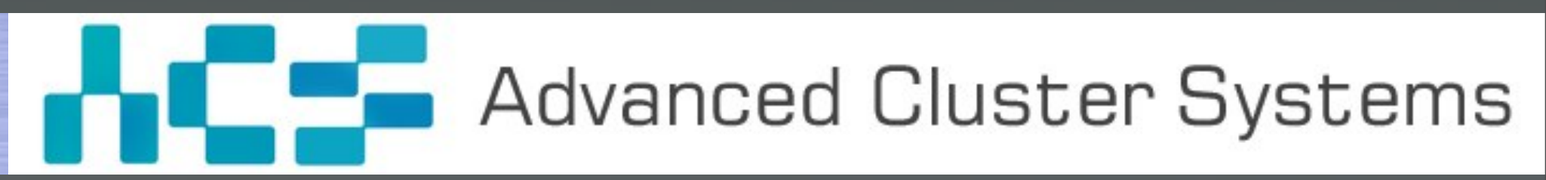
Supercomputing Engine for *Mathematica*

Supercomputing 2008
Austin, Texas



Supercomputing Engine for Mathematica

Dean E. Dauger, Ph. D.
President, Dauger Research, Inc.
d@daugerresearch.com

The First Mac Cluster - established 1998

The Dawson Cluster

High-Performance, Scientific, and Cluster Computing

- Software
  - "Plug-and-Play" Supercomputer-Compatible Clusters
    - Pooch Application
    - Source-Code Tutorials
  - Visualization & Simulation

- Consulting Services
  - Optimization
  - Parallelization
  - Vectorization

Why Parallel Computing?

Problems too large to solve on one computer

- Takes too much time
- Requires more memory
  - can outgrow RAM capacity

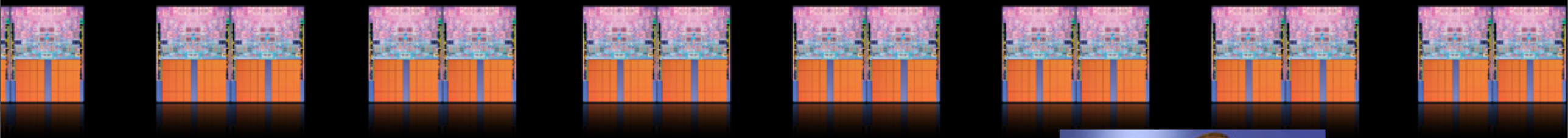
Programming API standardized

- Message-Passing Interface (MPI)
  - specification established in 1994
  - dominant software interface at supercomputing centers
  - portable MPI code in Fortran and C

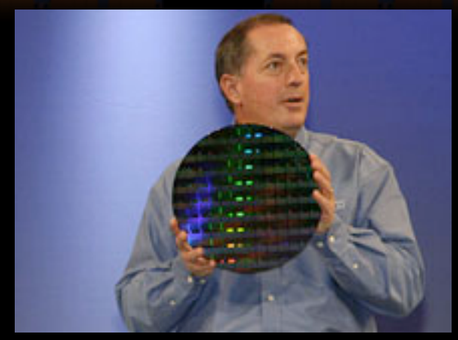
Why Parallel Computing?

Problems too large to solve on one ~~computer~~ "core"

- Computational power *per core* no longer doubles
- Doubling cores is how Moore's Law technically holds
- Multicore utilization is not automatic
- Software must make up for where hardware leaves off
  - **Choose your parallel programming paradigm wisely**



