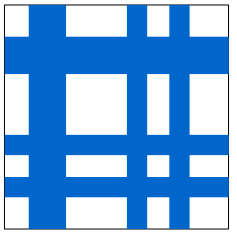
- fixed or very fine coarse grids (detailed geometry, structured grids, ...)
- temporal dynamics restricted to small parts of the domain

limit SDC improvements to spatial regions with significant dynamics by selecting DoFs

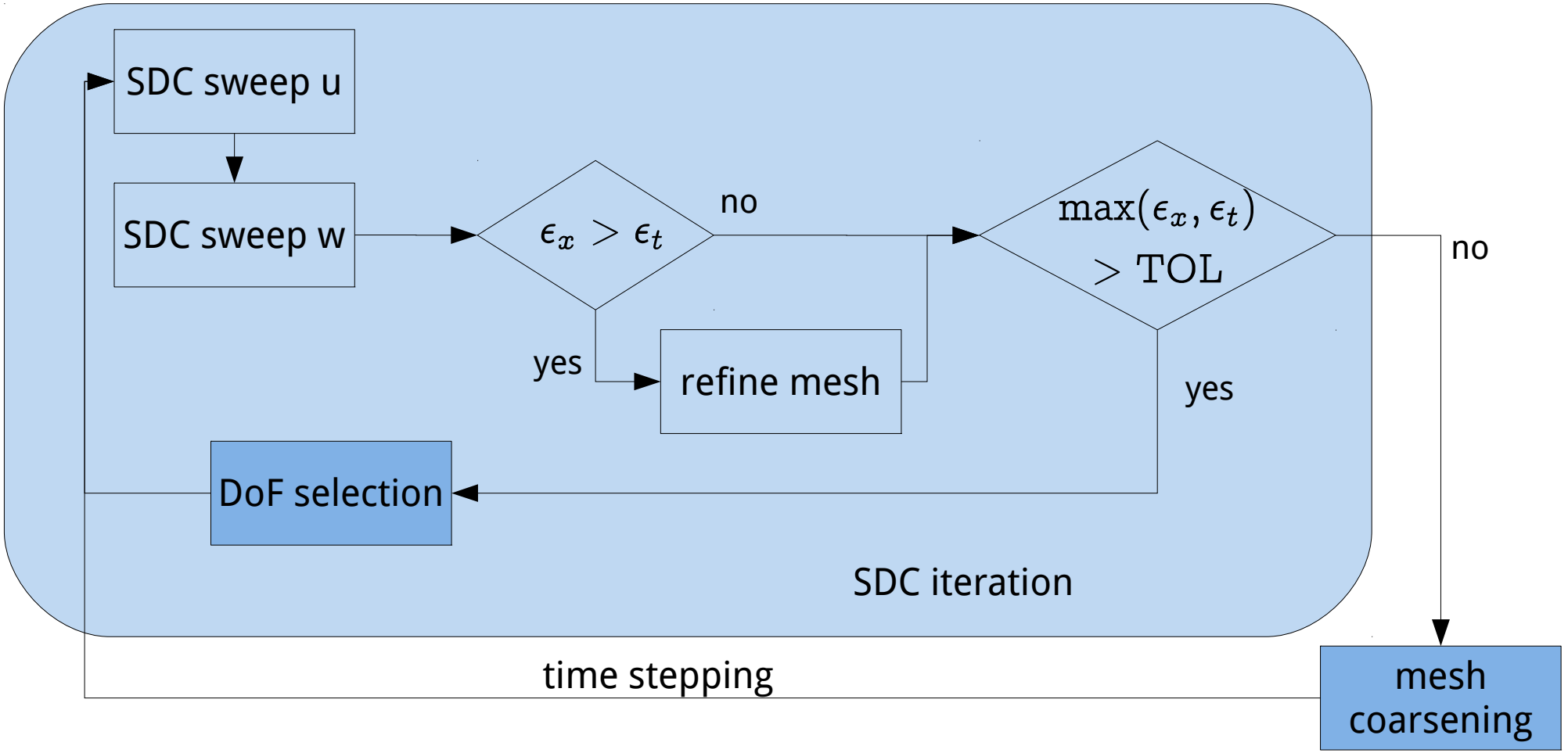


DoF selection

- include node only if its last SDC correction exceeds the drop tolerance → nested subspaces
- cheap projection of mass/stiffness matrices

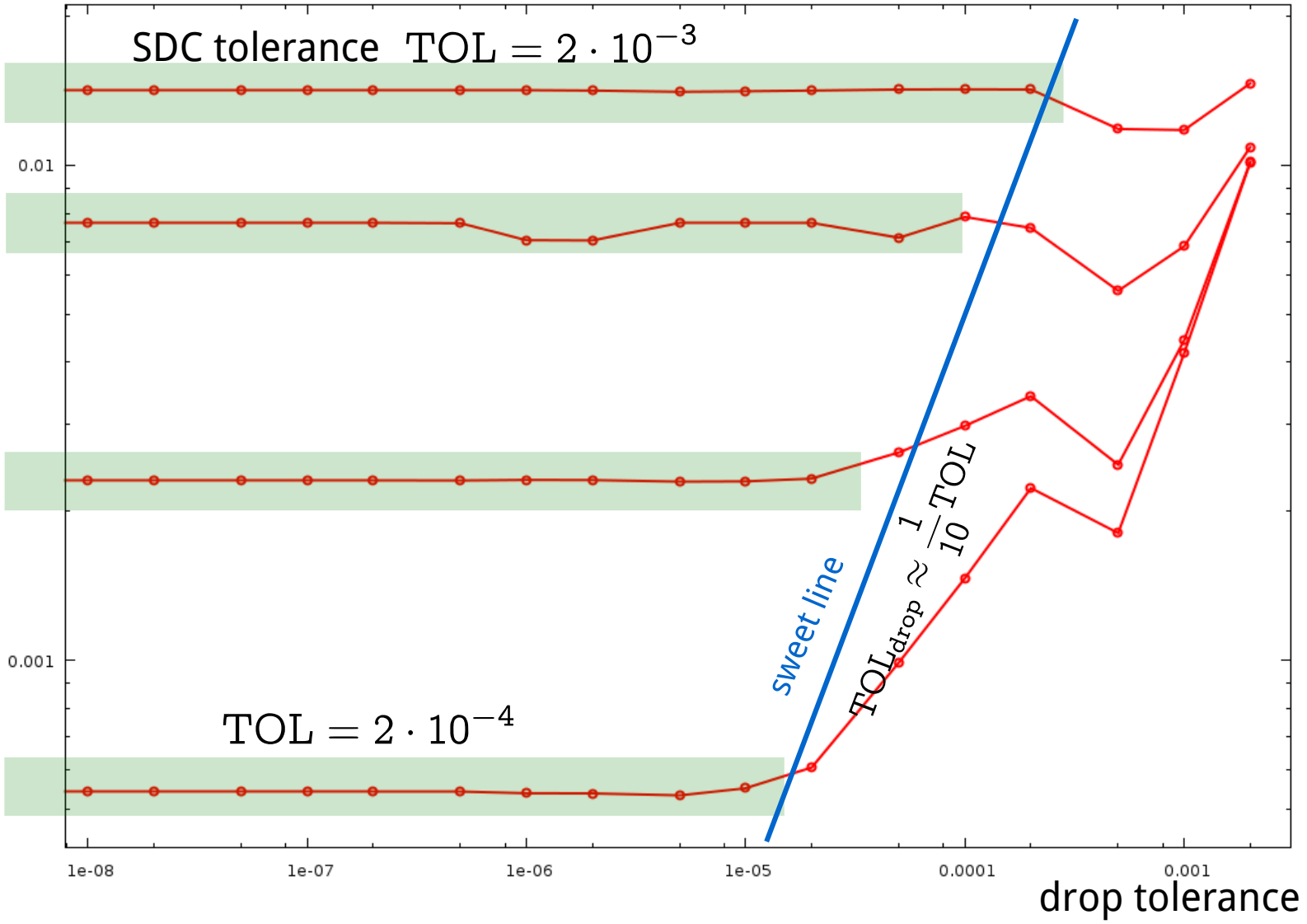


Spatial Multirate SDC



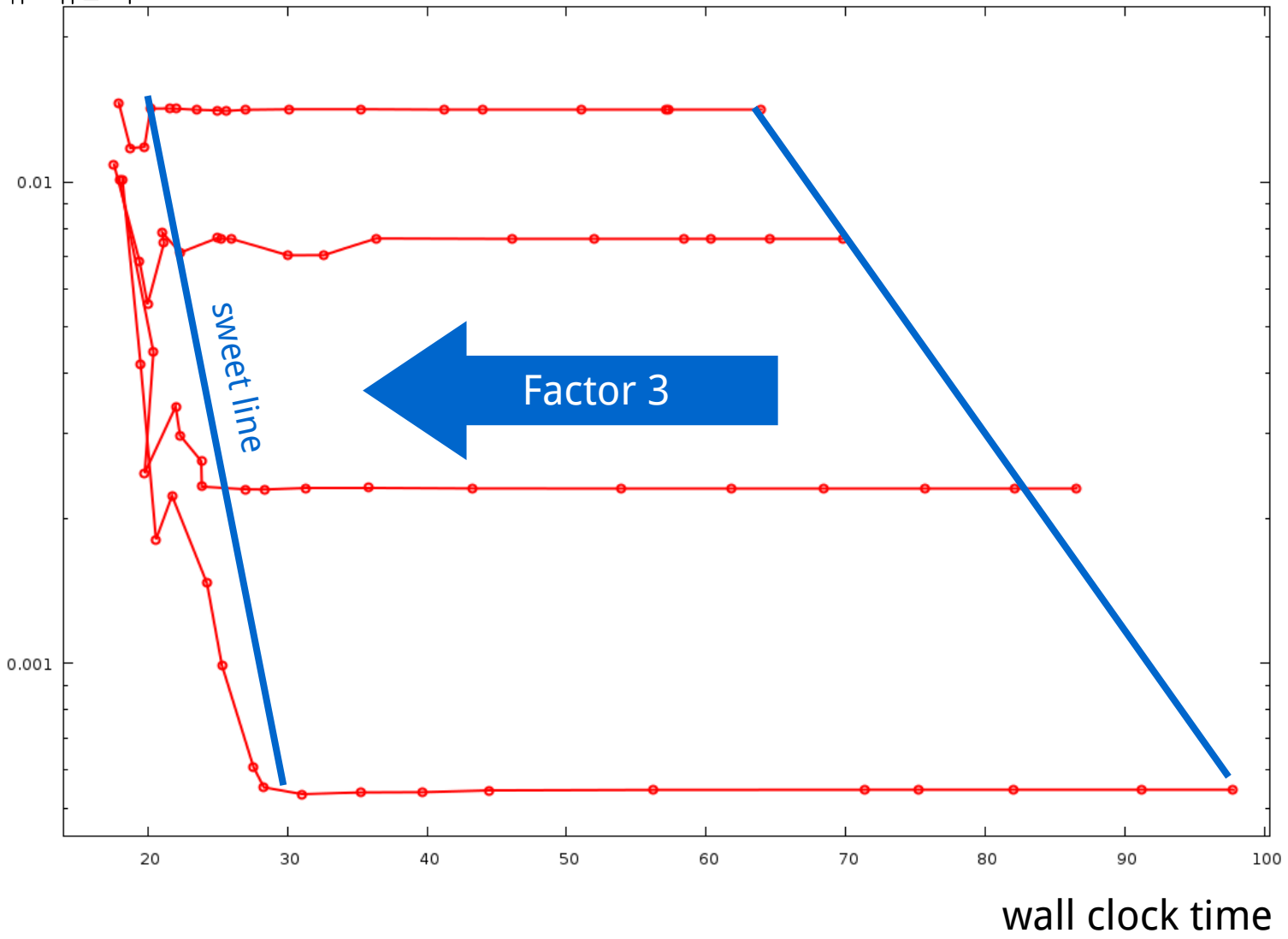
Algebraic Adaptivity (fixed grid)

