Investigating implicit and explicit methods for stiff ionic models

Serial and first steps towards parallel-in-time integration

Giacomo Rosilho de Souza, Simone Pezzuto, Rolf Krause

Università della Svizzera Italiana, Lugano, Switzerland

Microcard annual meeting, July 2023, Strasbourg

Contents

Serial integration: New explicit methods and comparison with the popular IMEX+Rush-Larsen scheme

- Monodomain equation and notation,
- Standard IMEX+Rush-Larsen approach (IMEX-RL),
- Multirate Explicit stabilized methods (mES),
- Exponential Multirate Explicit Stabilized methods (exp-mES),
- A numerical comparison.

Towards parallel-in-time integration: Combining serial methods with SDC and exponential SDC

- Spectral Deferred Correction (SDC),
- Numerical results,
- Exponential SDC (ESDC),
- Numerical results.

Monodomain equation for cardiac electrophysiology

Solve problem:

$$\begin{cases} \chi\left(C_m\partial_t V + I_{ion}(V,z)\right) = \nabla\cdot(\sigma\nabla V) & \text{in } \Omega\times[0,T] \\ \partial_t z = g(V,z) & \text{in } \Omega\times[0,T] \\ -\sigma\nabla V\cdot n = 0 & \text{on } \partial\Omega\times[0,T] \\ u = u_0 & \text{on } \Omega\times\{0\} \end{cases}$$

The ionic model $\partial_t z = g(V,z)$ can be written as

$$\begin{cases} \partial_t z_E & = g_E(V,z_E,z_e), \\ \partial_t z_e & = g_e(V,z_e) \end{cases}$$

Where g_E is a nonlinear generally non stiff term and

$$g_e(V,z_e) = \tilde{\Lambda}_e(V)(z_e - z_{e,\infty}(V))$$

is a usually a stiff term.

After space discretization the PDE becomes

$$\mathbf{V}' = A\mathbf{V} - C_m^{-1}I_{ion}(\mathbf{V},\mathbf{z}_E,\mathbf{z}_e),$$
$$\mathbf{z}_E' = g_E(\mathbf{V},\mathbf{z}_E,\mathbf{z}_e),$$
$$\mathbf{z}_e' = \tilde{\Lambda}_e(\mathbf{V})(\mathbf{z}_e - \mathbf{z}_e(\mathbf{V}))\,.$$

Noting $y = (\mathbf{V},\mathbf{z}_E,\mathbf{z}_e)$ and

$$f_I(y) = (A\mathbf{V},0,0),$$
$$f_E(y) = (-C_m^{-1}I_{ion}(y), g_E(y),0),$$
$$f_e(y) = (0,0,\tilde{\Lambda}(\mathbf{V})(\mathbf{z}_e - \mathbf{z}_{e,\infty}(\mathbf{V}))) = \Lambda(y)(y - y_\infty(y)),$$

with $\Lambda(y)$ a diagonal matrix having zeros in the coordinates corresponding to $\mathbf{V},\mathbf{z}_E$, we can write the ODE as:

$$y' = f_I(y) + f_E(y) + f_e(y).$$

Where I,E,e stand for implicit, explicit, exponential terms, respectively.

IMEX—Rush-Larsen scheme

The IMEX-RL scheme consists in applying sequentially the explicit Euler, exponential Euler, and implicit Euler schemes as follows:

1. $y^1 = y_n + \Delta t \phi_1(\Delta t \Lambda(y_n)) f_e(y_n)$,
2. $y^2 = y^1 + \Delta t f_E(y^1)$,
3. $y^3 = y^2 + \Delta t f_I(y^3)$,
4. $y_{n+1} = y^3$,

where $\phi_1(z) = \dfrac{e^z - 1}{z}$.

The severe stiffness of f_e is smoothed out thanks to ϕ_1, the one from the Laplacian is dealt by the implicit Euler method.

The exponential term is very cheap to evaluate due to its diagonal form.

Explicit Stabilized methods

Consider again

$$y' = f(y), \qquad y(0) = y_0 .$$

One step of any explicit stabilized (ES) method is:

i) Compute approximation $\rho = \rho\left(\frac{\partial f}{\partial y}\right)$ (power method)

ii) Choose the number of stages s such that

$$\Delta t \rho \leq \beta s^2$$

iii) Iterate:

$$g_0 = y_0, \qquad g_1 = g_0 + \mu_1 \Delta t\, f(g_0)$$
$$g_j = \nu_j\, g_{j-1} + \kappa_j\, g_{j-2} + \mu_j \Delta t\, f(g_{j-1}) \quad j = 2, \ldots, s$$
$$y_1 = g_s$$

with $y_1 \approx y(\Delta t)$ and $g_i \approx y(c_i \Delta t)$, $0 < c_1 < \ldots < c_s = 1$.

Properties:

- The purpose of adding stages is to increase stability, not order.
- Stability grows quadratically with the number of stages s.

Pros:

- No step size restriction: just increase s,
- Workload proportional to $s \propto \sqrt{\rho} \propto 1/\Delta x$,
- Fully explicit and easy to implement.

Cons: s still depends on ρ.

Van der Houwen, Sommeijer, Verwer, Lebedev, Abdulle, Medovikov, Vilmart, Rosilho, Almuslimani,...

Multirate Explicit Stabilized methods

Consider the multirate problem

$$y' = f_F(y) + f_S(y),$$

with:

- f_F stiff but cheap,
- f_S mildly stiff but expensive.

Define the *modified equation*

$$y_\eta' = f_\eta(y_\eta)$$

with f_η the *averaged force*

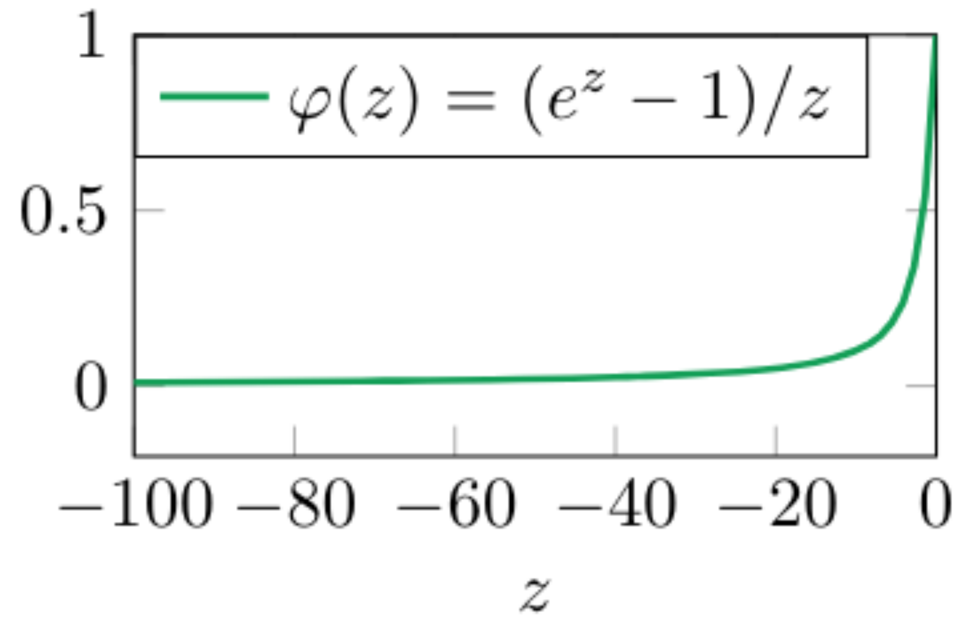
$$f_\eta(y) = \frac{1}{\eta}(u(\eta) - y)$$

defined via the *auxiliary solution* u

$$u' = f_F(u) + f_S(y), \quad u(0) = y\,.$$

Properties of f_η:

- Stiffness of f_η decreases as η grows. For instance:
  - For $f_F(y) = Ay$
  - then $f_\eta(y) = \varphi(\eta A) f(y)$
  - with $\varphi(z) = \dfrac{e^z - 1}{z} \le e^z$.



- For $\eta = 2/\rho_S$ then $\rho_\eta \le \rho_S = \rho(\partial f_S / \partial y)$.
- Stiffness of f_η depends on f_S only.
- No need for scale separation.

Abdulle A., Grote M., Rosilho G. 2022.

