Parallel integration of linear ODEs

Martin J. Gander Stefan Güttel

Department of Mathematics
University of Geneva

ILAS Pisa, 2010

UNIVERSITÉ
DE GENÈVE

Motivation

Let us consider the linear initial value problem

$$u'(t) = \mathbf{A}u(t) + g(t), \quad t \in [0, T], \quad u(0) = u_0,$$

where $\mathbf{A} \in \mathbb{C}^{d\times d}$ is large and $u, g : [0, T] \to \mathbb{C}^d$ and $u_0 \in \mathbb{C}^d$.

Applications

- Input–output simulations of dynamical systems,
- control problems,
- quasi-Monte Carlo methods for SPDEs,
- fluid flow problems, ...

Aim: Speed up solution on a parallel computer.

Usual approach: Time-stepping

Partition the time-interval $0 = T_0 < T_1 < \cdots < T_N = T$ and let $\mathcal{F}(T_n, T_{n-1}) \circ X$ denote some numerical integrator that returns an approximation of $u(T_n)$ satisfying

$$u'(t) = \mathbf{A}u(t) + g(t), \quad t \in [T_{n-1}, T_n], \quad u(T_{n-1}) = X.$$

The numerical solution $U_n \approx u(T_n)$ which satisfies $u(T_0) = U_0$ is then given as

$$U_n = \mathcal{F}(T_n, T_{n-1}) \circ \cdots \circ \mathcal{F}(T_2, T_1) \circ \mathcal{F}(T_1, T_0) \circ U_0.$$

This decomposition is sequential! How to parallelize?

Parallelization

There have been developed various forms of parallelism, e.g.,

- **Parallelism across the method:** Compute stages of a Runge–Kutta method simultaneously (cf. book [Burrage 95]),
- **Parallelism across space:** Domain decomposition methods,
- **Parallelism across space–time:** Waveform relaxation methods,
- **Parallelism in linear algebra:** ScaLapack, PLapack,
- **Parallelism across time:** Parareal [Lions, Maday & Turincini 01].

Parallelization

There have been developed various forms of parallelism, e.g.,

- **Parallelism across the method:** Compute stages of a Runge–Kutta method simultaneously (cf. book [Burrage 95]),
- **Parallelism across space:** Domain decomposition methods,
- **Parallelism across space–time:** Waveform relaxation methods,
- **Parallelism in linear algebra:** ScaLapack, PLapack,
- **Parallelism across time:** Parareal [Lions, Maday & Turincini 01].

Parallelism across time is usually considered less efficient for general IVPs $u' = f(t, u)$. [Gander & Hairer 08] about parareal:

"These examples show that parallel speedup in time is possible, although not at the same level as in space, where one often asks for perfect speedup, i.e. the computation with one hundred processors should be one hundred times faster."

However, for linear problems the situation is different!

Overlapping time-decomposition

The problem

$$u'(t) = \mathbf{A}u(t) + g(t), \quad t \in [0, T], \quad u(0) = u_0,$$

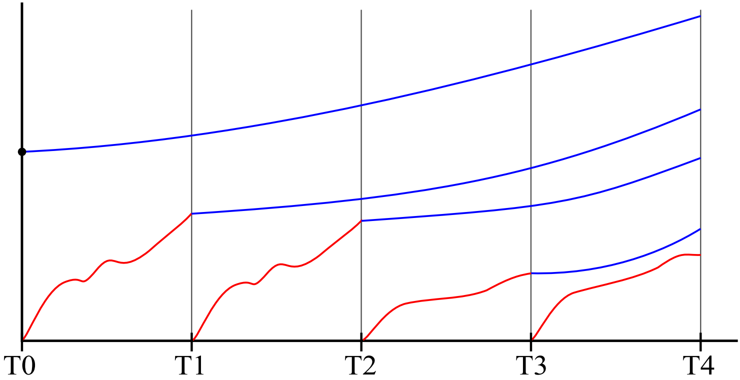
can be decoupled into N subproblems $(n = 1, \ldots, N)$

$$\begin{aligned} v_n'(t) &= \mathbf{A}v_n(t) + g(t), \qquad & t \in [T_{n-1}, T_n], \quad & v_n(T_{n-1}) = 0, \\ w_n'(t) &= \mathbf{A}w_n(t), \qquad & t \in [T_{n-1}, T_N], \quad & w_n(T_{n-1}) = v_{n-1}(T_{n-1}), \end{aligned}$$

where $v_0(T_0) = u_0$, such that

$$u(t) = \sum_{n=1}^{N} v_n(t) + w_n(t).$$

If processor $P(n)$ computes the approximations $V_{n-1}(t)$ and $W_n(t)$, this summation is the only communication point.



