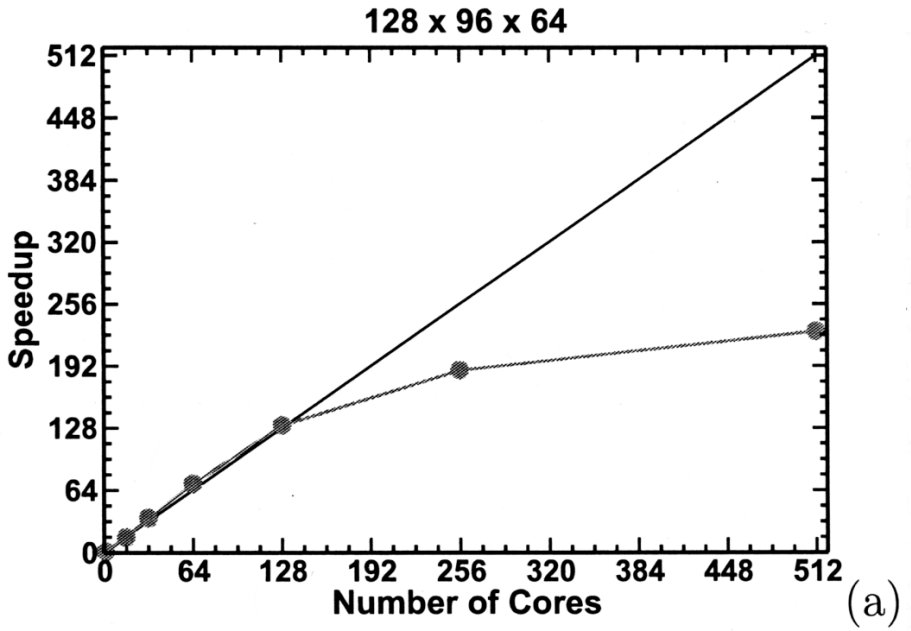
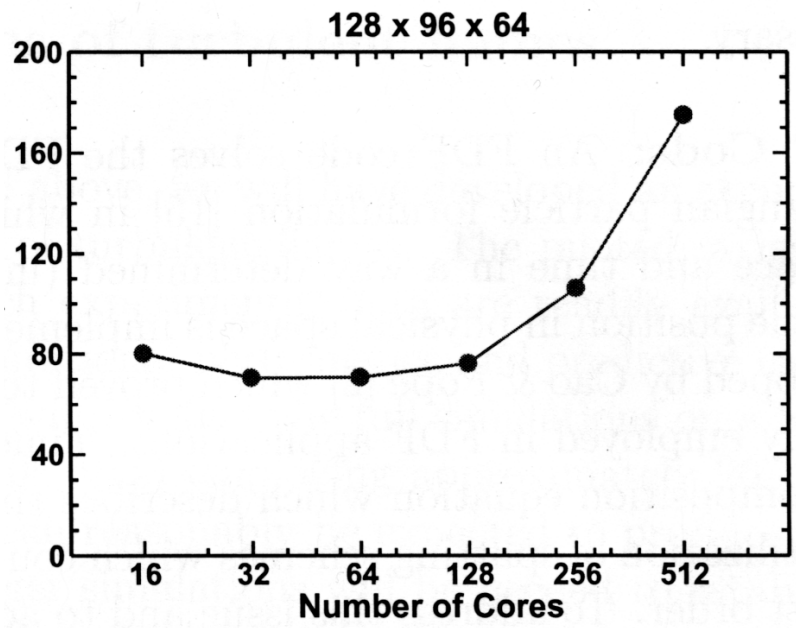
Ranger

Scalability and Optimization

Drew Dolgert



(a)

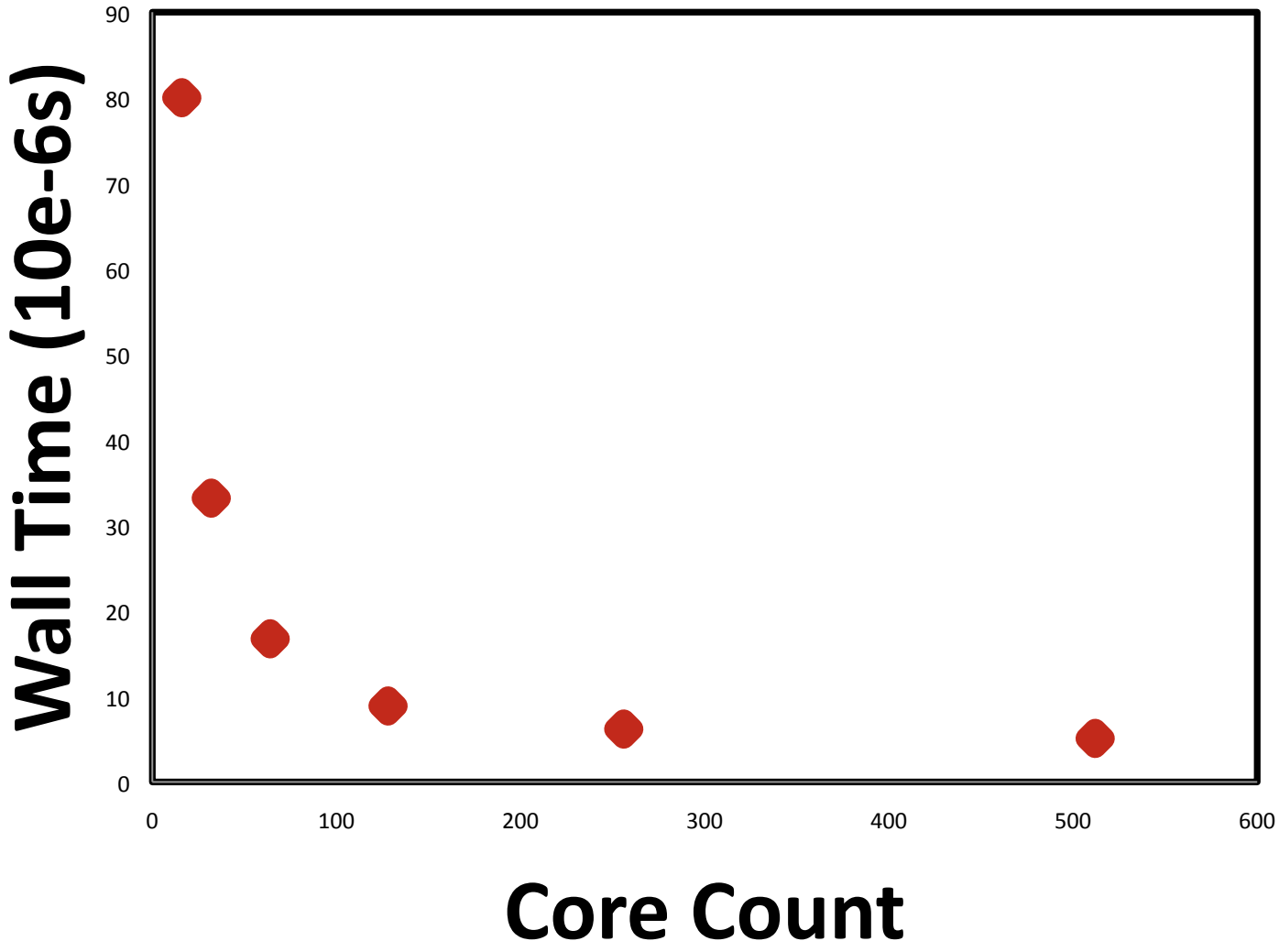


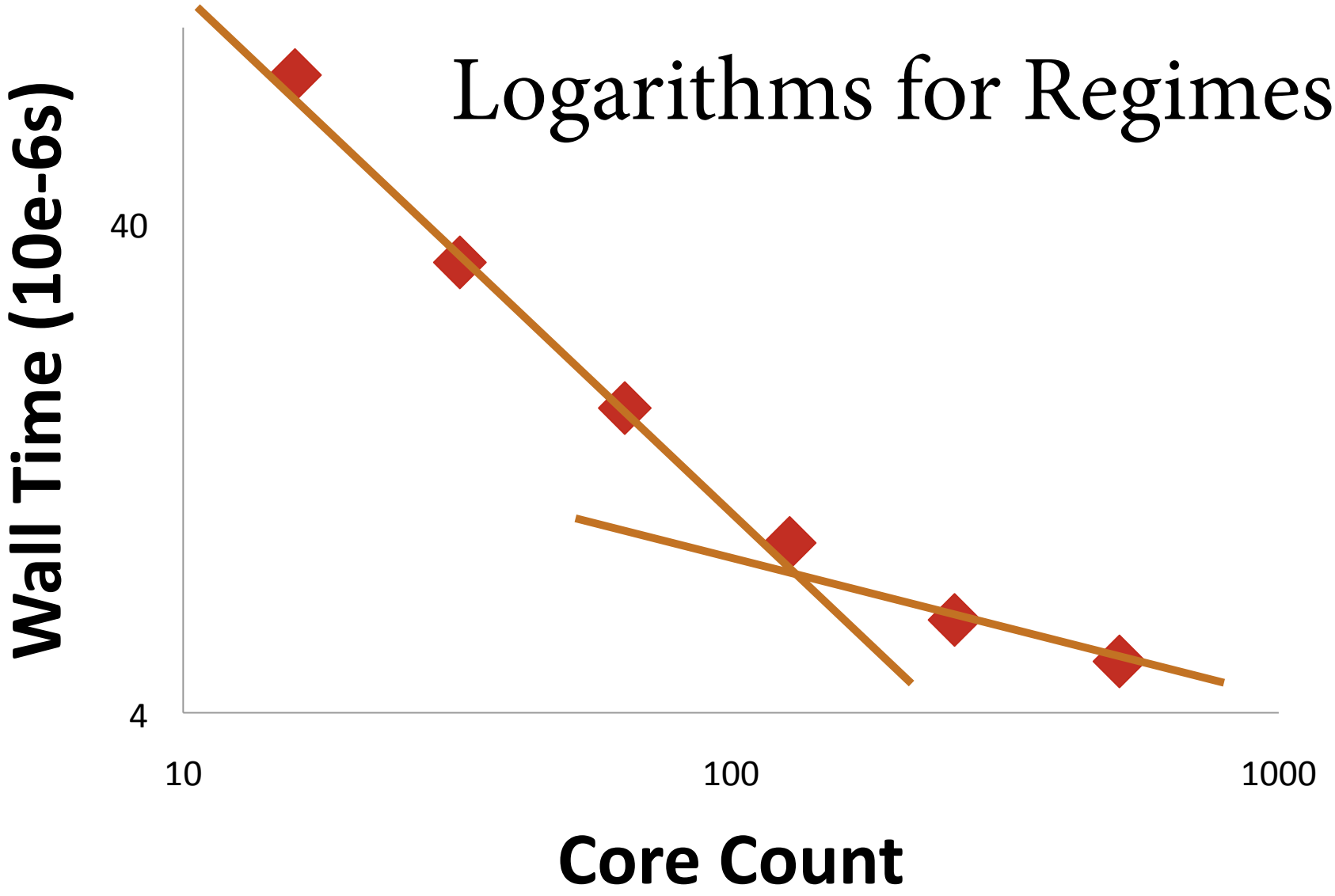
(b)

It is seen that for the chosen grid resolution the LES code exhibits linear scalability up to 128 processors and reasonable scalability up to 256 processors.