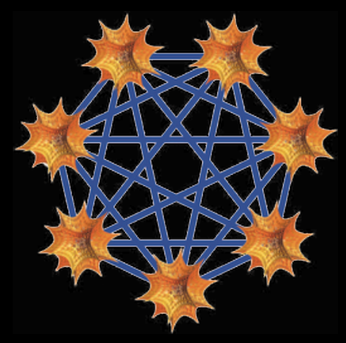
Multicore Eroding Moore's Law



- http://macresearch.org/multicore_eroding_moores_law

MPI in *Mathematica*

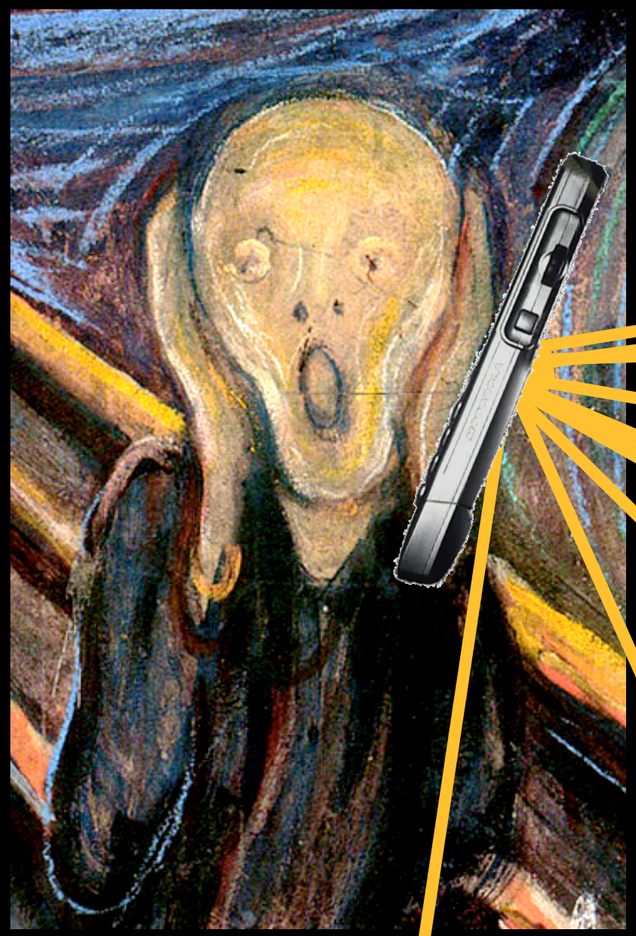
Supercomputing Engine for *Mathematica*

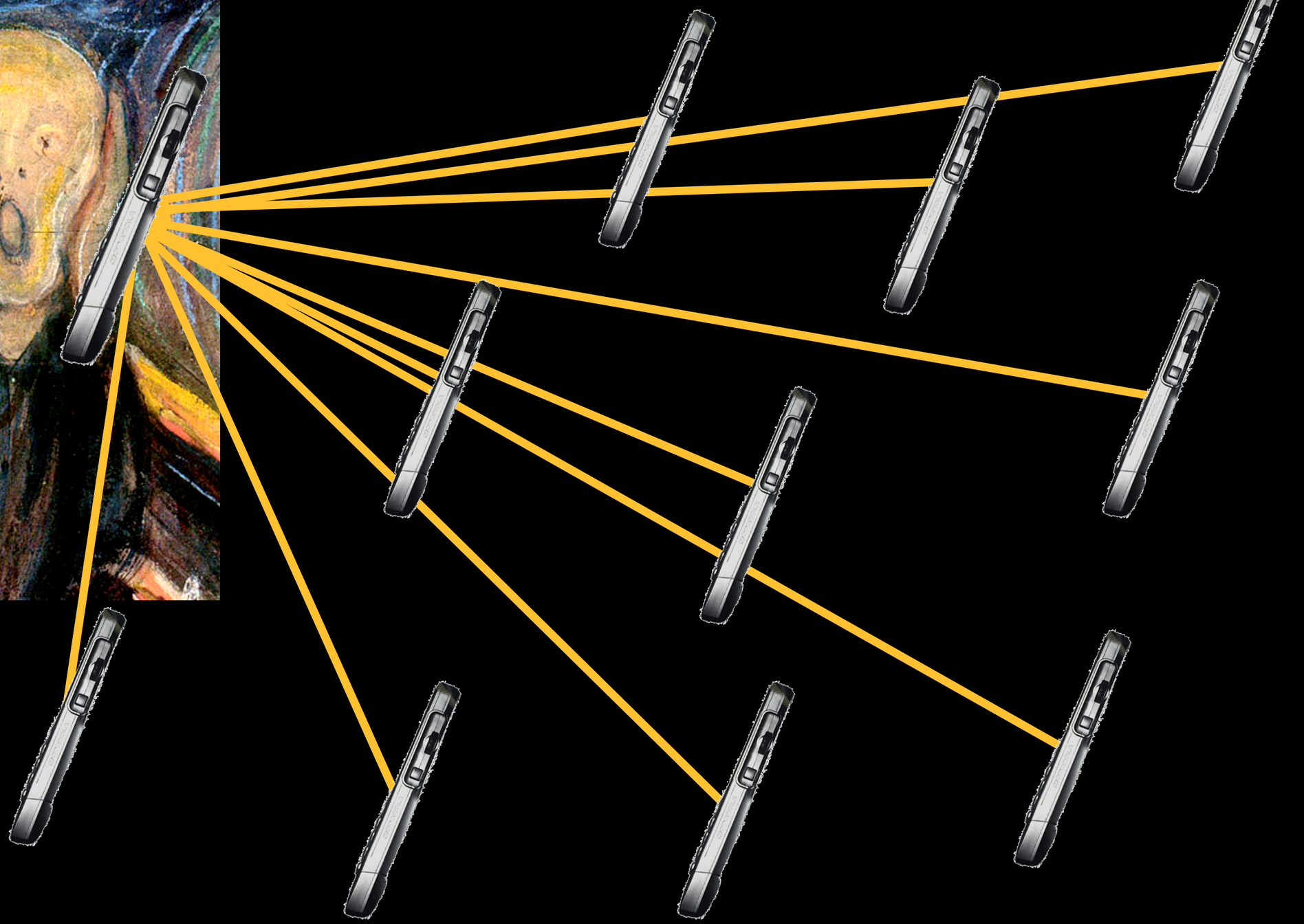


- Closely follows MPI standard in *Mathematica* environment
  - Basic MPI calls (mpiSend, mpiRecv)
  - Asynchronous MPI calls (mpiISend, mpiIRecv, mpiTest)
  - Collective MPI calls (mpiBcast, mpiGather, mpiAlltoall)
- High-level parallel calls for common tasks
  - ParallelTable, EdgeCell, ParallelFourier, ElementManage
- Basic Parallel I/O
- Automatically locates, launches, configures, and coordinates *Mathematica* kernels via Pooch
  - from command line or *Mathematica*'s Front End
- Builds on any licensed grid*Mathematica*

Why MPI instead of “grid”?