$|\|u_h\|_{L^2} - \|u\|_{L^2}|$



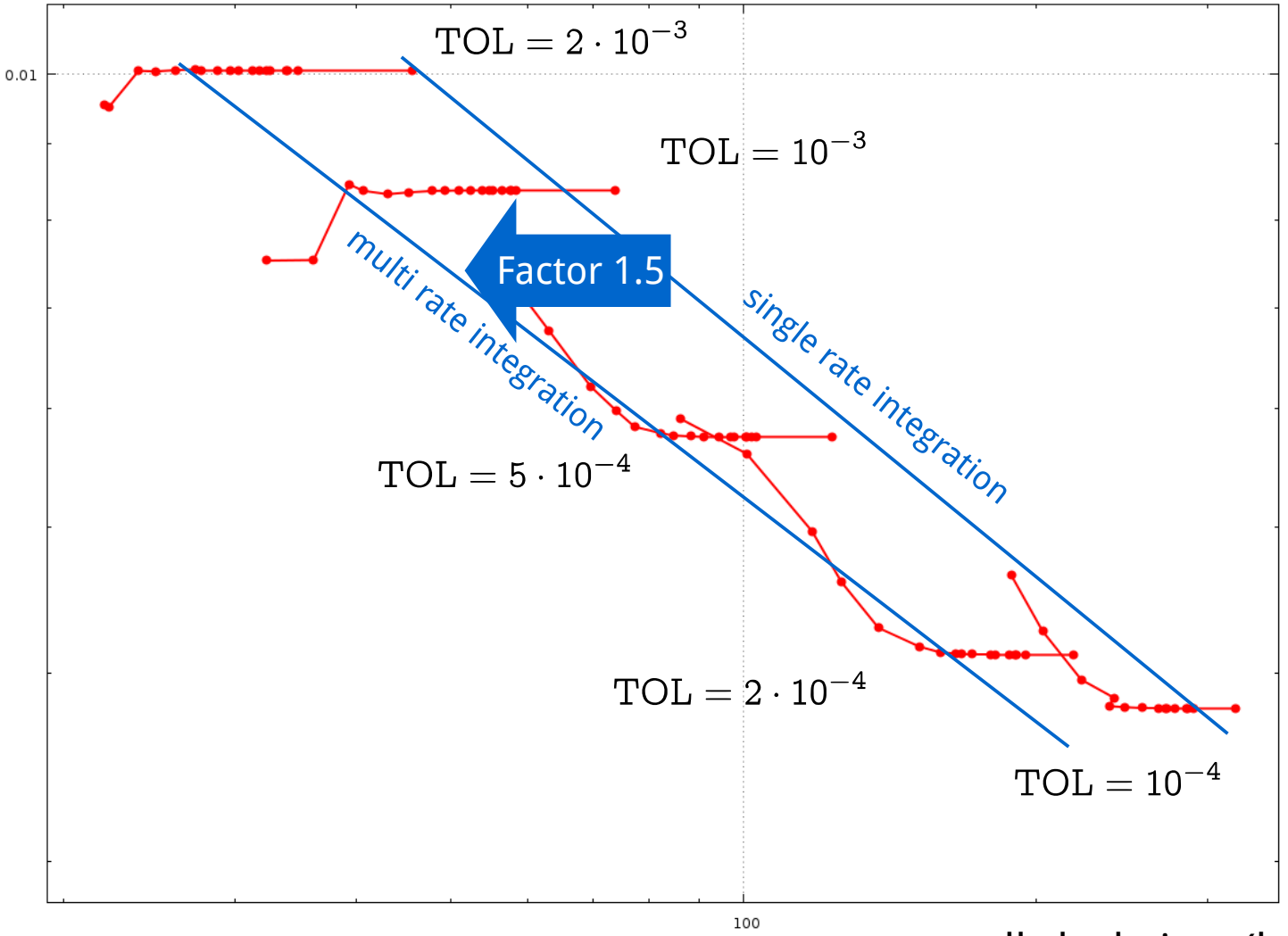
Time Savings (fixed grid)

$|\,|\|u_h\|_{L^2} - \|u\|_{L^2}|$



Time Savings (adaptive mesh refinement)

$|\|u_h\|_{L^2} - \|u\|_{L^2}|$



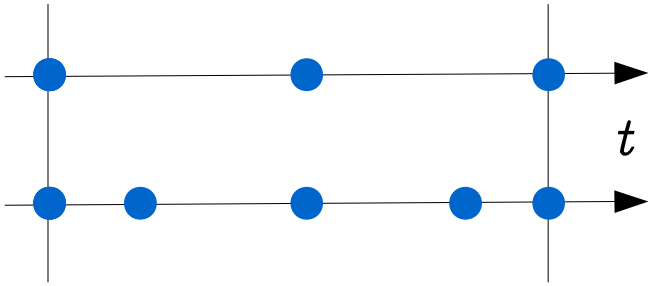
wall clock time (log scale)

Multirate Integration II

Component splitting

slow components

fast components



[Bourlioux/Layton/Minion 2003, Emmett/Zhang/Bell 2014]

Variables

- transmembrane voltage: fast
- gating variables: slow (in phenomenological models)
- mechanical contraction: slow

Electromechanic Coupling

Electrophysiology: monodomain equations

transmembrane voltage $\dot{u} = \mathrm{div}(D(y)\nabla u) + I_{\mathrm{ion}}(u, w)$

gating variables $\dot{w} = f(u, w)$

fast & local, but cheap:
- fine space discretization: cubic FE
- fine time discretization: Radau(N)

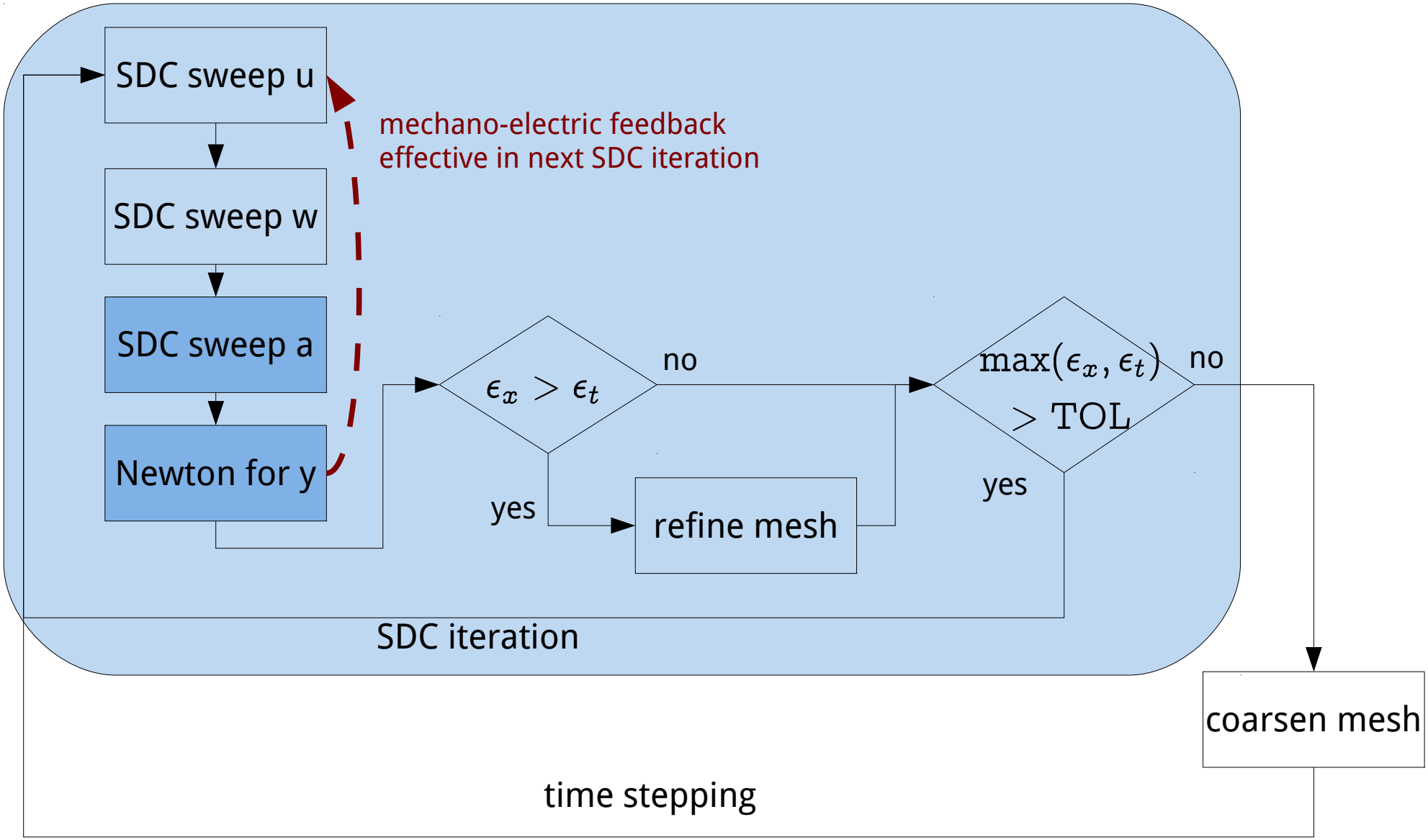
Mechanics: quasistatic hyperelasticity [Nash, Panfilov 2004]

active stress $\dot{a} = g(a, u)$

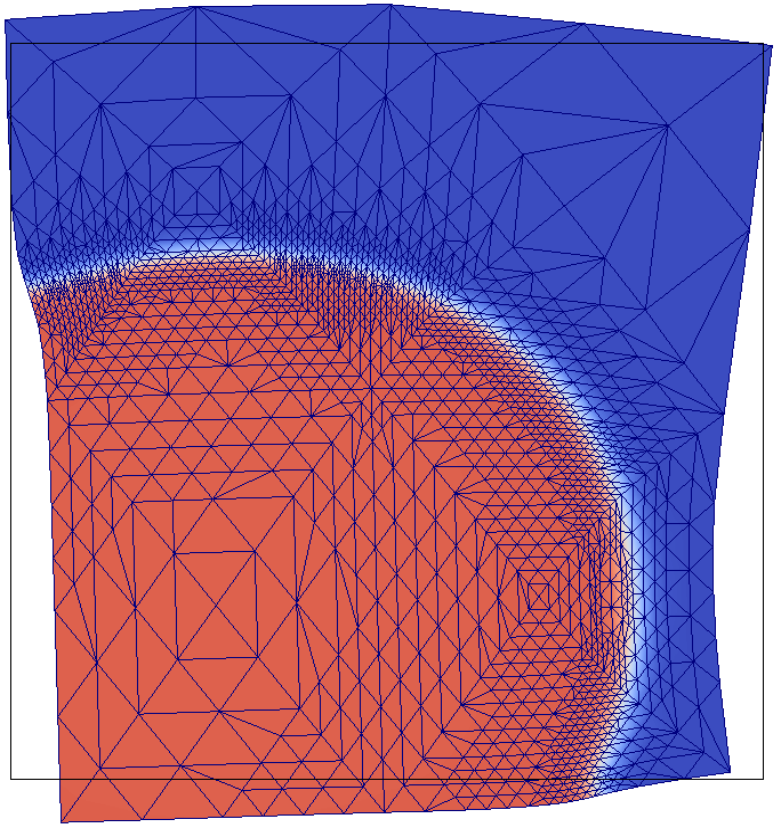
displacement $y = \arg\min_{\hat{y}} W(E(\hat{y})) + aE(\hat{y}) : (e_1^T e_1)$

slow & smooth, but expensive:
- coarse space discretization: linear FE
- coarse time discretization: Radau(n)

