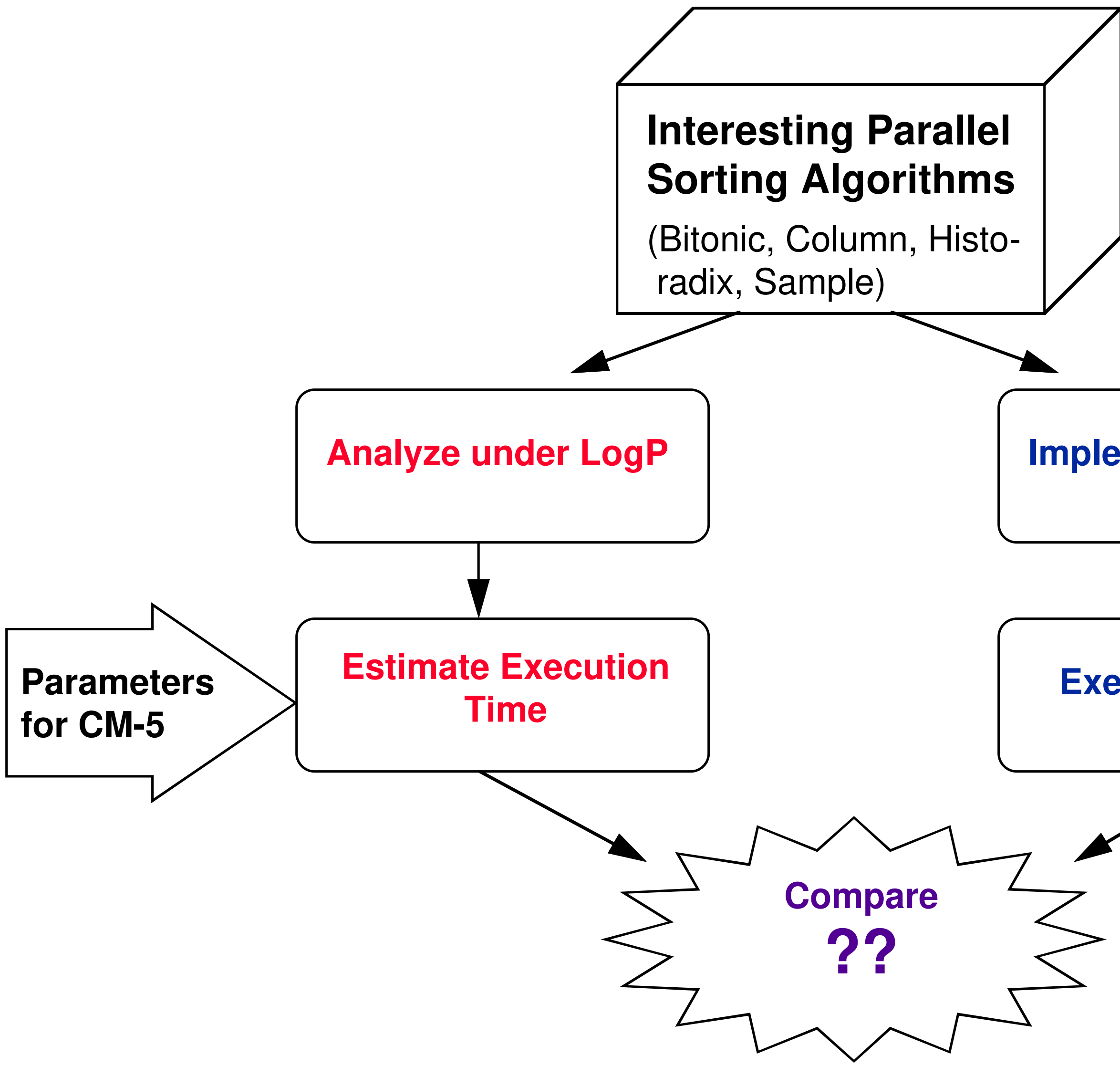
Implementation and Modeling of Parallel Sorts on Current and Future Machines

David E. Culler
(Andrea Dusseau, Klaus Schauser, Rich Martin)
Computer Science Division
University of California, Berkeley
culler@cs.berkeley.edu

DIMACS 94 Parallel Implementation Challenge

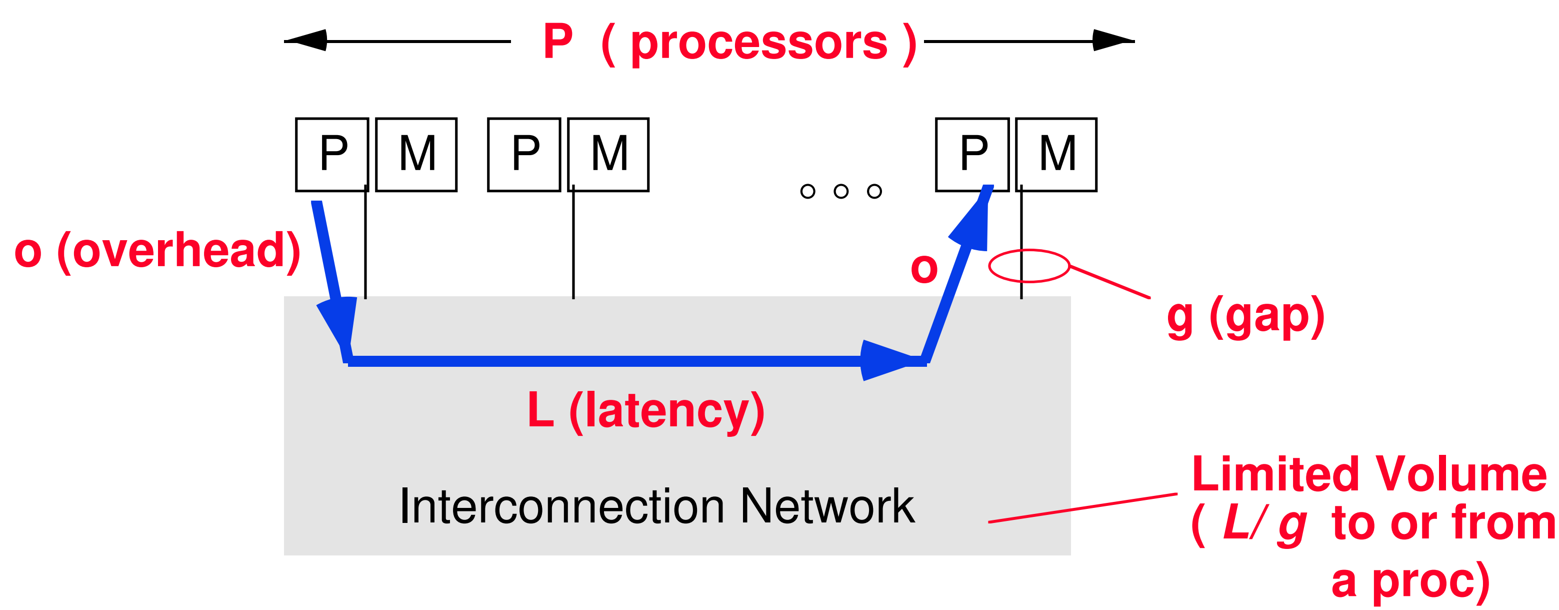
© Culler 1994

The Study



© Culler 1994

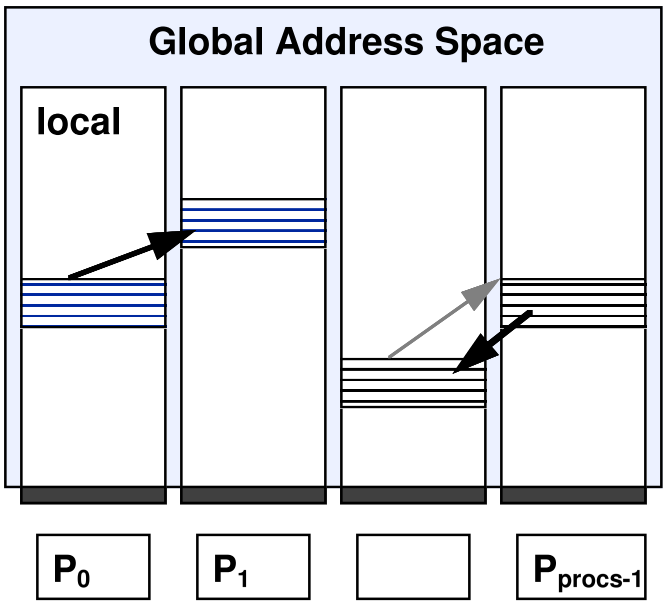
LogP



- Latency in sending a (small) mesage between modules
- overhead felt by the processor on sending or receiving msg
- gap between successive sends or receives (1/BW)
- Processors

© Culler 1994

Split-C



- Explicitly parallel C
- 2D global address space
  - linear ordering on local spaces
- Local and Global pointers
  - spread arrays too
- Read/Write
- Get/Put (overap compute and comm)

  x := G; . . .

  sync();
- Signaling store (one-way)

  G :– x; . . .

  store_sync(); or all_store_sync();
- Bulk transfer
- Global comm.