Multirate Explicit Stabilized methods

Consider the multirate problem

$$y' = f_F(y) + f_S(y),$$

with:

- f_F stiff but cheap,
- f_S mildly stiff but expensive.

Define the *modified equation*

$$y_\eta' = f_\eta(y_\eta)$$

with f_η the *averaged force*

$$f_\eta(y) = \frac{1}{\eta}(u(\eta) - y)$$

defined via the *auxiliary solution* u

$$u' = f_F(u) + f_S(y), \quad u(0) = y\,.$$

Discretization via ES methods:

Solve

$$y_\eta' = f_\eta(y_\eta)$$

With s stages, where $\Delta t \rho_S \leq \beta s^2$.

Whenever f_η needs to be evaluated, solve

$$u' = f_F(u) + f_S(y), \quad u(0) = y$$

With m stages, where $\Delta t \rho_F \leq \beta s^2 m^2$.

- The number of evaluations of f_F and f_S depends only on their stiffness.
- No interpolations.
- No need for scale separation.

Abdulle A., Grote M., Rosilho G. 2022.

Multirate Explicit Stabilized for Monodomain model

We solve

$$y' = f_I(y) + f_E(y) + f_e(y).$$

To solve it with mES we set

$$f_S(y) = f_E(y) + P_S f_e(y),$$
$$f_F(y) = f_I(y) + P_F f_e(y),$$

Where $P_F + P_S = I$ and P_F selects only the very stiff components of $\Lambda(y)$ (one or two). The others become part of the mildly stiff term $f_S(y)$.

One step of mES is given by:

$$g_0 = y_n, \qquad g_1 = g_0 + \mu_1 \Delta t f_\eta(g_0),$$
$$g_j = \nu_j g_{j-1} + \kappa_j g_{j-1} + \mu_j \Delta t f_\eta(g_{j-1}), \quad j = 2, \dots, s,$$
$$y_{n+1} = g_s,$$

With $f_\eta(y)$ defined as

$$u_0 = y, \qquad u_1 = u_0 + \alpha_1 \eta (f_F(u_0) + f_S(y)),$$
$$u_j = \beta_j u_{j-1} + \gamma_j u_{j-2} + \alpha_j \eta (f_F(u_{j-1}) + f_S(y)), \quad j = 2, \dots, m,$$
$$f_\eta(y) = (u_m - u_0)/\eta\,.$$

- This is an efficient approach when $P_F f_e(y)$ is not overly stiff. For instance for the Courtemanche, Ramirez, Nattel, 1998 ionic model.
- For stiffer models, as Ten Tusscher, Panfilov 2006, this entails a too high number of $f_F(y)$ evaluations and efficiency deterioration.

Exponential Multirate Explicit Stabilized

To recover efficiency even for stiff models, we remove the stiffness arising from $P_F f_E(y)$ employing exponentials:

$$\tau P_F f_e(y) \longrightarrow = \tau\phi_1(\tau P_F \Lambda(y)) f_e(y)$$

We still solve

$$\begin{aligned} g_0 &= y_n, \qquad g_1 = g_0 + \mu_1 \Delta t f_\eta(g_0),\\ g_i &= \nu_i g_{i-1} + \kappa_i g_{i-1} + \mu_i \Delta t f_\eta(g_{i-1}),\\ y_{n+1} &= g_s, \end{aligned}$$

For $i = 2,\ldots,s,$ with s depending on f_E only.

For the multirate stabilized method (mES):

$$\begin{aligned} u_0 &= y,\\ u_1 &= u_0 + \alpha_1\eta(f_I(u_0) + P_F f_e(u_0)\\ &\qquad + P_S f_e(y) + f_E(y)),\\ u_j &= \beta_j u_{j-1} + \gamma_j u_{j-2}\\ &\quad + \alpha_j\eta(f_I(u_{j-1}) + P_F f_e(u_{j-1})\\ &\qquad + P_S f_e(y) + f_E(y)),\\ f_\eta(y) &= (u_m - u_0)/\eta\,. \end{aligned}$$

For $j = 2,\ldots,m,$ with m depending on f_I and $P_F f_e$

For the exponential multirate stabilized method (exp-mES):

$$\begin{aligned} u_0 &= y,\\ u_1 &= u_0 + \alpha_1\eta(f_I(u_0) + \phi_1(\alpha_1\eta P_F\Lambda(u_0))P_F f_e(u_0)\\ &\qquad + P_S f_e(y) + f_E(y)),\\ u_j &= \beta_j u_{j-1} + \gamma_j u_{j-2}\\ &\quad + \alpha_j\eta(f_I(u_{j-1}) + \phi_1(\alpha_j\eta P_F\Lambda(u_{j-1}))P_F f_e(u_{j-1})\\ &\qquad + P_S f_e(y) + f_E(y)),\\ f_\eta(y) &= (u_m - u_0)/\eta\,. \end{aligned}$$

For $j = 2,\ldots,m,$ with m depending on f_I only.

Comparison on the benchmark problem

Solve problem

$$\begin{cases} \chi \left(C_m \partial_t V + I_{ion}(V,z) \right) = \nabla \cdot (\sigma \nabla V) & \text{in } \Omega \times [0,T] \\ \partial_t z = g(V,z) & \text{in } \Omega \times [0,T] \\ -\sigma \nabla V \cdot n = 0 & \text{on } \partial\Omega \times [0,T] \\ u = u_0 & \text{on } \Omega \times \{0\} \end{cases}$$

with usual parameters from Niederer et al 2011[1] N-version benchmark, hence

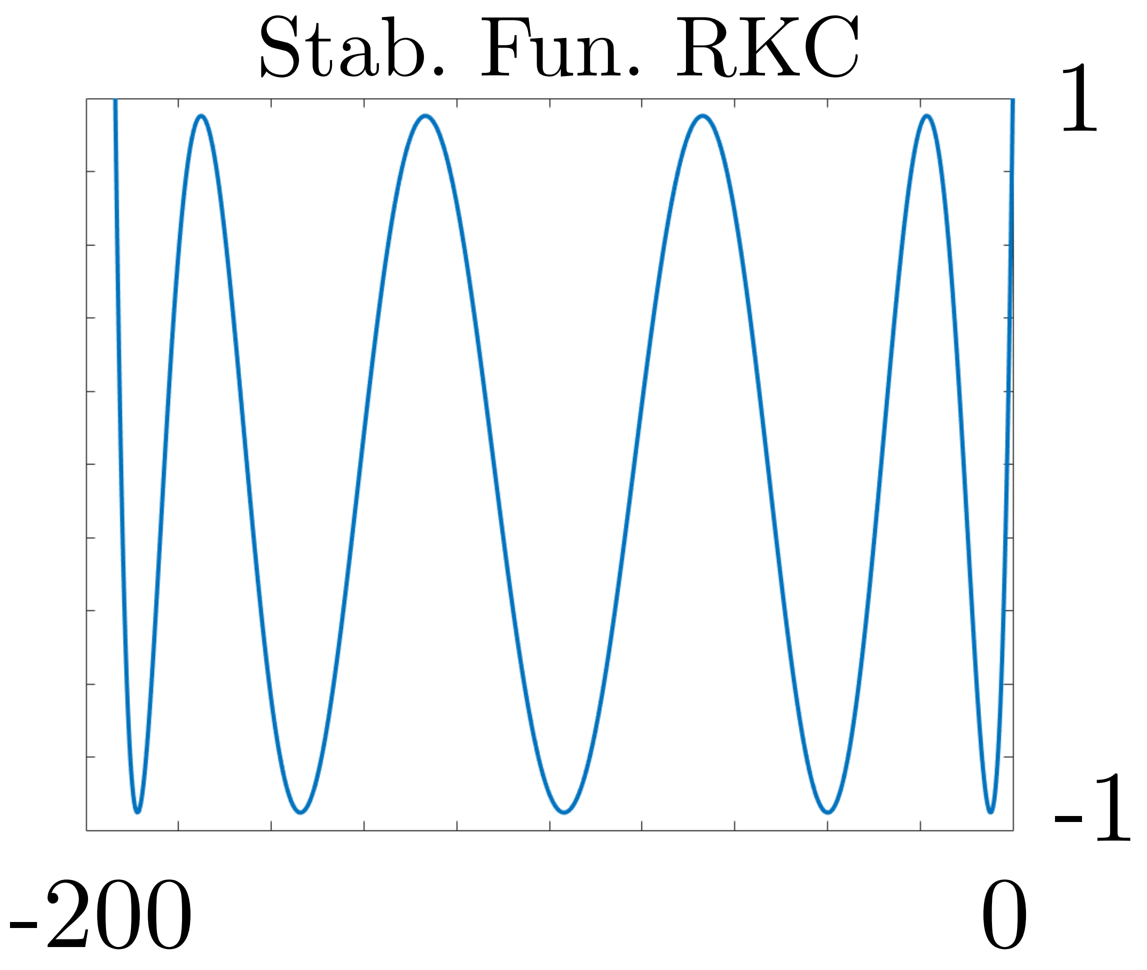
- $\Omega = 20 \times 7 \times 3\ mm^3$,
- Stimulus in a 1.5 mm^3 corner for $2ms$.
- We solve it with Ten Tusscher, Panfilov (TTP) 2006 Epi model.
- $\Delta x = 0.1mm$ and $\Delta t = 0.1\ ms$ or $\Delta t = 0.05\ ms$.