1. On processor $P(n)$ compute $V_{n-1}(T_{n-1}) = \mathcal{F}(T_{n-1}, T_{n-2}) \circ 0$.
2. On processor $P(n)$ compute $W_n(t) = e^{(t-T_{n-1})\mathbf{A}} V_{n-1}(T_{n-1})$.
3. Summation $U(T_j) = \sum_{n=1}^{N} V_n(T_j) + W_n(T_j)$ at desired T_j.

Note: The introduced redundancy removes almost all need for communication and synchronization between the processors.

All parallel gain stems from parallelization of the integrator $\mathcal{F}$, hence the cost for solving the homogeneous problem

$$W'(t) = \mathbf{A}W(t), \quad t \in [T_0, T_1], \quad W(T_0) = W^{[0]}$$

should be small in comparison.

$\Longrightarrow$ **Compute $W(t) = \exp(t\mathbf{A})W^{[0]}$ by a rational Krylov method!**

Rational Chebyshev Method

To approximate $\exp(\mathbf{A})v$ efficiently, we combine

- the Chebyshev method [Druskin & Knizhnerman 89] [Schaefer 90] with
- the shift-and-invert idea [Moret & Novati 04] [Hochbruck & vd Eshof 06],
- see also [Grosch & Orszag 77] [Boyd 82/87].

Rational Chebyshev Method

To approximate $\exp(\mathbf{A})v$ efficiently, we combine

- the Chebyshev method [Druskin & Knizhnerman 89] [Schaefer 90] with
- the shift-and-invert idea [Moret & Novati 04] [Hochbruck & vd Eshof 06],
- see also [Grosch & Orszag 77] [Boyd 82/87].

Assume that $\mathbf{A}$ is symmetric negative-semidefinite, i.e., $\Lambda(\mathbf{A}) \in (-\infty, 0]$.

Map its spectrum into $(-1, 1]$,

$$\mathbf{B} := (\xi\mathbf{I} + \mathbf{A})(\xi\mathbf{I} - \mathbf{A})^{-1} \qquad \text{for } \xi > 0,$$

and note that $\exp(\mathbf{A})v = g(\mathbf{B})v$ with $g(x) := \exp(\xi \frac{x-1}{x+1})$.

Now approximate $g(x)$ by Chebyshev expansion

$$g(x) = \sum_{j=0}^{\infty} \gamma_j C_j(x), \quad x \in [-1, 1],$$

where $C_j(x) = \cos(j \arccos(x))$.

Chebyshev polynomials $C_j(x)$ are orthogonal w.r.t. the inner product

$$\langle f(x), g(x)\rangle_w := \int_{-1}^{1} \frac{f(x)g(x)}{\sqrt{1-x^2}}\,\mathrm{d}x$$

and the γ_j are the coordinates of $g(x)$, i.e.,

$$\gamma_j := \langle g(x), C_j(x)\rangle_w.$$

Since

$$\begin{aligned}\langle g(x), C_j(x)\rangle_w &= \int_{-1}^{1} \frac{\cos(j\arccos(x))g(x)}{\sqrt{1-x^2}}\,\mathrm{d}x \\ &= \int_{0}^{\pi} \cos(j\theta)g(\cos(\theta))\,\mathrm{d}\theta,\end{aligned}$$

the coefficients γ_j are easily approximated via FFT.

The truncated Chebyshev sum

$$S_m(x) := \sum_{j=0}^{m-1} \gamma_j C_j(x), \quad x \in [-1,1],$$

is the best approximation of $g(x)$ from $\mathcal{P}_{m-1} \subset L^2_w([-1,1])$.

$$\boxed{\text{We consider } S_m(\mathbf{B})v \text{ as an approximation to } g(\mathbf{B})v = \exp(\mathbf{A})v.}$$

Note that

$$\|g(\mathbf{B})v - S_m(\mathbf{B})v\|_2 \le \|g(x) - S_m(x)\|_\infty \, \|v\|_2,$$

where $\|f\|_\infty := \max\{|f(x)| : x \in [-1,1]\}$.

What about convergence $\|g(x) - S_m(x)\|_\infty \to 0$?

Dini–Lipschitz Test: If $g(x) \in C[-1,1]$ satisfies

$$\lim_{m\to\infty} \omega(1/m)\log m \to 0, \quad \text{where} \;\; \omega(\delta) := \sup_{\substack{-1\leq a,b\leq 1\\ |a-b|\leq\delta}} |g(a)-g(b)|,$$

then $\|g(x) - S_m(x)\|_\infty \to 0$.

Hence: $S_m(x)$ converges uniformly to $g(x)$ and

$$\|\exp(\mathbf{A})v - S_m(\mathbf{B})v\|_2 \leq \|g(x) - S_m(x)\|_\infty \, \|v\|_2 \to 0.$$

The convergence is $O(m^{-k})$ for arbitrary k, and independent of $\Lambda(\mathbf{A})$!