© Culler 1994

Basic Costs

• Read, Write	x = *G, *G = x	2 (L + 2o)
• Store	*G :– x	L + 2o
• Get	x := *G	o
	....	2L + 2o
	sync();	o

– with interval g

- Bulk store (N/ω comm. words) — 2o + (n-1)g + L
- Exchange — $2o + 2L + (\lceil n/\omega \rceil - 1 - L/g)\ \max(g, 2o)$

- One to many
- Many to one

© Culler 1994

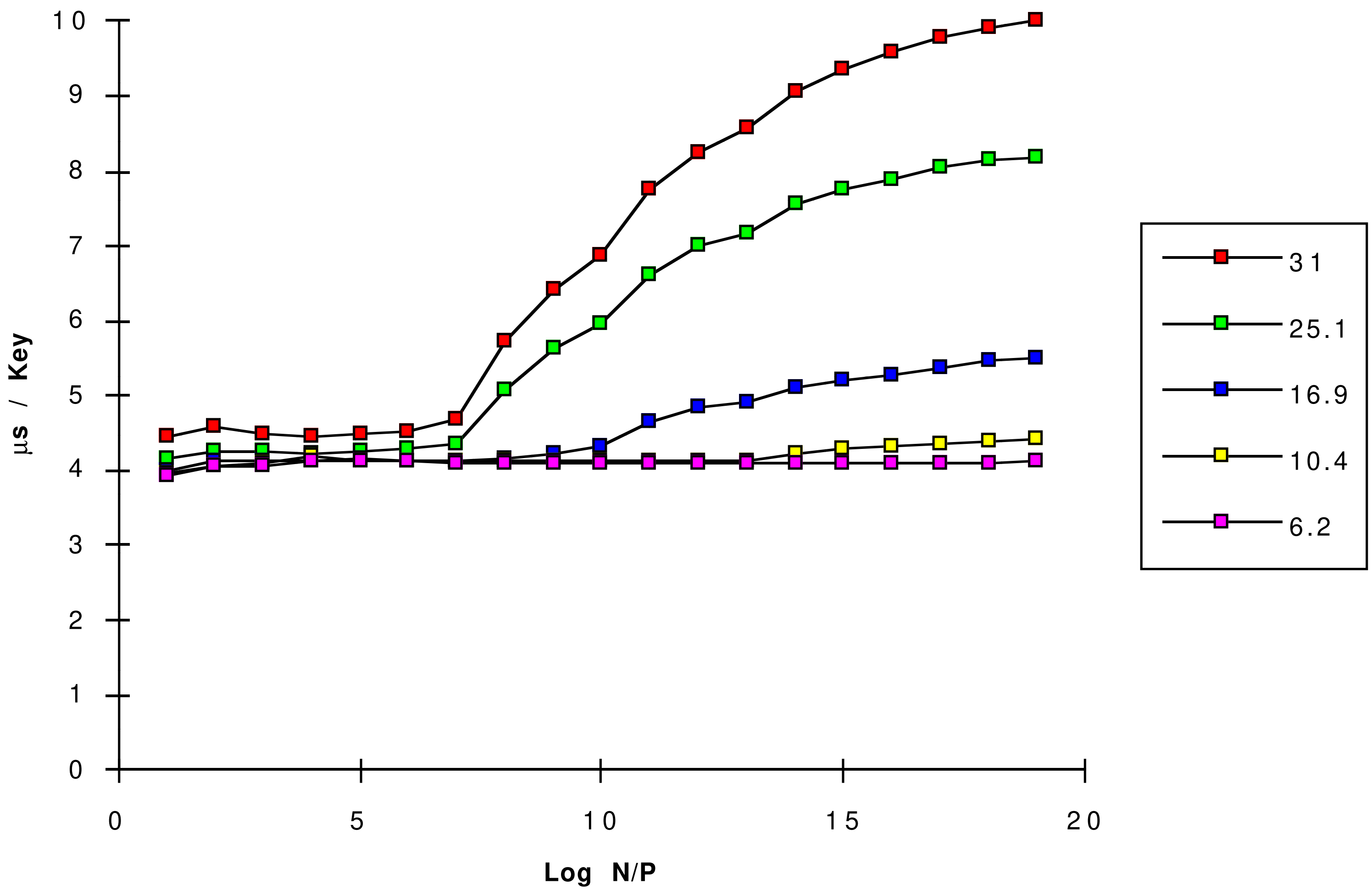
LogP model of the CM-5

- $L = 6\ \mu s$
- $o = 2.2\ \mu s$
- $g = 4\ \mu s$
- P varies from 32 to 1024

- What is the processor performance?

© Culler 1994

Local Sort Performance

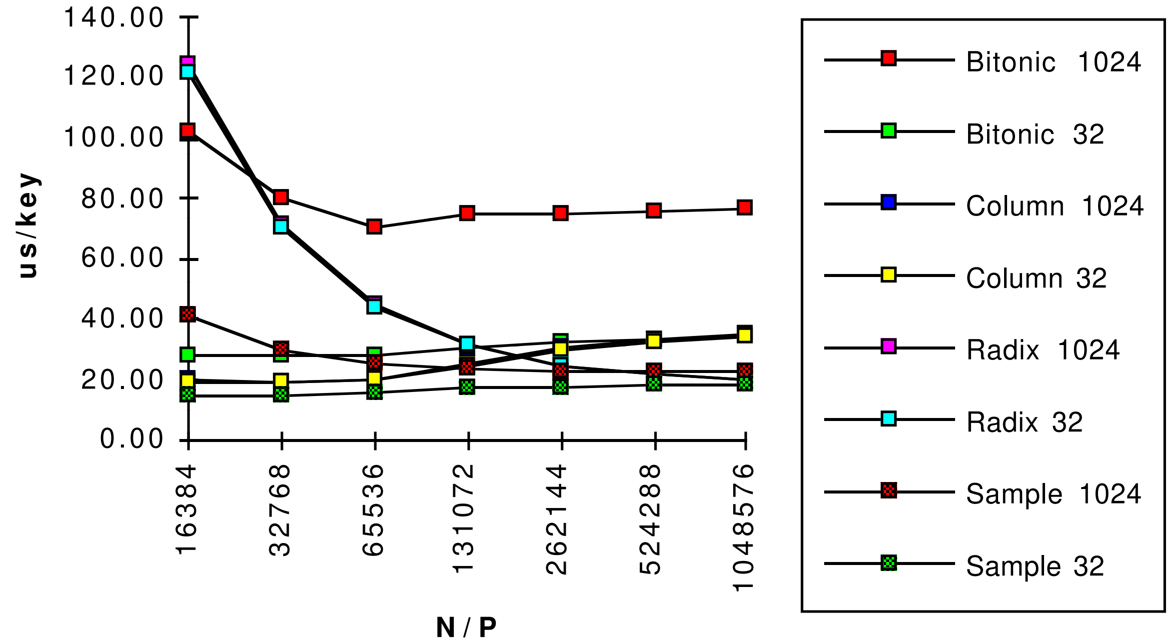


© Culler 1994

Local Computation Parameters

Parameter	Operation	μs per key	Sort
Swap	Simulate cycle butterfly per key	0.025 lg N	Bitonic
mergesort	Sort bitonic sequence	1.0	
scatter	Move key for Cyclic-to-block	0.46	
gather	Move key for Block-to-cyclic	0.52 if $n \leq 64k$ or $P \leq 64$	Bitonic & Column
		1.1 otherwise	
local sort	Local radix sort (11 bit)	4.5 if $n < 64K$	
		9.0 - (281000/n)	
merge	Merge sorted lists	1.5	Column
copy	Shift Key	0.5	
zero	Clear histogram bin	0.2	Radix
hist	produce histogram	1.2	
add	produce scan value	1.0	
bsum	adjust scan of bins	2.5	
address	determine desitination	4.7	
compare	compare key to splitter	0.9	Sample
localsort8	local radix sort of samples	5.0	