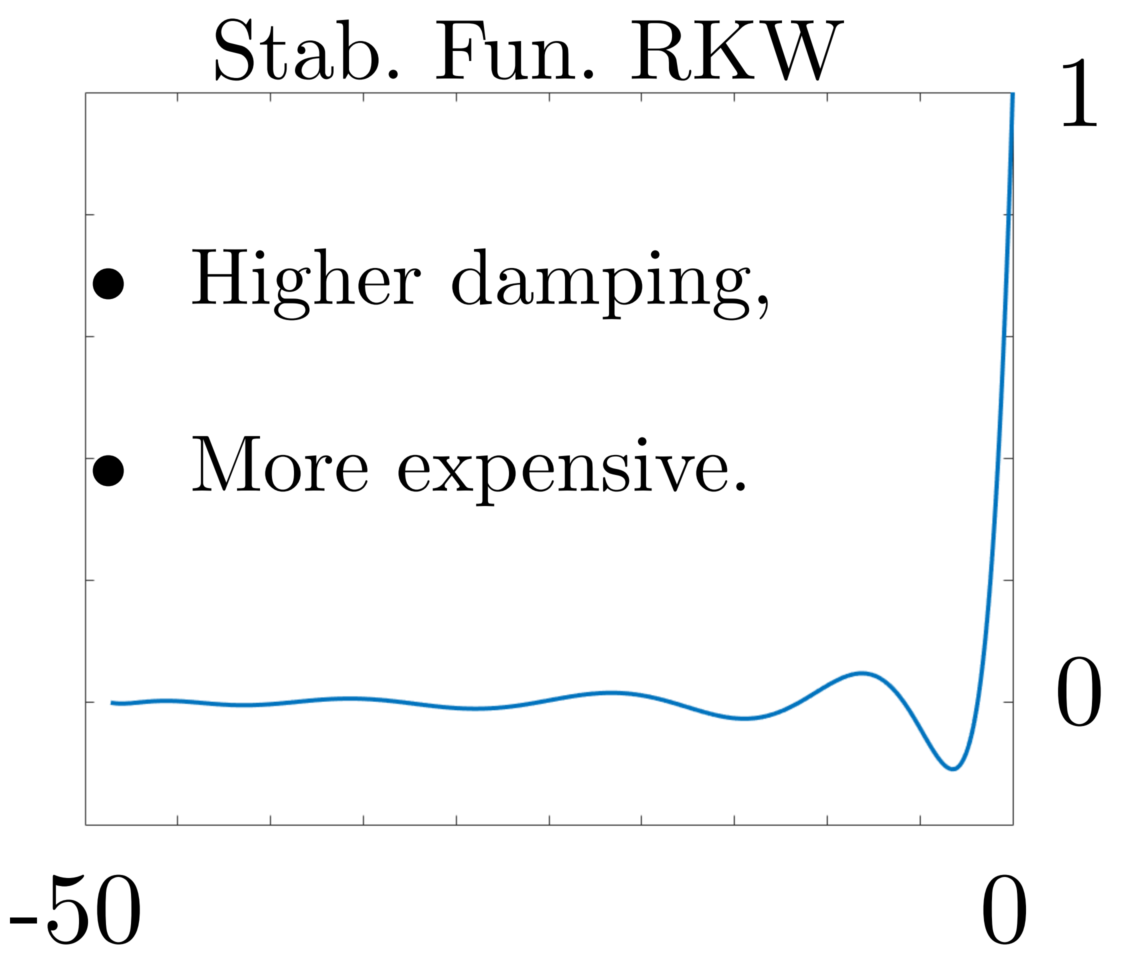
- Compare results of IMEX-RL, mES, and exp-mES methods.
- We consider two versions of both mES and exp-mES, with different stability polynomials defining the ES methods.
  - RKC: $R(z)$ oscillates in $[-0.95, 0.95]$,
  - RKW: satisfies $|zR(z)| \leq 1$, i.e. mimics L-stability as long as z is in the stability domain.
- Show activation times on 10 points along a diagonal line going from (1.5,1.5,1.5) to (20,7,3).
- Compare CV and
- CPU time.

[1]Niederer, S. et al. Verification of cardiac tissue electrophysiology simulators using an N-version benchmark. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *369*, 4331–4351.

Comparing CV and activation times

With Ten Tusscher, Panfilov 2006 Epi model.