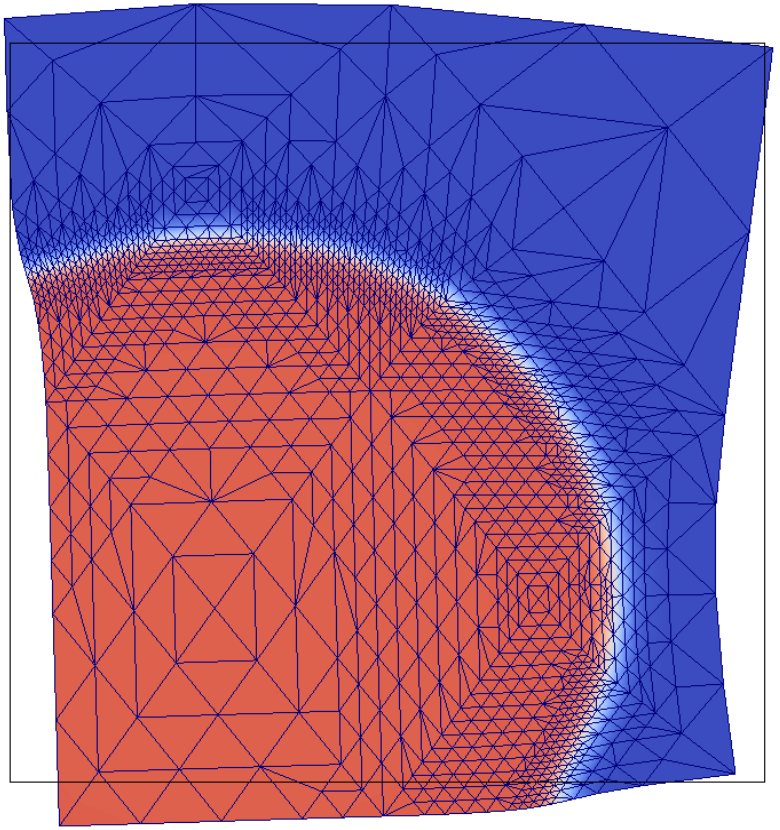
Electromechanic Coupling



Mechano-Electric Feedback



with feedback
$\|u\|_{L^2}^2 = 2.94$



without feedback
$\|u\|_{L^2}^2 = 3.01$

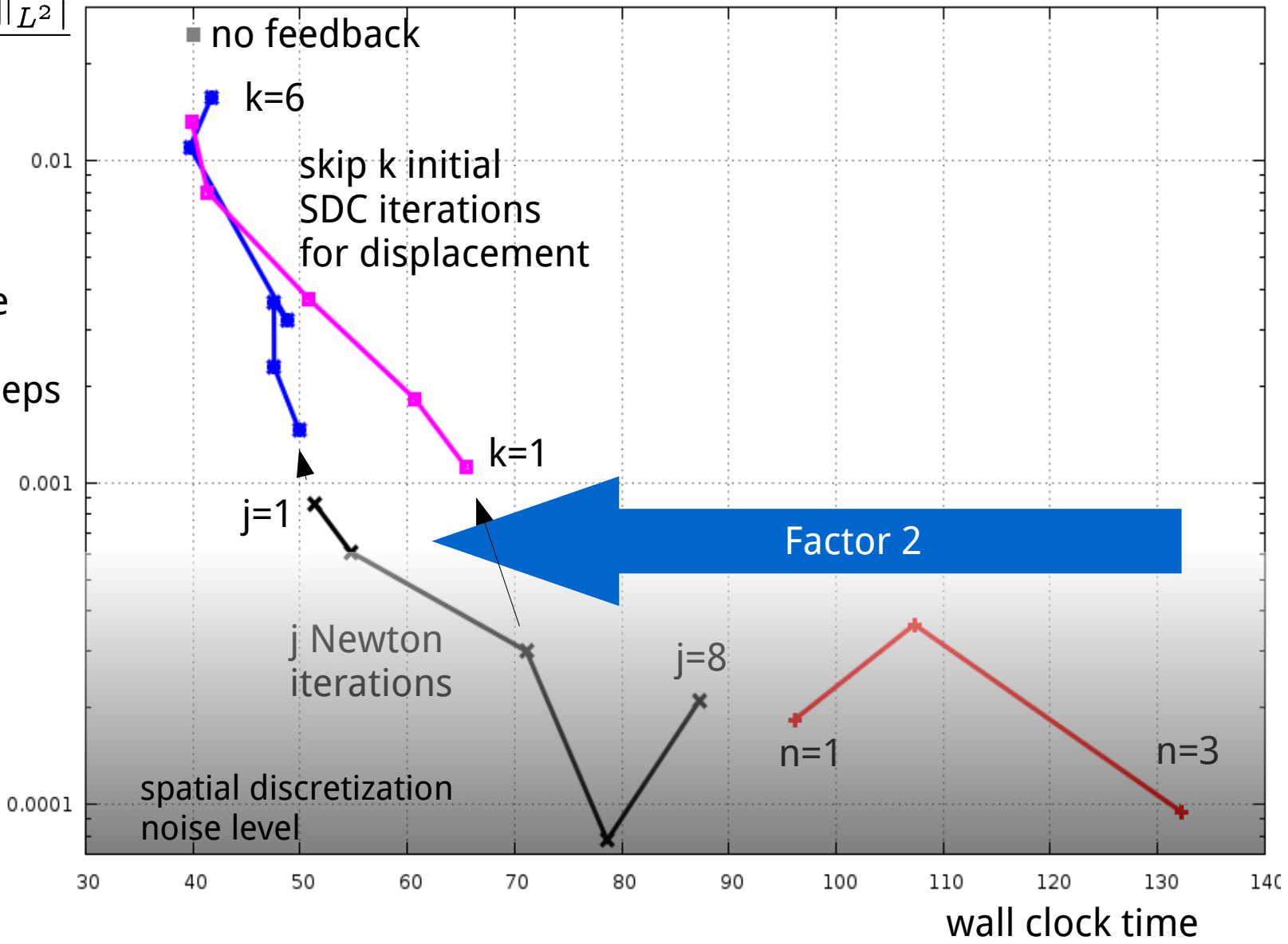
Radau(4) for transmembrane voltage $\tau = 1\text{ms}$
Radau(3) for displacement, 10 Newton-like steps

62% of time for elasticity

Feedback in SDC Iterations

$$\frac{\left| \|u_k\|_{L^2}^2 - \|u_*\|_{L^2}^2 \right|}{\|u_*\|_{L^2}^2}$$

transmembrane voltage:
Radau(4), 7 sweeps
$\tau = 2\text{ms}$



Acknowledgements

Conclusions

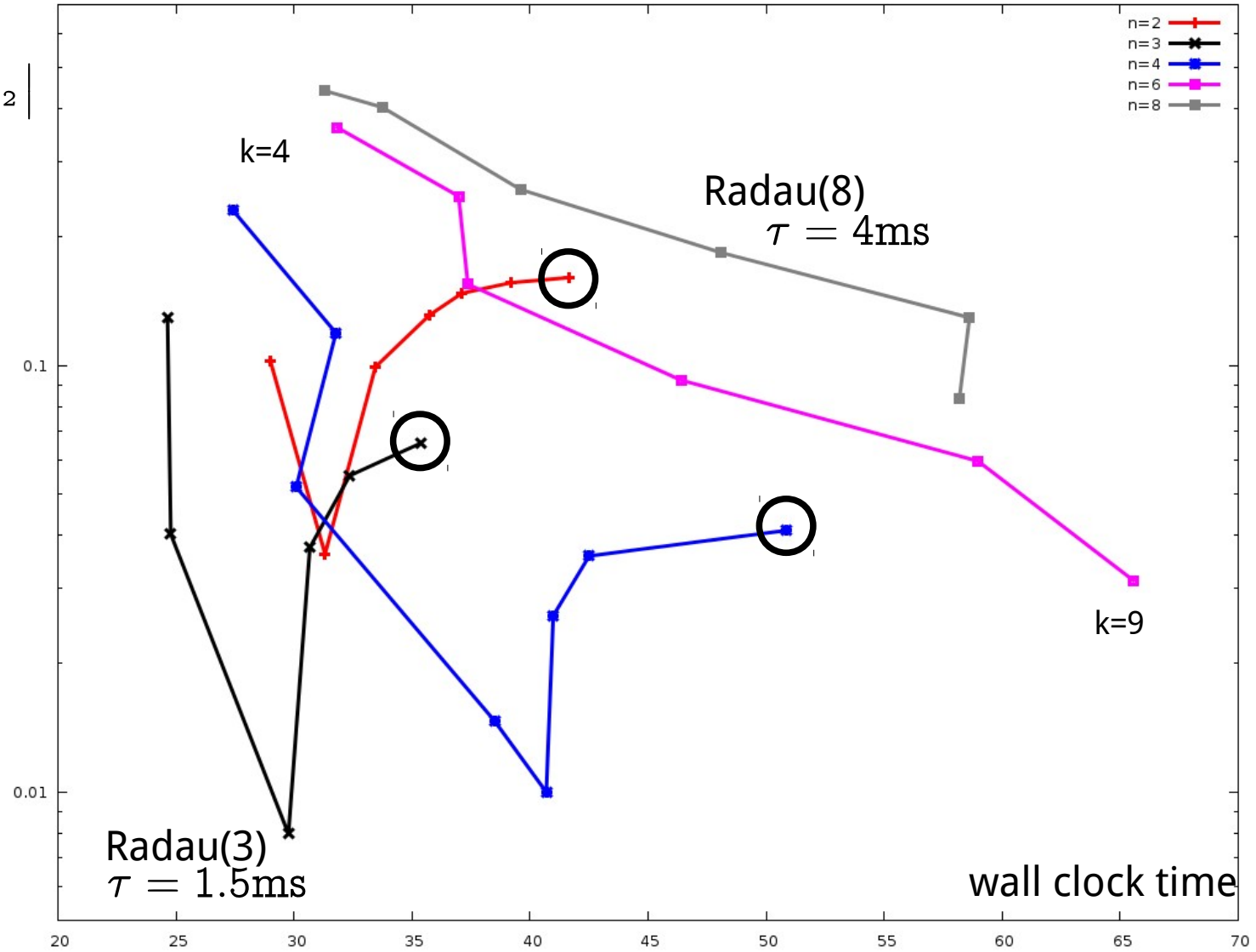
- SDC methods are attractive in combination with adaptivity, operator splitting, multi-rate integration, and multi-physics coupling
- adaptive simultaneous control of order, sweep count, mesh refinement, linear solver accuracy, algebraic coarsening, and time step size needed
- other iterative approaches may be equally attractive

Thanks for your attention!