This code runs well.

It has to run this many times.


Wall Time (10e-6s)
90
80
70
60
50
40
30
20
10
0
0
100
200
300
400
500
600
Core Count


Logarithms for Regimes
Wall Time (10e-6s)
40
4
10
100
1000
Core Count

Efficiency

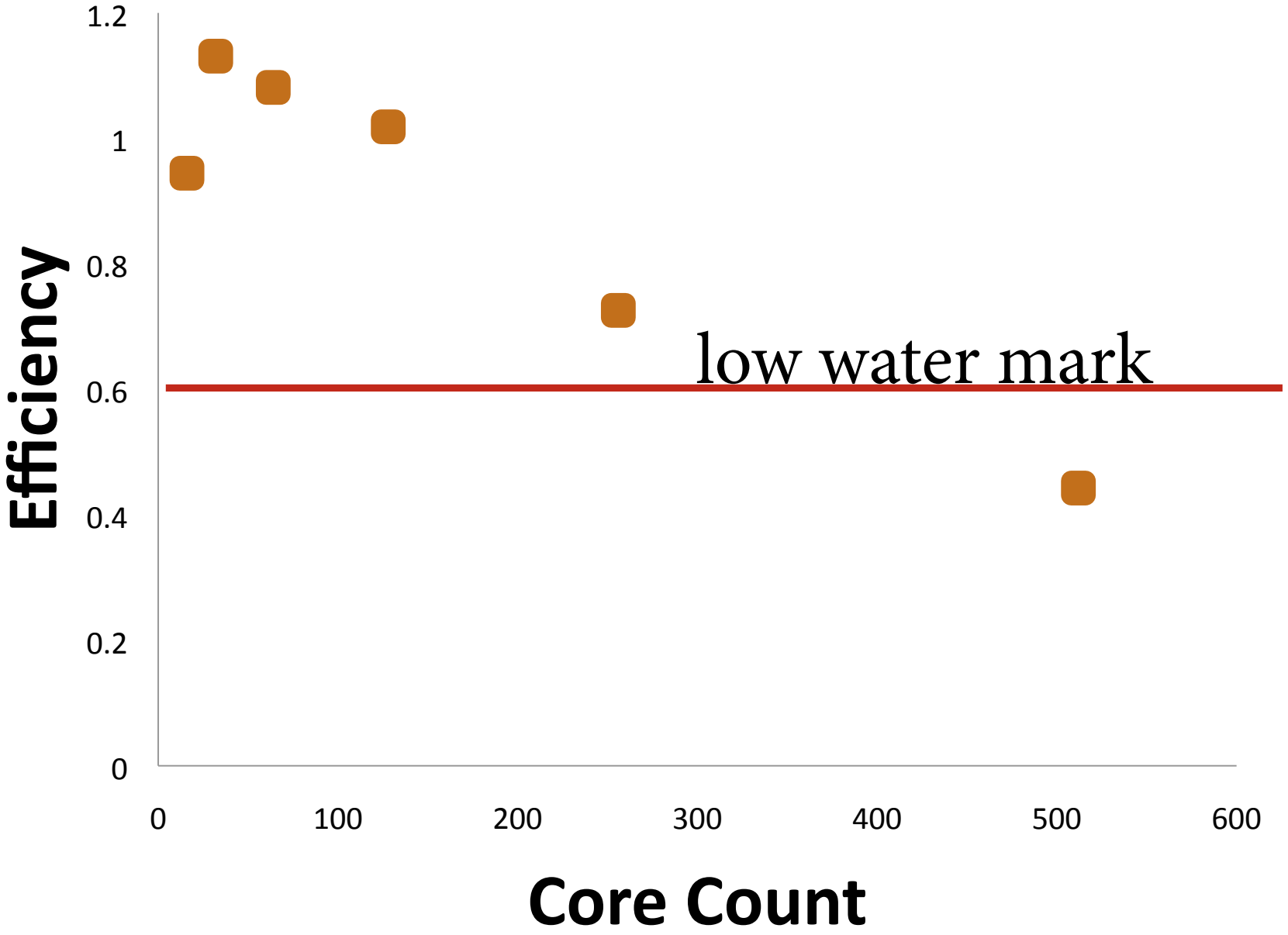
How much an N-way parallel job does

How much an N serial jobs do

Efficiency

$$\frac{\text{Time for 1-way job}}{N}$$

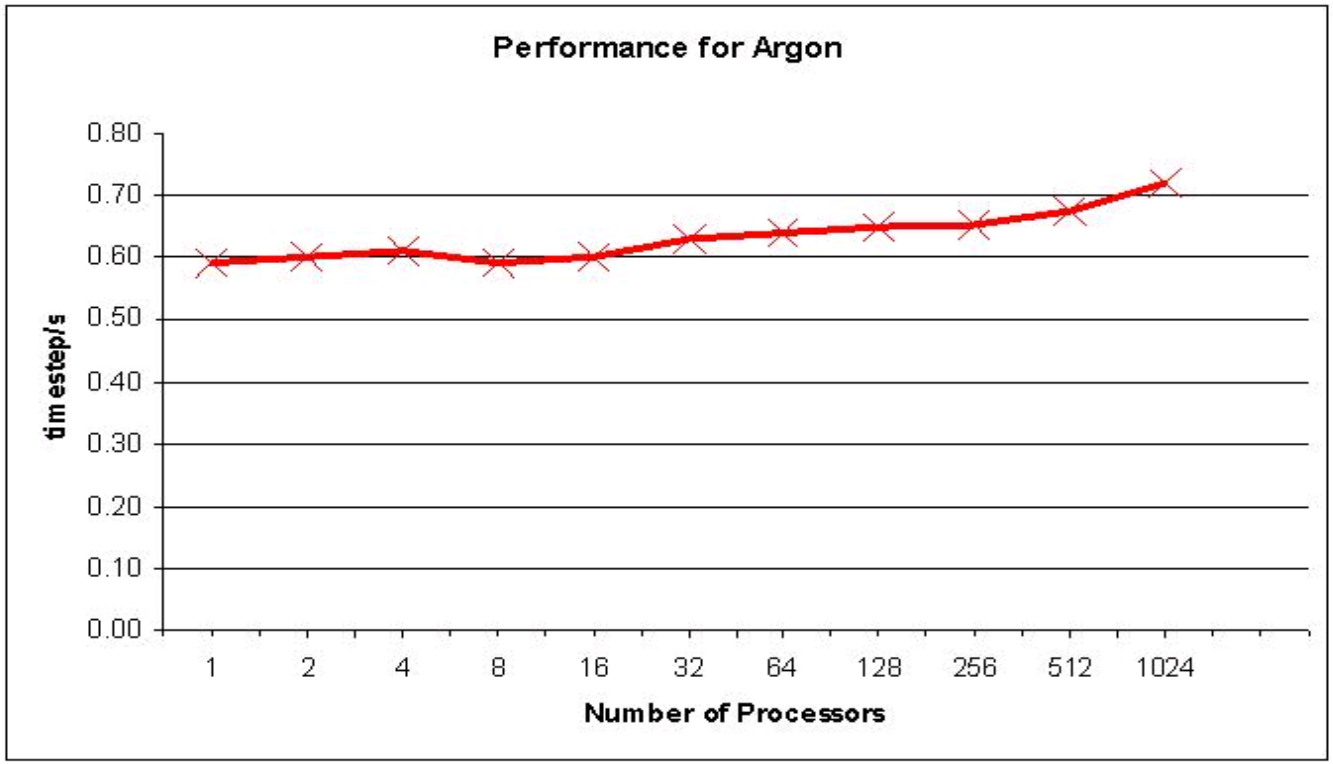
$$\text{Time for N-way job}$$


Efficiency
1.2
1
0.8
0.6
0.4
0.2
0
low water mark
0
100
200
300
400
500
600
Core Count

We expect improvements in scalability with increasing problem size.