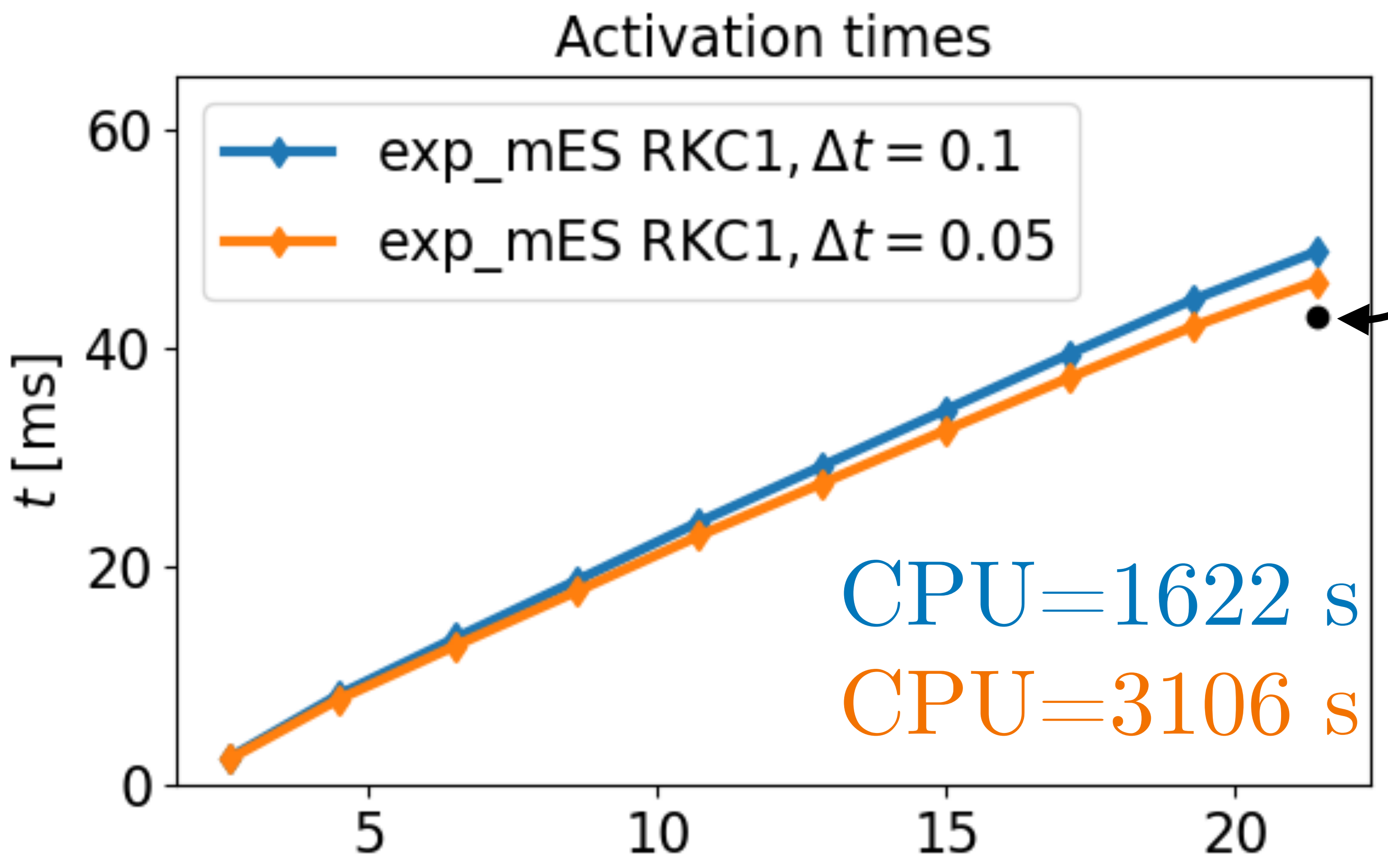


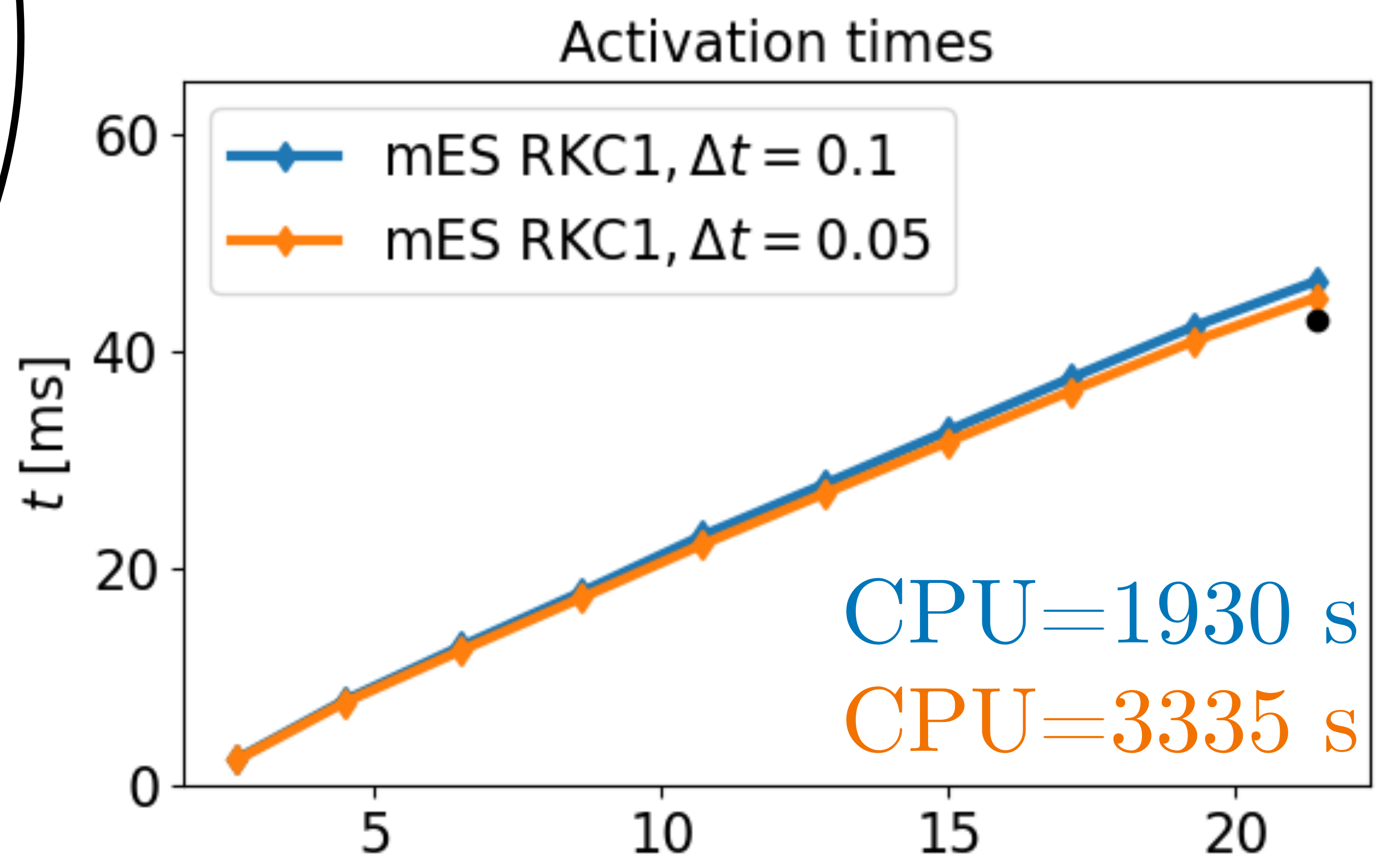
- Higher damping,
- More expensive.