Bottom Line (sort of)

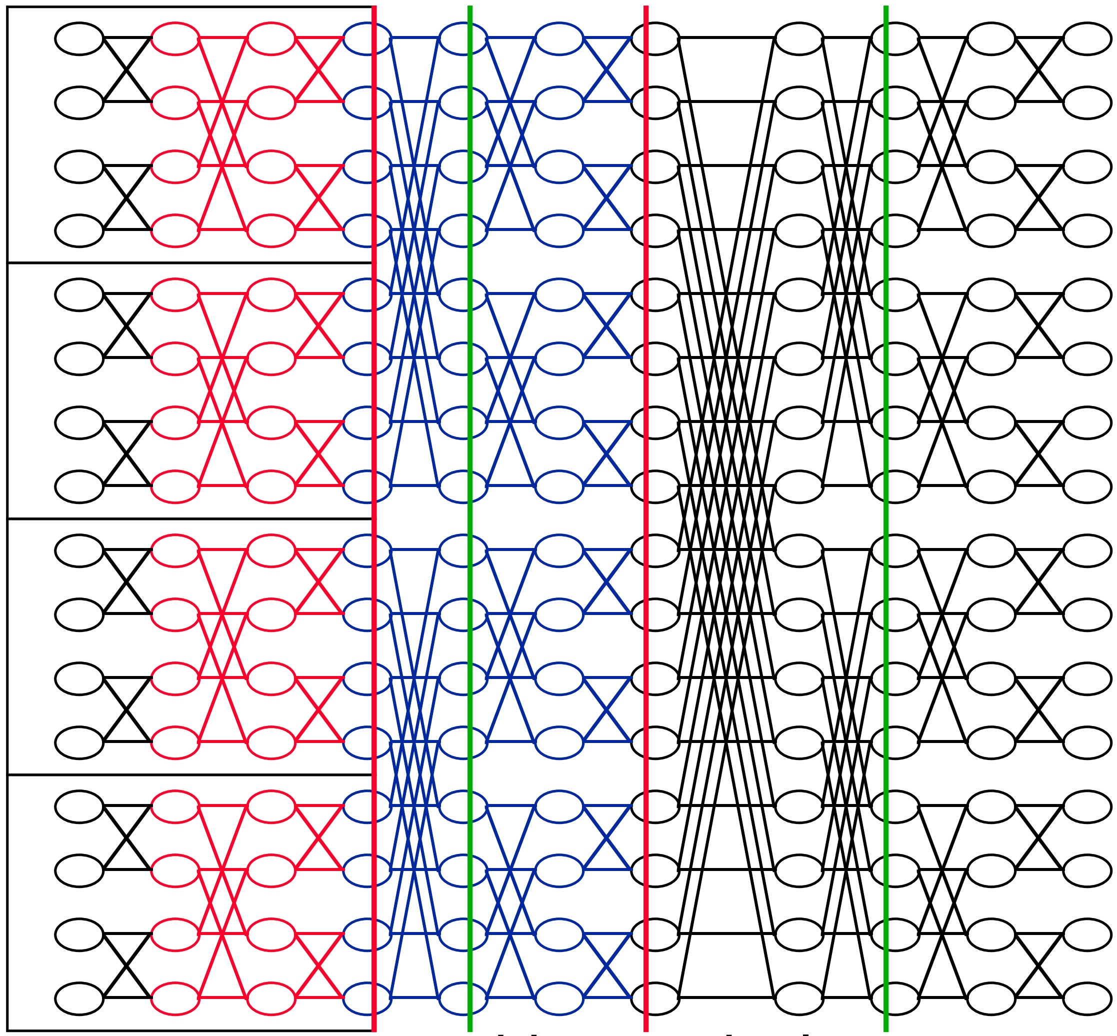


- Good fit between predicted and measured (10%)
- Different sorts for different sorts
  – scaling by processor, input size, sensitivity
- All are global / local hybrids
  – the local part is hard to implement and model

© Culler 1994

Bitonic Sort

Block Layout



lg N/p stages are local sort

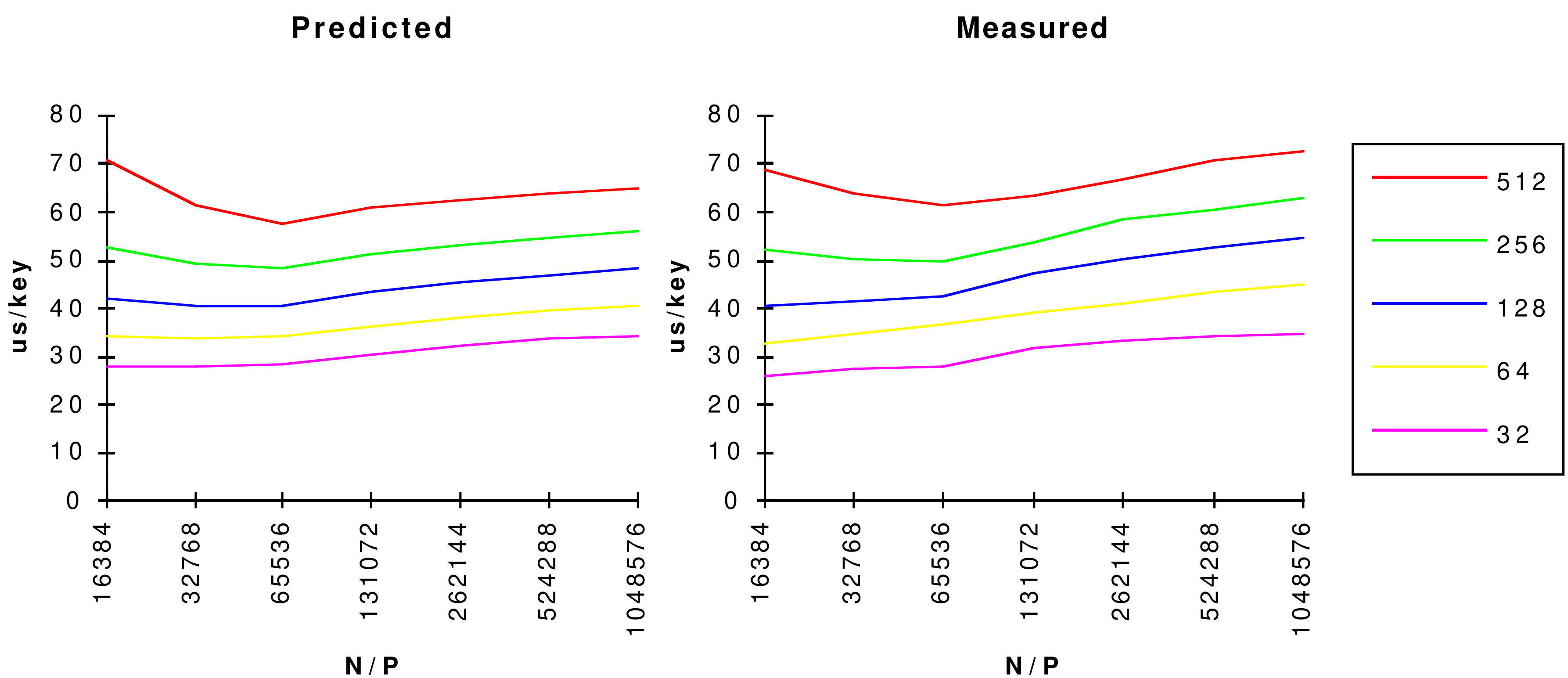
remaining stages involve

Block-to-cyclic, local merges (i - lg N/P cols)

cyclic-to-block, local merges (lg N/p cols within stage)