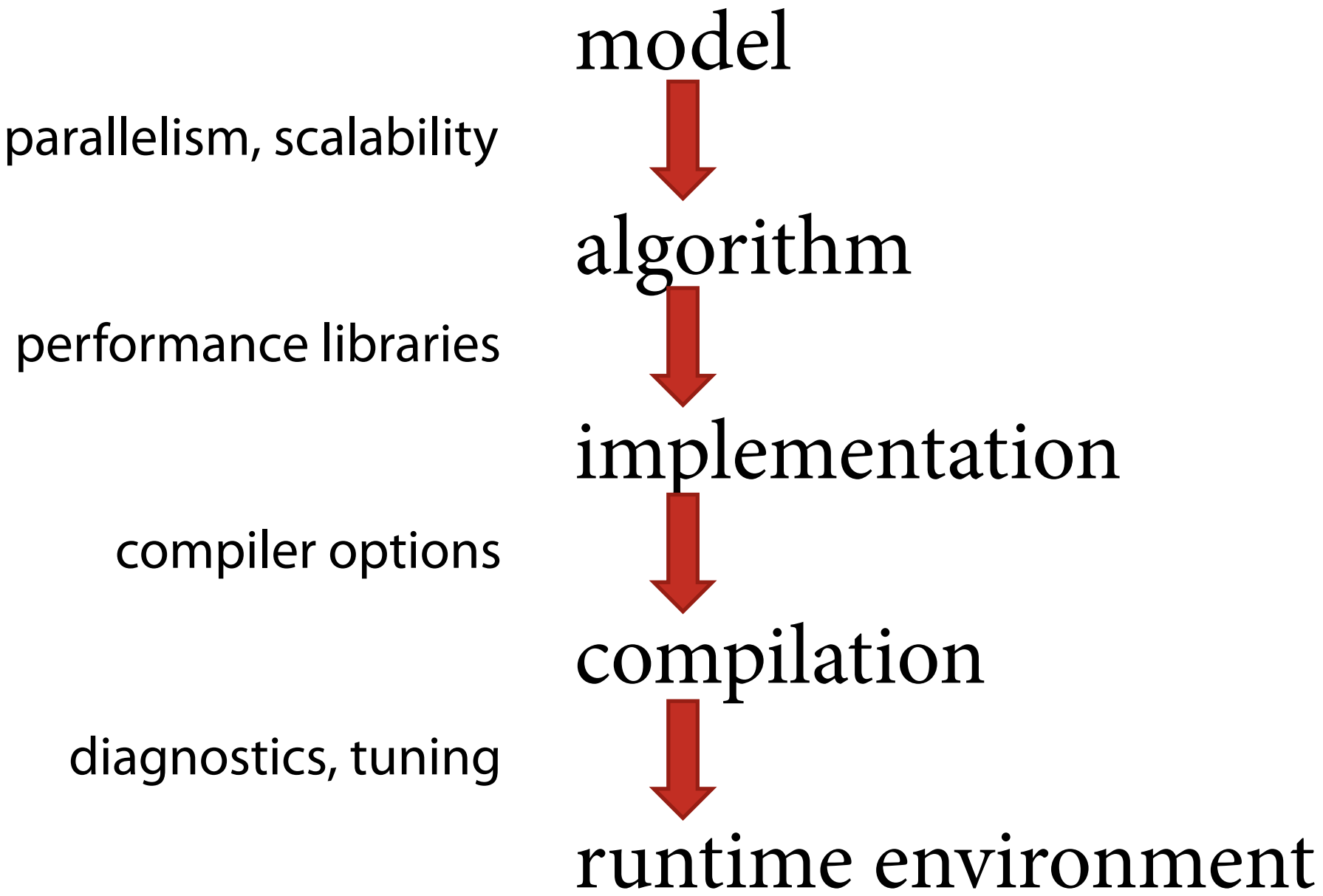
Communication *versus* **Computation**

Weak Scaling



SU ≈ cpu-hour

1.2 Allocation MadLib


model
parallelism, scalability
algorithm
performance libraries
implementation
compiler options
compilation
diagnostics, tuning
runtime environment

Libraries

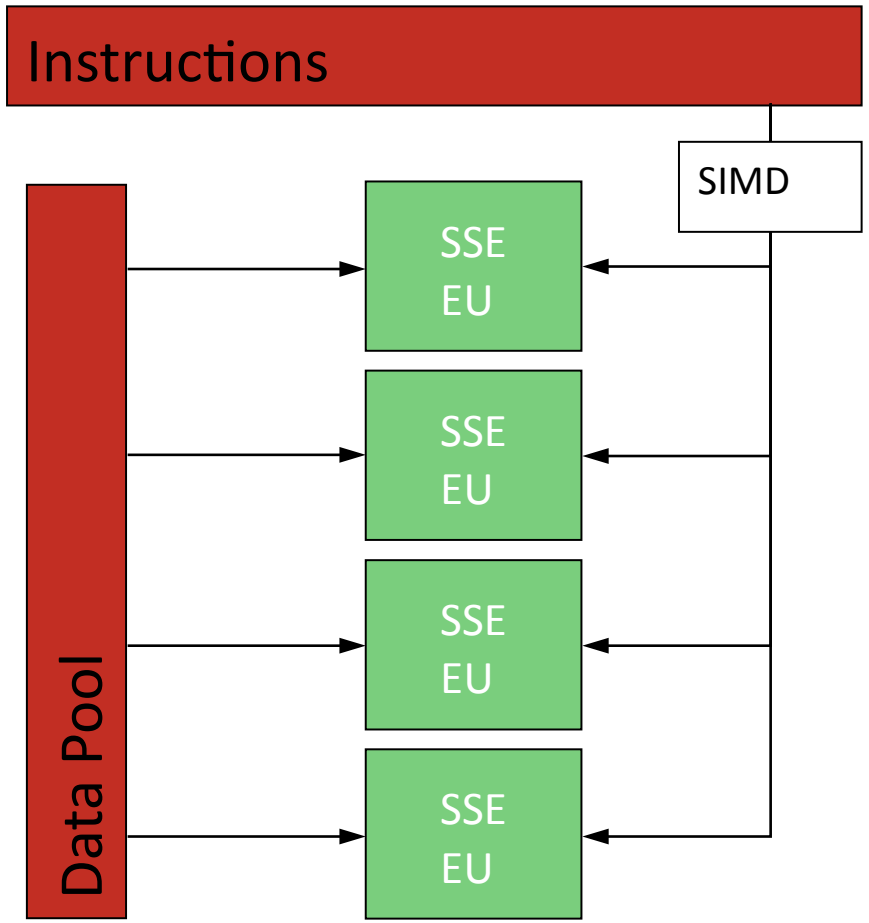
Performance	Math Libs	Method Libs	Applications	I/O
gprof	fftw	petsc	Amber	netcdf
tau	GotoBLAS	scalapack	NAMD	hdf5
papi	Metis/parmetis		charm++	
	MKL 10.0		Gamess	
	Gnu Scientific Library			

Exercise Libraries

2.2 Compare libraries and hand-written code

Use your chip – vectorize.

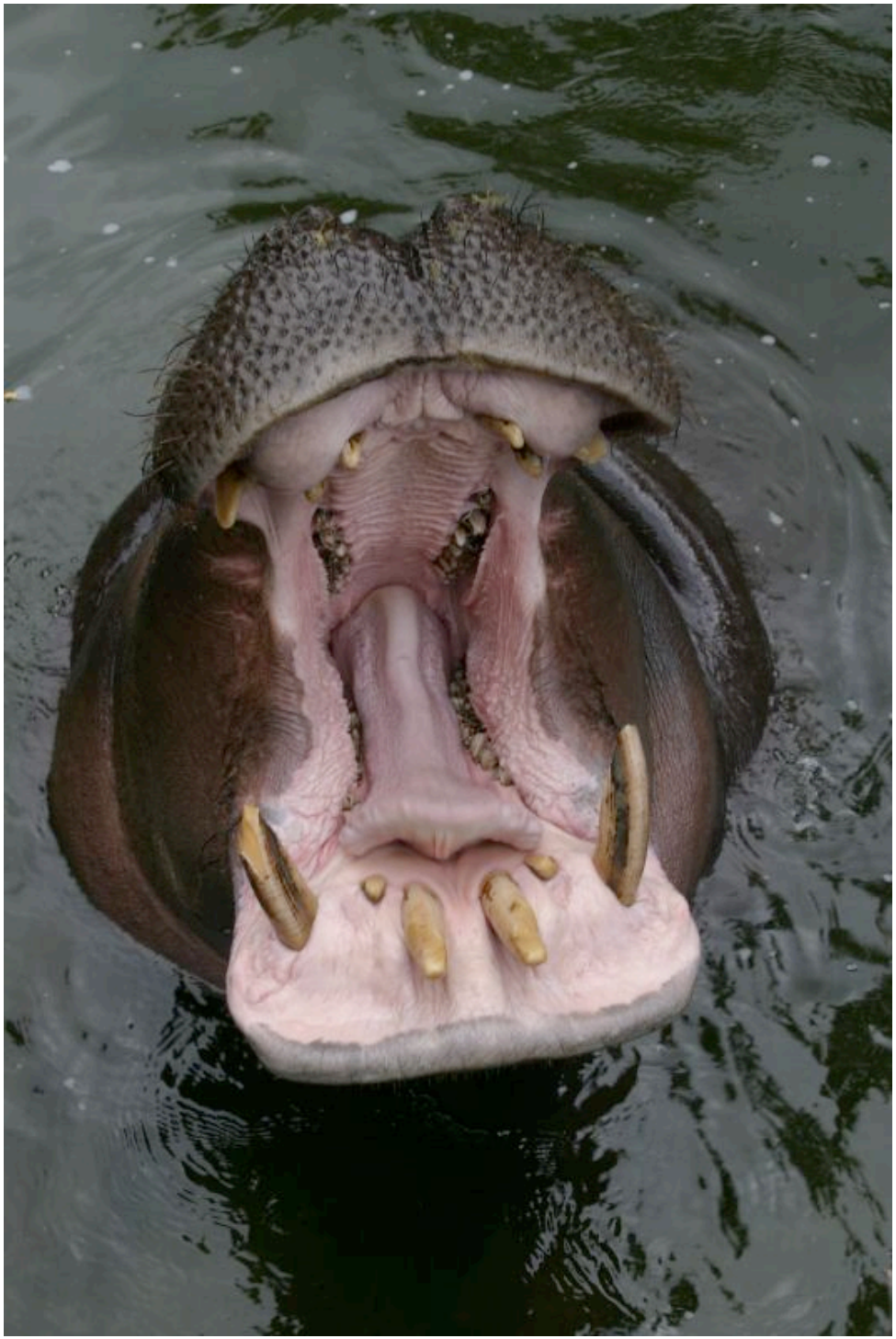
```
for [xIdx in 1:N]
   y[xIdx] = func(xIdx)
```




```
cat /etc/procinfo
```