A practical method for computing $S_m(\mathbf{B})v \approx \exp(\mathbf{A})v$:

- Precompute Chebyshev coefficients $\gamma_0, \gamma_1, \ldots$ via FFT.
- Initialize $C_0 = v$, $C_1 = \mathbf{B}v$.
- Set $S_2 = \gamma_0 C_0 + \gamma_1 C_1$.
- For $j = 2, \ldots, m-1$ compute $C_j = 2\mathbf{B}C_{j-1} - C_{j-2}$ and set $S_{j+1} = S_j + \gamma_j C_j$.

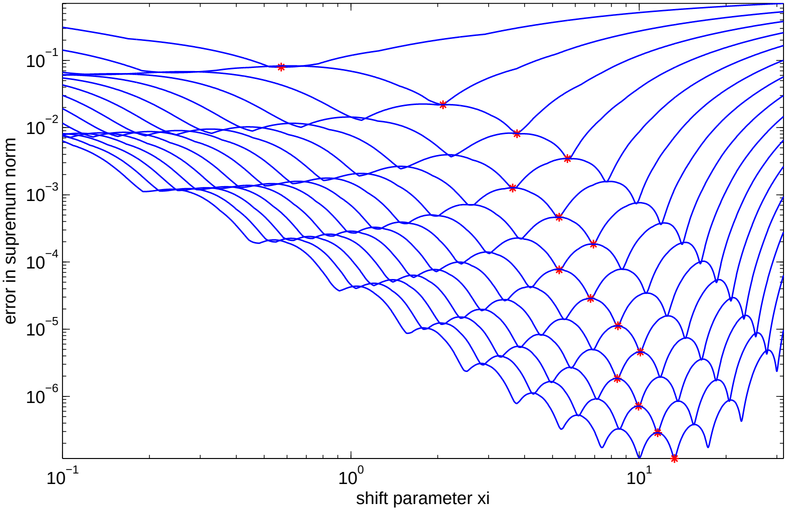
Remarks:

- Storage for only 3 vectors needed, no inner products.
- Can be applied with a block $v = [v_1, \ldots, v_J]$, allowing the simultaneous solution of

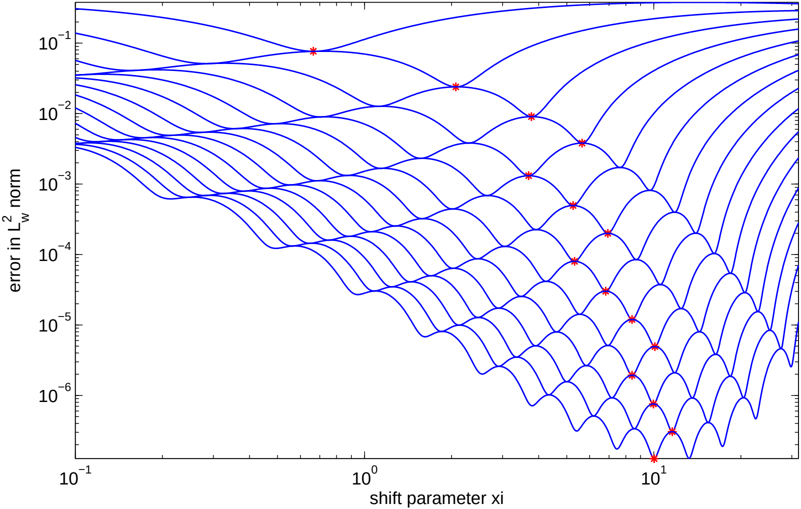
$$v_j'(t) = \mathbf{A}v_j(t), \quad v_j(0) = v_{j,0} \quad (j = 1, \ldots, J).$$

- Each action of $\mathbf{B} = (\xi\mathbf{I} + \mathbf{A})(\xi\mathbf{I} - \mathbf{A})^{-1}$ involves linear system solve.
- **How to choose ξ?**

Recall: S_m is the truncated Chebyshev expansion of $g^\xi(x) = \exp(\xi \frac{x-1}{x+1})$, so in fact $S_m = S_m^\xi$. Minimize $\|g^\xi(x) - S_m^\xi(x)\|_\infty$ as a function of ξ.



The picture looks even nicer if we measure in the appropriate norm, i.e., we plot $\|g^{\xi}(x) - S_m^{\xi}(x)\|_{L_w^2}$.



Conjecture: The function $\xi \mapsto \|g^{\xi}(x) - S_m^{\xi}(x)\|_{L^2_w}$ has $m-1$ minima and touches the function $\xi \mapsto \|g^{\xi}(x) - S_{m+1}^{\xi}(x)\|_{L^2_w}$ at $m-1$ points.

Remarks:

- A similar conjecture was made by [Hochbruck & van den Eshof 06] for the polynomial uniform best approximation

$$E_m^{\xi} := \arg \min_{p \in \mathcal{P}_{m-1}} \|g^{\xi}(x) - p(x)\|_{\infty}.$$

- It is known that the minimizer ξ of $\|g^{\xi}(x) - E_m^{\xi}(x)\|_{\infty}$ behaves asymptotically like $m/\sqrt{2}$ (for $m \to \infty$). [Anderson 81]
- We conjecture that the same is true for the minimizer ξ of $\|g^{\xi}(x) - S_m^{\xi}(x)\|_{L^2_w}$.