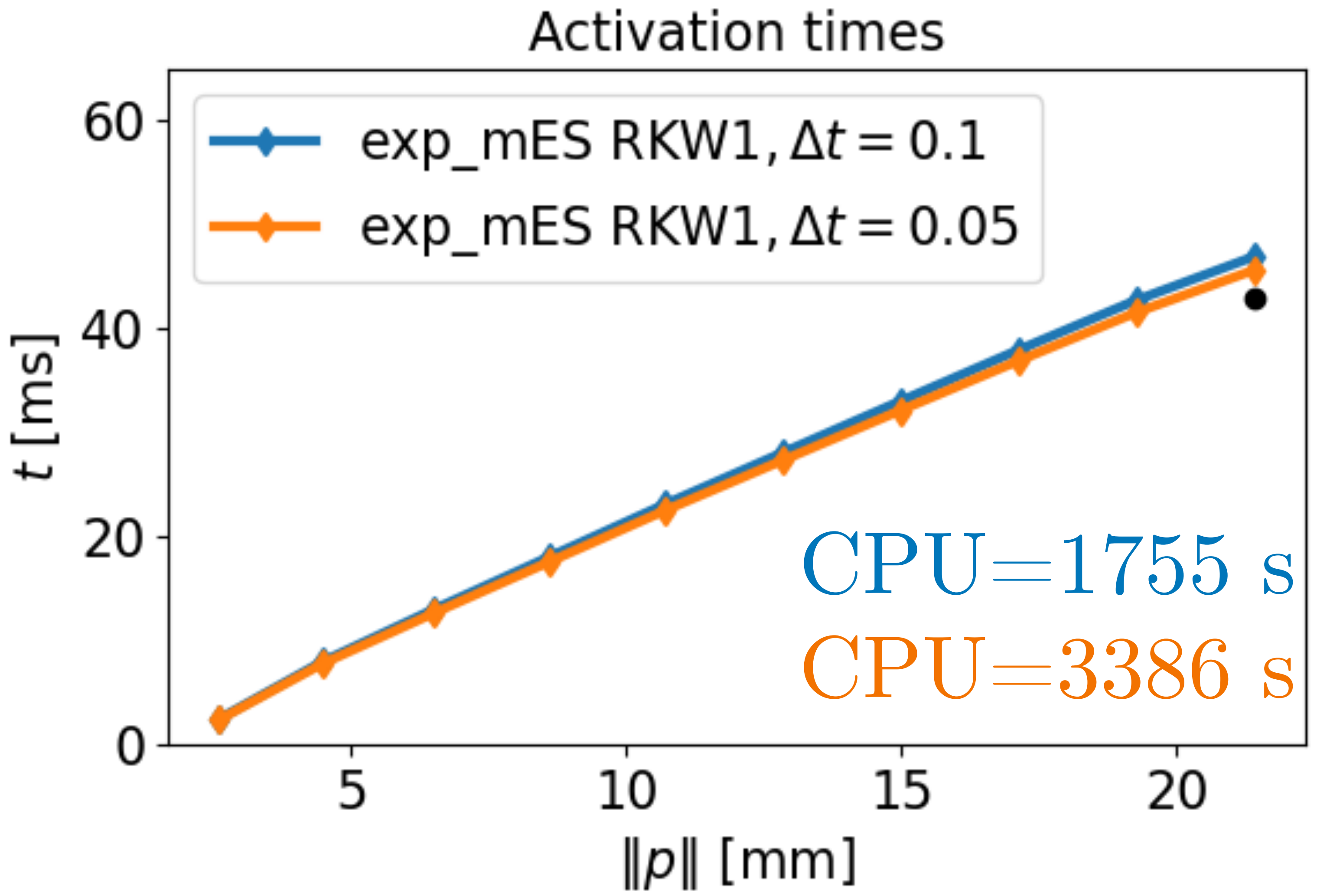
exp-mES

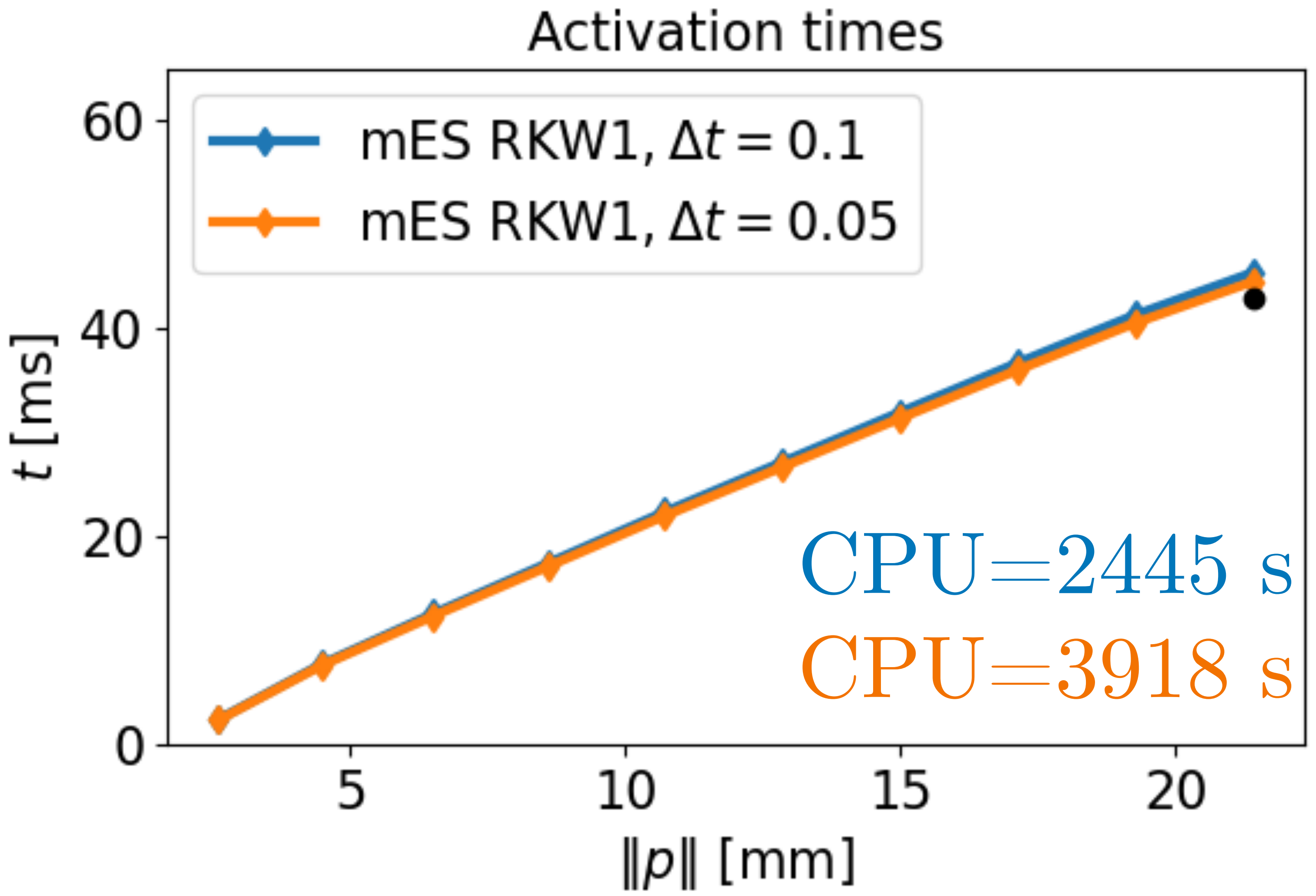
Reference value

mES





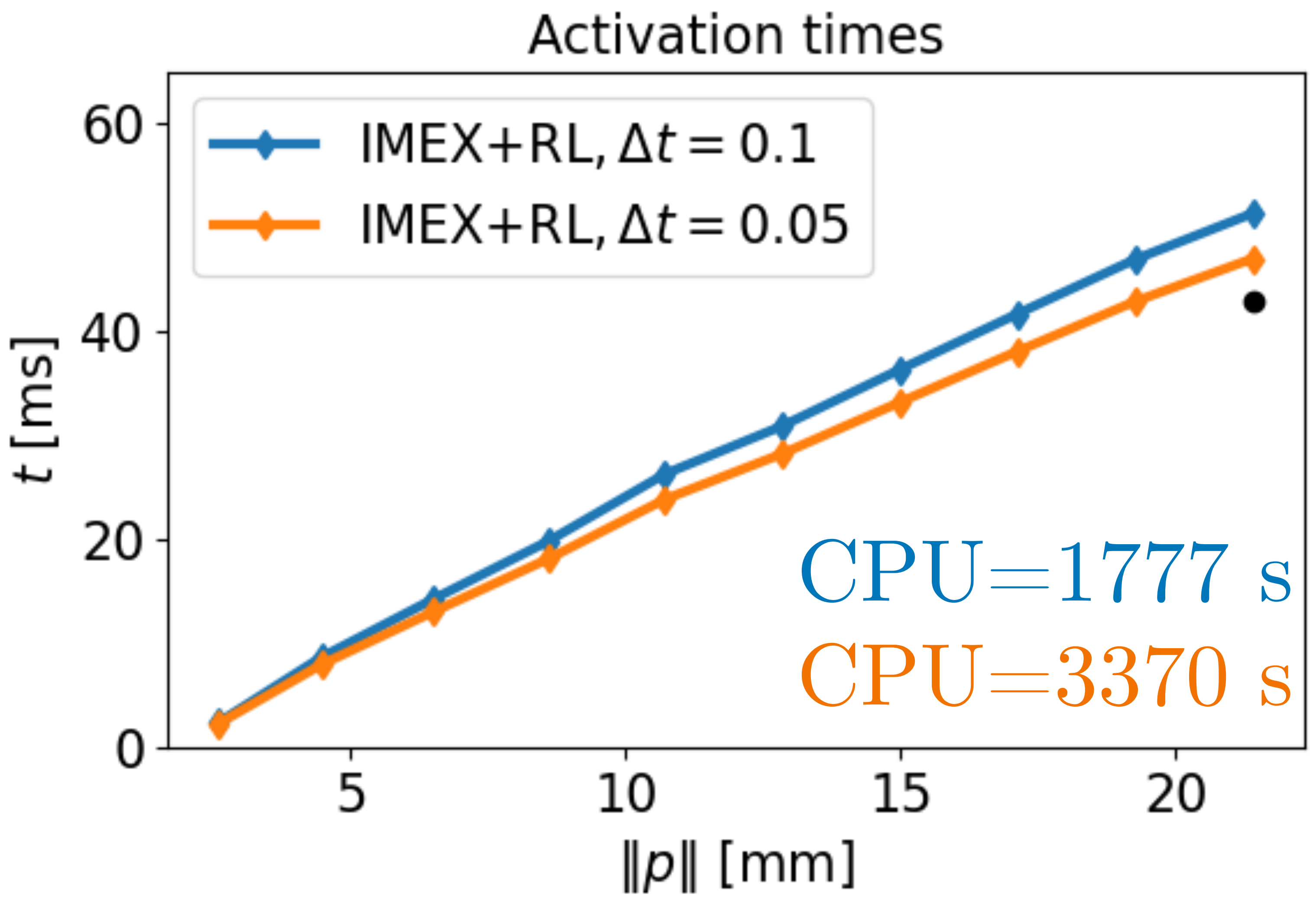