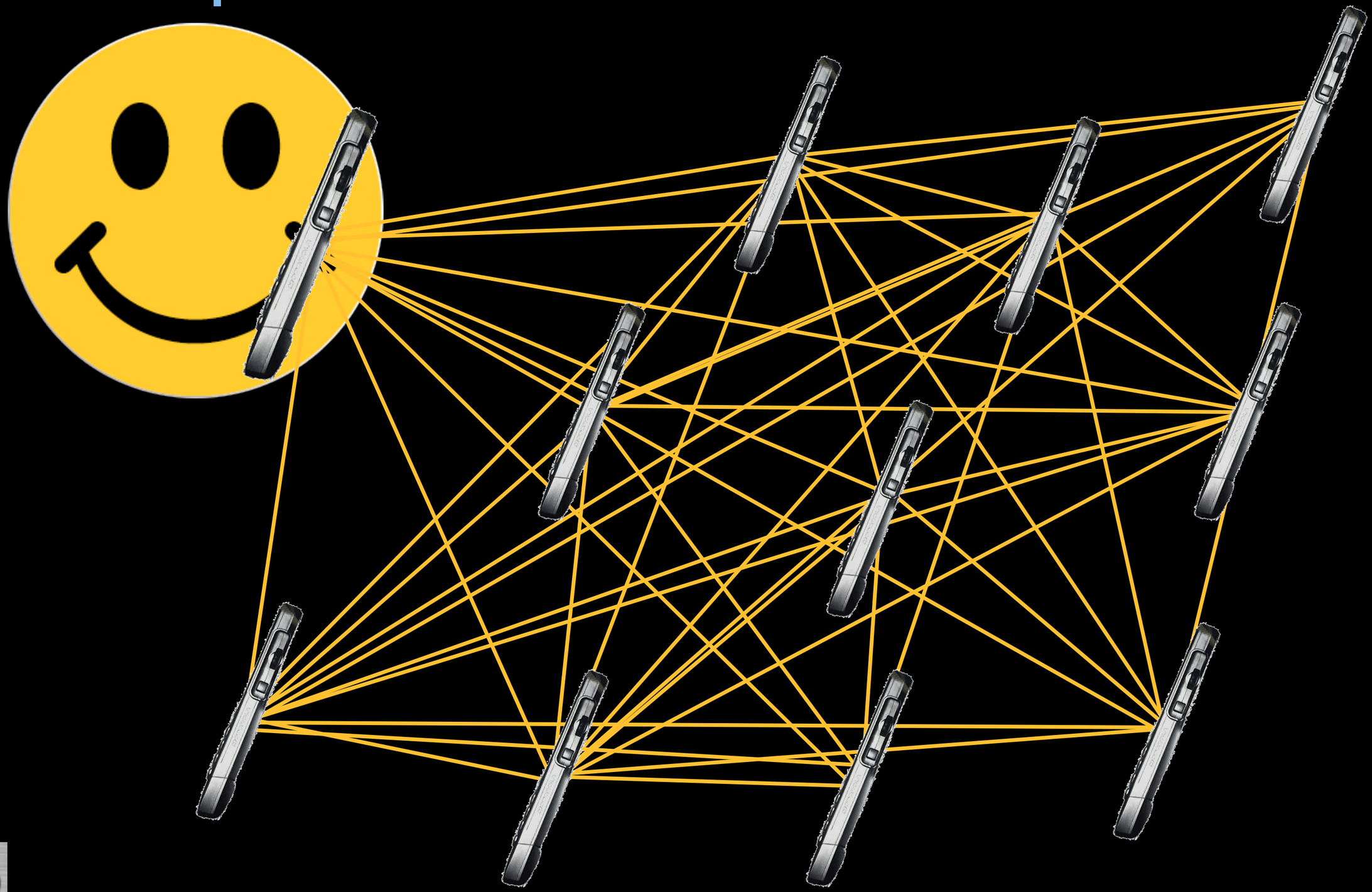
Typical “grid” implementation





Why MPI instead of “grid”?

MPI implementation



Parallel Code using MPI

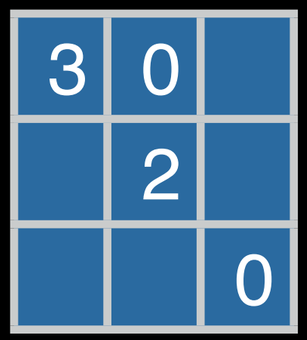
Code coordinating parallel work using messages

- N tasks or “virtual processors” running simultaneously, labeled 0 through N-1
- executables often use identification data to determine algorithmically what part of the problem on which to work
- tasks pass messages amongst themselves to organize data and coordinate work
- any group of tasks can communicate ($\Rightarrow O(N^2)$ connections)
  - simple sends and receives
  - collective calls - broadcast, gather, reduce, transpose, etc.
- synchronization not required, but often implied by messages

“Game of Life”