Conjecture: The function $\xi \mapsto \|g^\xi(x) - S_m^\xi(x)\|_{L^2_w}$ has $m-1$ minima and touches the function $\xi \mapsto \|g^\xi(x) - S_{m+1}^\xi(x)\|_{L^2_w}$ at $m-1$ points.

Remarks:

- A similar conjecture was made by [Hochbruck & van den Eshof 06] for the polynomial uniform best approximation

$$E_m^\xi := \arg \min_{p \in \mathcal{P}_{m-1}} \|g^\xi(x) - p(x)\|_\infty .$$

- It is known that the minimizer ξ of $\|g^\xi(x) - E_m^\xi(x)\|_\infty$ behaves asymptotically like $m/\sqrt{2}$ (for $m \to \infty$). [Anderson 81]
- We conjecture that the same is true for the minimizer ξ of $\|g^\xi(x) - S_m^\xi(x)\|_{L^2_w}$.
- In fact, S_m^ξ and E_m^ξ cannot be arbitrarily far: (cf. book [Rivlin 90])

$$\begin{aligned}\|g^\xi(x) - E_m^\xi(x)\|_\infty &\le \|g^\xi(x) - S_m^\xi(x)\|_\infty \\ &\le \left(4 + \frac{4}{\pi^2}\log m\right) \|g^\xi(x) - E_m^\xi(x)\|_\infty .\end{aligned}$$

Inexact solves

Recall: In each iteration of RCM a linear system is solved:

- $C_0 = v$,
- $C_1 = \mathbf{B}v$, where $\mathbf{B} = (\xi\mathbf{I} + \mathbf{A})(\xi\mathbf{I} - \mathbf{A})^{-1}$,
- $C_j = 2\mathbf{B}C_{j-1} - C_{j-2}$ (for $j = 2, 3, \ldots$).

Often in practice, $\widetilde{C}_j = C_j + E_j$ is an approximation to C_j and we have

$$(C_j + E_j) = 2\mathbf{B}(C_{j-1} + E_{j-1} + R_{j-1}) - (C_{j-2} + E_{j-2}),$$

where R_{j-1} is a residual. Substracting the exact recursion yields

$$E_j = 2\mathbf{B}(E_{j-1} + R_{j-1}) - E_{j-2}.$$

Note that $\|\mathbf{B}\|_2 \leq 1$ and $\mathbf{B} = \mathbf{UDU}^*$. With $\widehat{E}_j = \mathbf{U}^* E_j$ and $\widehat{R}_j = \mathbf{U}^* R_j$ we obtain recurrence formulas for the entries

$$\hat{e}_j^{(i)} = 2\lambda^{(i)} (\hat{e}_{j-1}^{(i)} + \hat{r}_{j-1}^{(i)}) - \hat{e}_{j-2}^{(i)}.$$

We consider the recurrence (entry indizes removed)

$$\hat{e}_0 = 0, \quad \hat{e}_1 = r, \quad \hat{e}_j = 2\lambda(\hat{e}_{j-1} + r) - \hat{e}_{j-2},$$

whose fastest growing solution (for $\lambda = 1$) is $\hat{e}_j = r\, j^2$.

If we assume $\|R_j\|_2 = \|\widehat{R}_j\|_2 \le \mathrm{tol}\, \|v\|_2$, we find that

$$\|E_j\|_2 = \|\widehat{E}_j\|_2 \le \mathrm{tol}\, \|v\|_2\, j^2.$$

Finally,

$$\left\| \widetilde{S}_m(\mathbf{B})v - S_m(\mathbf{B})v \right\|_2 = \left\| \sum_{j=0}^{m-1} \gamma_j \widetilde{C}_j - \sum_{j=0}^{m-1} \gamma_j C_j \right\|_2 \le \mathrm{tol}\, \|v\|_2 \sum_{j=0}^{m-1} j^2\, |\gamma_j|.$$

The rational Chebyshev method is stable w.r.t. inexact solves:

ξ	5	10	15	20	25
$\sum_{j=0}^{\infty} j^2\, \lvert\gamma_j\rvert$	2.90	5.11	7.54	10.02	12.50

Numerical Example

1D heat equation with homogeneous Dirichlet boundary condition and "oscillating hat" source (half-width $w = 0.05$, height $h = 50$, frequency $f = 23$):

$$\begin{aligned} \partial_t u(t,x) &= \partial_{xx} u(t,x) + q(t,x) \qquad && \text{on } x \in (0,1), \\ u(t,0) &= u(t,1) \;=\; 0, && \\ u(0,x) &= u_0(x) \;=\; x(1-x), && \\ q(t,x) &= h \max\{1 - |c-x|/w, 0\}, && c = .5 + (.5 - w)\sin(2\pi f t). \end{aligned}$$