SDC Maze: Order vs. Sweep Tradeoff

error

$$\left| \|u\|_{L^2}^2 - \|u^*\|_{L^2}^2 \right|$$



Spectral Deferred Corrections

Interpretation

SDC methods are
- ex- or diagonally implicit Runge-Kutta schemes
- matrix-decomposition fixed-point solvers for collocation systems
- inexact Newton methods for collocation systems

Why SDC?

- amortize overhead of mesh adaptivity over more stages / longer time steps
- re-use of computed values on mesh refinement & time step reduction
- multi rate integration
- construct high-order, highly accurate schemes
- efficient parallelization in time
- efficient multiphysics coupling

A Linear Algebra View

Collocation $Du = If(u) \Leftrightarrow Iu = Sf(u) \qquad u \in \mathbb{P}_n$

differentiation quadrature

Newton $(D - f')\delta u = f - Du \Leftrightarrow (I - Sf')\delta u = Sf(u) - u$

DSDC $(\hat{D} - If')\delta u = If - Du, \quad \hat{D}$ lower triangular

QSDC $(I - \hat{S}f')\delta u = Sf - Iu, \quad \hat{S}$ lower triangular

Joint notation $(\hat{D} - \hat{S}f')\delta u = Sf - Du, \quad \hat{D}, \hat{S}$ lower triangular

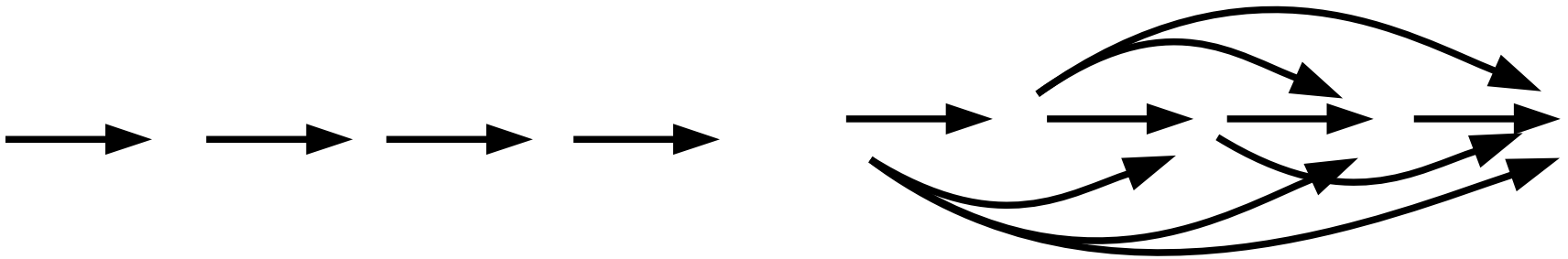
SDC with DIRK Sweeps

choice of arbitrary $\hat{D}, \hat{S}$

design decisios / restrictions:

- lower triangular → sweep structure
- lower triangular → one sweep is one DIRK step
- aim at PDEs/iterative solvers (no re-use of factorizations) → no **S**DIRK

Euler sweep ➡ DIRK sweep [Ws '13]



Euler sweep DIRK sweep

Nilpotent DIRK sweeps

SDC on Dahlquist equation $\dot{u} = -\lambda u,\ \lambda \geq 0$

$(\hat{D} - \hat{S} f')\delta u = Sf - Du$ ➡ fixed point iteration with iteration matrix

$$G(\lambda) = I - (\hat{D} + \lambda \hat{S})^{-1}(D + \lambda S)$$

Asymptotic convergence

spectral radius $\rho(G(\lambda))$ ➡ construct $\hat{D}, \hat{S}$ such that $\rho(G(\lambda)) = 0$

Idea $(D + \lambda S)^T = LU \ \Rightarrow\ U^{-T}(D + \lambda S) = L^T$

choose $\hat{D} + \lambda \hat{S} = U^T \ \Rightarrow\ G(\lambda) = N$ nilpotent of order n

DSDC: $S = I, \hat{S} = S \ \Rightarrow \rho(G(\infty)) = 0$

nilpotence for $\lambda = 0$: $\hat{D} = U^T,\ D^T = LU$

QSDC: $D = I, \hat{D} = D \ \Rightarrow \rho(G(0)) = 0$

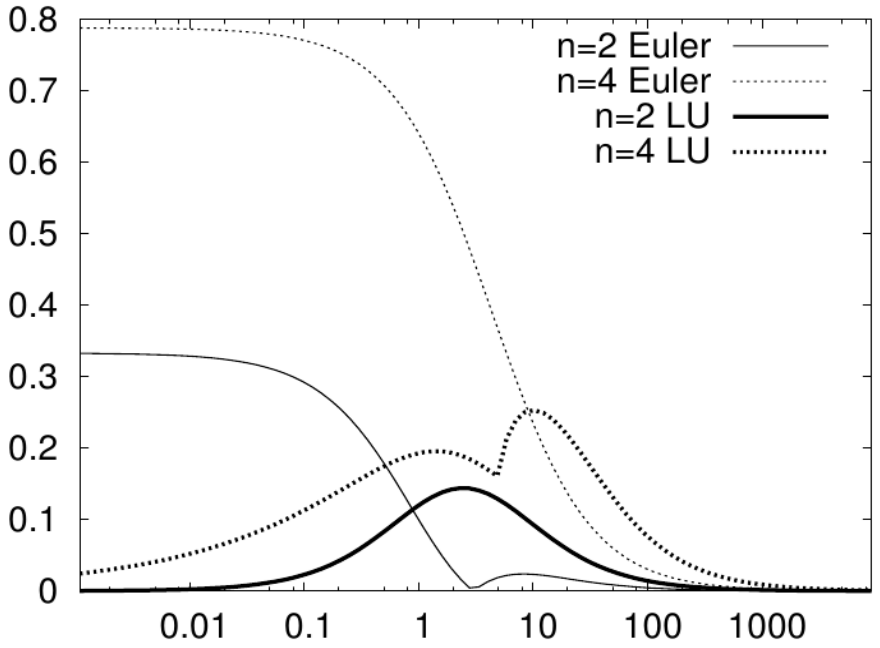
nilpotence for $\lambda = \infty$: $\hat{S} = U^T,\ S^T = LU$

Nilpotent DIRK Sweeps by LU

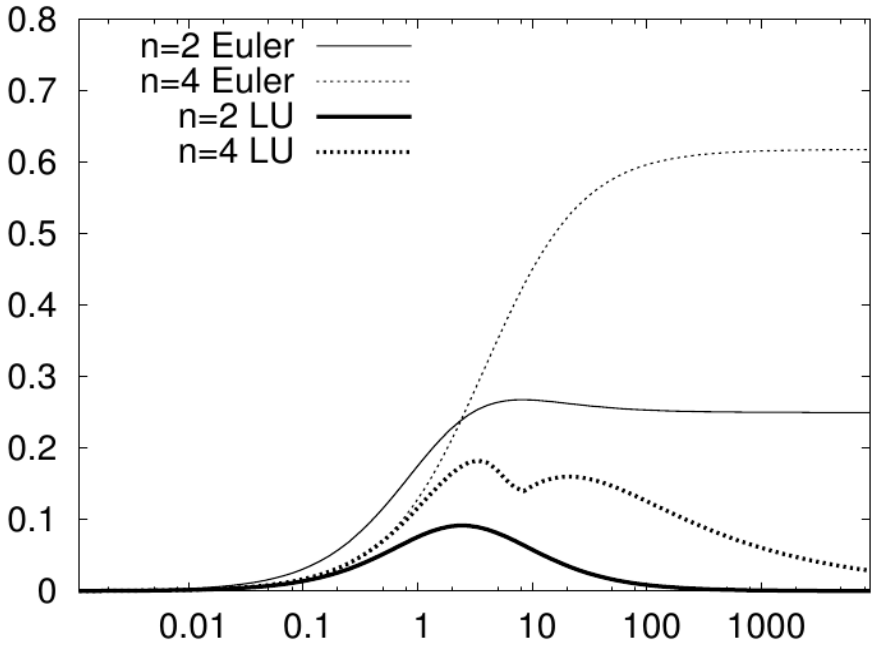
Theorem In DSDC, with $\hat{D} = U^T,\ D^T = LU$, we obtain $\rho(G(\lambda)) = \mathcal{O}(\lambda^{1/n})$

➡ one order every n steps

An analogous result holds for QSDC.



DSDC on RadauIIa



QSDC on RadauIIa

Direct Optimization of DIRK Sweeps

Idea Choose $\hat{D}, \hat{S}$ such as to minimize the integration error.

Objectives

spectral radius $\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda)\rho(G(\lambda))$

error norm $\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda)\|G(\lambda)\|$

final time error $\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda)\|e_n^T G(\lambda)\|$

Sweep blocks perform m sweeps

final time error $\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda)\|e_n^T G(\lambda)^m\|^{1/m}$

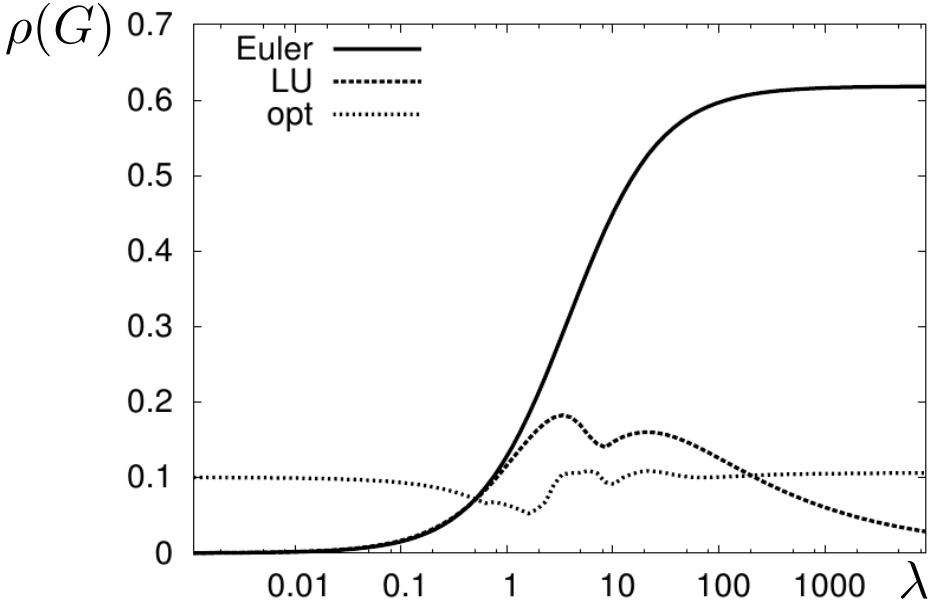
Flexible sweep blocks

final time error $\min_{\hat{D}_k,\hat{S}_k} \max_{\lambda \geq 0} w(\lambda)\|e_n^T \prod_{k=1}^{m} G_k(\lambda)\|^{1/m}$