Cellular Automata

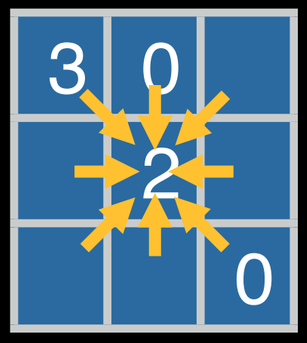
- J. Conway, Princeton

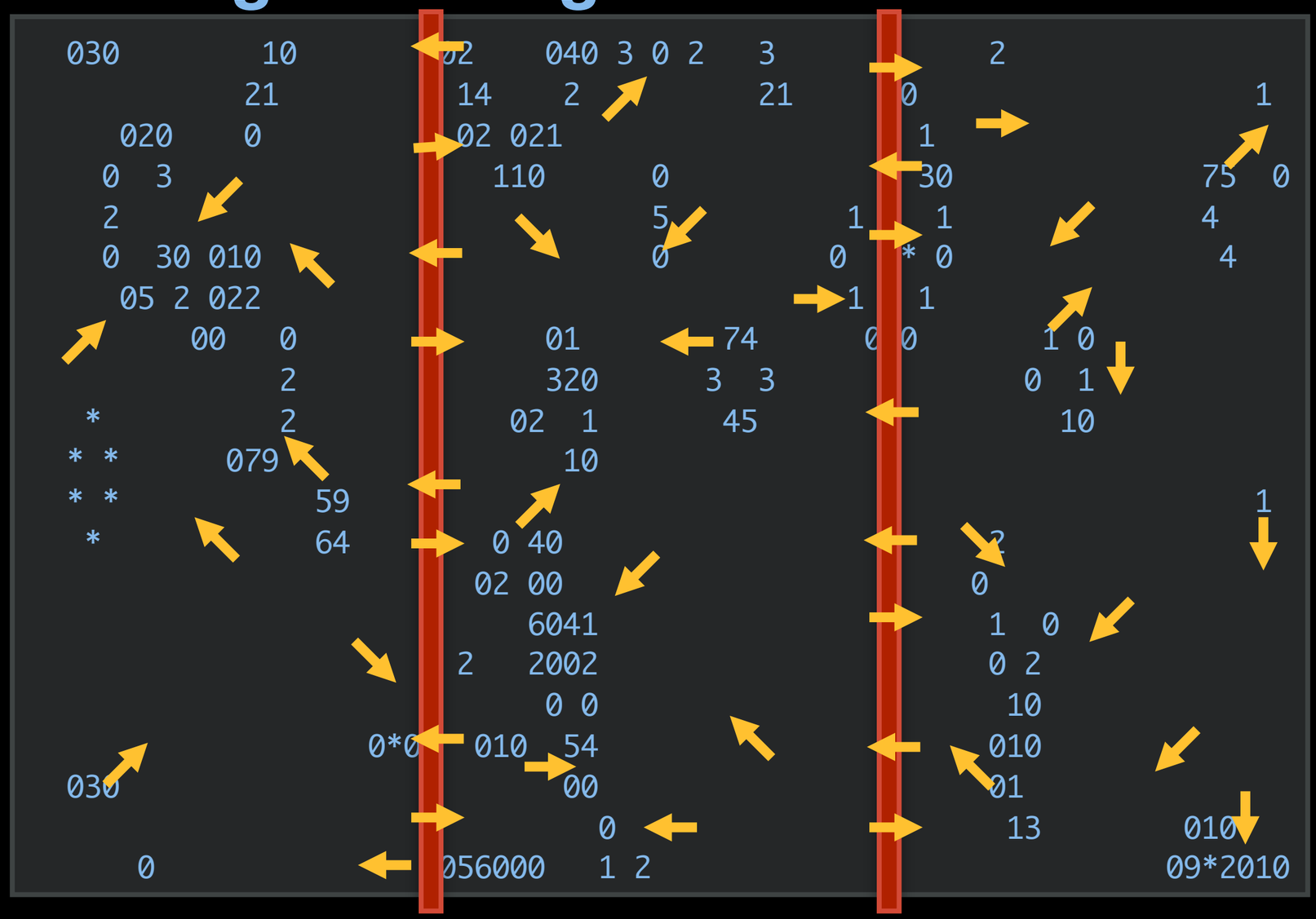


Life of one cell depends on its neighbors

Parallel “Life”