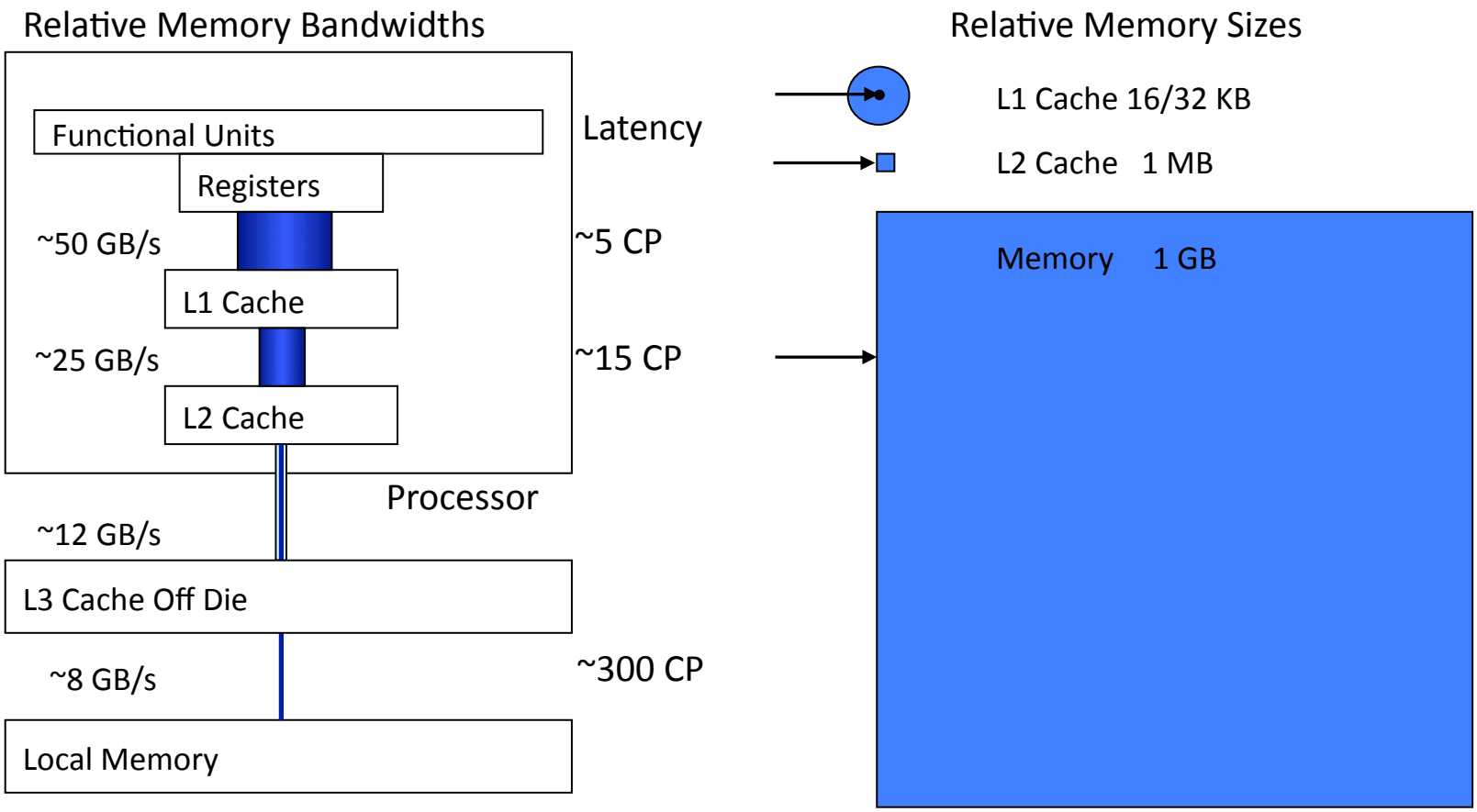
Data

Instructions

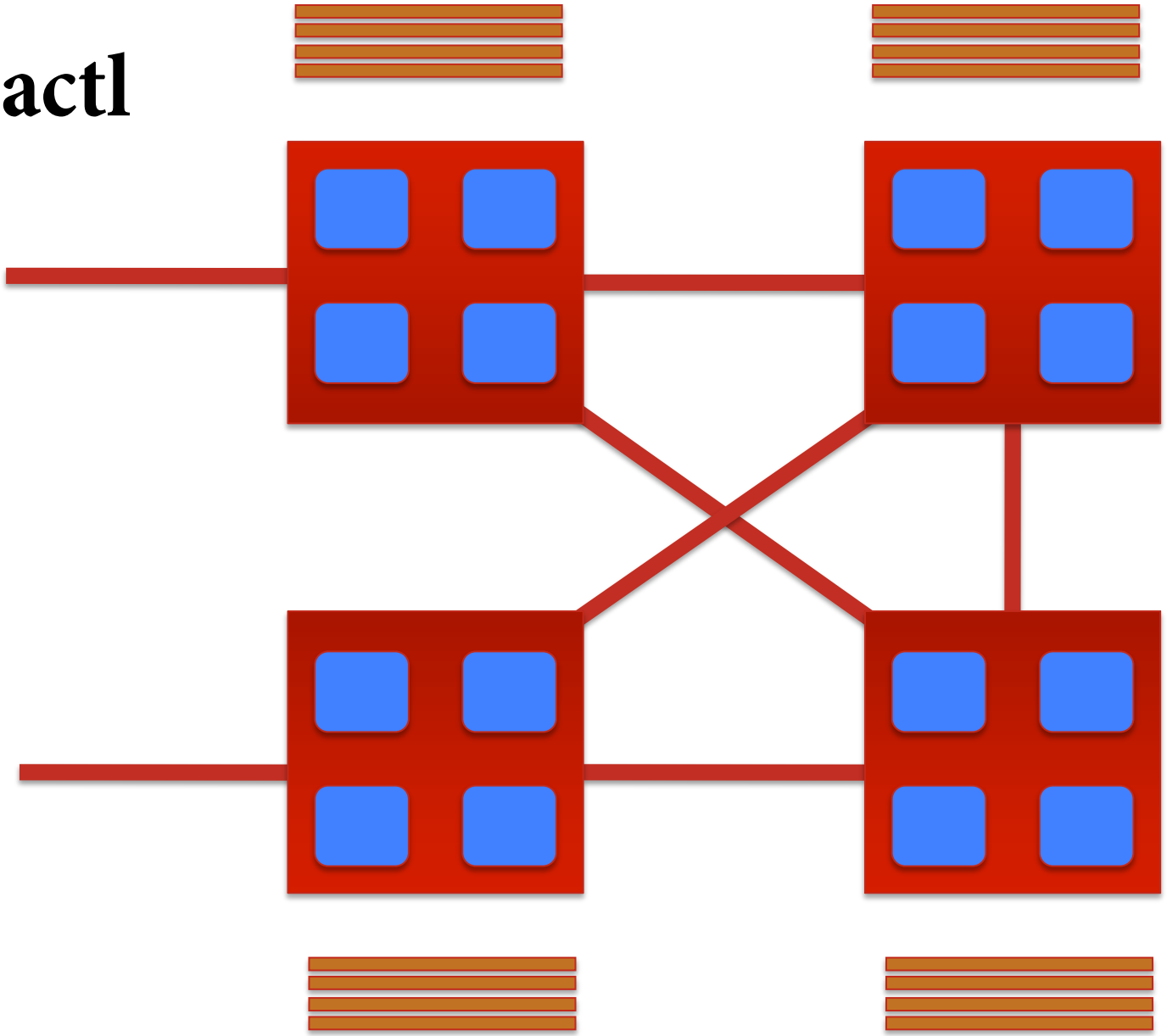


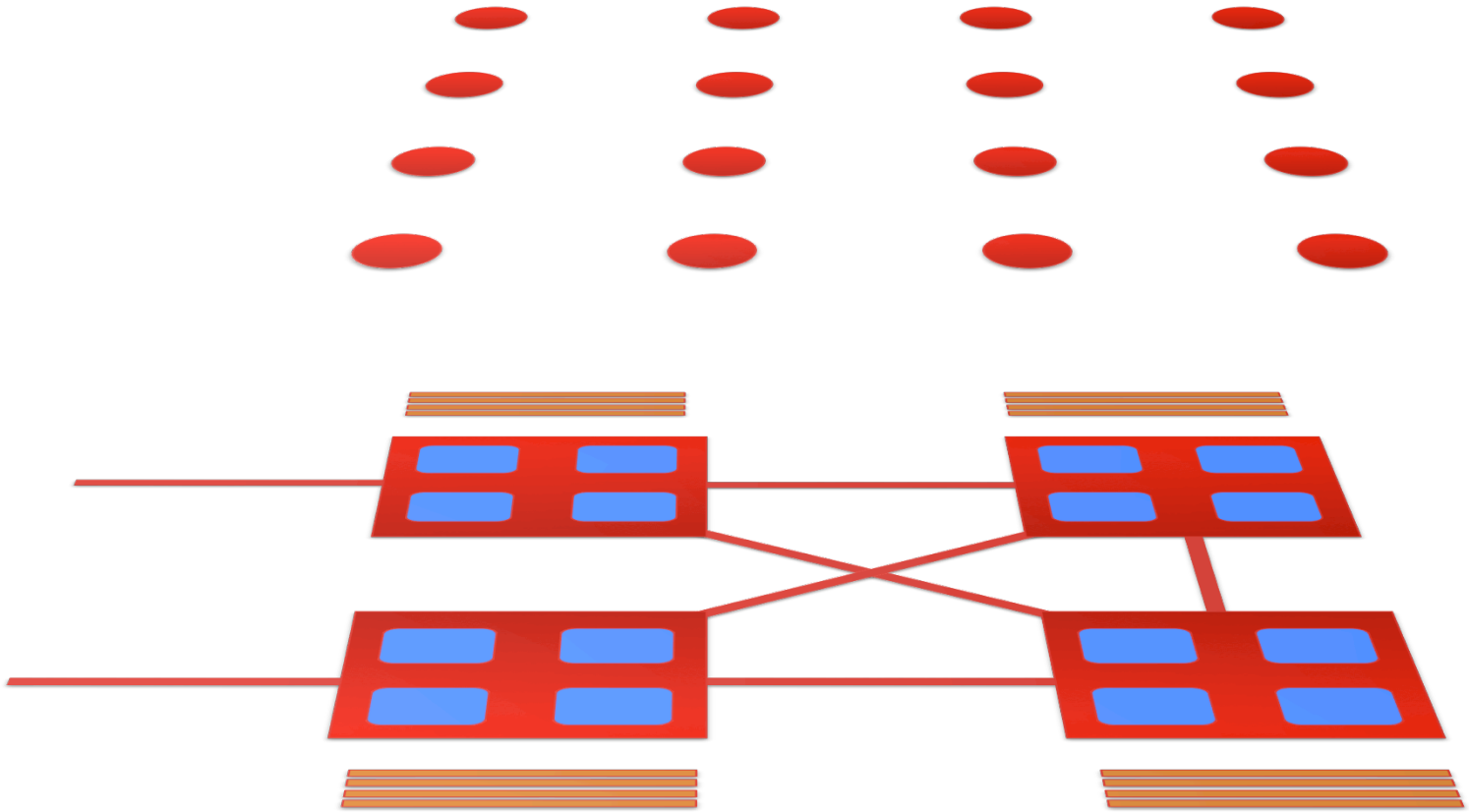
Network

Memory: The Long Pole in the Tent



numactl