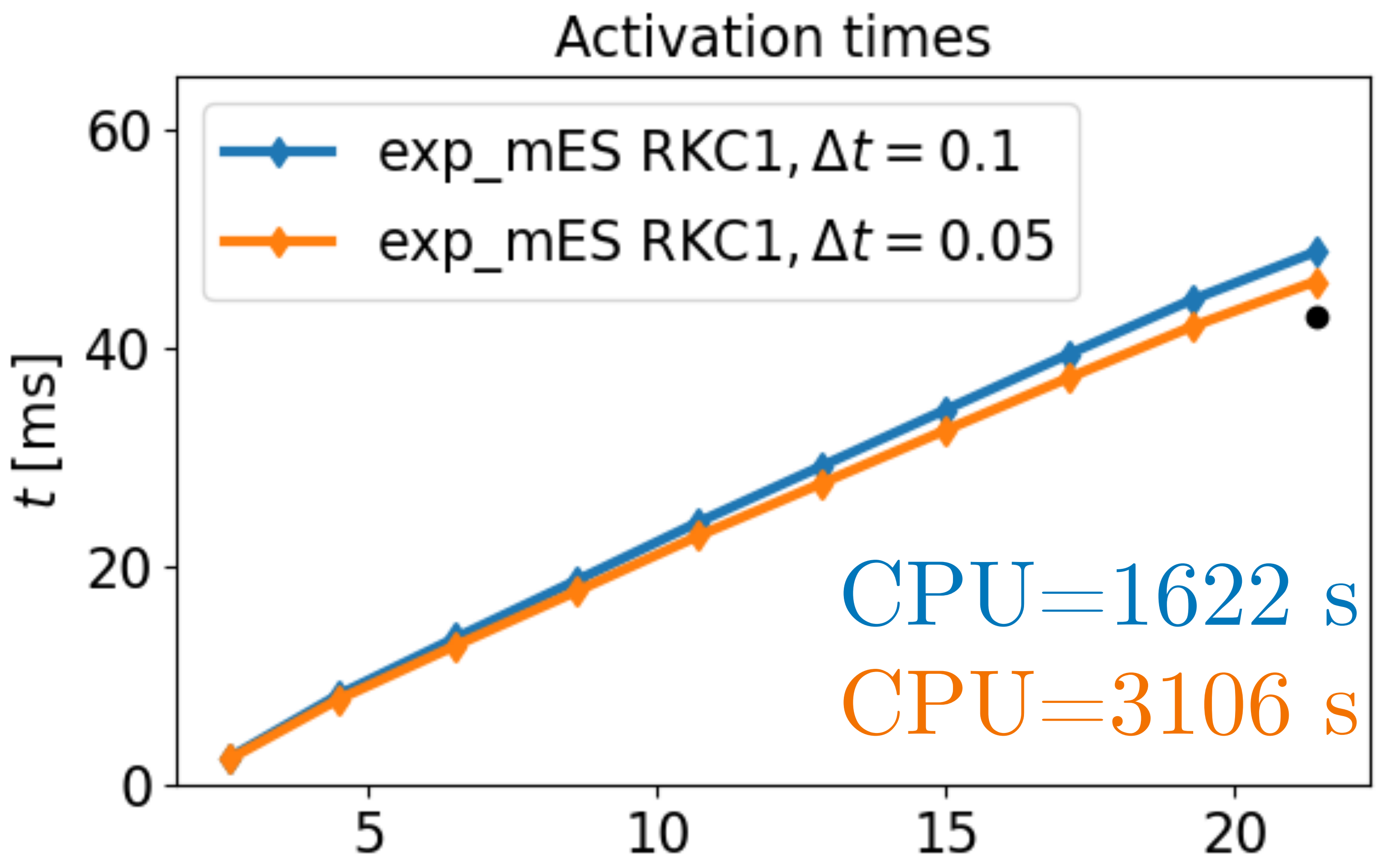
Comparing CV and activation times

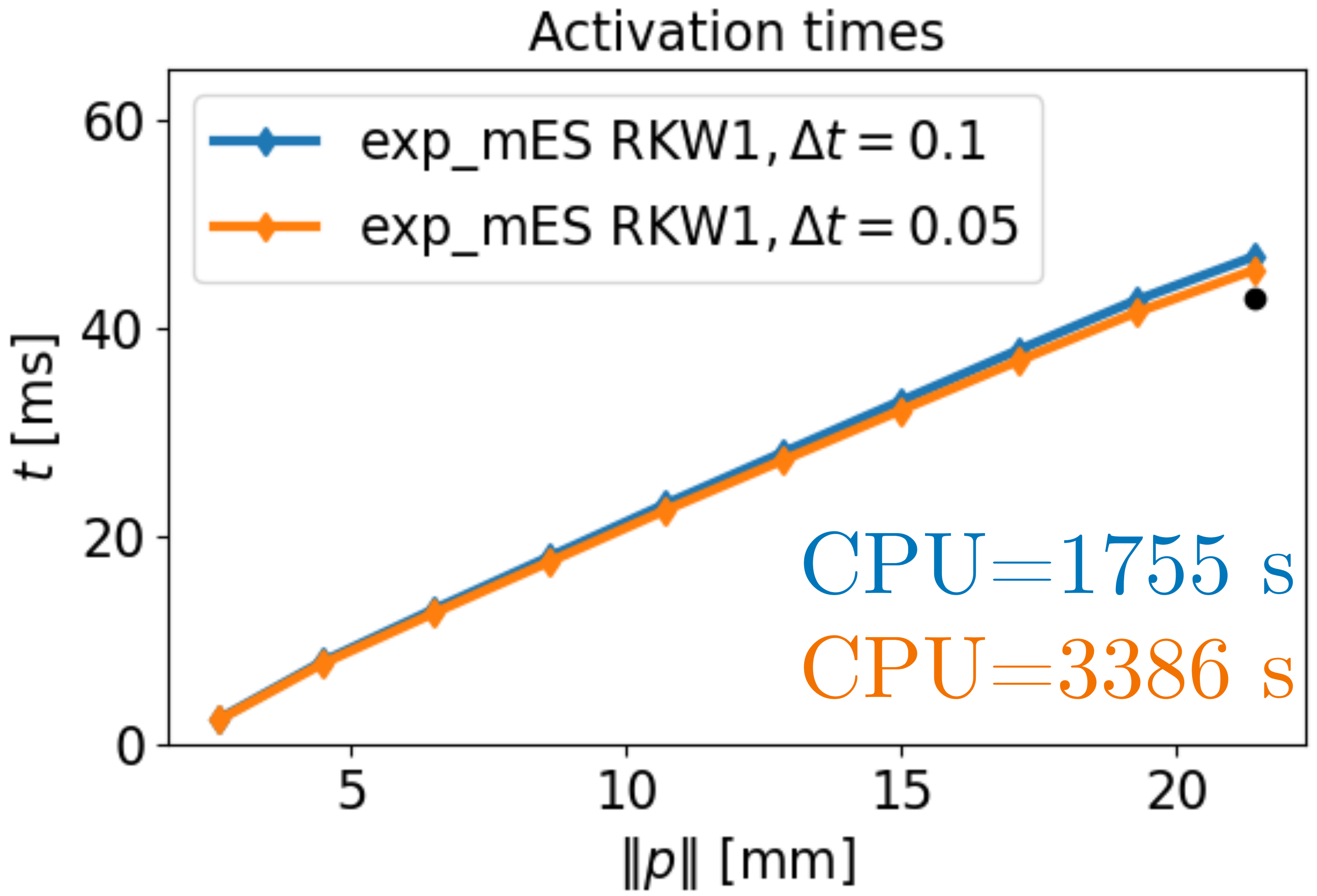
With Ten Tusscher 2006 Epi model.