Message-Passing for Cellular Automata

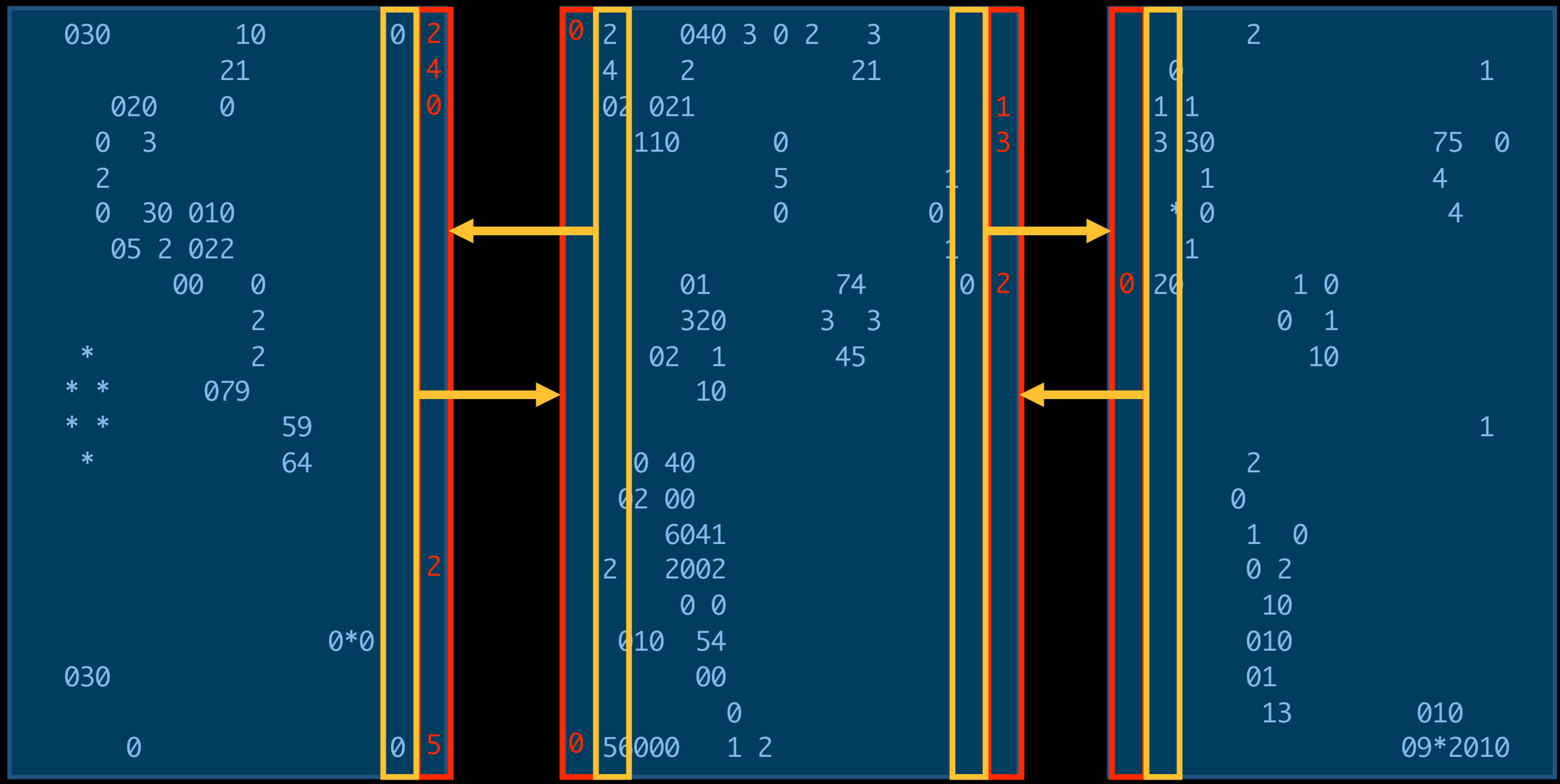




Data to propagate between partitions

Parallel “Life”

