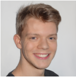
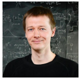
Probabilistic ODE Solutions in Millions of Dimensions



*Nico Krämer** Nathanael Bosch* Jonathan Schmidt* Philipp Hennig

* Equal contribution

June 27, 2022

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Faculty of Science
Department of Computer Science
Chair for the Methods of Machine Learning

MAX-PLANCK-GESELLSCHAFT
erc
ellis
CyberValley
DFG
Federal Ministry of Education and Research
Baden-Württemberg MINISTERIUM FÜR WISSENSCHAFT, FORSCHUNG UND KUNST
imprs-is

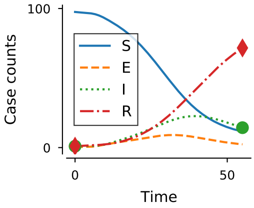
Simulation of ordinary differential equations

Simulation of ordinary differential equations

- ODE: $\dot{y}(t) = f(y(t), t),\ y(t_0) = y_0 \in \mathbb{R}^d$

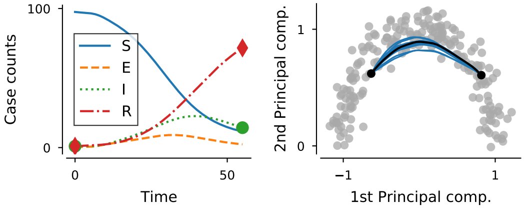
Simulation of ordinary differential equations

- ODE: $\dot{y}(t) = f(y(t), t),\ y(t_0) = y_0 \in \mathbb{R}^d$
- Applications *all over the natural sciences:* biology, geosciences, fluid dynamics, ...



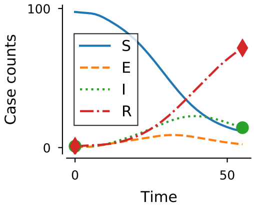
Simulation of ordinary differential equations

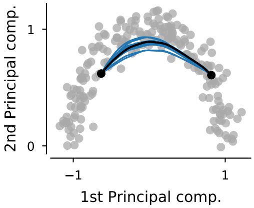
- ODE: $\dot{y}(t) = f(y(t), t), \; y(t_0) = y_0 \in \mathbb{R}^d$
- Applications *all over the natural sciences:* biology, geosciences, fluid dynamics, ...
- Applications *all over machine learning:* neural {O, S, P}DEs, PINNs, ...



Simulation of ordinary differential equations

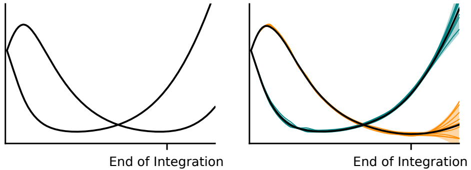
- ODE: $\dot{y}(t) = f(y(t), t),\; y(t_0) = y_0 \in \mathbb{R}^d$
- Applications *all over the natural sciences:* biology, geosciences, fluid dynamics, ...
- Applications *all over machine learning:* neural {O, S, P}DEs, PINNs, ...
- Many dimensions, $d \gg 100$. Think: $d \sim 10^6$

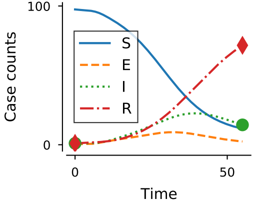


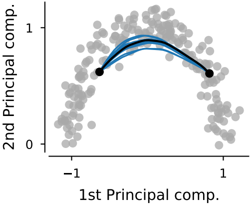


Simulation of ordinary differential equations

- ODE: $\dot{y}(t) = f(y(t), t),\ y(t_0) = y_0 \in \mathbb{R}^d$
- Applications *all over the natural sciences:* biology, geosciences, fluid dynamics, ...
- Applications *all over machine learning:* neural {O, S, P}DEs, PINNs, ...
- Many dimensions, $d \gg 100$. Think: $d \sim 10^6$







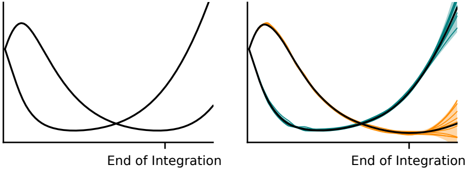
- Any approximation of the posterior distribution

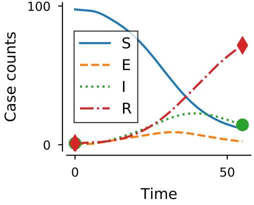
$$p\left(y \;\middle|\; \{\dot{y}(t_n) = f(y(t_n), t_n)\}_{n=0}^{N}, y(t_0) = y_0\right)$$

