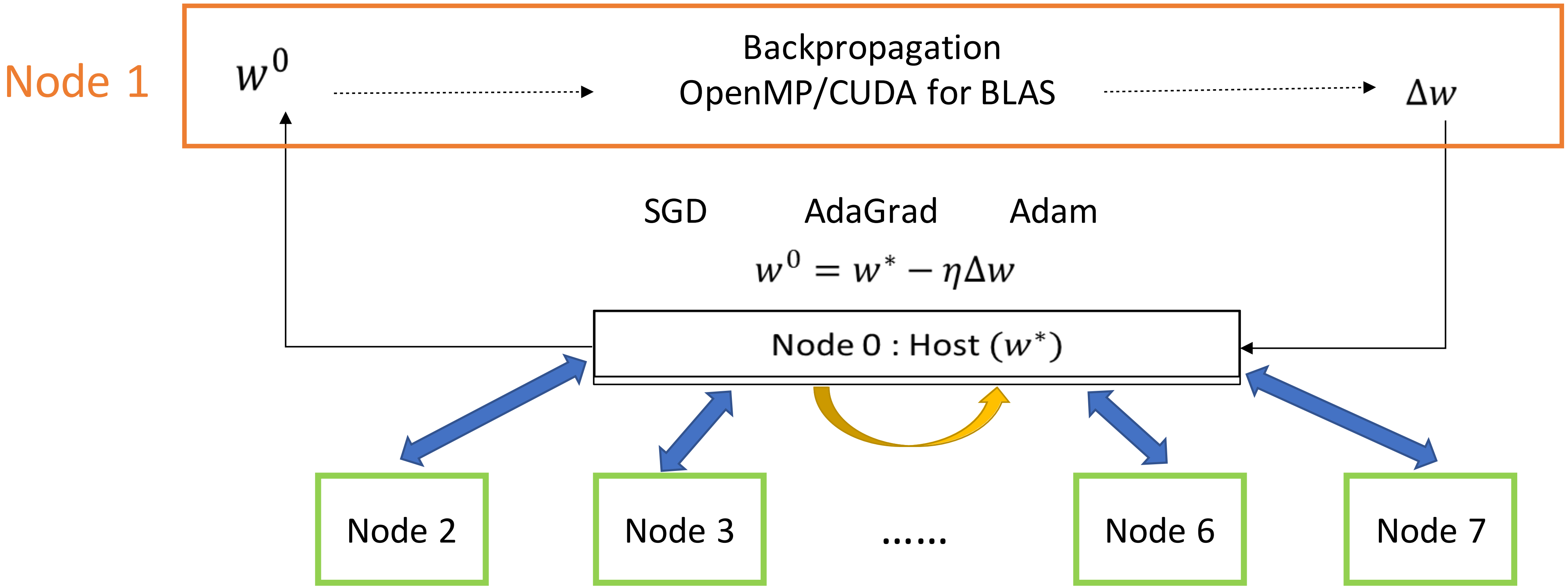
Parallelizing Neural Network for High-frequency Stock Prediction

Linglin Huang, Chang Liu, Greyson Liu, Kamrine Poels

Gradient Descent Methods



Improvements

Hessian-free Optimization (CUDA)

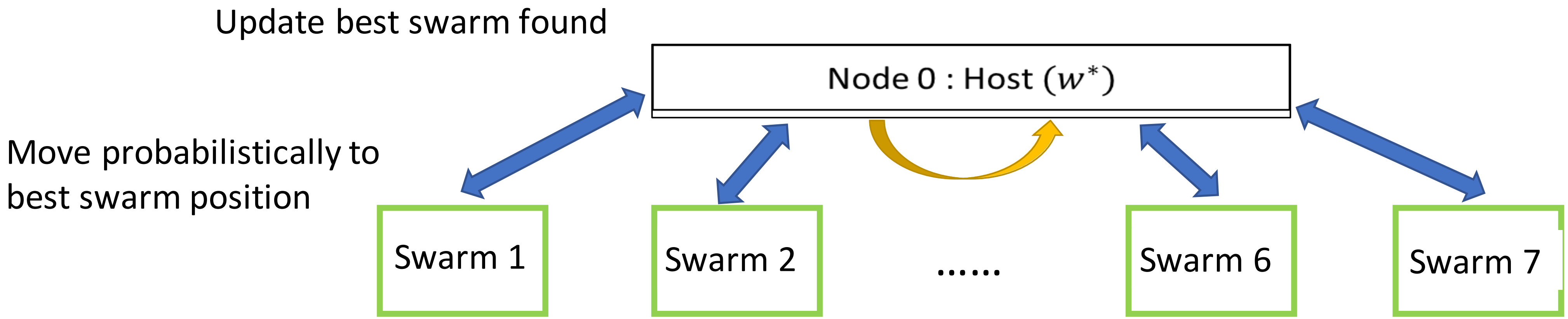
1. Initialize x_i for iterations $i = 0$
2. Use second-order Taylor expansion:

$$f(x + \Delta x) \approx f(x) + f(x)^T \Delta x + \Delta x^T H(f) \Delta x$$

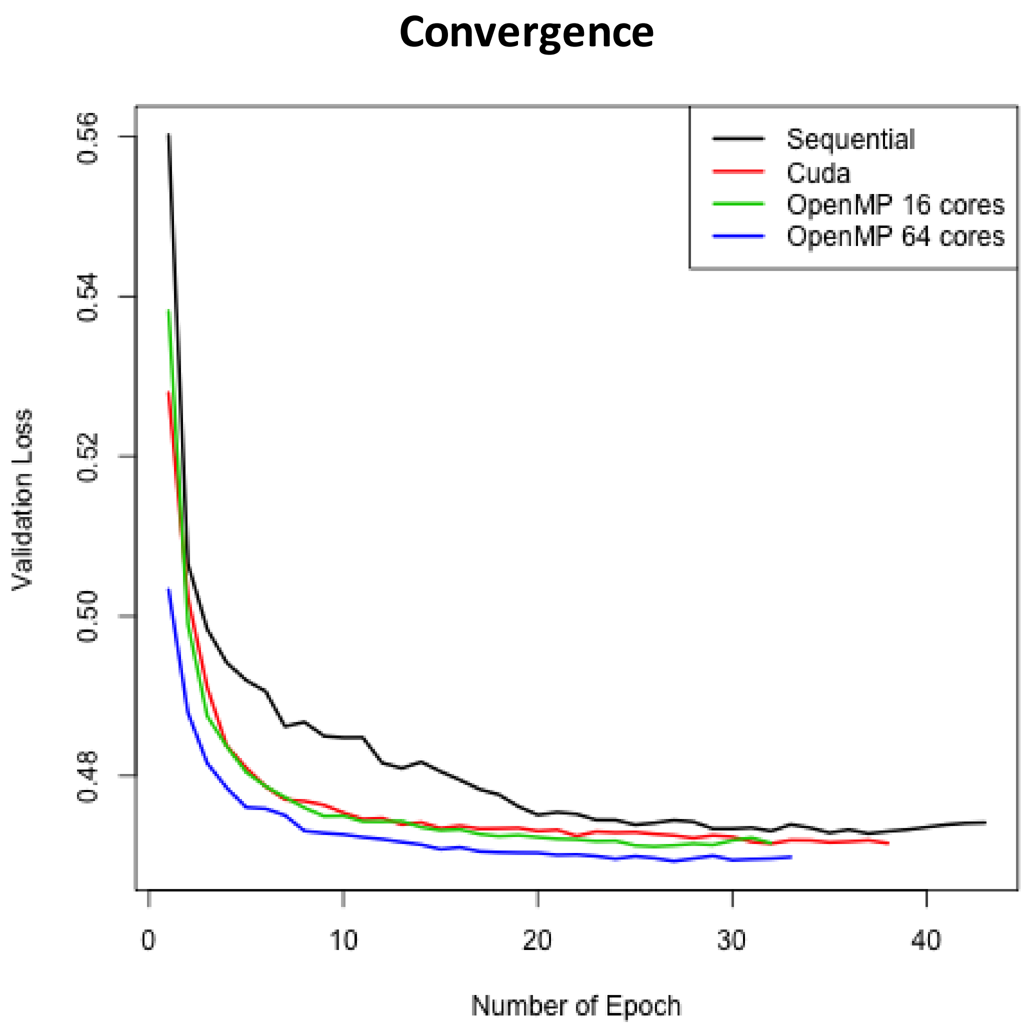
3. Compute x_{i+1} using Conjugate Gradient algorithm for quadratic functions on current expansion.
4. Iterate until convergence

