Exploiting "time-domain" parallelism to accelerate neural network training and PDE constrained optimization

Eric C. Cyr and Denis Ridzal (SNL)

Stefanie Guenther (LLNL), Lars Ruthotto (Emory), Jacob B. Schroder (UNM), Nico R. Gauger (TU Kaiserslautern)

Problem

Optimization with some time-domain PDEs and training very deep neural networks is intractable. Bigger computers will not help!

Evolutionary processes are defined by the step:

$$u_{n+1} = A_z(u_n) \text{ for } n = 0 \dots N$$
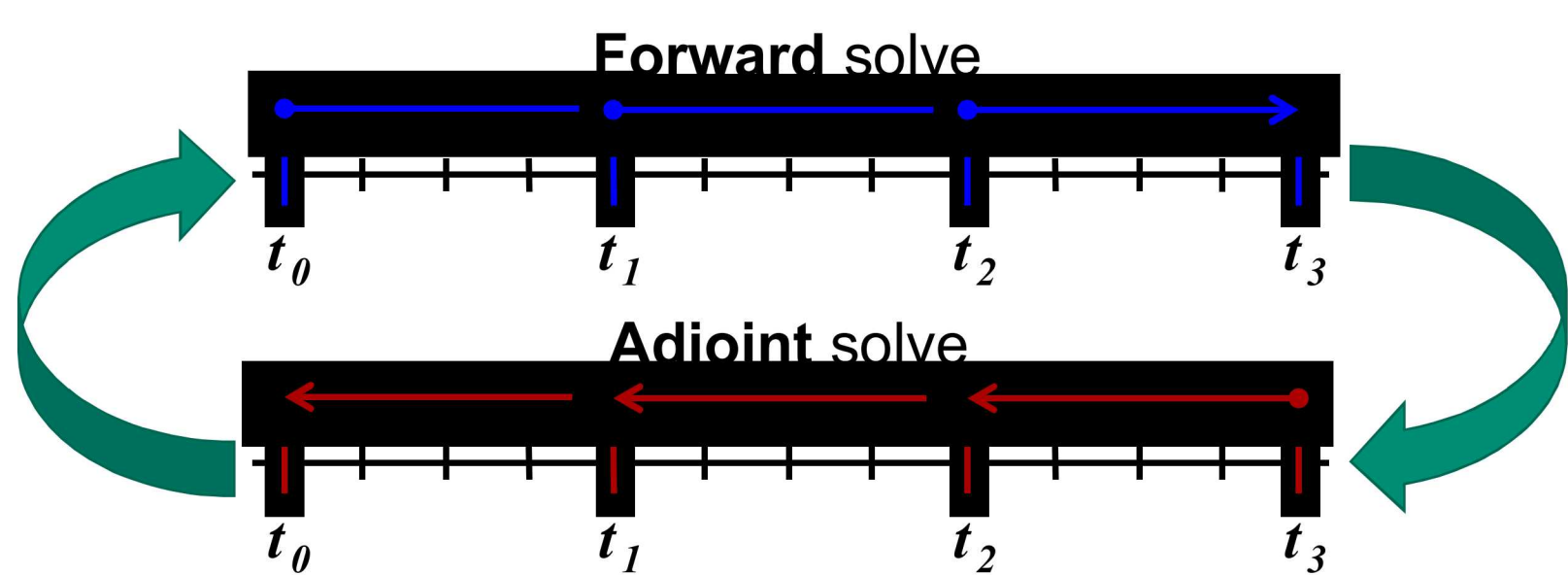
For Example:

1) Numerical PDEs (Maxwell's Equation):

$$[E_{n+1}, B_{n+1}] = [E_n, B_n] + \Delta t[\nabla \times B_n - J, -\nabla \times E_n]$$

2) Neural Networks (ResNet):