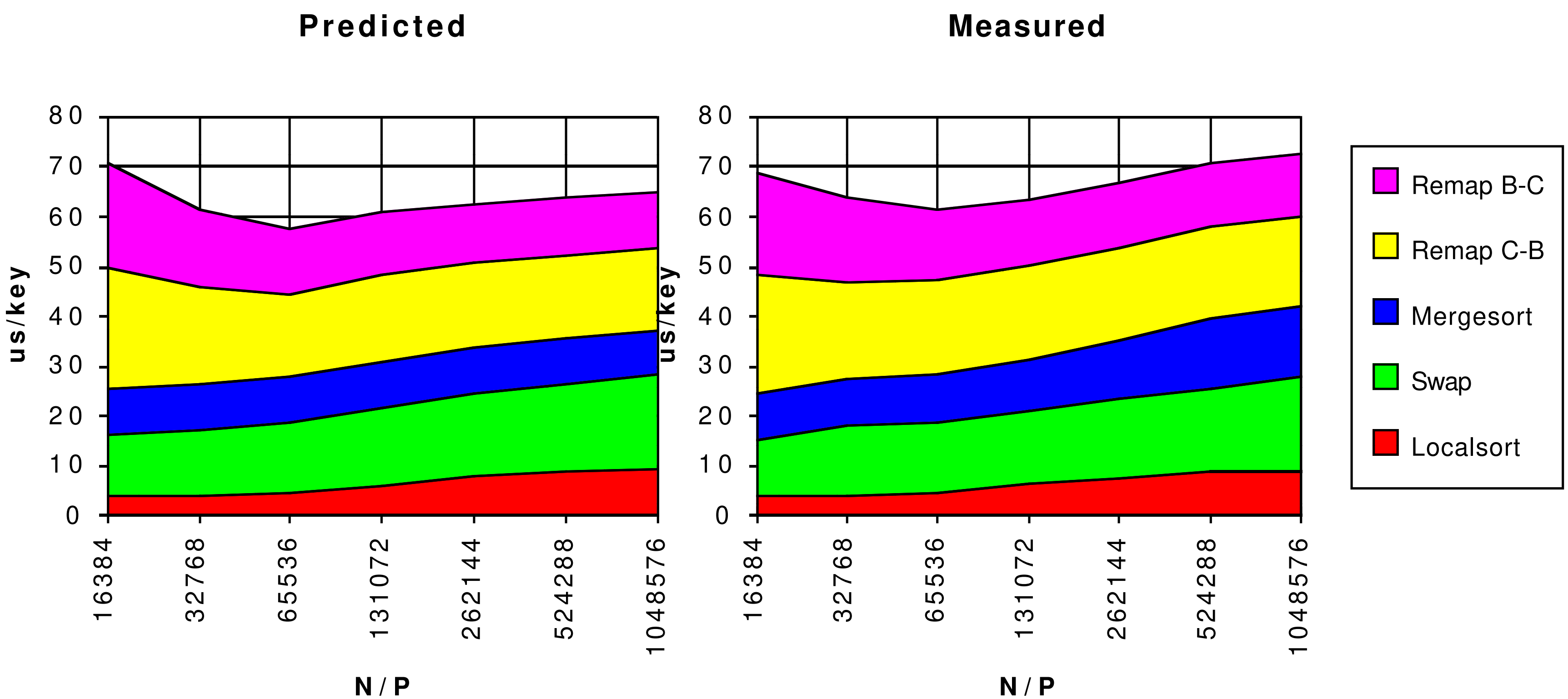
© Culler 1994

Bitonic Sort: time per key



© Culler 1994

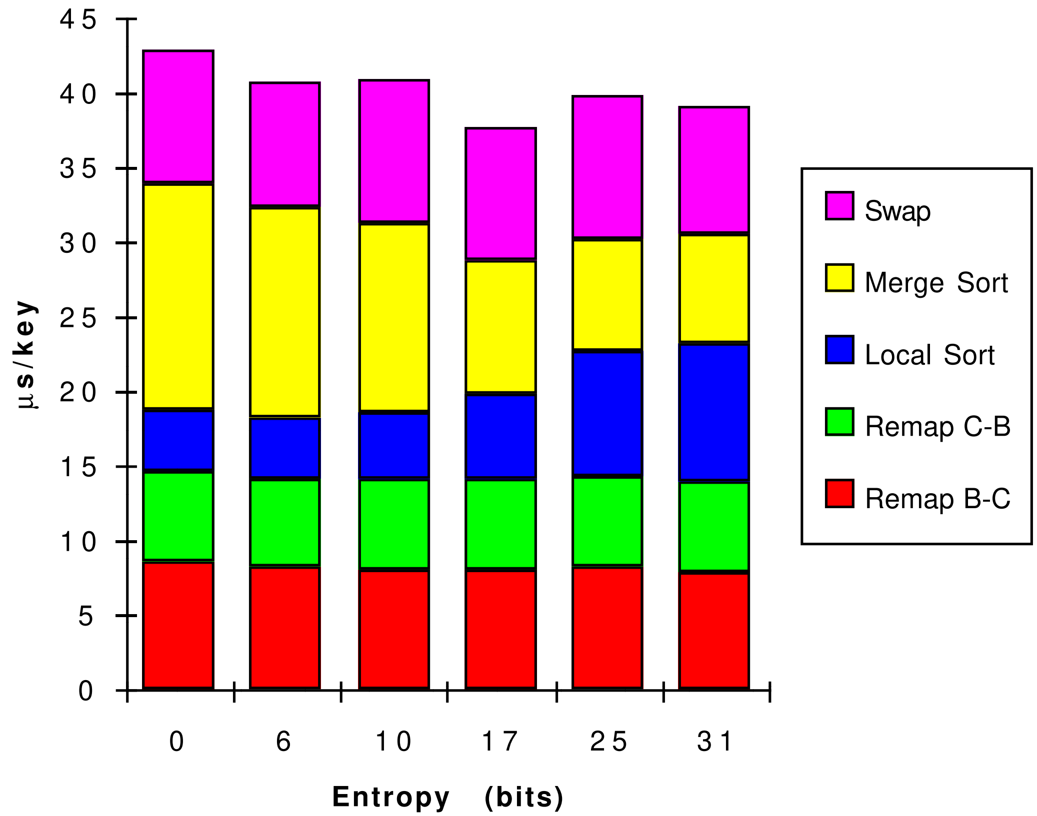
Bitonic: Breakdown



P= 512, random

© Culler 1994

Bitonic: Effect of Key Distributions

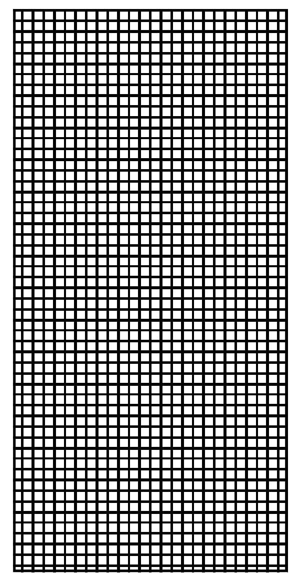


P = 64, N/P = 1 M

© Culler 1994

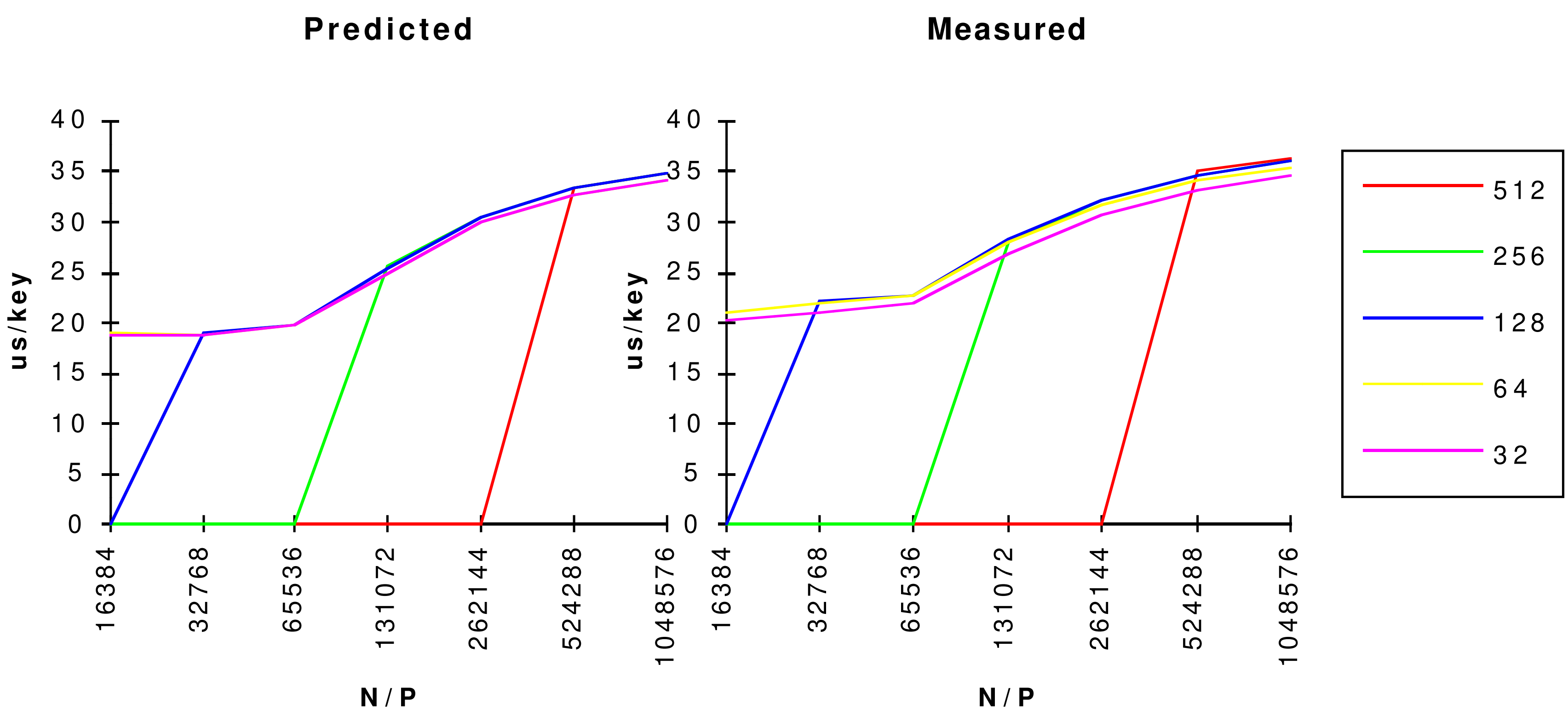
Column Sort

- Local Sort
- Transpose (block-to-cyclic)
- Local Sort
- Untranspose (cyclic-to-block w/o the scatter)
- Local Sort
- Cyclic shift
- Merge Sort
- Unshift