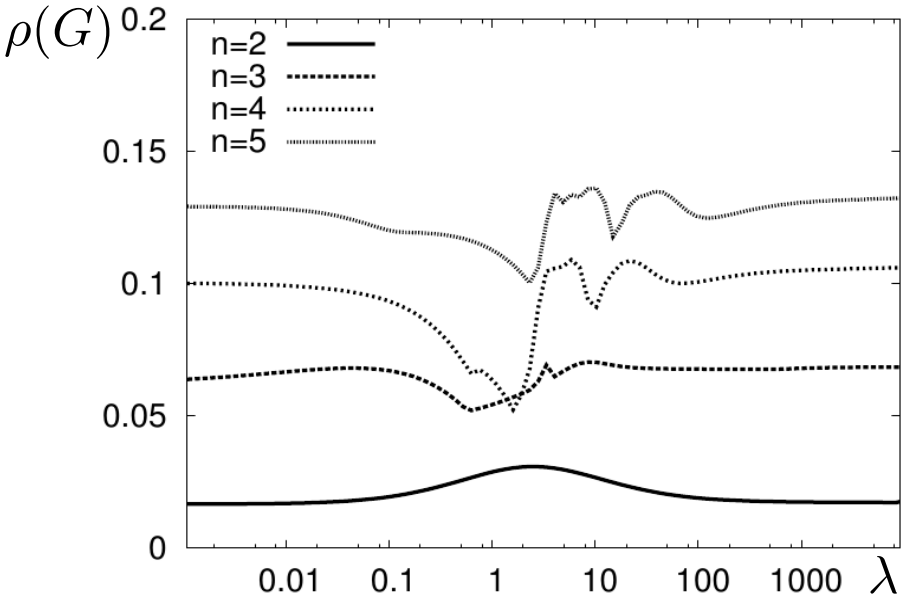
Spectral Radius Minimization

$$\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda)\rho(G(\lambda))$$

QSDC on RadauIIa points



$n = 4$

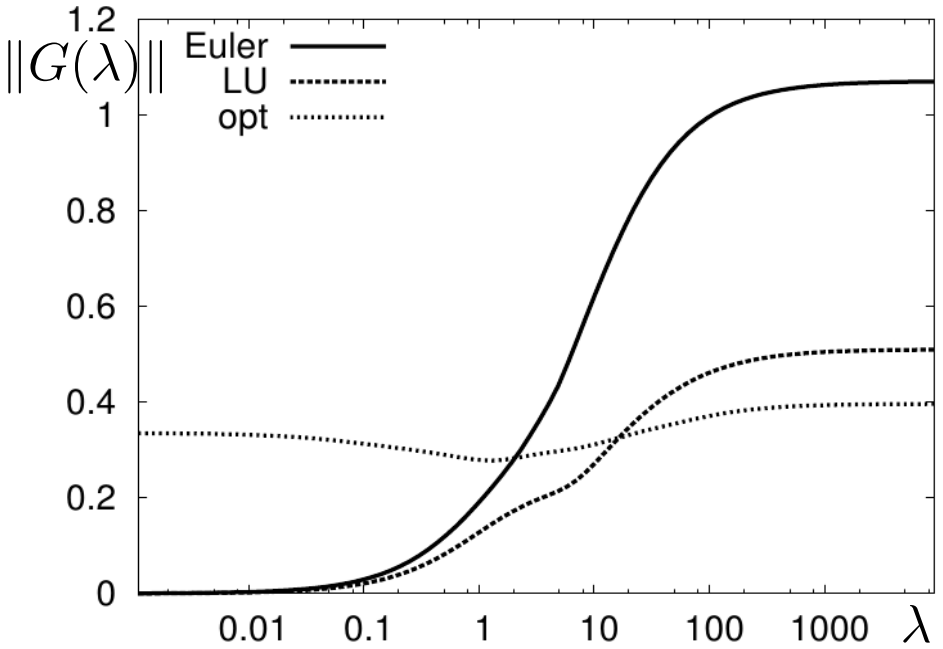


$n = 2, \ldots, 5$

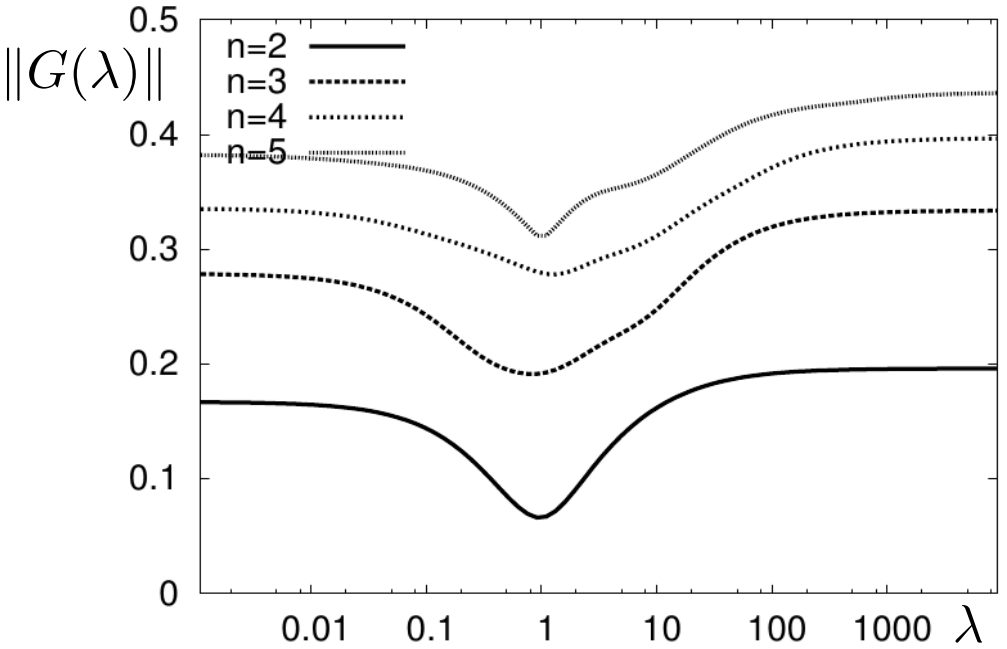
Error Norm Minimization

$$\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda) \|G(\lambda)\|$$

QSDC on RadauIIa points



$n = 4$

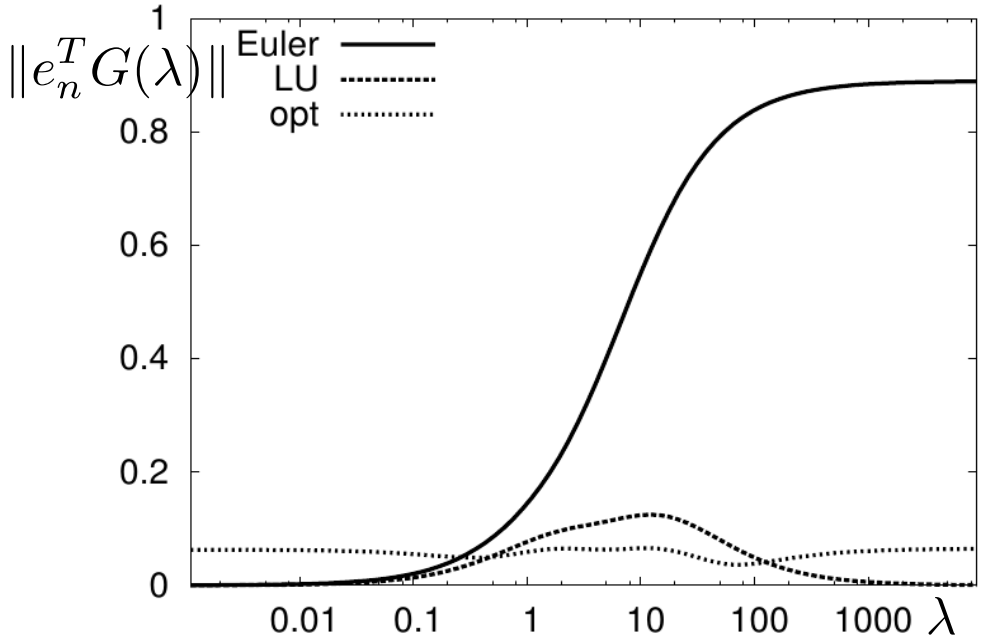


$n = 2, \ldots, 5$

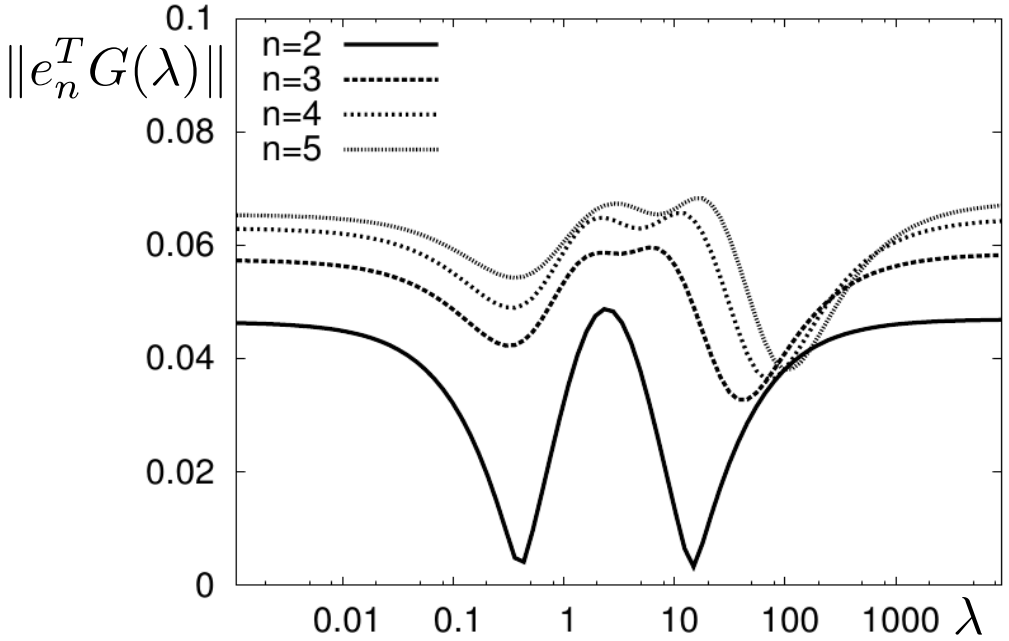
Final Time Error Minimization

$$\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda) \| e_n^T G(\lambda) \|$$