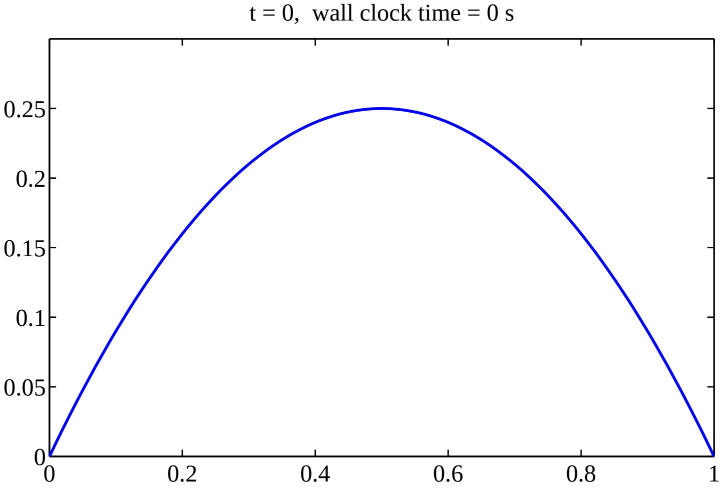
Finite-difference discretization in space yields

$$u'(t) = \mathbf{A} u(t) + q(t), \quad u(0) = u_0,$$

where $\mathbf{A} = (d+1)^2 \operatorname{tridiag}(1,-2,1) \in \mathbb{R}^{d \times d}$, $d = 100$.

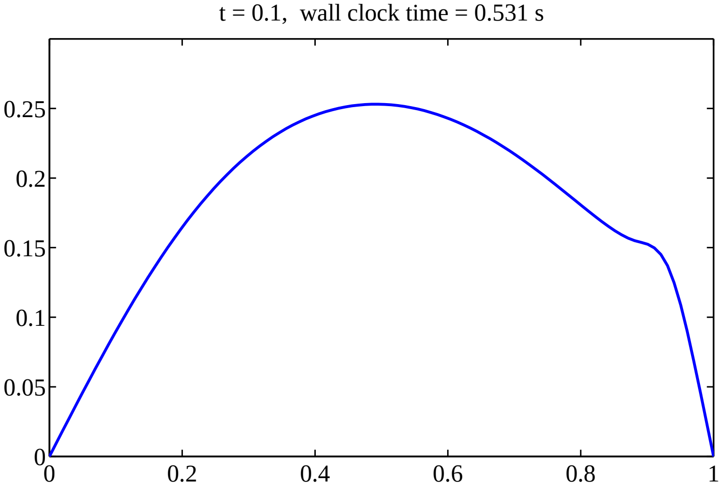
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



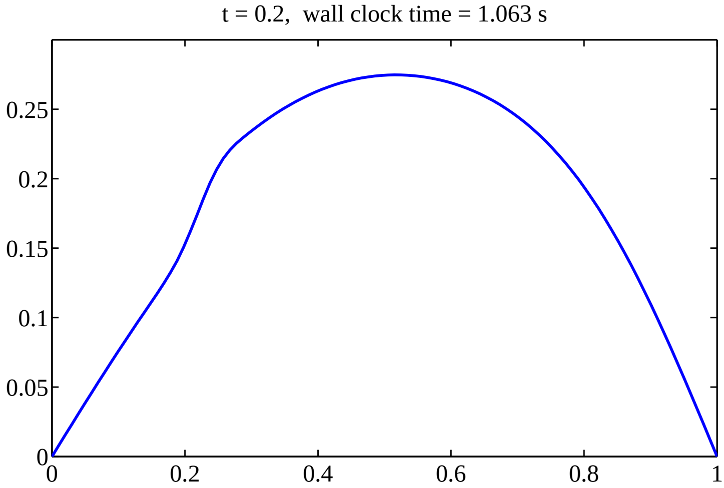
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



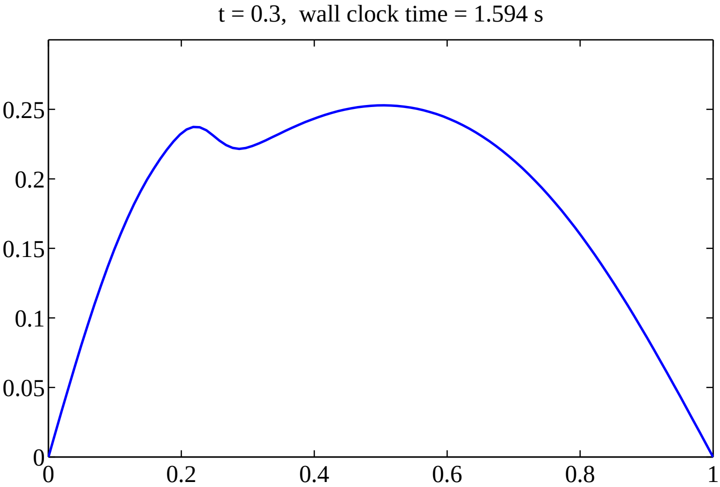
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