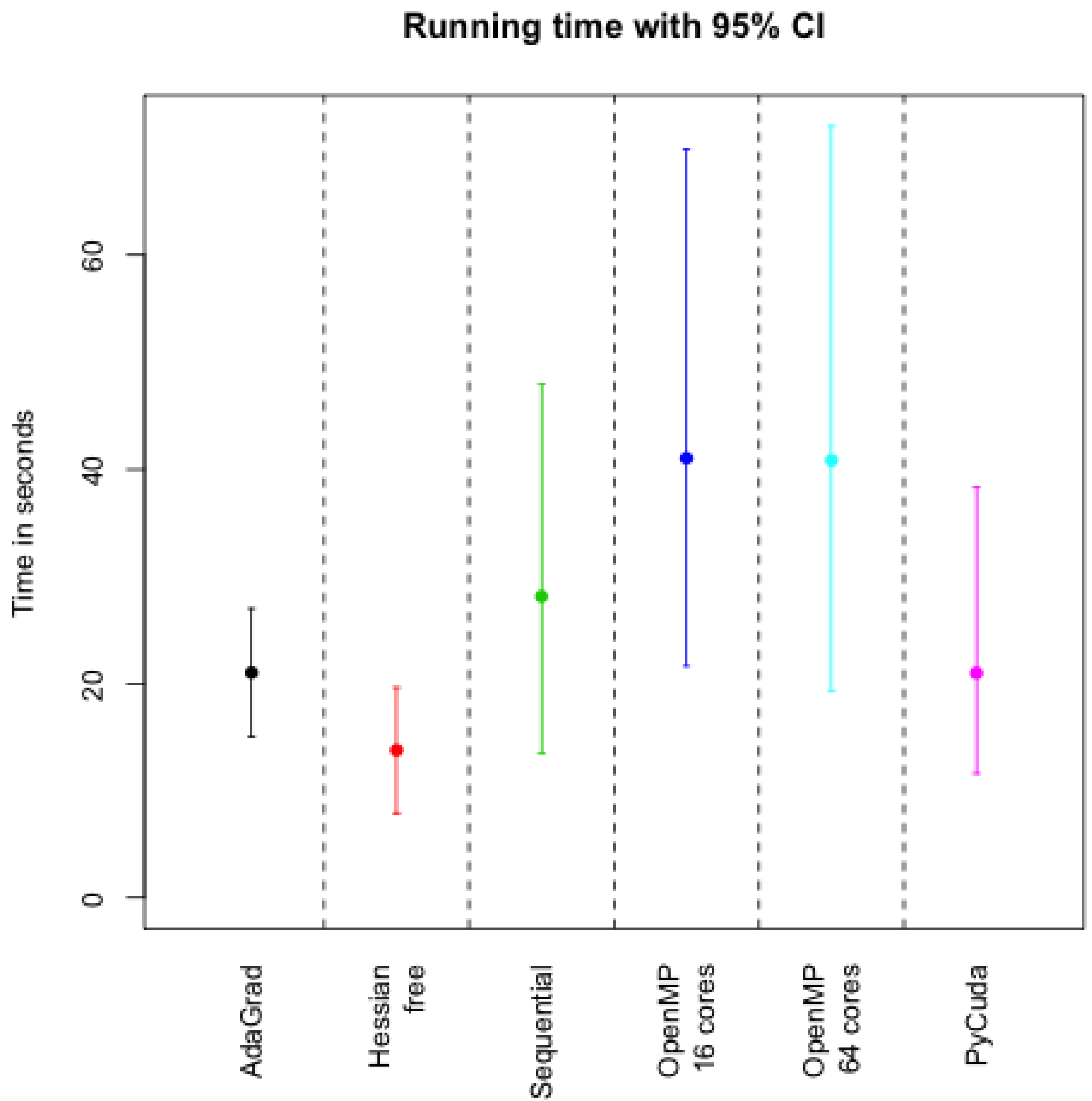
Particle Swarm Optimization

Heuristic random search: Update each model (a swarm) influenced by the best swarms and iterate until convergence



Efficiency Results for Model Replica





Prediction Accuracy of Direction of Next-second Price Movement

Sequential	Adagrad	Hessian Free	Adam
Mean	53%	50%	53%
lower bound	46%	41%	45%
upper bound	58%	58%	57%

Parallel Adam	16 cores	64 cores	CUDA
Mean	53%	53%	52%
lower bound	45%	45%	35%
upper bound	57%	57%	58%

Baseline on test dataset: 41%

Appendix

- http://andrew.gibiansky.com/blog/machine-learning/hessian-free-optimization/

- http://www.cs.toronto.edu/~jmartens/docs/Deep_HessianFree.pdf

- https://static.googleusercontent.com/media/research.google.com/en//archive/large_deep_networks_nips2012.pdf

Conjugate Gradient

Let f be a quadratic function $f(x) = \frac{1}{2}x^T Ax + b^T x + c$ which we wish to minimize.

1. **Initialize:** Let $i = 0$ and $x_i = x_0$ be our initial guess, and compute $d_i = d_0 = -\nabla f(x_0)$.
2. **Find best step size:** Compute α to minimize the function $f(x_i + \alpha d_i)$ via the equation
$$\alpha = -\frac{d_i{}^T (Ax_i + b)}{d_i{}^T Ad_i}.$$
3. **Update the current guess:** Let $x_{i+1} = x_i + \alpha d_i$.
4. **Update the direction:** Let $d_{i+1} = -\nabla f(x_{i+1}) + \beta_i d_i$ where β_i is given by
$$\beta_i = \frac{\nabla f(x_{i+1})^T Ad_i}{d_i{}^T Ad_i}.$$
5. **Iterate:** Repeat steps 2-4 until we have looked in n directions, where n is the size of your vector space (the dimension of x).