QSDC on RadauIIa points



$n = 4$

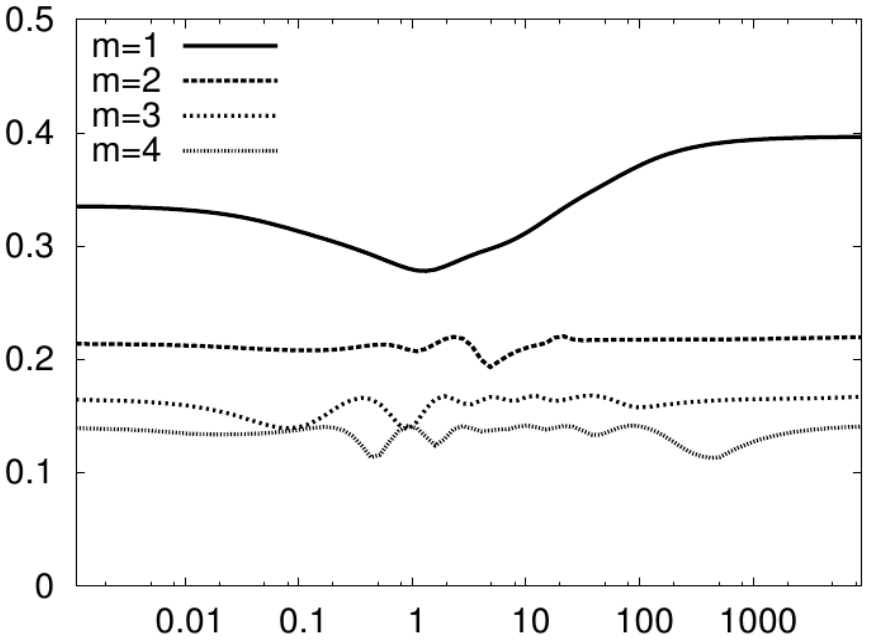


$n = 2, \ldots, 5$

Sweep Blocks

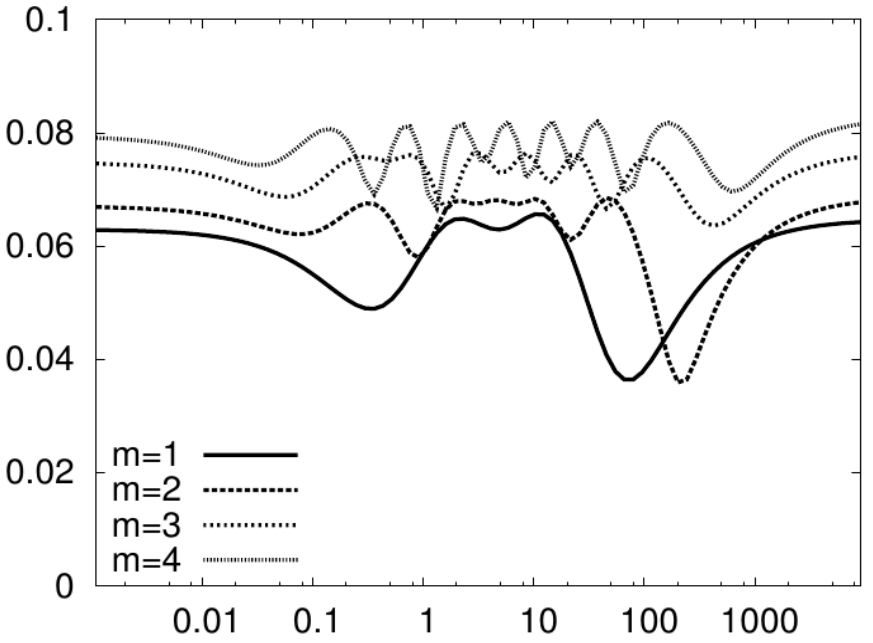
Optimize for not one, but several SDC iterations.

QSDC on RadauIIa(4) points



$$\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda) \| G(\lambda)^m \|^{1/m}$$

average error norm reduction



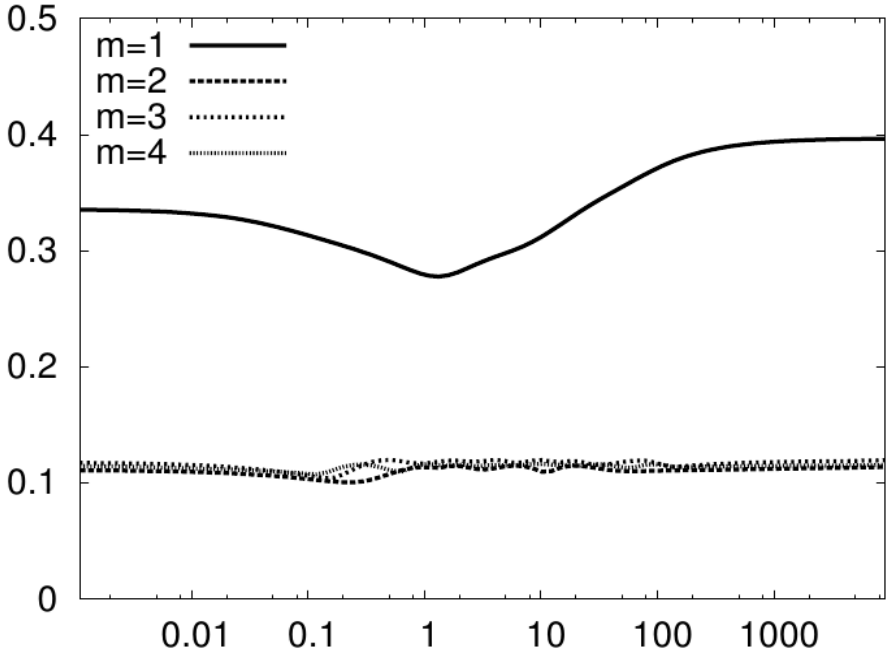
$$\min_{\hat{D},\hat{S}} \max_{\lambda \geq 0} w(\lambda) \| e_n^T G(\lambda)^m \|^{1/m}$$

average final time error reduction

Flexible Sweep Blocks

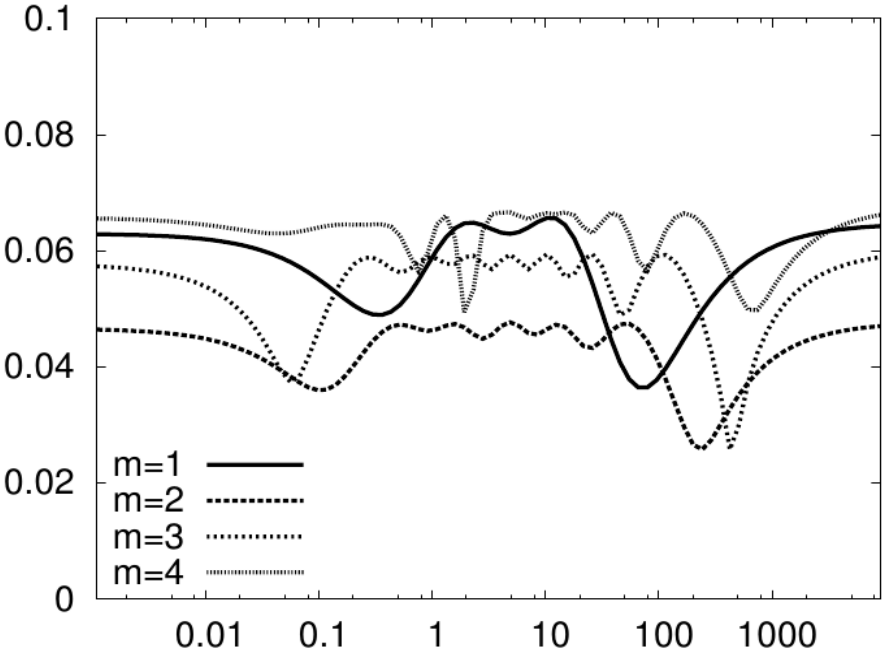
Optimize for not one, but several SDC iterations, each with its own DIRK sweep.

QSDC on RadauIIa(4) points



$$\min_{\hat{D}_k,\hat{S}_k} \max_{\lambda \geq 0} w(\lambda) \| \prod_{k=1}^{m} G_k(\lambda) \|^{1/m}$$

average error norm reduction



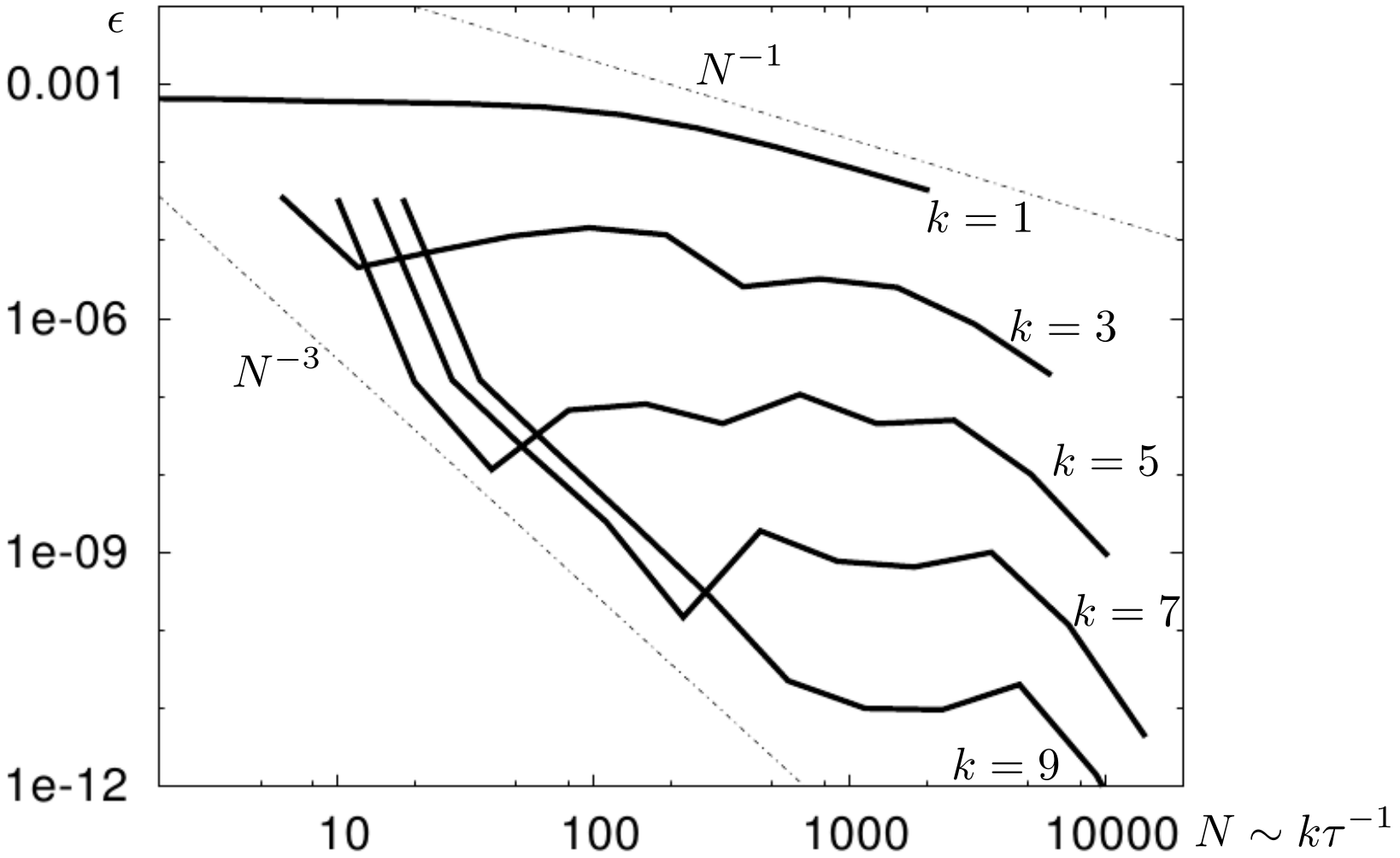
$$\min_{\hat{D}_k,\hat{S}_k} \max_{\lambda \geq 0} w(\lambda) \| e_n^T \prod_{k=1}^{m} G_k(\lambda) \|^{1/m}$$