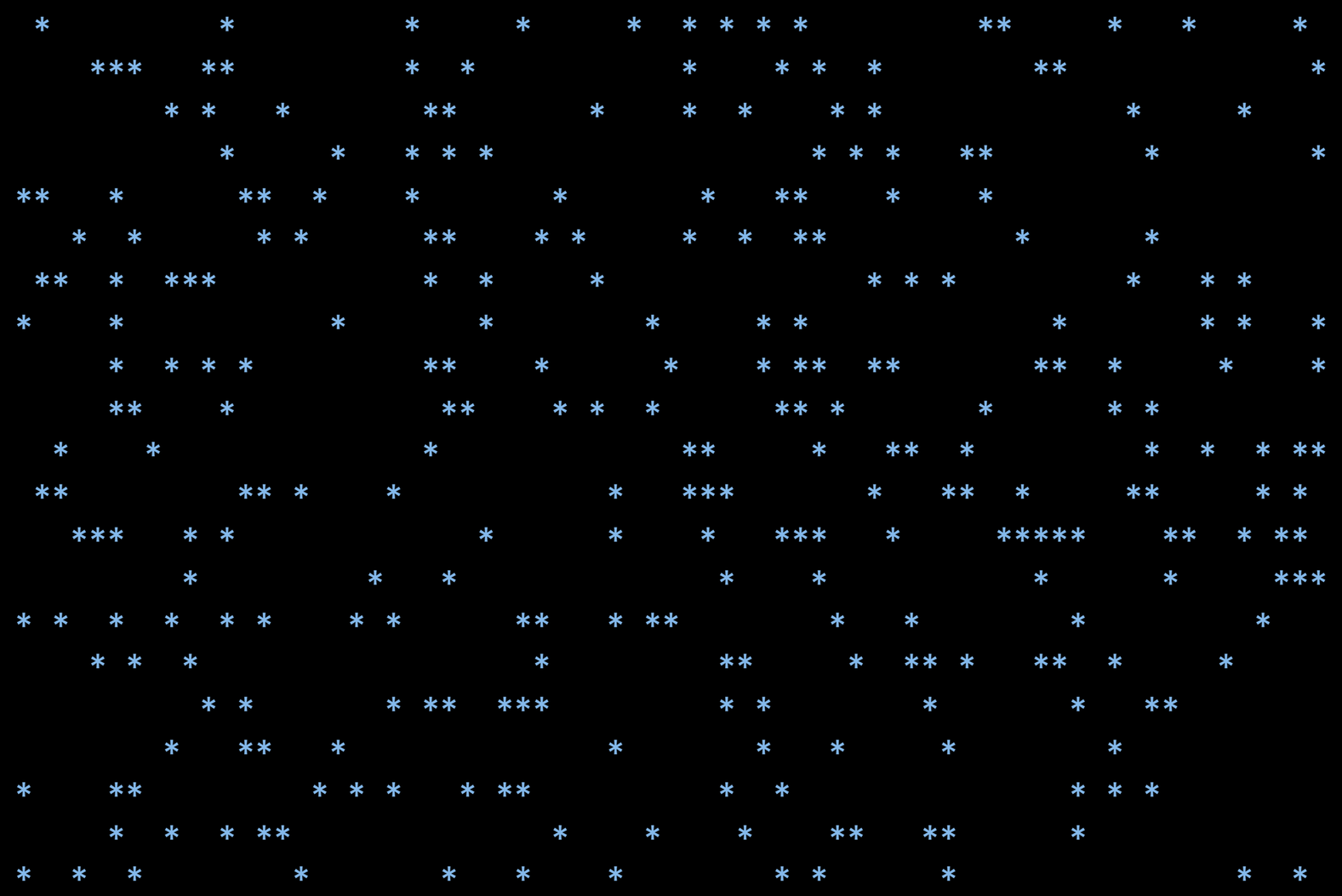
Message-Passing for Cellular Automata



Message exchange maintains “guard cells”