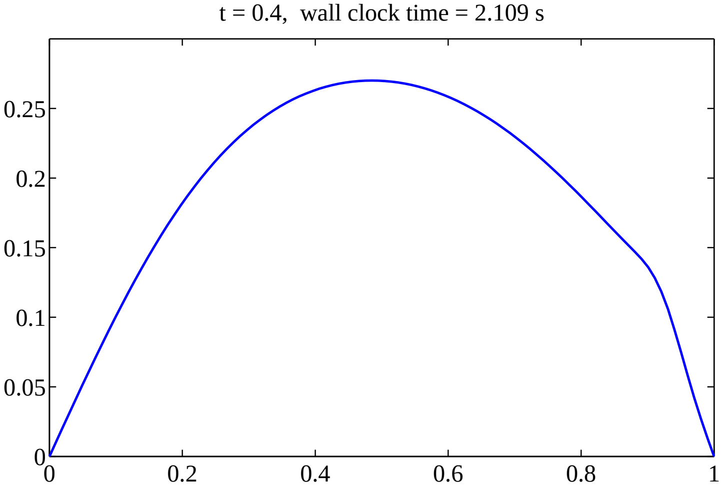
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



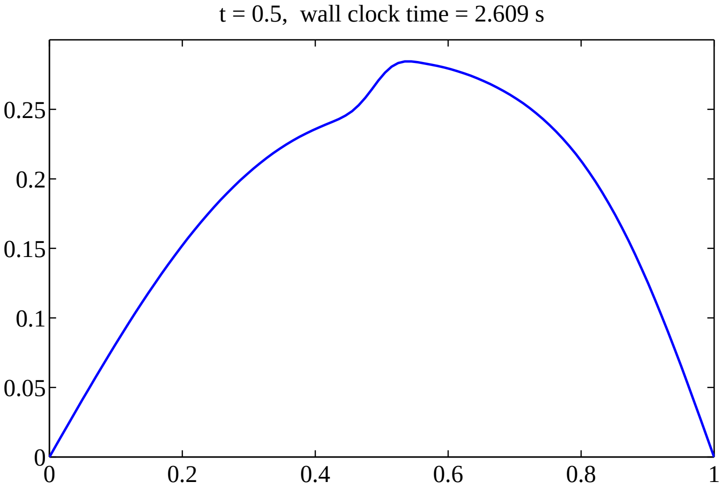
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



Serial integration

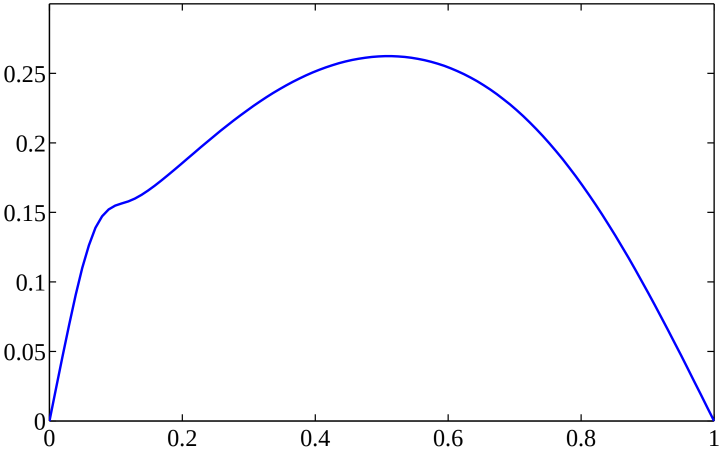
We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



Serial integration

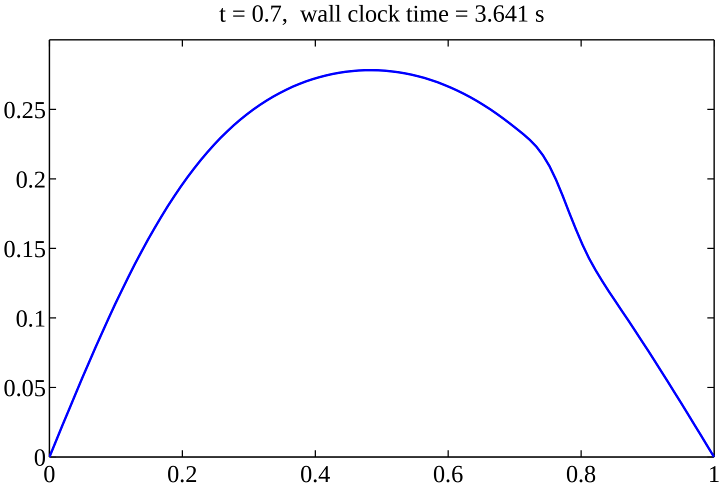
We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s