IMEX-RL

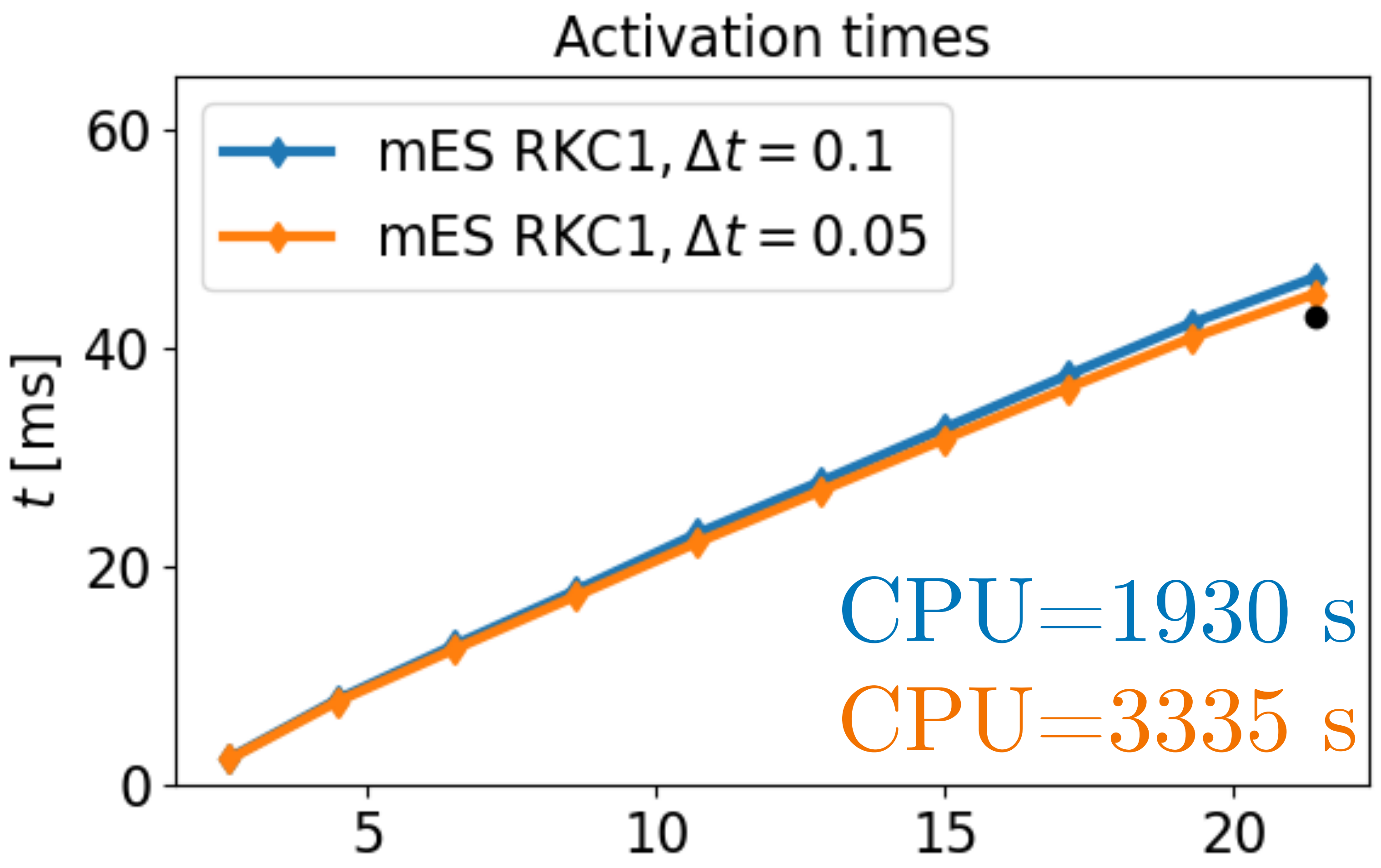


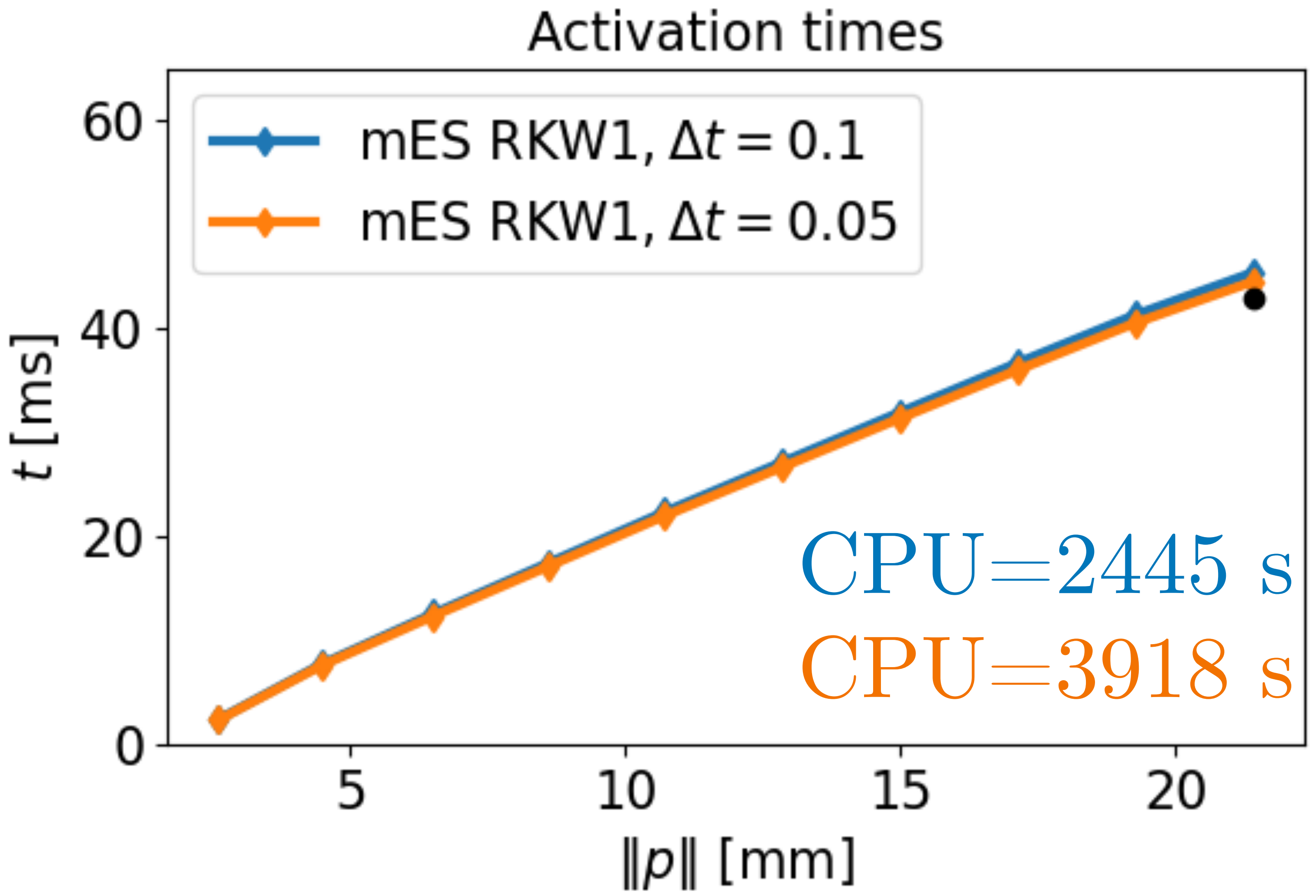
exp-mES





mES





Contents

Serial integration: New explicit methods and comparison with the popular IMEX+Rush-Larsen scheme

- Monodomain equation and notation,
- Standard IMEX+Rush-Larsen approach (IMEX-RL),
- Multirate Explicit stabilized methods (mES),
- Exponential Multirate Explicit Stabilized methods (exp-mES),
- A numerical comparison.

Towards parallel-in-time integration: Combining serial methods with SDC and exponential SDC

- Spectral Deferred Correction (SDC),
- Numerical results,
- Exponential SDC (ESDC),
- Numerical results.

Spectral Deferred Correction (SDC)

Collocation method

Consider

$$y' = f(y), \qquad y(0) = y_0$$

and fix collocation nodes $0 \le c_1 < \ldots < c_m \le 1$.