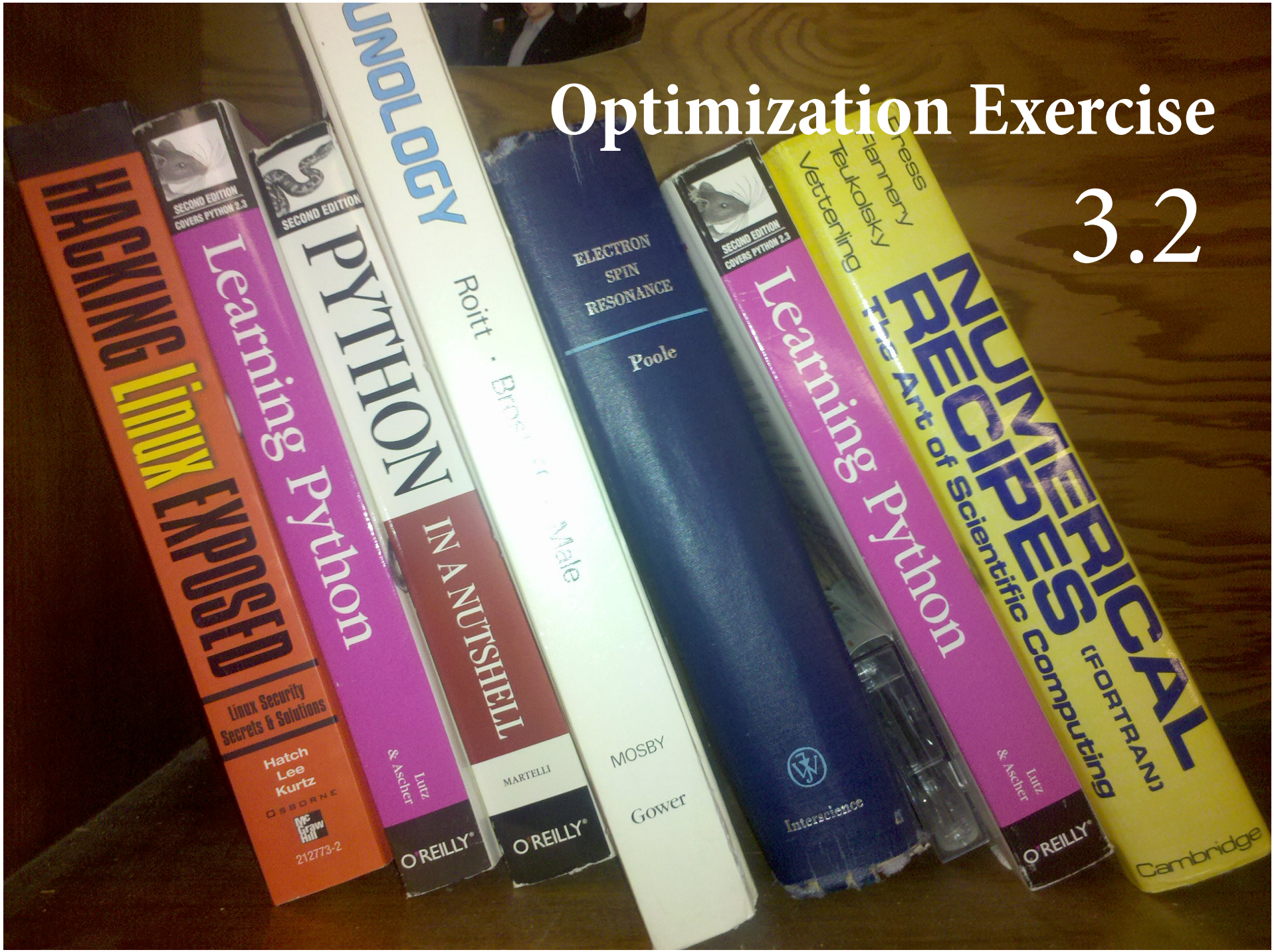


Let the compiler roam.

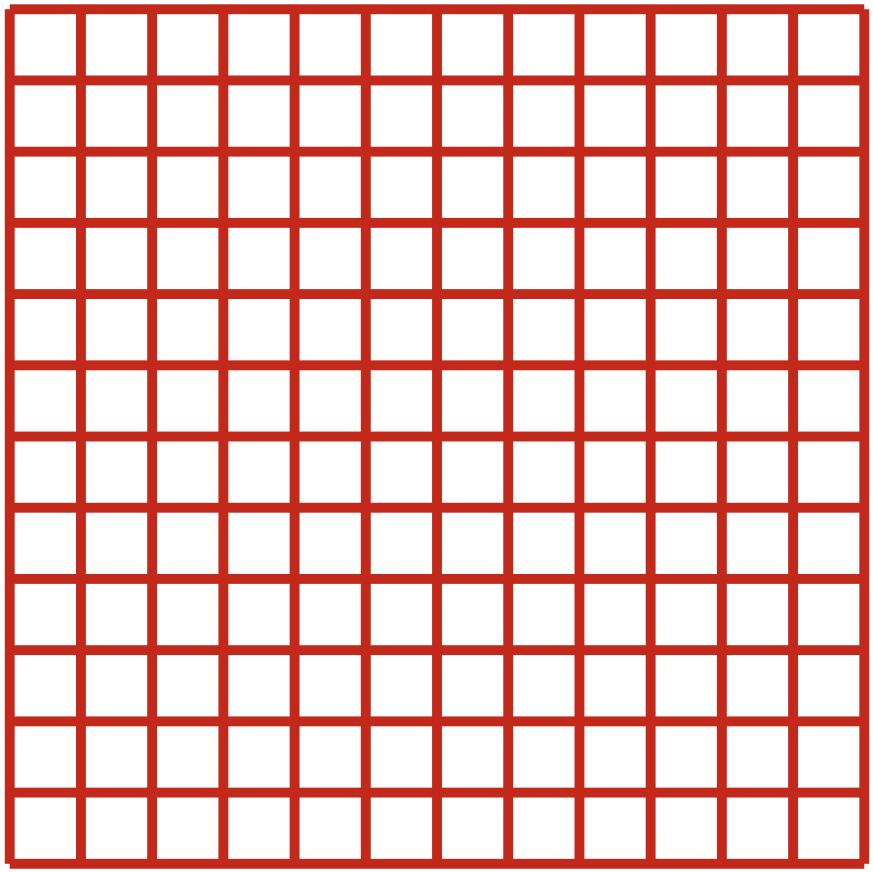
Interprocedural Optimization - IP,IPO

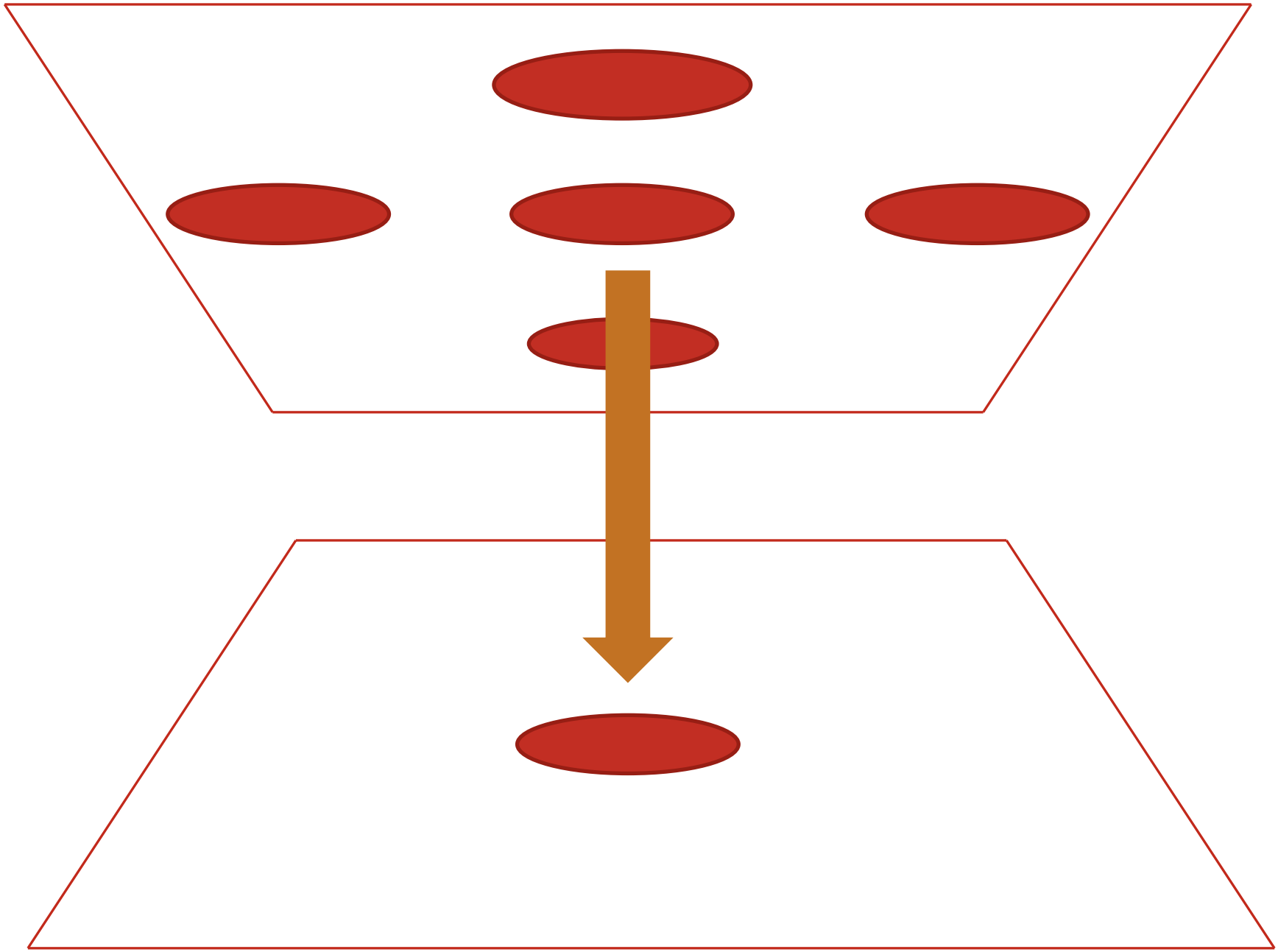
But watch it.