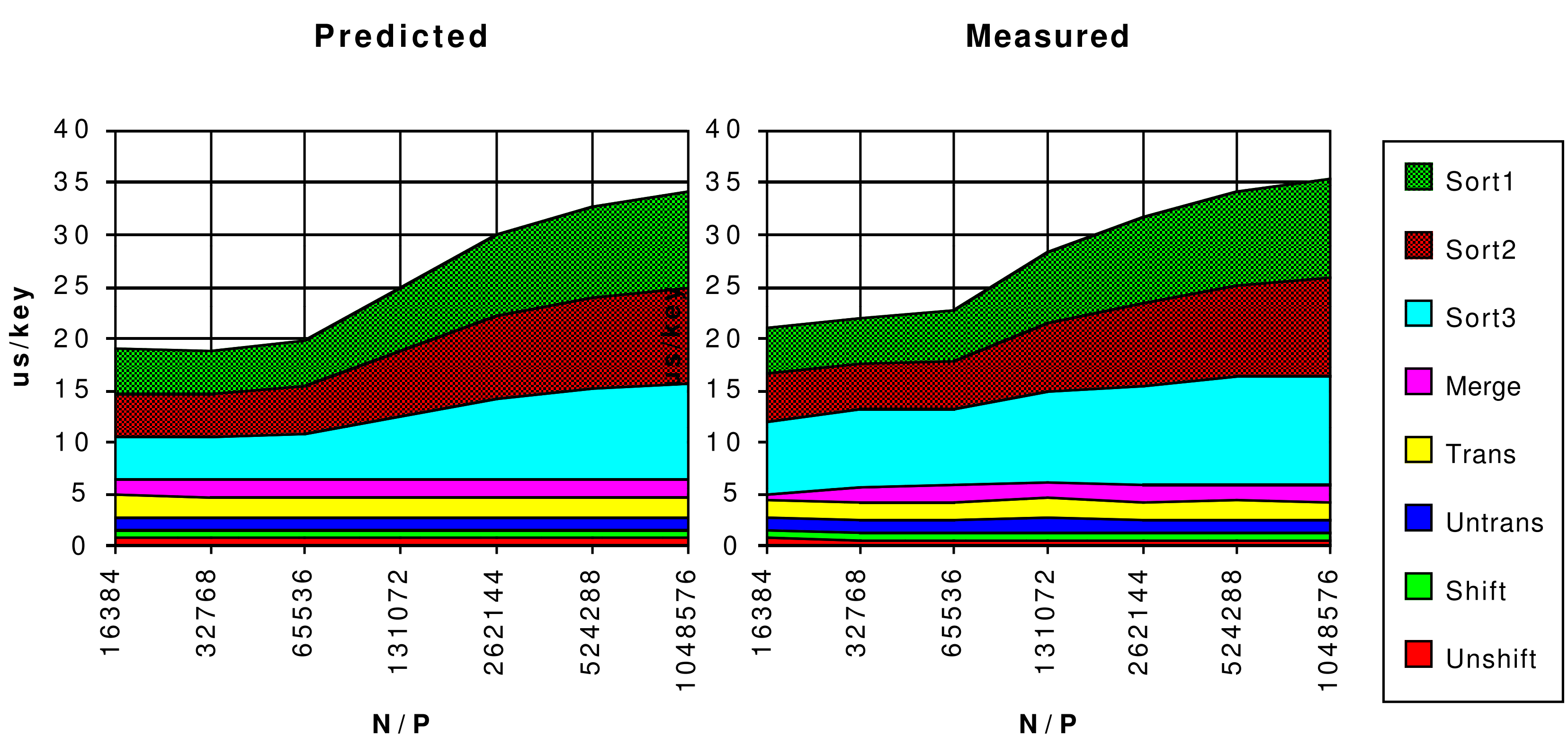
© Culler 1994

Column Sort: Times



© Culler 1994

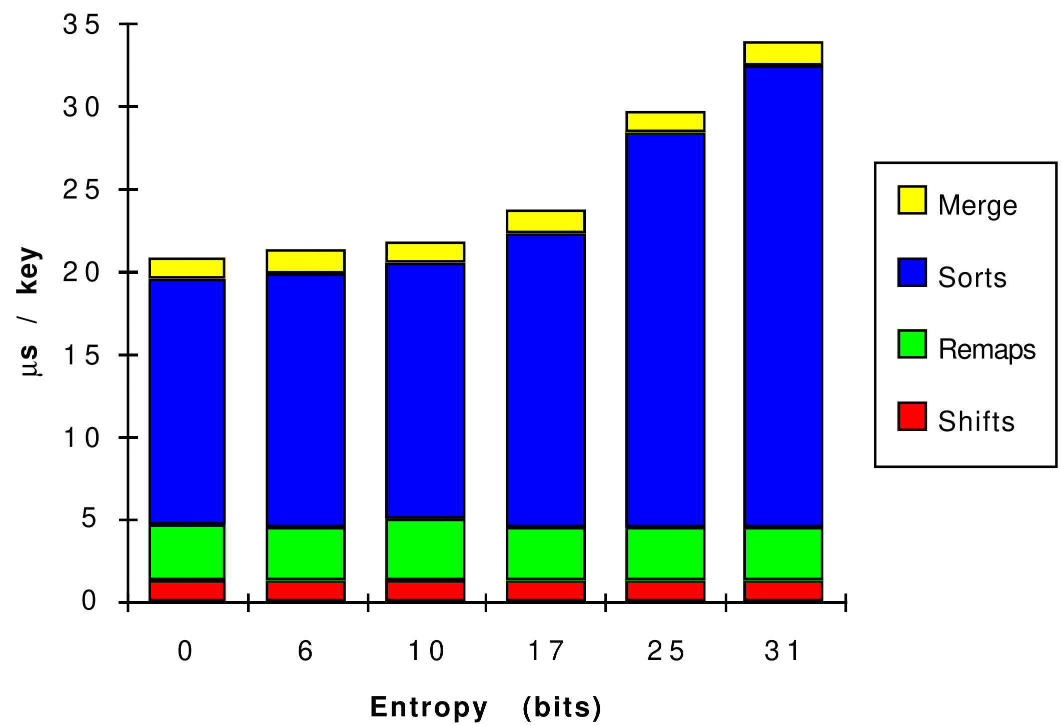
Column: Breakdown



P= 64, random

© Culler 1994

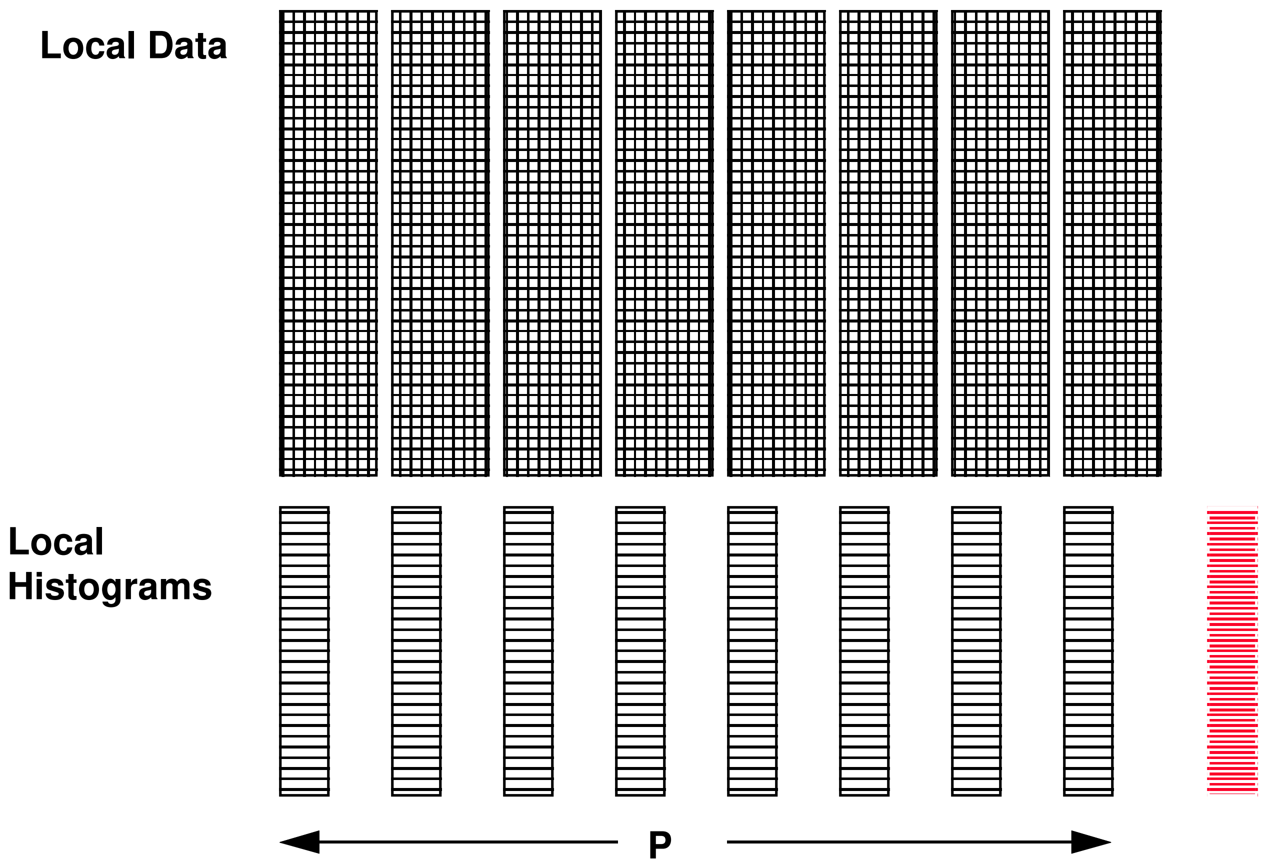