t = 0.6, wall clock time = 3.125 s



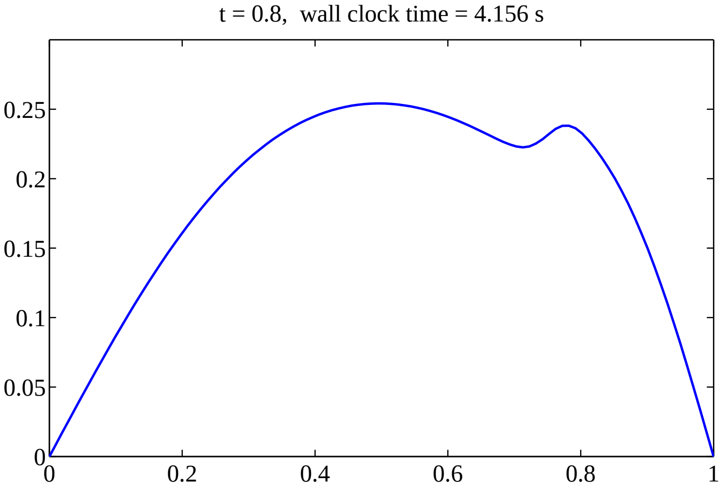
Serial integration

We integrate for $t \in [0,1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



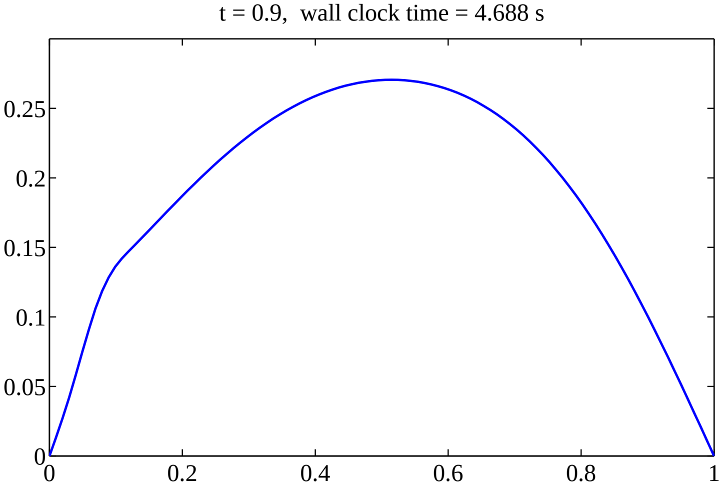
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



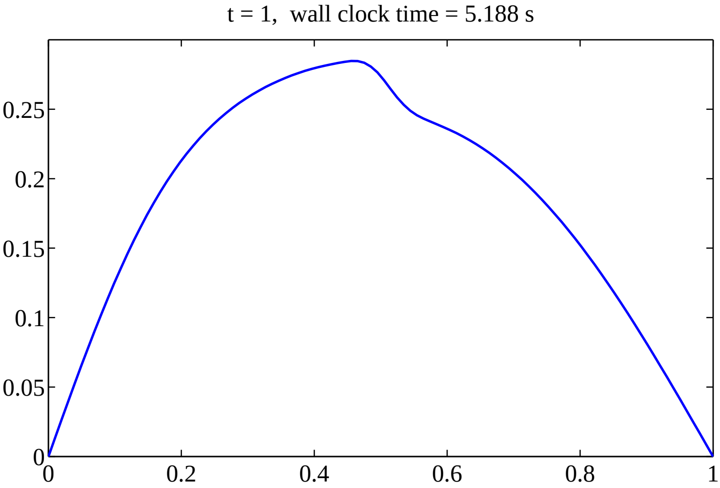
Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s

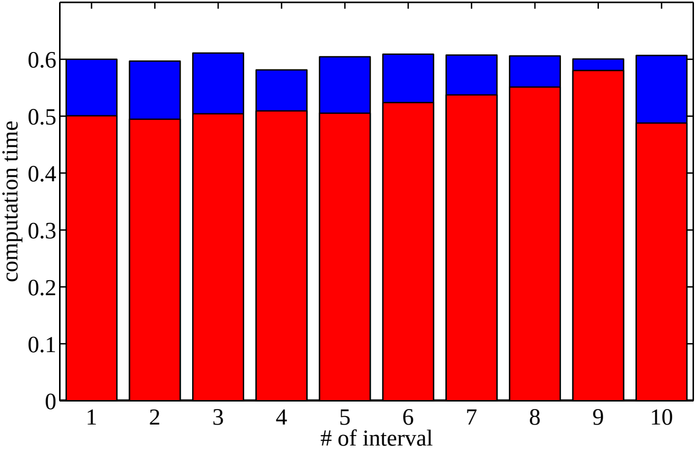


Serial integration

We integrate for $t \in [0, 1]$ using `ode15s` with error tolerance 10^{-3}.
Overall computation time: 5.2 s