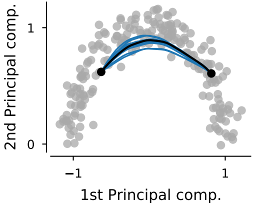
approximately solves the ODE

Simulation of ordinary differential equations

- ODE: $\dot{y}(t) = f(y(t), t),\ y(t_0) = y_0 \in \mathbb{R}^d$
- Applications *all over the natural sciences:* biology, geosciences, fluid dynamics, ...
- Applications *all over machine learning:* neural {O, S, P}DEs, PINNs, ...
- Many dimensions, $d \gg 100$. Think: $d \sim 10^6$







- Any approximation of the posterior distribution

$$p\left(y \;\middle|\; \{\dot{y}(t_n) = f(y(t_n), t_n)\}_{n=0}^{N}, y(t_0) = y_0\right)$$

approximately solves the ODE
- *Complexity $O(d^3)$ per step ϟϟ*

Solution: more structure in the state space

Solution: more structure in the state space

Theorem (Kronecker; simplified)

If the νth-order prior has Kronecker structure, a single step with the prob. ODE solver costs $O(\nu^3 + d\nu^2)$ in float-ops and $O(d\nu + d^2 + \nu^2)$ in memory.

Solution: more structure in the state space

Theorem (Kronecker; simplified)

If the νth-order prior has Kronecker structure, a single step with the prob. ODE solver costs $O(\nu^3 + d\nu^2)$ in float-ops and $O(d\nu + d^2 + \nu^2)$ in memory.

Theorem (Independence; simplified)

If the νth-order prior independent prior dimensions, a single step with the prob. ODE solver costs $O(d\nu^3)$ in float-ops and $O(d\nu^2)$ in memory.

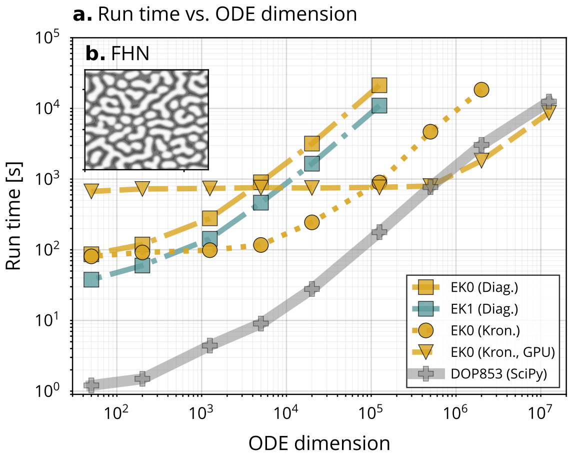
Solution: more structure in the state space

Theorem (Kronecker; simplified)

If the νth-order prior has Kronecker structure, a single step with the prob. ODE solver costs $O(\nu^3 + d\nu^2)$ in float-ops and $O(d\nu + d^2 + \nu^2)$ in memory.

Theorem (Independence; simplified)

If the νth-order prior independent prior dimensions, a single step with the prob. ODE solver costs $O(d\nu^3)$ in float-ops and $O(d\nu^2)$ in memory.



Conclusion

Conclusion

- Structured state spaces accelerate ODE solvers

Conclusion

- Structured state spaces accelerate ODE solvers
- Simulation of millions of dimensions

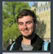
Conclusion

- Structured state spaces accelerate ODE solvers
- Simulation of millions of dimensions
- Simulation of PDEs and more

Conclusion

- Structured state spaces accelerate ODE solvers
- Simulation of millions of dimensions
- Simulation of PDEs and more


Nico Krämer (**`@pnkraemer`**)


Nathanael Bosch (**`@NathanaelBosch`**)


Jonathan Schmidt

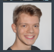

Philipp Hennig (**`@PhilippHennig5`**)

Conclusion

- Structured state spaces accelerate ODE solvers
- Simulation of millions of dimensions
- Simulation of PDEs and more


Nico Krämer (`@pnkraemer`)


Nathanael Bosch (`@NathanaelBosch`)


Jonathan Schmidt


Philipp Hennig (`@PhilippHennig5`)

Software: `https://github.com/pnkraemer/tornadox`

```
pip install tornadox
```

Paper: `https://arxiv.org/abs/2110.11812`

Come and talk to us. Visit the poster.

Experiment code