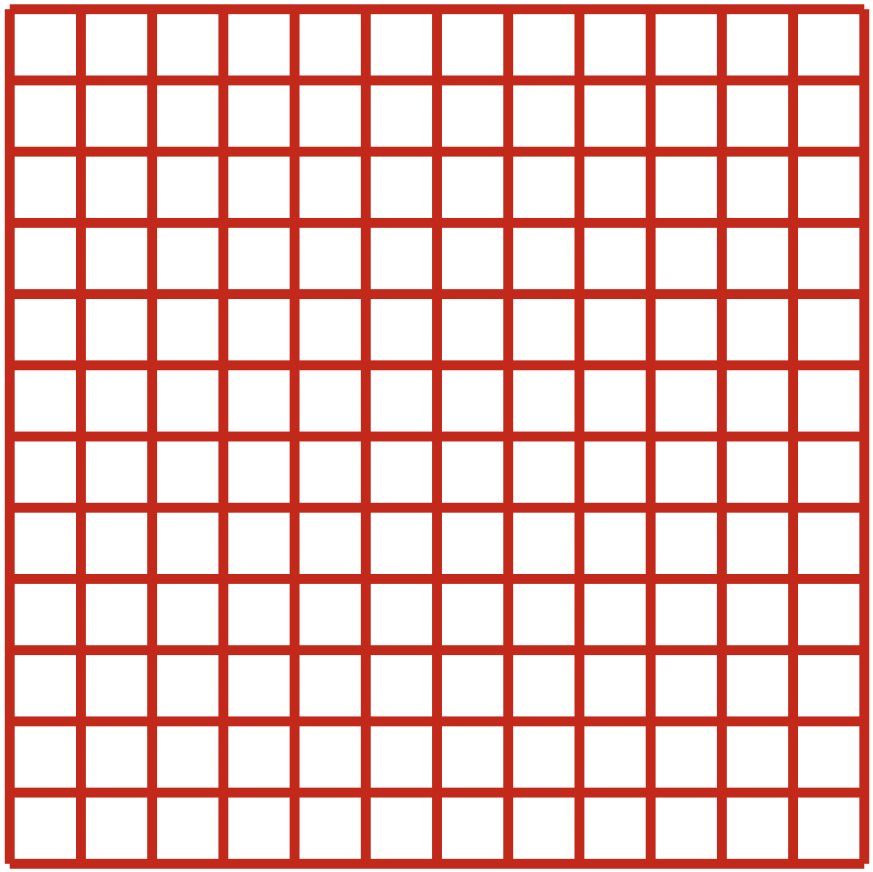
-g -O1 -O2 -O3

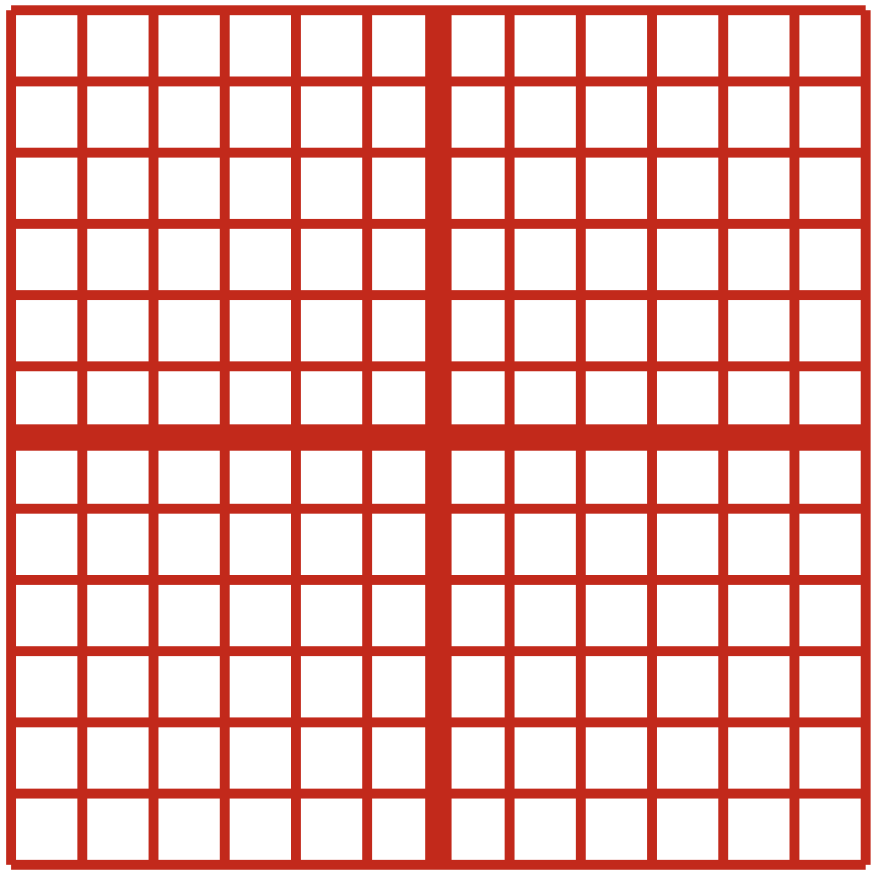

Optimization Exercise
3.2
HACKING Linux EXPOSED
Linux Security Secrets & Solutions
Hatch
Lee
Kurtz
OSBORNE
212773-2
SECOND EDITION
COVERS PYTHON 2.3
Learning Python
Lutz & Ascher
O'REILLY
SECOND EDITION
PYTHON
IN A NUTSHELL
MARTELLI
O'REILLY
UNOLOGY
Roitt
ELECTRON SPIN RESONANCE
Poole
MOSBY
Gower
SECOND EDITION
COVERS PYTHON 2.3
Learning Python
Lutz & Ascher
O'REILLY
Interscience
Press
Flannery
Teukolsky
Vetterling
NUMERICAL RECIPES
The Art of Scientific Computing
(FORTRAN)
Cambridge