$$x_{n+1} = x_n + \sigma(A_n x_n + b_n)$$

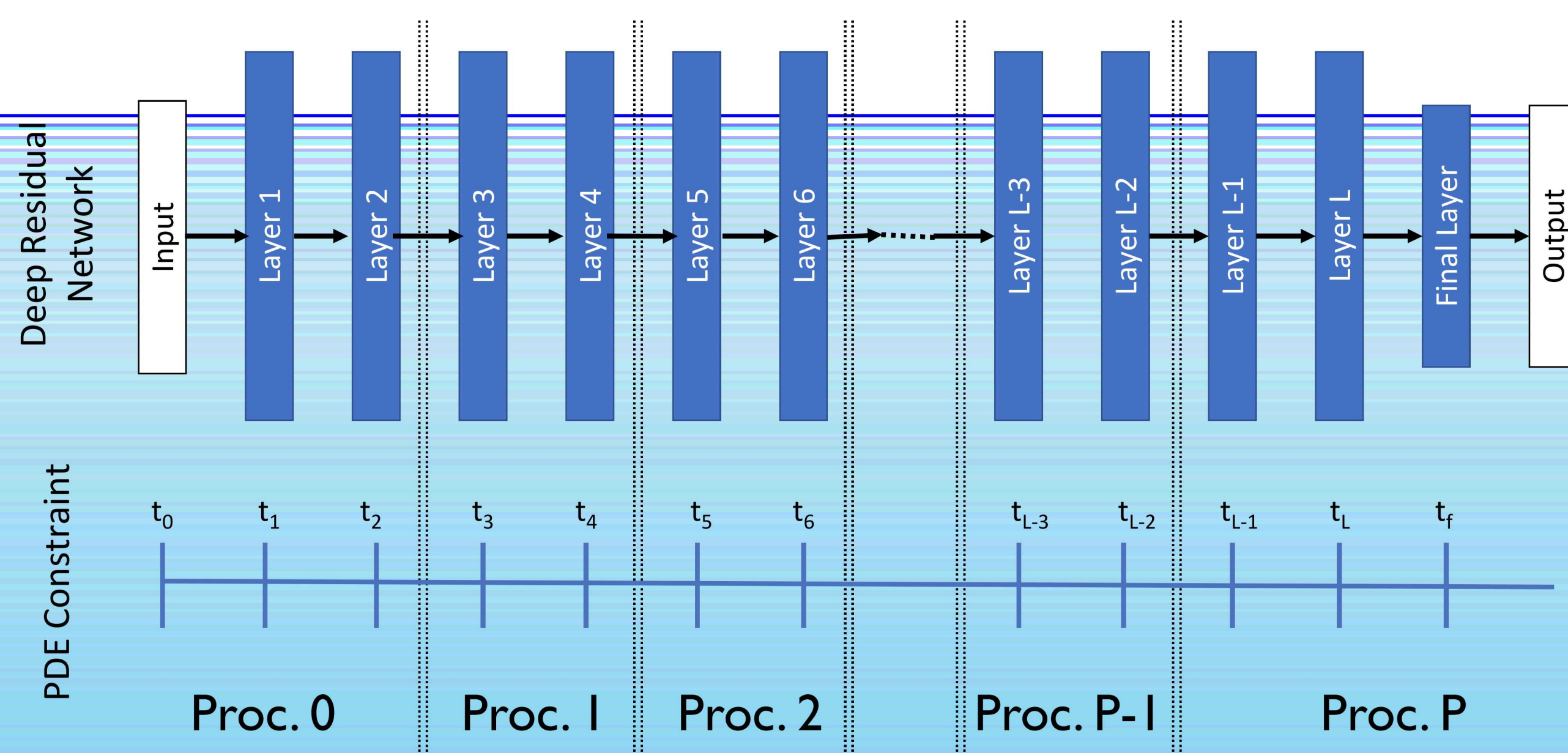Optimization constrained by these processes requires a forward and backward sweep:

Forward and adjoint solves are serial bottlenecks, leading to unreasonable computational times
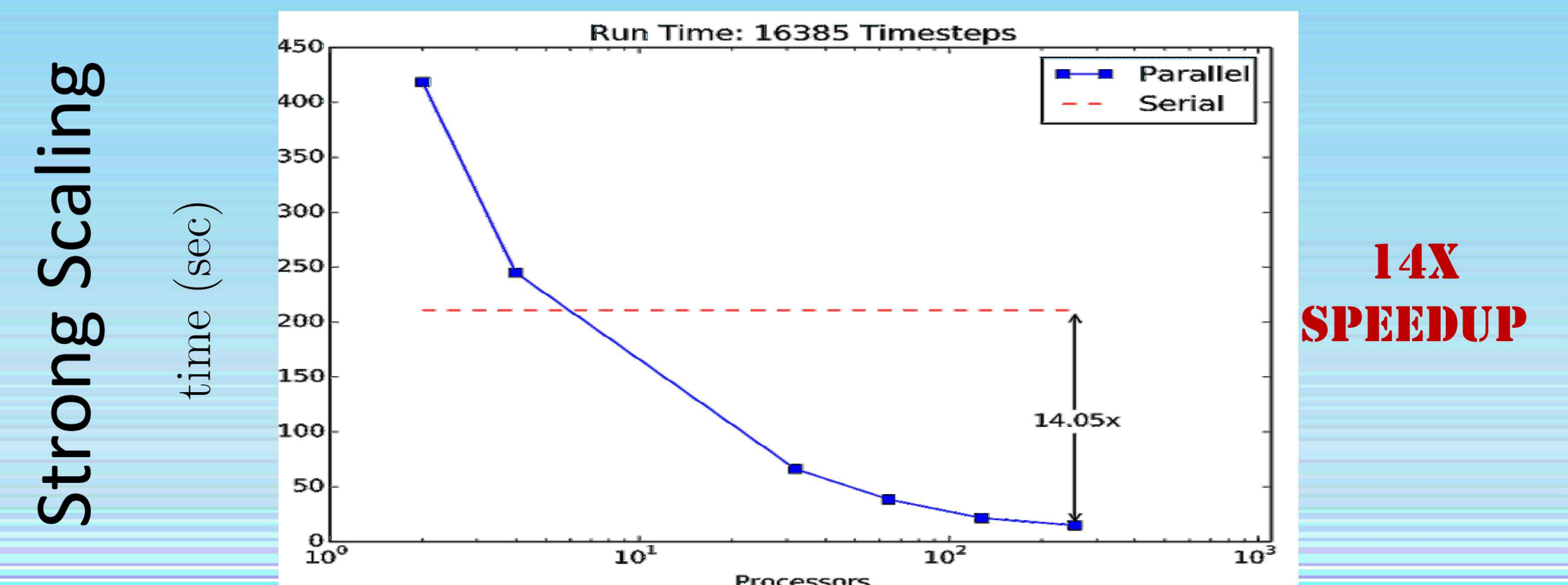- Using a bigger computer will not solve this!