Column: Key distributions



P = 64, N/P = 1M

© Culler 1994

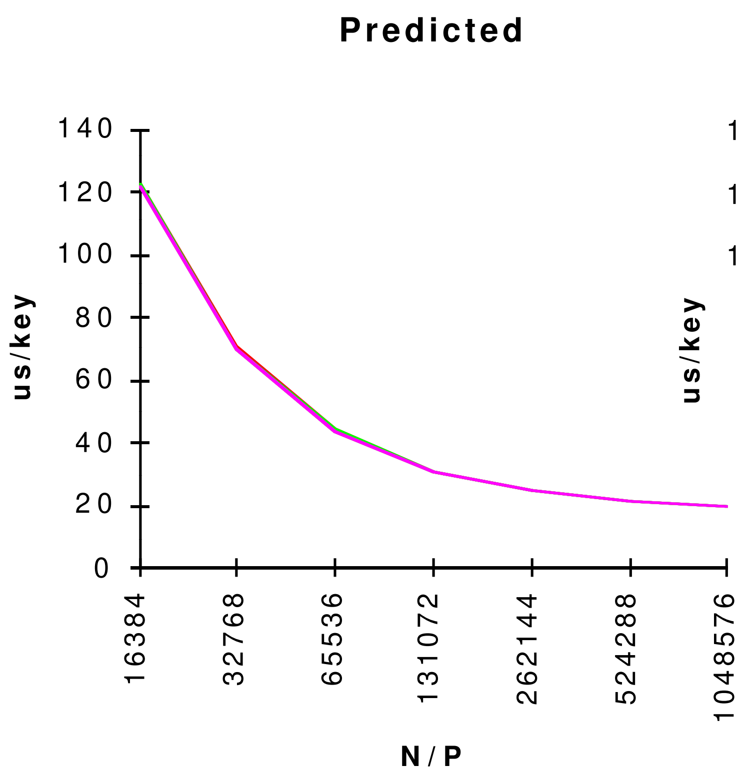
Histo-Radix Sort

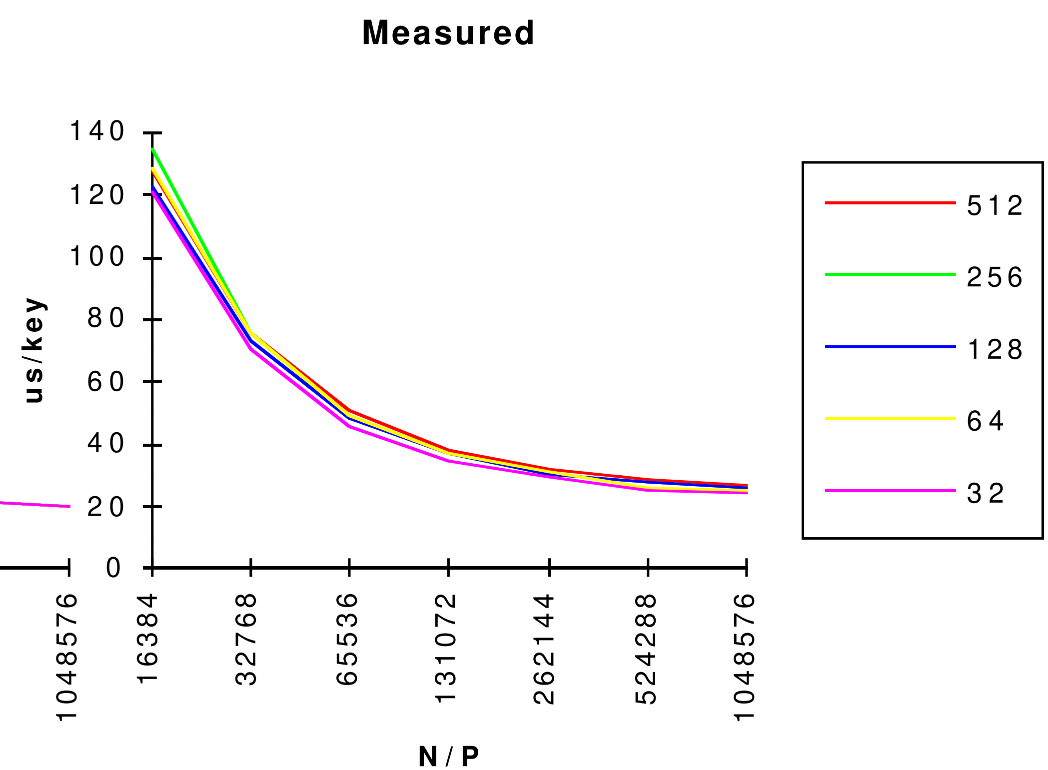


Each Pass
- form local histograms
- form global histogram
- globally distribute data