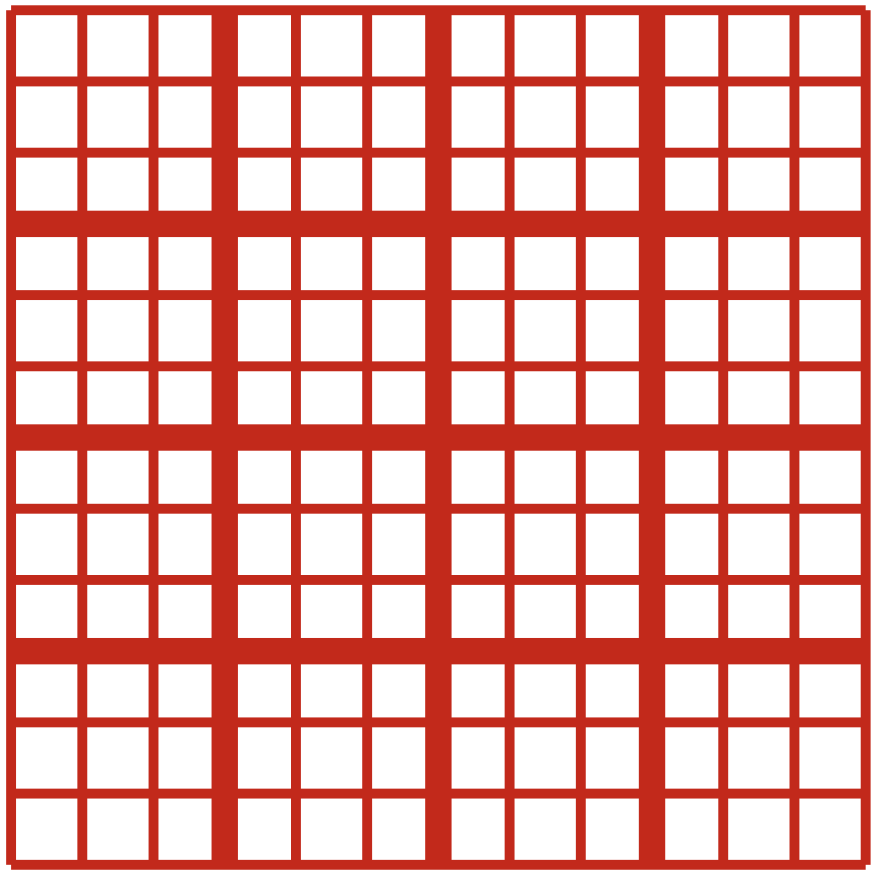
time = _ + _ + _

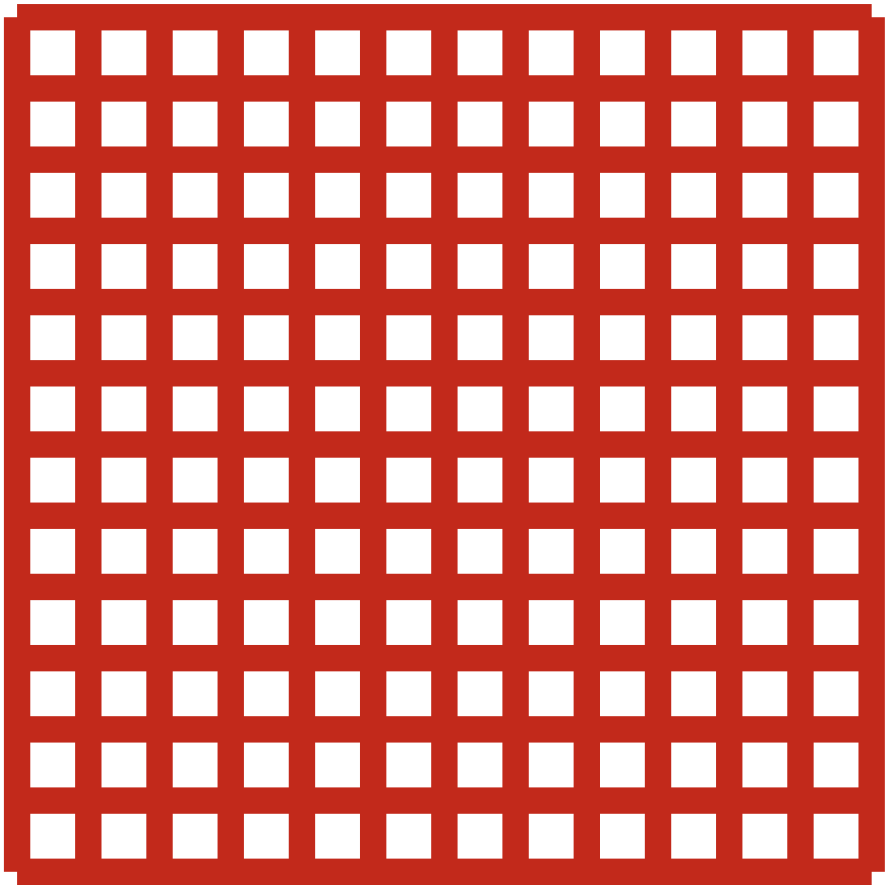


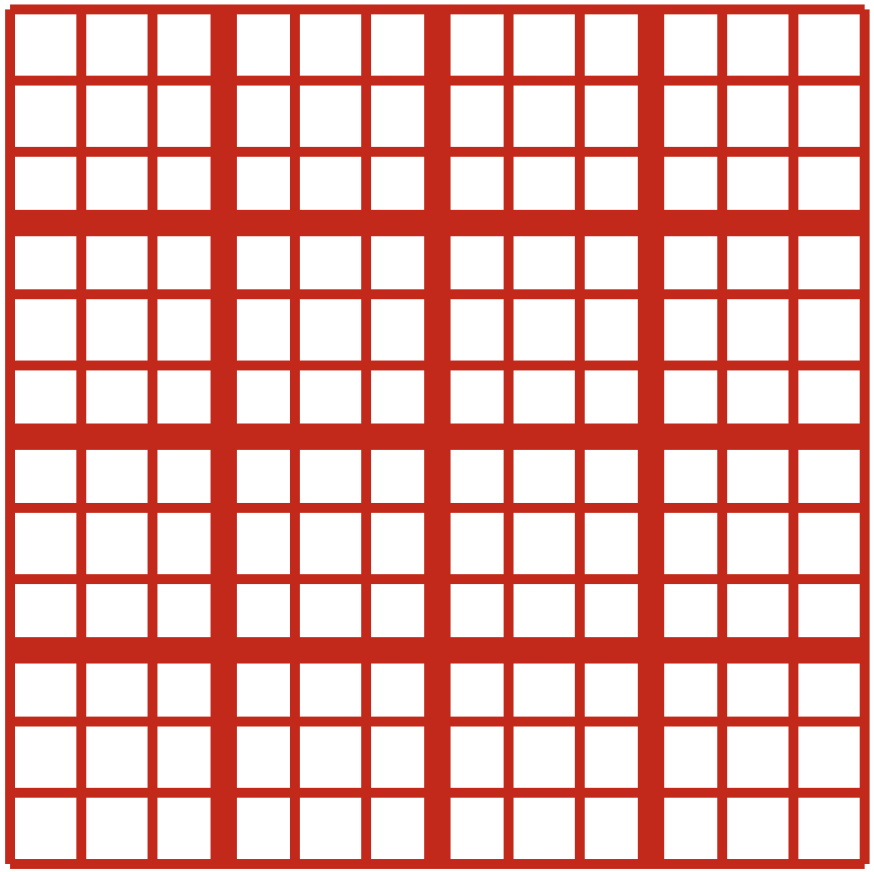






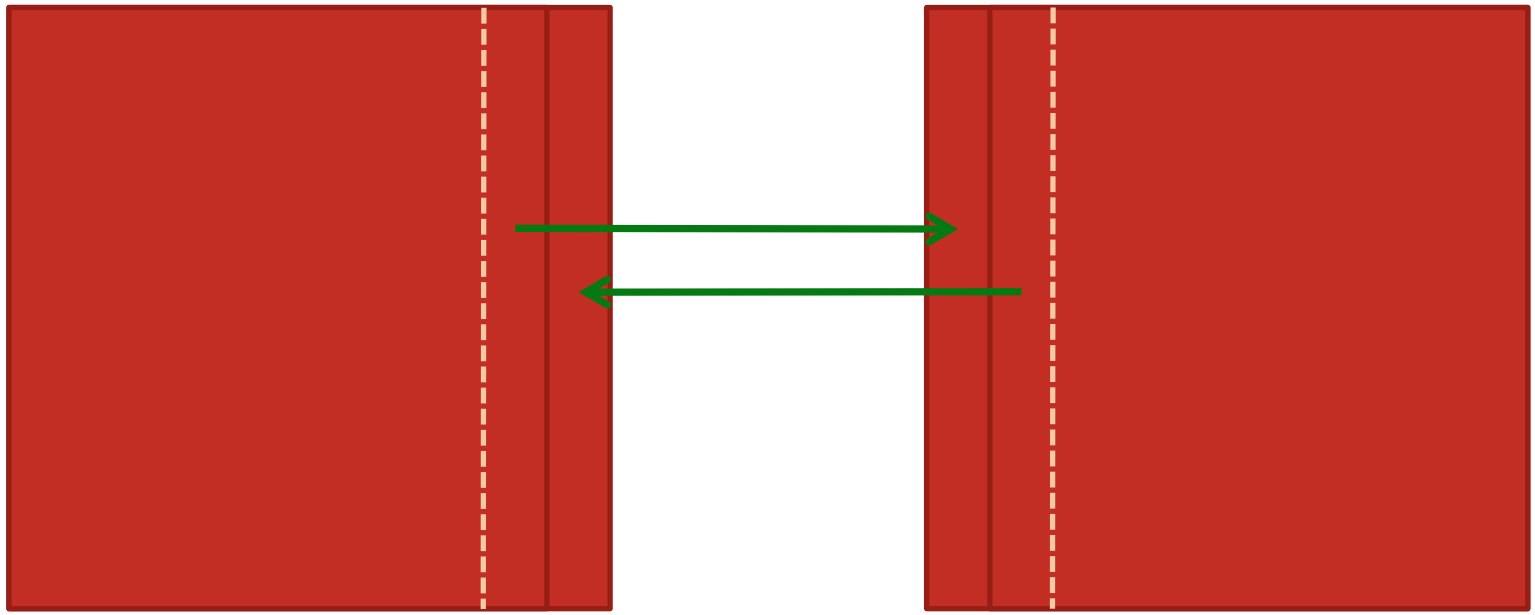




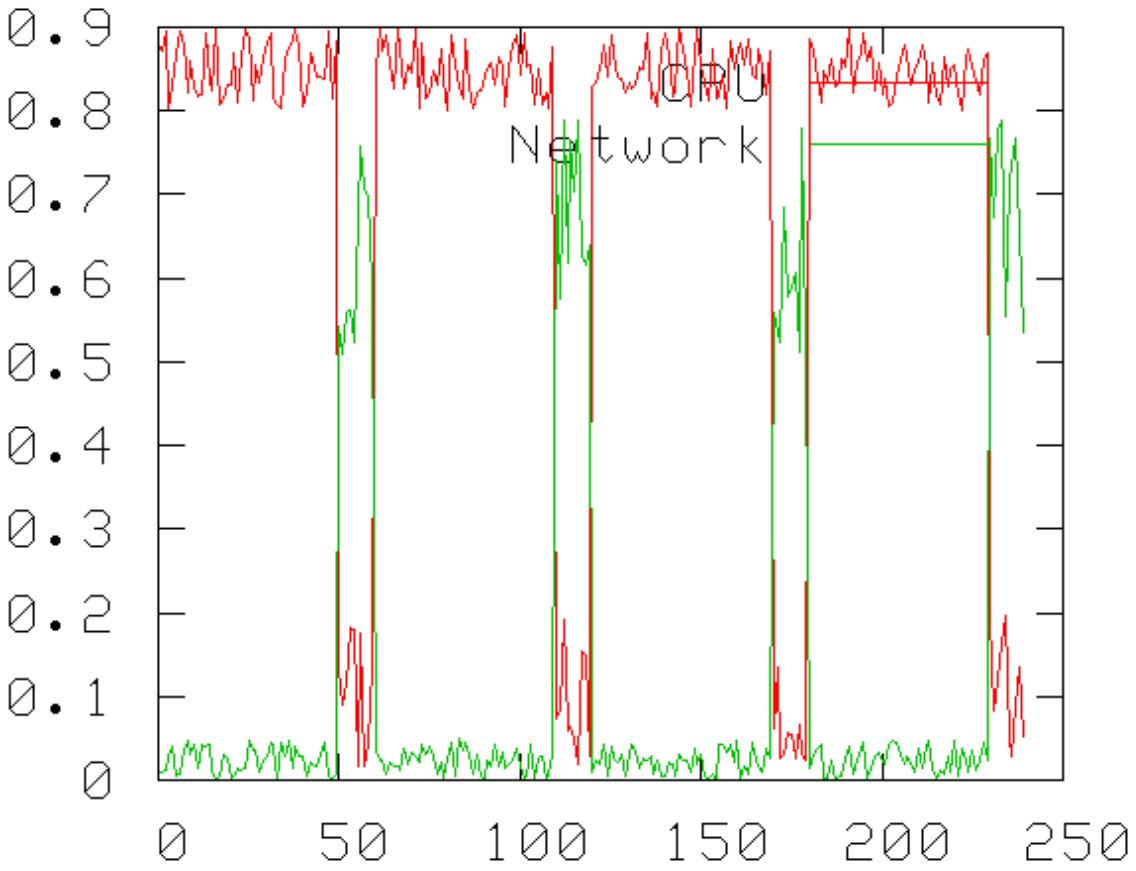


W

W/N



Communication Pattern in Time



Amount sent / bandwidth

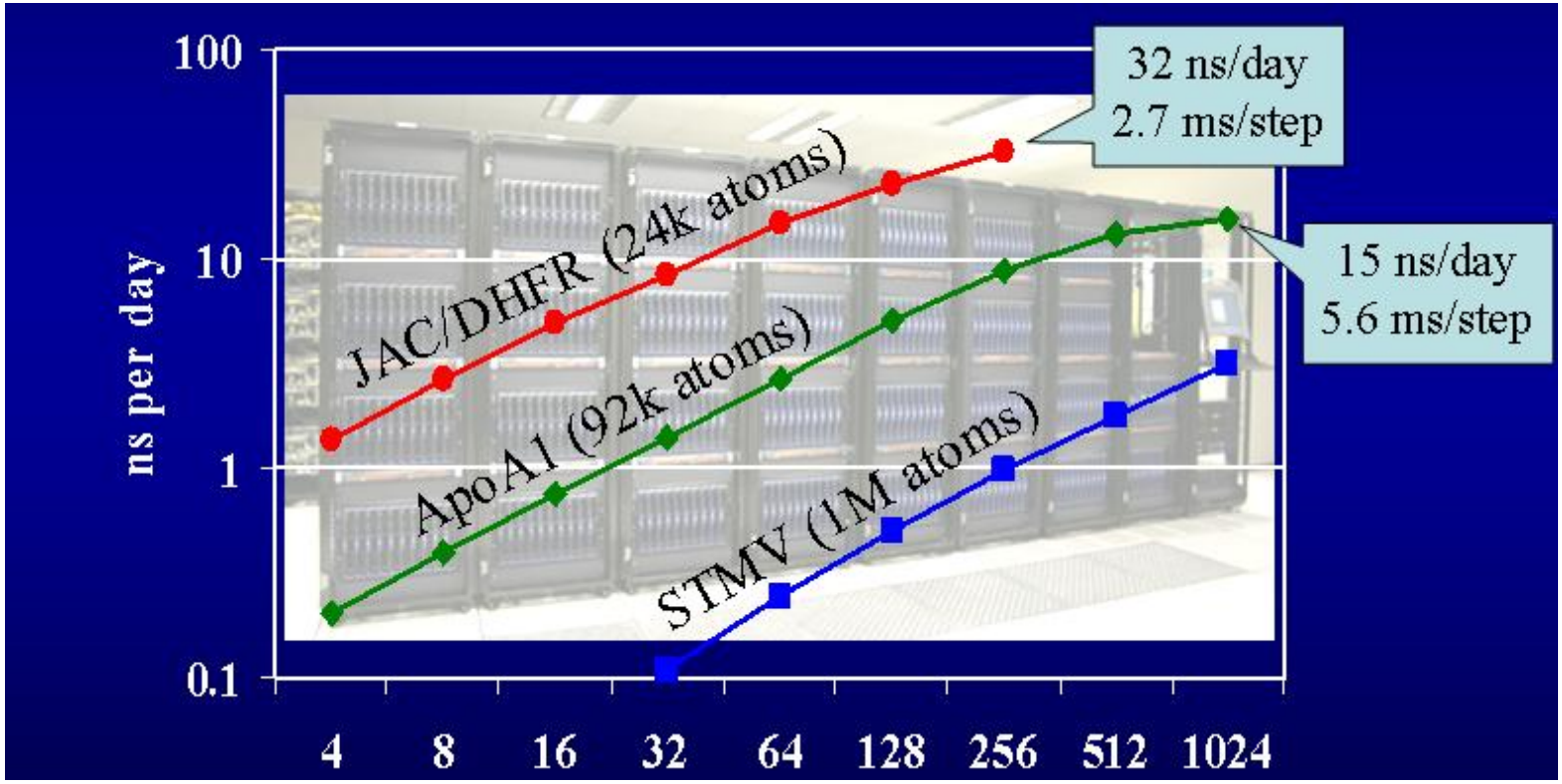
Number of messages × latency

What is the time per iteration?

$$\text{time} = W/N + N/bw + 2 \times \text{latency}$$

$$\mathcal{O}(f(N))$$

It's LOG!