Approach

Distribute "time steps" across multiple processors, multigrid allows uses parallel solves
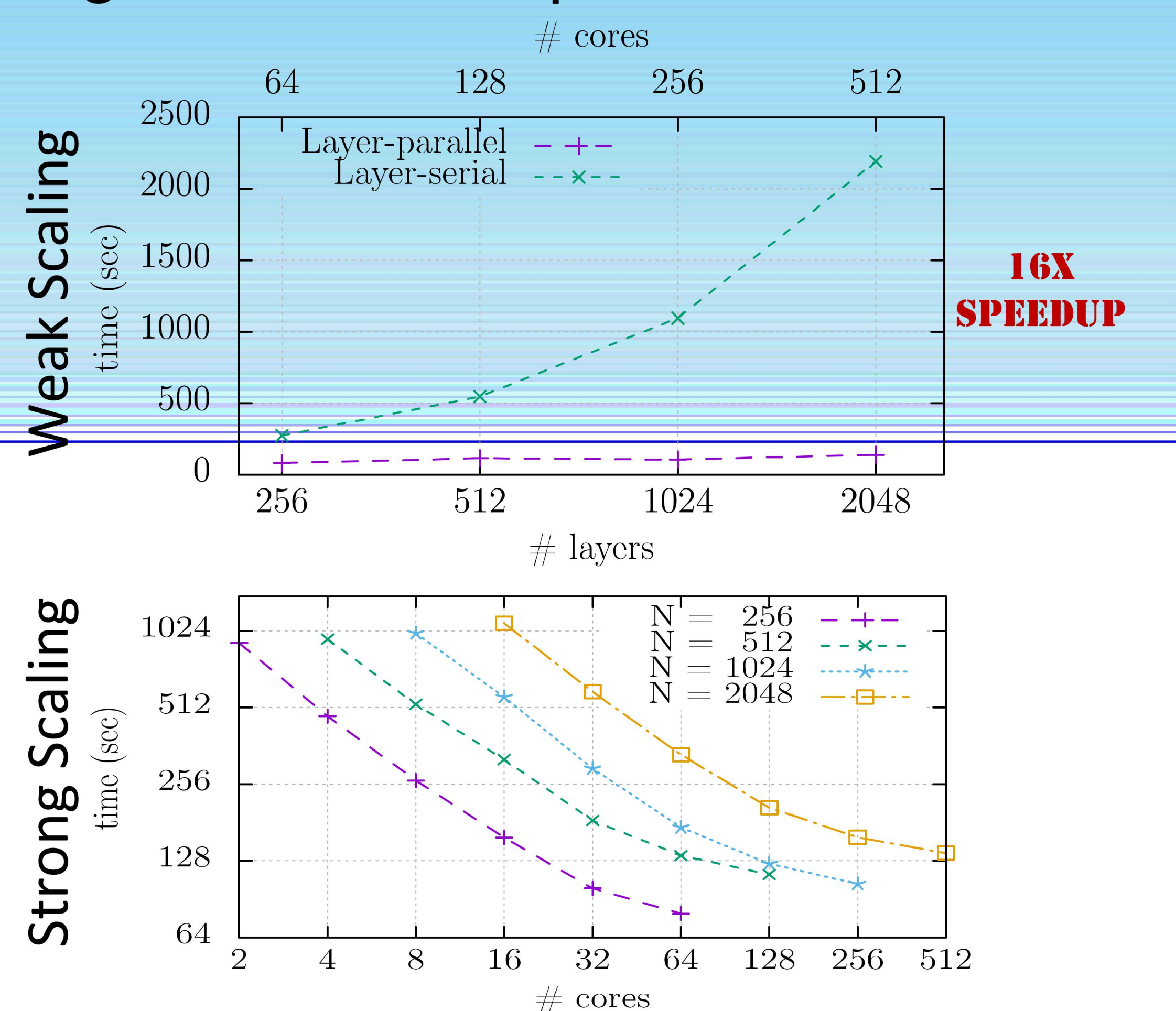

Results

Over 10x Speedup: PDE-Opt and DNN Training!

1) Source inversion for the heat equation

14X SPEEDUP

2) Training for the MNIST problem

16X SPEEDUP

Significance

Parallel-in-time/Layer-parallel algorithms for PDE-constr. optimization and training neural networks:
- Order of magnitude speedups for both neural networks and PDE constrained optimization
- Algorithms can be developed to improve both PDE-Opt and Training DNN

Pubs: Arxiv, SIMOD (sub.), Talks: Copper, SIAM CSE, ICIAM, PinT Workshop, ICCOPT, DD

Funding