© Culler 1994

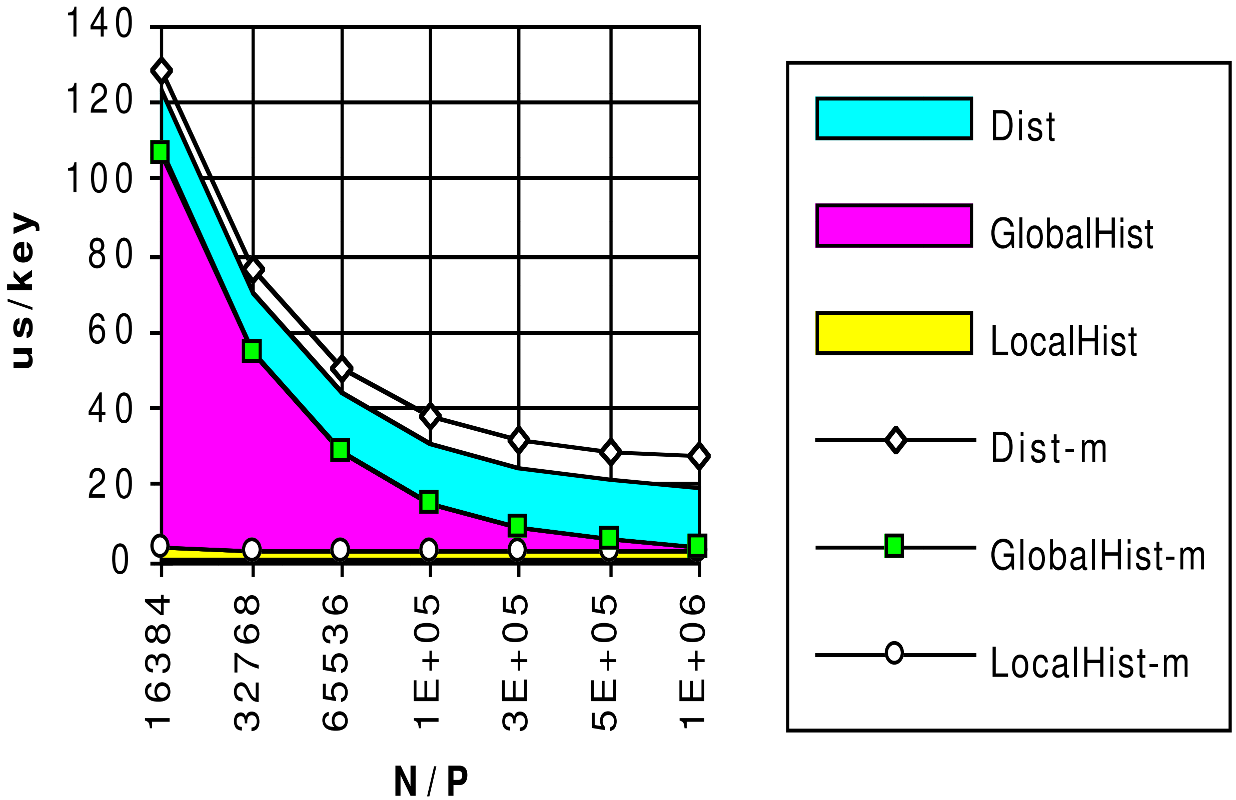
Radix Sort: Times





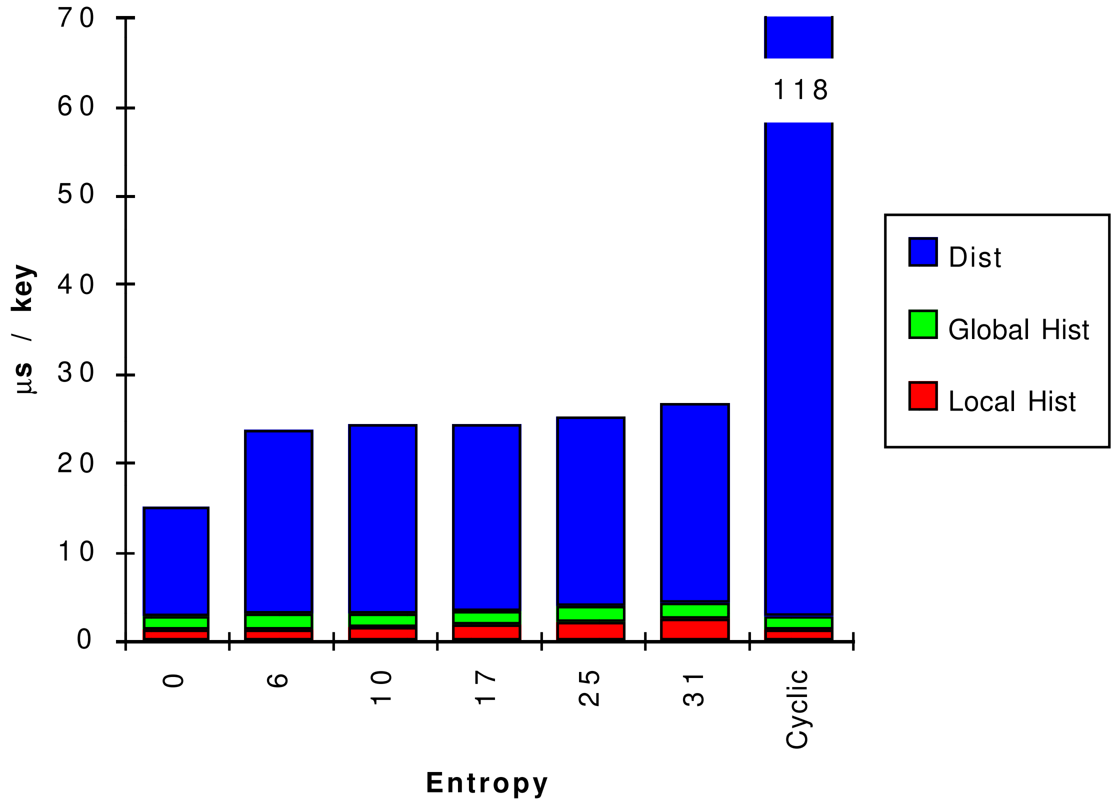
© Culler 1994

Radix: Breakdown



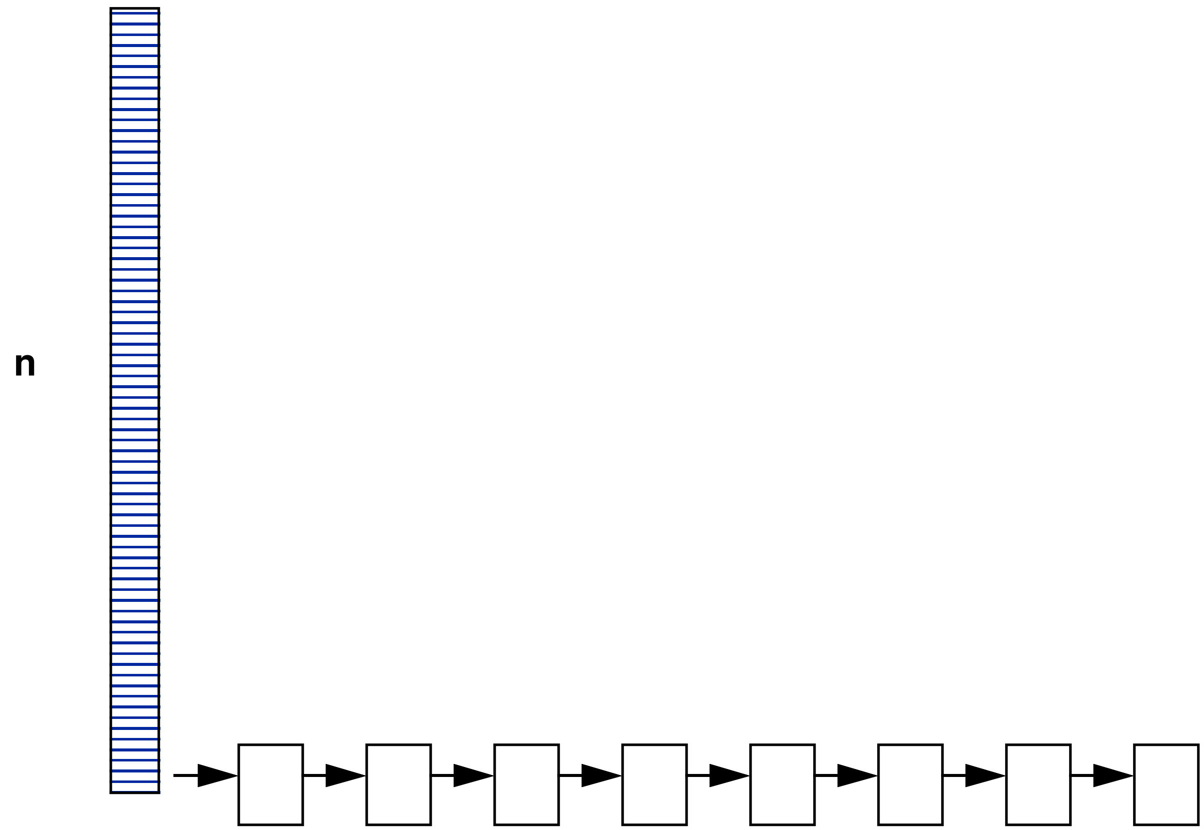
© Culler 1994

Radix: Key distribution



© Culler 1994

Radix: Stream Broadcast Problem



$(P-1)\,(\,2o + L + (n-1)\,g\,)$?

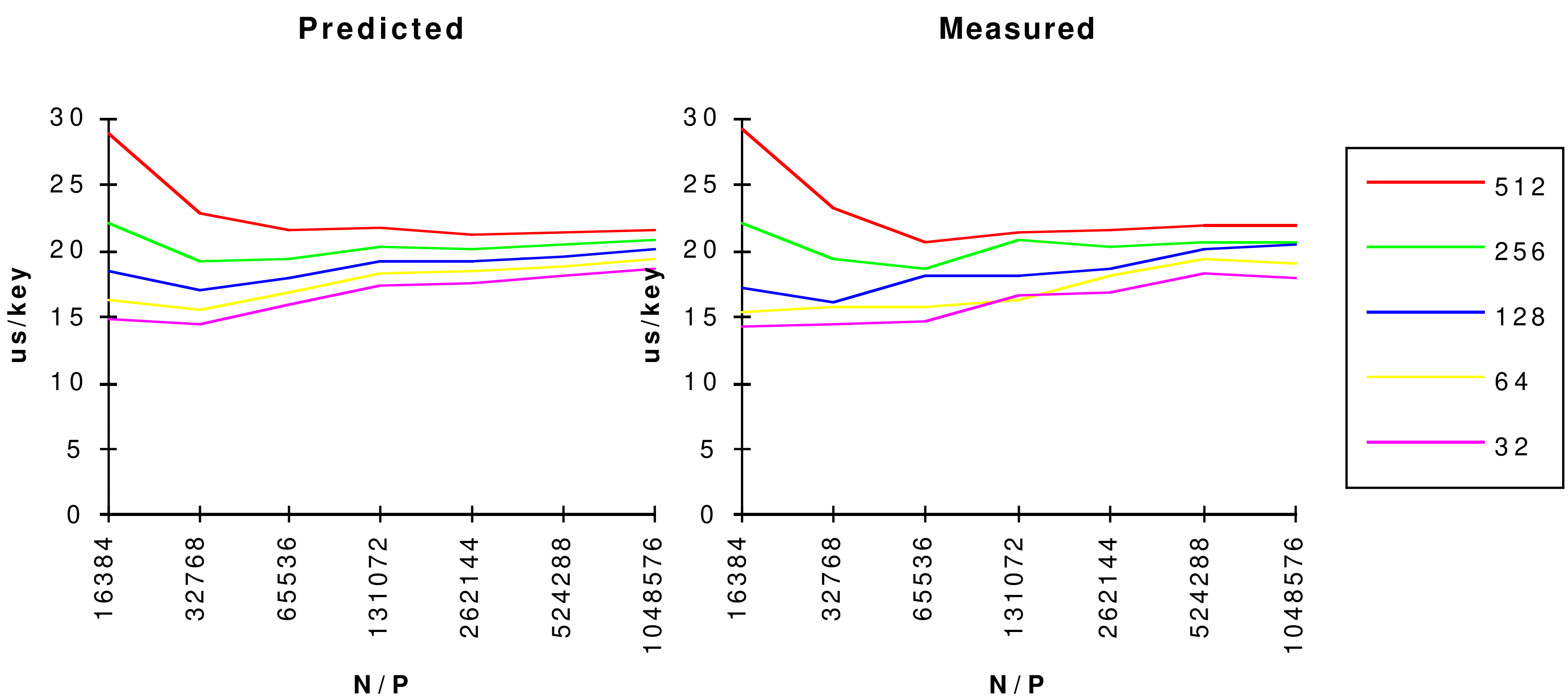
© Culler 1994

Sample Sort

- Sample local data sets
- Gather potential splitters, sort, pick splitters
- Broadcast splitters
- Redistribute data
- Local Sort
- Possibly reshift

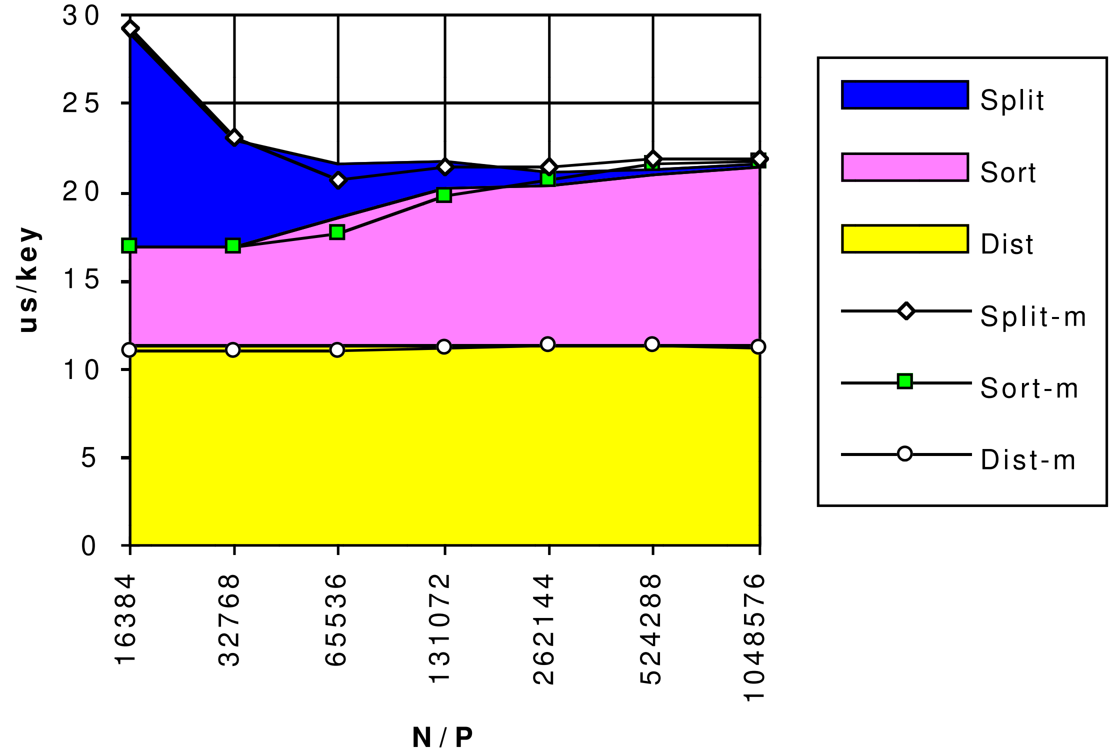
© Culler 1994

Sample Sort: Times



© Culler 1994

Sample Breakdown



© Culler 1994

Conclusions

- Distributed memory model leads to hybrid global / local algorithms
- LogP model is good enough for the global part
  - bandwidth (g) or overhead (o) matter most
  - including end-point contention
  - latency (L) only matters when BW doesn't
  - g is going to be what really matters in the days ahead (NOW)
- Local computational performance is hard!
  - dominated by effects of storage hierarchy (TLBs)
  - getting trickier with multilevels
    - » physical address determines L2 cache behavior
  - and with real computers at the nodes (VM)
  - and with variations in model
    - » cycle time, caches, . . .

© Culler 1994