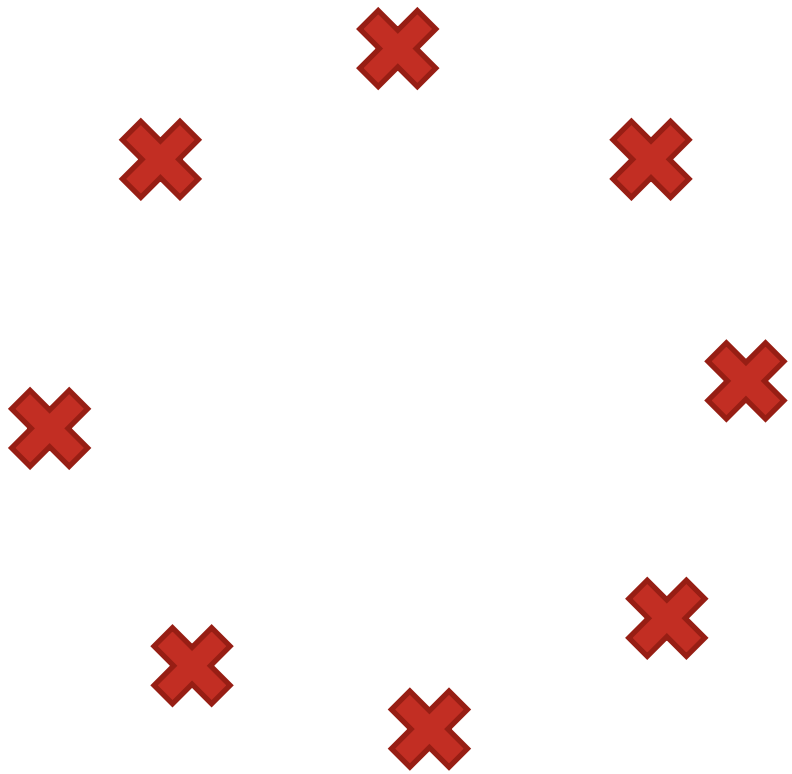
It's LOG!

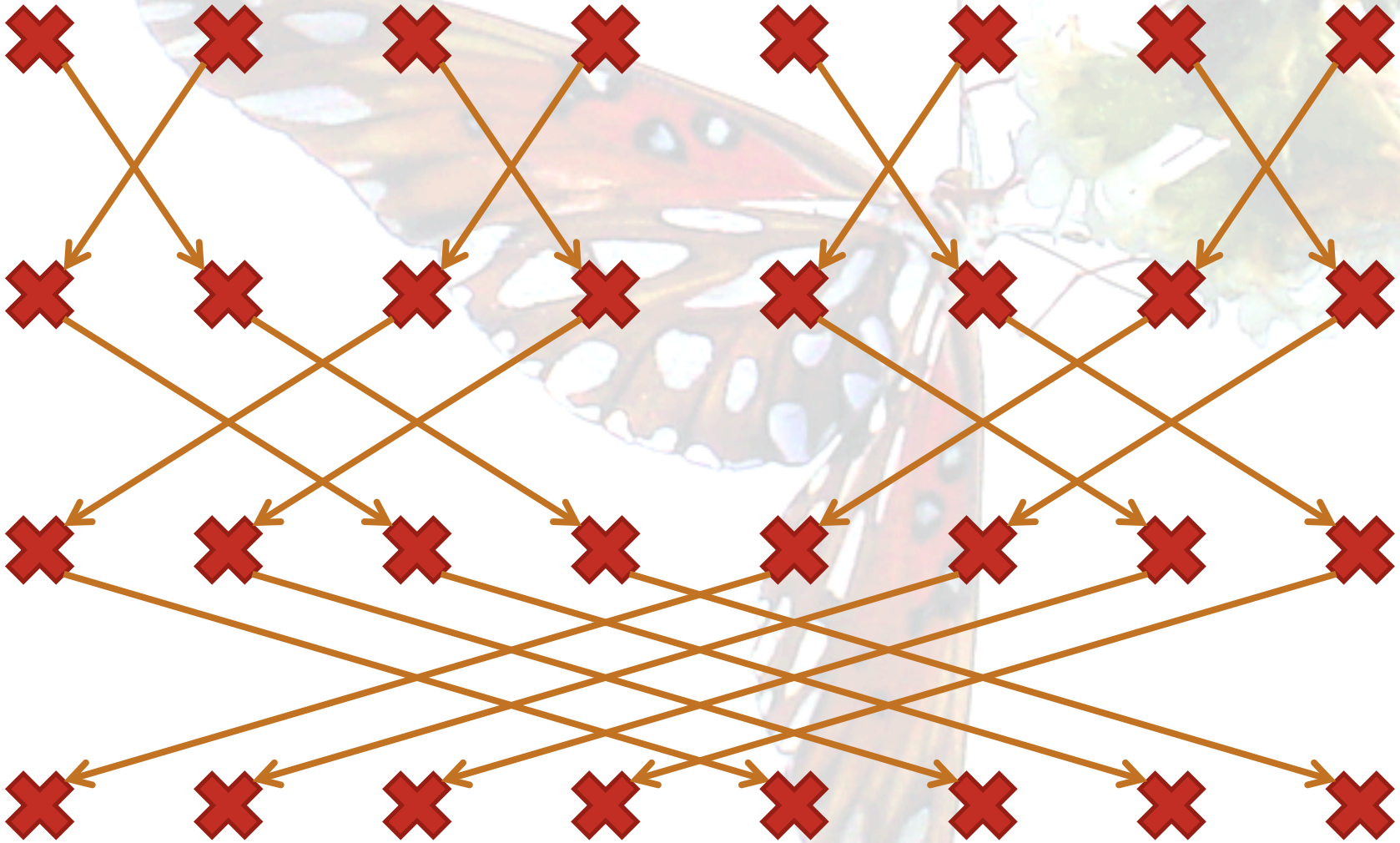
Excel of Spectral Code

Section 4.1 Fluent Model


Sun
microsystems
Sun
microsystems
Sun
microsystems
Sun
microsystems
2.3 μs

All-to-All Communication



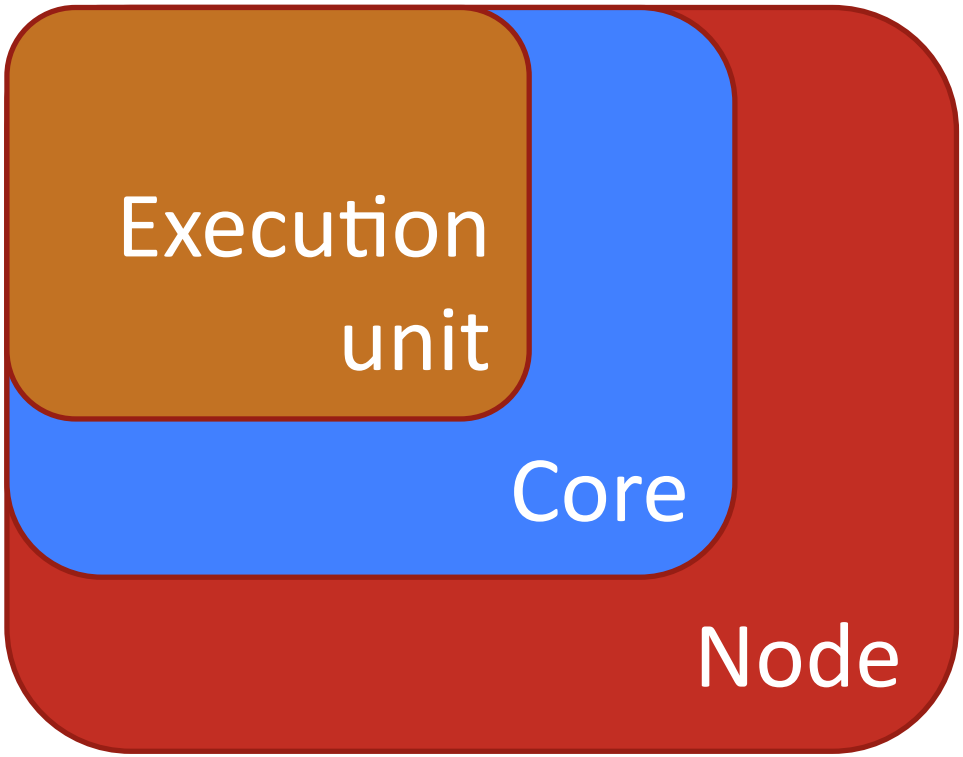


Right Resources

High Performance Systems									
Name	Institution	System	*Peak TFlops*	Memory TBytes	Status	Load	Running Jobs	Queued Jobs	Other Jobs
Kraken	NICS	Cray XT5	608.00	129.00	Up		24	5	3
Ranger	TACC	Sun Constellation	579.40	123.00	Up		297	406	100
Abe	NCSA	Dell Intel 64 Linux Cluster	89.47	9.38	Up*		194	170	136
Lonestar	TACC	Dell PowerEdge Linux Cluster	62.16	11.60	Up		40	90	1
Steele	Purdue	Dell Intel 64 Linux Cluster	60.00	12.40	Up		813	189	25
Queen Bee	LONI	Dell Intel 64 Linux Cluster	50.70	5.31	Up		119	5	1
Lincoln	NCSA	Dell/Intel PowerEdge 1950	47.50	3.00	Up		1	0	0
Big Red	IU	IBM e1350	30.60	6.00	Up*		611	903	43
BigBen	PSC	Cray XT3	21.50	4.04	Up		13	56	48
TeraGrid Cluster	NCSA	IBM Itanium2 Cluster	10.23	4.47	Up		45	4	0
Cobalt	NCSA	SGI Altix	6.55	3.00	Up		63	473	40
Frost	NCAR	IBM BlueGene/L	5.73	0.51	Up		8	0	10
Pople	PSC	SGI Altix 4700	5.00	1.54	Up		38	0	16
TeraGrid Cluster	SDSC	IBM Itanium2 Cluster	3.10	1.02	Up*		42	6	0
TeraGrid Cluster	UC/ANL	IBM Itanium2 Cluster	0.61	0.24	Up		1	0	0
NSTG	ORNL	IBM IA-32 Cluster	0.34	0.07	Up		1	0	0
		Total:	**1580.89**	**314.58**			**2310**	**2307**	**423**

Profiling and Presents

Speeds and Feeds at Scales



Your Model

Unhalted computation.
Data movement.
Data transfer.

http://www.flickr.com/photos/35188692@N00/83775147/

http://www.flickr.com/photos/avdleeuw/48388892/