average final time error reduction

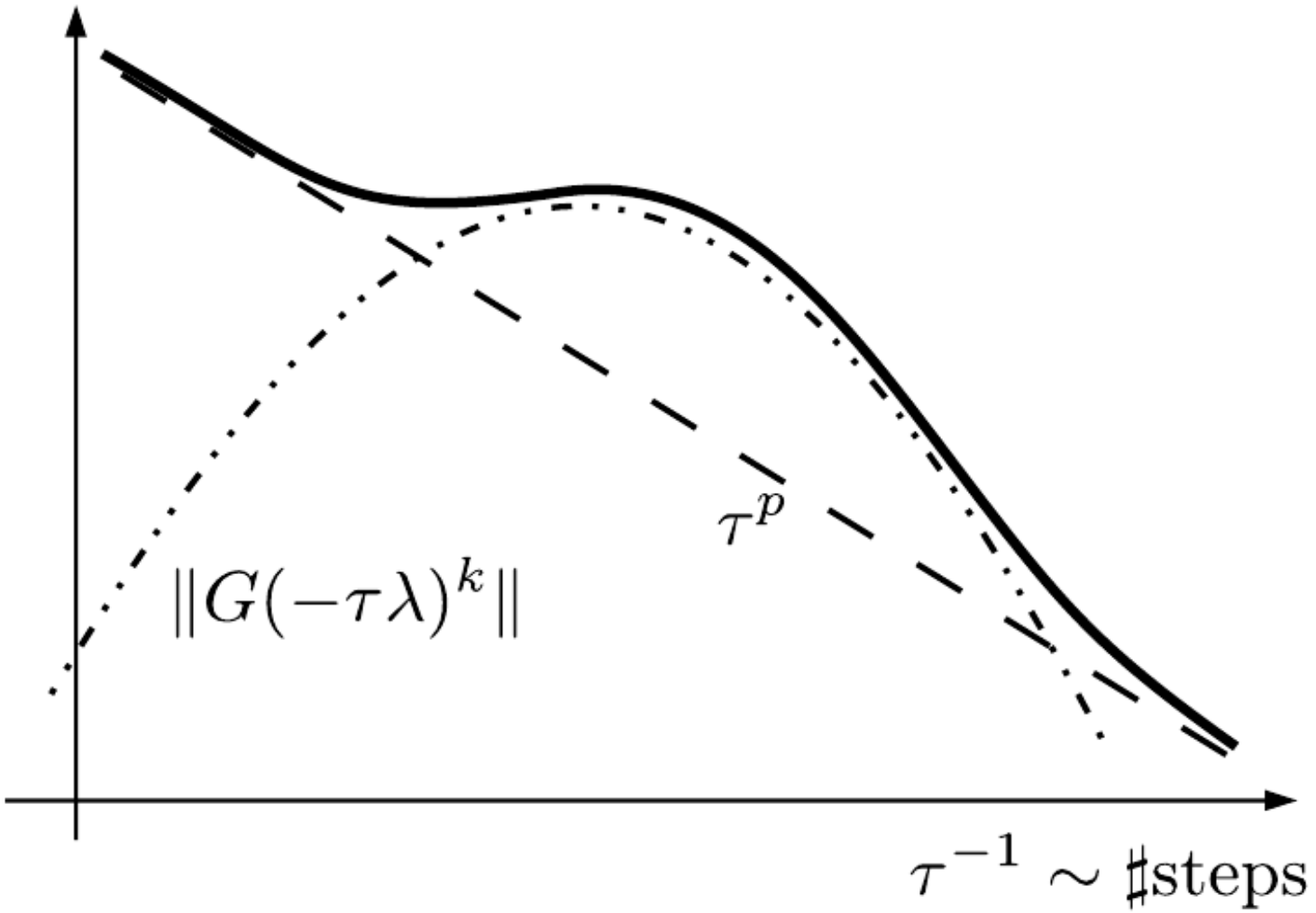
Numerical Example: Prothero-Robinson

$$\dot{u} = -10^3(u - \sin) + \cos,\ u(0) = 0$$

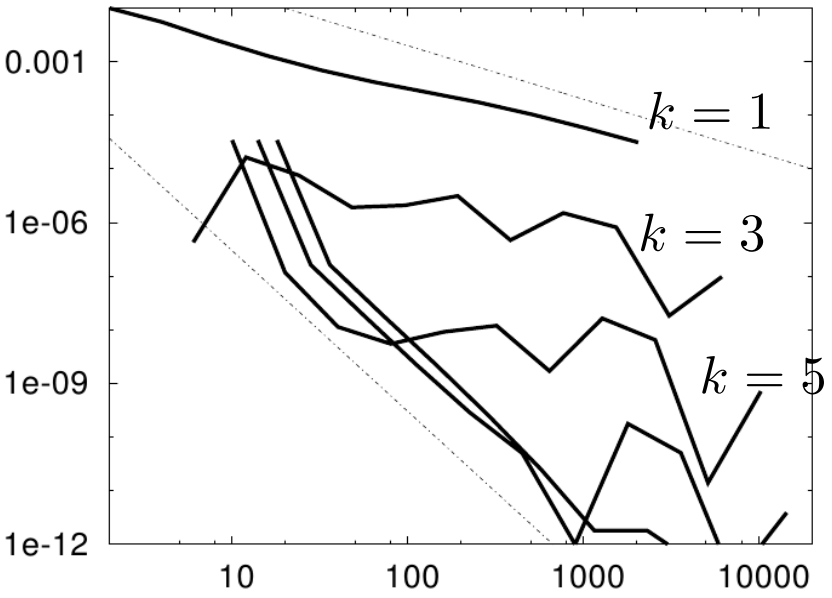
LU-based QSDC on RadauIIa(3)