An approximation $y_i \approx y(\Delta t c_i) \in \mathbb{R}^n$ satisfies

$$y_i = y_0 + \Delta t \sum_{j=1}^{m} a_{ij} f(y_j) \quad i = 1, \ldots, m$$

$$\mathbf{y} = \mathbf{y}_0 + \mathbf{K}(\mathbf{y}) \quad \in \mathbb{R}^{m \cdot n}$$

with $\mathbf{y} = (y_1, \ldots, y_m)$, $\mathbf{y}_0 = (y_0, \ldots, y_0)$ and

$$\mathbf{K}(\mathbf{y}) = \left(\Delta t \sum_{j=1}^{m} a_{1j} f(y_j), \ldots, \Delta t \sum_{j=1}^{m} a_{mj} f(y_j) \right)$$

Deferred correction method

Let $\tilde{\mathbf{y}} = (\tilde{y}_1, \ldots, \tilde{y}_m)$ be an approximation $\tilde{y}_i \approx y(\Delta t c_i)$ computed via a cheaper (preconditioner) method

$$\tilde{\mathbf{y}} = \mathbf{y}_0 + \widetilde{\mathbf{K}}(\tilde{\mathbf{y}})$$

Compared to the collocation method, we have

i) error: $\delta = \mathbf{y} - \tilde{\mathbf{y}}$,

ii) residual: $\varepsilon = \mathbf{y}_0 + \mathbf{K}(\tilde{\mathbf{y}}) - \tilde{\mathbf{y}}$,

iii) error equation: $\delta = \varepsilon + \mathbf{K}(\tilde{\mathbf{y}} + \delta) - \mathbf{K}(\tilde{\mathbf{y}})$.

The error δ is approximated with

$$\tilde{\delta} = \varepsilon + \widetilde{\mathbf{K}}(\tilde{\mathbf{y}} + \delta) - \widetilde{\mathbf{K}}(\tilde{\mathbf{y}})$$

and the approximation is updated as $\tilde{\mathbf{y}} \leftarrow \tilde{\mathbf{y}} + \tilde{\delta}$.

Dutt, A., Greengard, L., Rokhlin, V. (2000). BIT Numerical Mathematics, 40(2).

Spectral Deferred Correction (SDC)

Recipe for a (Spectral) Deferred Correction method:

1. Choose the collocation nodes $0 \leq c_1 < \ldots < c_m \leq 1$.
2. Choose the cheaper method $\tilde{\mathbf{y}} = \mathbf{y}_0 + \widetilde{\mathbf{K}}(\tilde{\mathbf{y}})$.
3. Iterate over the error equation

   i) $\tilde{\delta} = \varepsilon + \widetilde{\mathbf{K}}(\tilde{\mathbf{y}} + \delta) - \widetilde{\mathbf{K}}(\tilde{\mathbf{y}})$

   ii) $\tilde{\mathbf{y}} \leftarrow \tilde{\mathbf{y}} + \tilde{\delta}$

Accuracy: gain Δt^p per iteration, with p the order of the cheap method. Limit: order of collocation method.

<u>Common choices:</u>

- For collocation method: Radau, Lobatto, Gauss, etc. due to their excellent superconvergence, stability, and geometric properties.
- For the cheaper method, in general, compute $\tilde{\delta}_{i+1}$ from $\tilde{\delta}_i$ by solving the error equation in $[c_i\Delta t, c_{i+1}\Delta t]$.
- Do so using IMEX-RL, mES, exp-mES.

<u>Pros:</u>

- No need to solve large nonlinear systems in $\mathbb{R}^{m\cdot n}$,
- Competes with standard high-order methods,
- Due to its iterative nature it's well suited for PinT (PFASST).

IMEX-RL, mES, exp-mES and SDC

We compare the IMEX-RL, mES, exp-mES methods combined with SDC.
For the time being, we only want to check whether the SDC sweeps converge or not.

Computational setup:

- Everything implemented in the pySDC library.
- Use three Radau IIA nodes,
- Sweep till convergence with relative tolerance $tol = 5 \cdot 10^{-8}$ on the residual. Norm is ℓ_2-norm on $y = (\mathbf{V}, \mathbf{z}_E, \mathbf{z}_e)$.
- Consider three ionic models with increasing stiffness:
  - Hodgkin-Huxley (HH), ◦ Courtemanche, Ramirez, Nattel 1998 (CRN), ◦ Ten Tusscher, Panfilov 2006 Epi (TTP).
- Different step sizes: $\Delta t = 0.1\ ms$, $\Delta t = 0.05\ ms$, $\Delta t = 0.01\ ms$.
- Here we solve the 2D version of the benchmark problem, for simplicity.

Summary of codes behaviour

Stiffness: $\rho(HH) \approx 55$, $\rho(CRN) \approx 130$, $\rho(TTP) \approx 950$.

	mES + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✓	✓	✓
TTP	✗	✓	✓

	exp-mES + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✗	✓	✓
TTP	✗	✗	✗

	IMEX-RL + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✗	✗	✓
TTP	✗	✗	✗

Not too bad

Could be better

Could be much better

These two schemes use exponentials.
So, let's try exponential Runge-Kutta as underlying method, instead of a standard collocation method.
This stabilizes multiplications with $\Lambda(y)$ in the residual computation: $\varepsilon = \mathbf{y}_0 + \mathbf{K}(\tilde{\mathbf{y}}) - \tilde{\mathbf{y}}$

Exponential Spectral Deferred Correction (ESDC)

Consider equation:

$$y' = \Lambda y + N(y)\,.$$

SDC is based on collocation methods. Hence, on formula:

$$y(t) = y_0 + \int_0^t f(y(s))\mathrm{d}s$$

and its discretization

$$y_i = y_0 + \Delta t \sum_{j=1}^{m} a_{ij} f(y_j) \quad i = 1,\dots,m,$$

with

$$a_{ij} = \int_0^{c_i} \ell_j(s)\mathrm{d}s$$

ESDC is based on exponential collocation methods. Hence, on formula:

$$y(t) = y_0 + \int_0^t e^{(t-s)\Lambda}(\Lambda y_0 + N(y(s)))\mathrm{d}s$$

and its discretization

$$y_i = y_0 + \Delta t \sum_{j=1}^{s} a_{ij}(\Delta t\Lambda)(\Lambda y_0 + N(y_j)) \quad i = 1,\dots,m,$$

with

$$a_{ij}(\Delta t\Lambda) = \int_0^{c_i} e^{(c_i - s)\Delta t\Lambda}\ell_j(s)\mathrm{d}s$$

Buvoli, T. (2020). A class of exponential integrators based on spectral deferred correction. *SIAM Journal on Scientific Computing*, *42*(1), A1–A27.

ESDC for monodomain equation

- Write

$$\begin{aligned} y' &= f_e(y) + f_I(y) + f_E(y) \\ &= \Lambda(y)(y - y_\infty(y)) + f_I(y) + f_E(y) \\ &= \underbrace{\Lambda(y_n)y}_{\Lambda y} + \underbrace{f_I(y) + f_E(y) + \Lambda(y)(y - y_\infty(y)) - \Lambda(y_n)y}_{N(y)} \end{aligned}$$

- Since

$$\Lambda(y_n)f_I(y) = \Lambda(y_n)f_E(y) = 0,$$

Then

$$a_{ij}(\Delta t\Lambda)f_I(y_j) = a_{ij}f_I y_j, \qquad a_{ij}(\Delta t\Lambda)f_E(y_j) = a_{ij}f_E y_j,$$

hence f_I, f_E terms are integrated with the standard SDC method, while f_e with ESDC.

Summary of codes behaviour

Università della Svizzera italiana

	mES + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✓	✓	✓
TTP	✗	✓	✓

	exp-mES + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✗	✓	✓
TTP	✗	✗	✗

	IMEX-RL + SDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✗	✗	✓
TTP	✗	✗	✗

	mES + ESDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✓	✓	✓
TTP	✗	✓	✓

	exp-mES + ESDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✓	✓	✓
TTP	✗	✓	✓

	IMEX-RL + ESDC		
	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
HH	✓	✓	✓
CRN	✓	✓	✓
TTP	✓	✓	✓

To be investigated....

WOW!! 🔥

Conclusions

- Explicit schemes, when properly stabilized, can be competitive and should be considered
- Standard SDC sweeps can become unstable as the stiffness of the ionic model increases
- It is probable that instabilities are introduced during residual computation, where exponentials are not employed.
- Preliminary results indicate that exponential SDC solves this issue.

The end

Thank you for your attention 😁

Funding: This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreements No 955495 (MICROCARD) and No 955701 (TIME-X). The JU receives support from the European Union's Horizon 2020 research and innovation programme and Belgium, France, Germany, Switzerland.