Parallel computation time with 10 processors (`ode15s` / cheby)



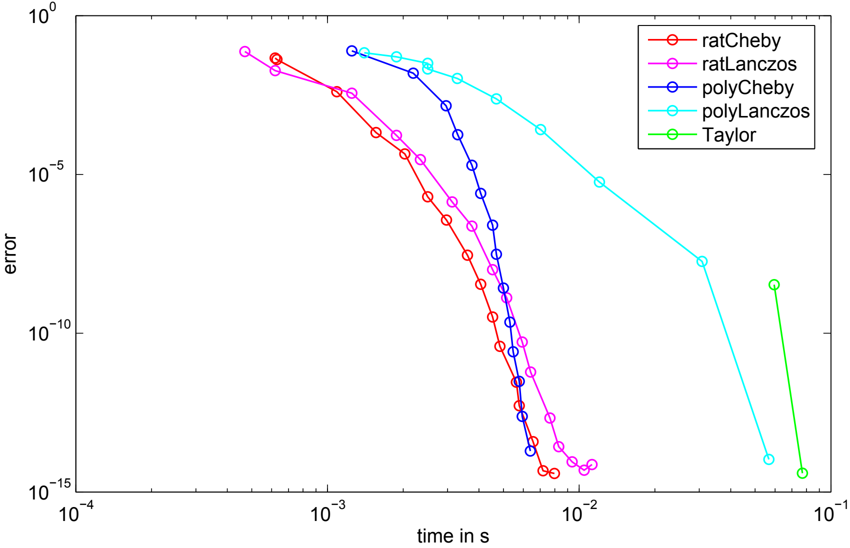
Speedup = 5.2 / 0.6 ≈ 8.7. (with balanced interval lengths)

Summary

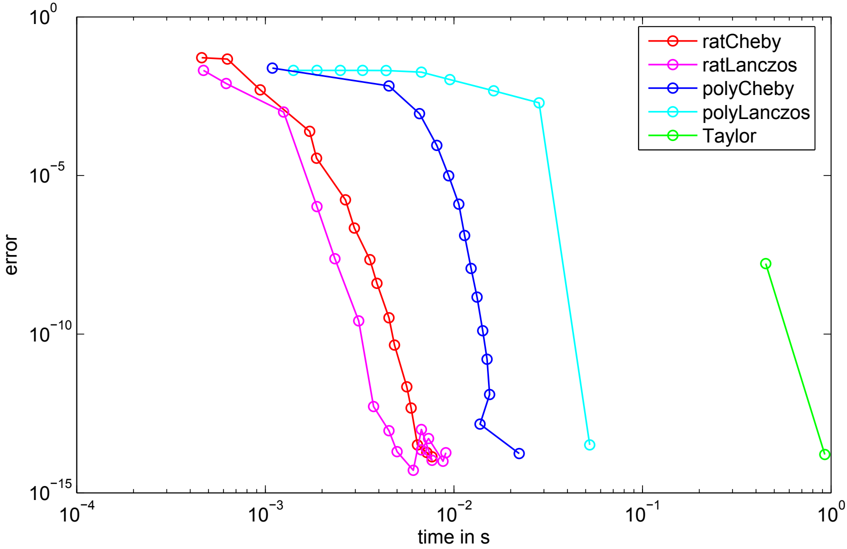
- Almost perfectly scaling solution methods can be derived for linear problems by appropriate decomposition into homogeneous and inhomogeneous subproblems.
- The proposed overlapping time-decomposition removes almost all communication need between processors.
- Efficient (in fact, near-optimal in some sense) rational Krylov methods are available to integrate the homogeneous subproblems.
- The proposed rational Chebyshev method is one of these methods, applicable if $\mathbf{A}$ is symmetric negative-semidefinite.
- This method can handle multiple vectors v simultaneously.
- For semilinear problems $u'(t) = \mathbf{A}u(t) + g(t, u(t))$ similar ideas can be applied. However, in this case an outer correction iteration (e.g., parareal) is required at extra cost.

Matlab function `rcexpmv(A,v,tol)` available: `www.guettel.com`

Compute $\exp(t\mathbf{A})v$, $\mathbf{A} = (d+1)^2\,\mathrm{tridiag}(1,-2,1) \in \mathbb{R}^{d\times d}$, $v = \mathrm{randn}$
$d = 100$, $t = 0.01$



Compute $\exp(t\mathbf{A})v$, $\mathbf{A} = (d+1)^2\,\mathrm{tridiag}(1,-2,1) \in \mathbb{R}^{d\times d}$, $v = \mathrm{randn}$
$d = 100$, $t = 0.1$



Compute $\exp(t\mathbf{A})v$, $\mathbf{A} = (d+1)^2\, \mathrm{tridiag}(1,-2,1) \in \mathbb{R}^{d\times d}$, $v = \mathrm{randn}$
$d = 100$, $t = 1.0$