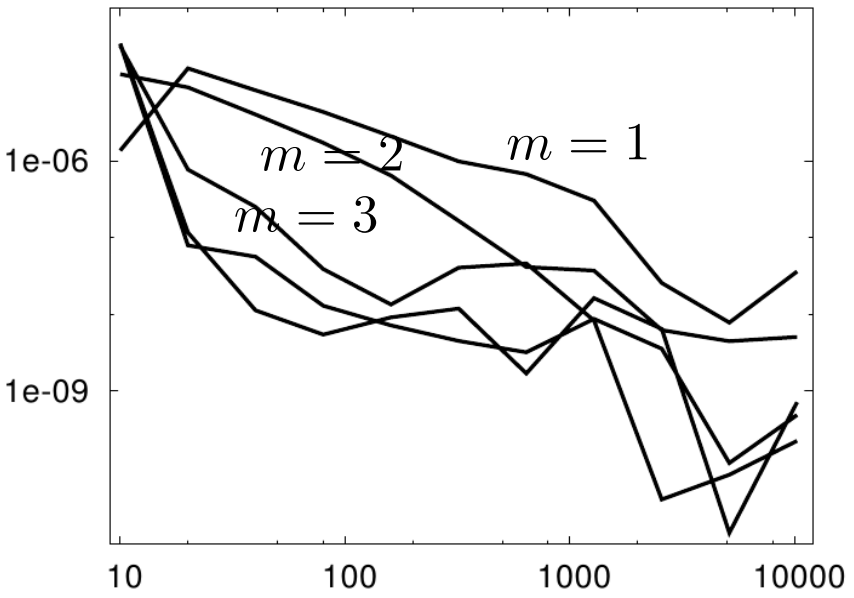
Understanding SDC Convergence



Different Sweep Blocks

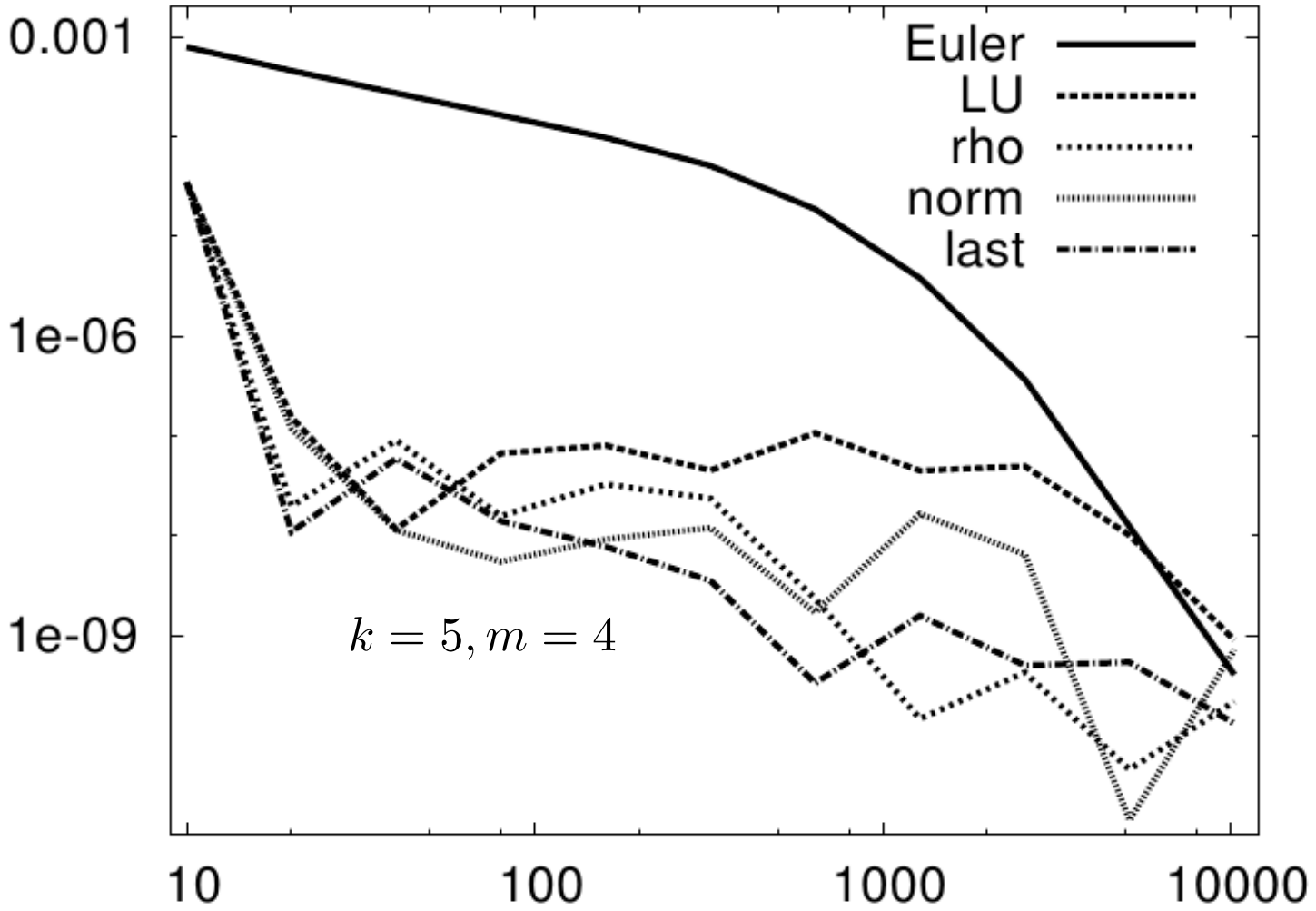


$$\min_{\hat{D},\hat{S}} \max_{\lambda} \|G(\lambda)^4\|^{1/4}$$



$$\min_{\hat{D},\hat{S}} \max_{\lambda} \|G(\lambda)^m\|^{1/m}$$

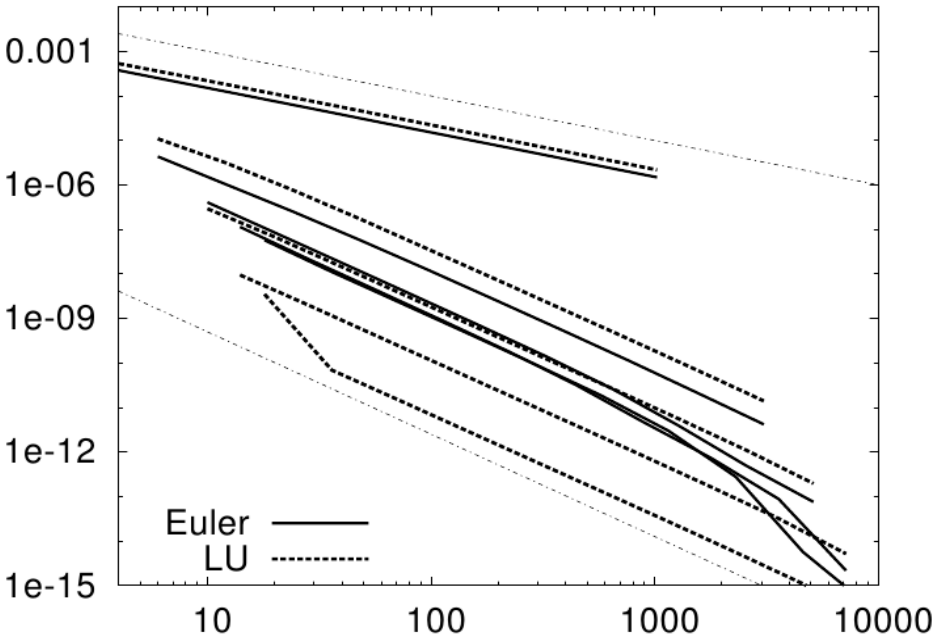
Different Optimization Criteria



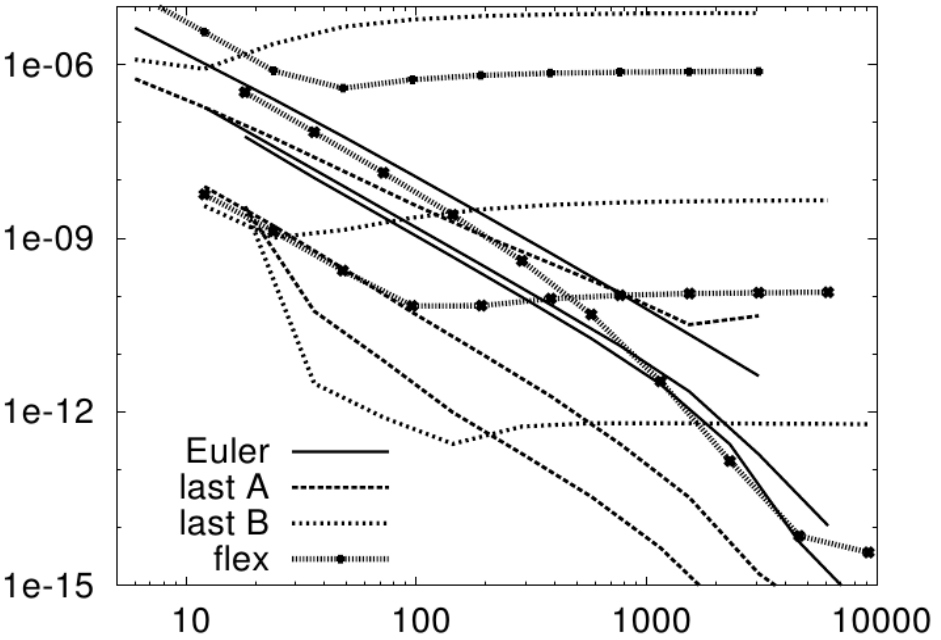
Example: NonautonomousHeat Equation

$$\dot{u} = \Delta u + xe^t, \quad x \in]0,1[, t \in [0, 0.1]$$

QSDC on RadauIIa(5)



$k = 1, 3, 5, 7, 9$

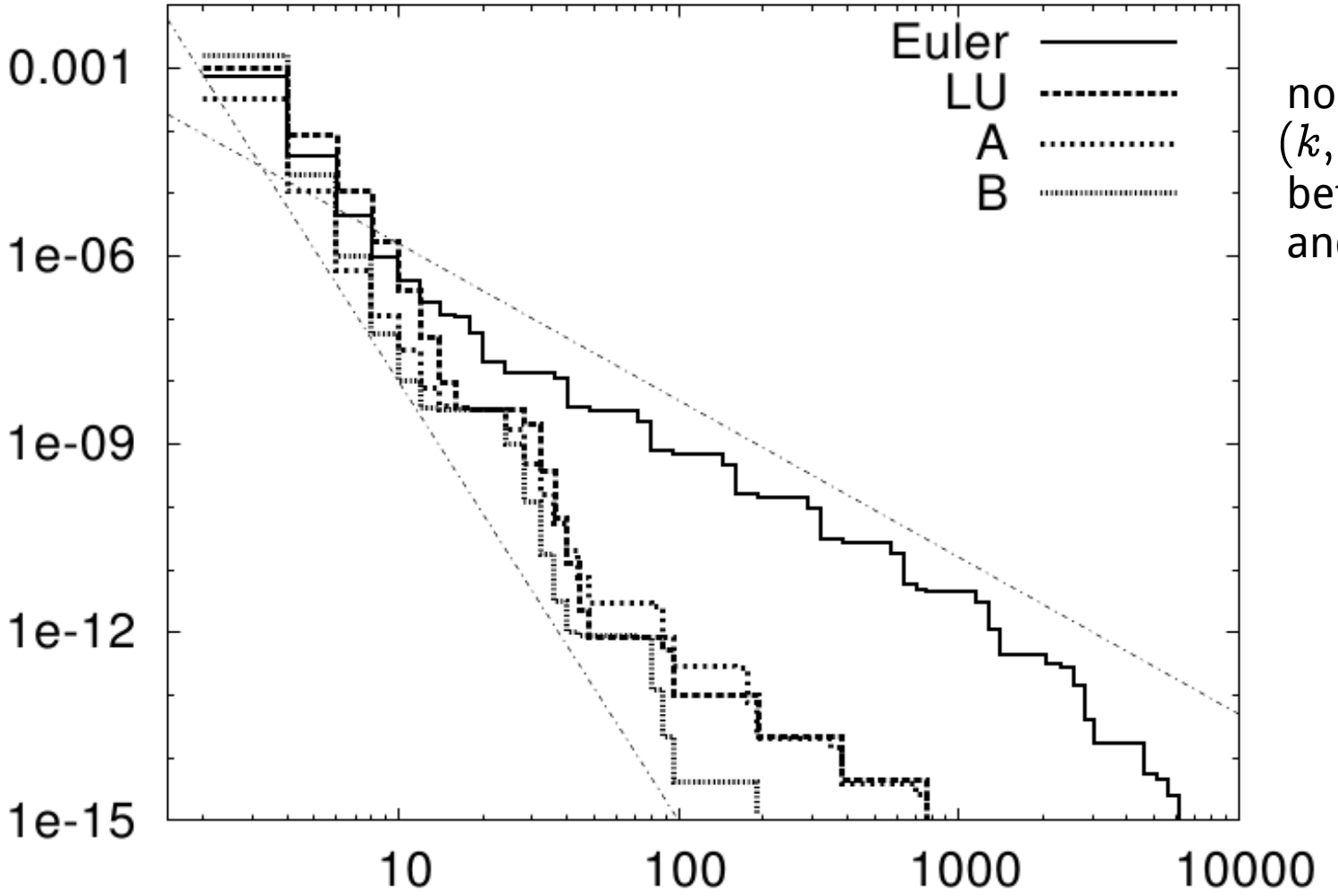


$k = 3, 6, 9$

$A : w = \tau\lambda^{-1/2}$

$B : w = 1$

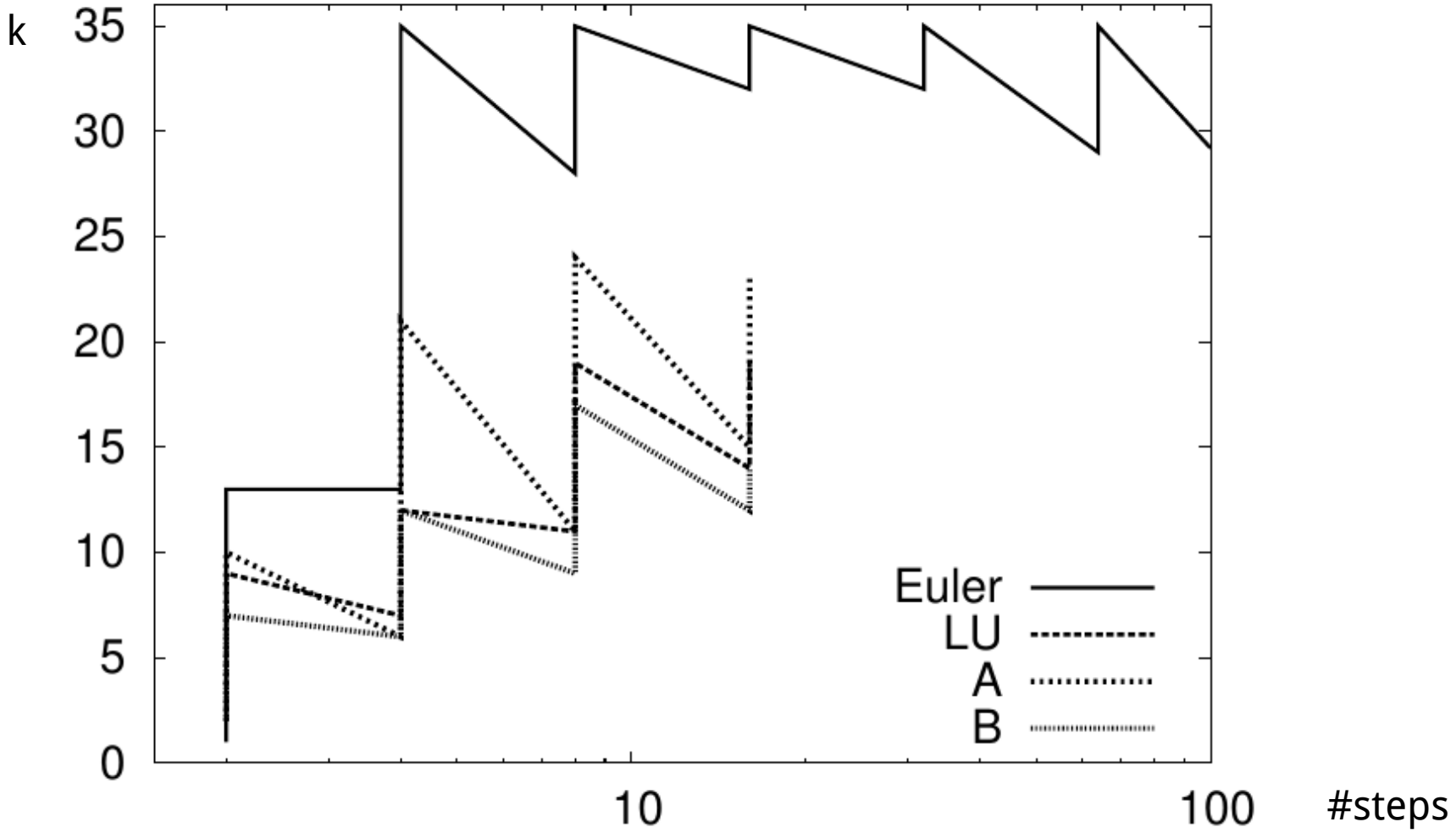
Pareto Front



non-dominated points: (k, τ) with no point better in both error and work

up to k=12 sweeps

Pareto Front



optimal number of sweeps for given time step size
(limited to 35)

The Road to Adaptivity

Time discretization

- adaptive simultaneous choice of collocation order, sweep count, and time step size
- nested iteration
- different basic schemes, different coupling of variables
- parallelization in time [Emmet/Minion '12, ...]

Spatio-temporal discretization

- local mesh refinement, local time stepping, local order
- multigrid – SDC coupling