Plasma Simulation

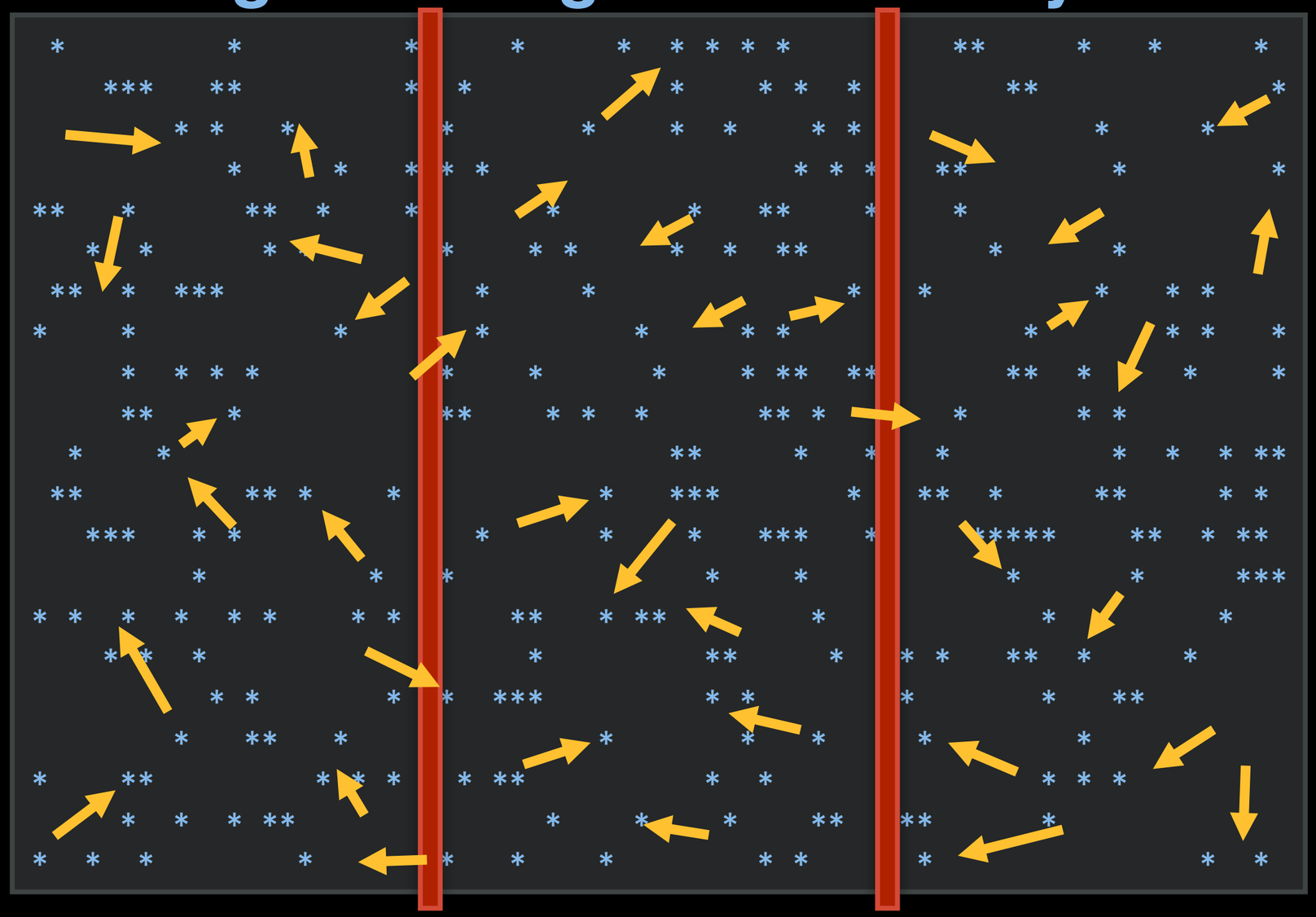
Plasma Dynamics



V. K. Decyk, C. D. Norton, *Comp. Phys. Communications* **164** (2004) 80-85

Parallel Plasma Simulation

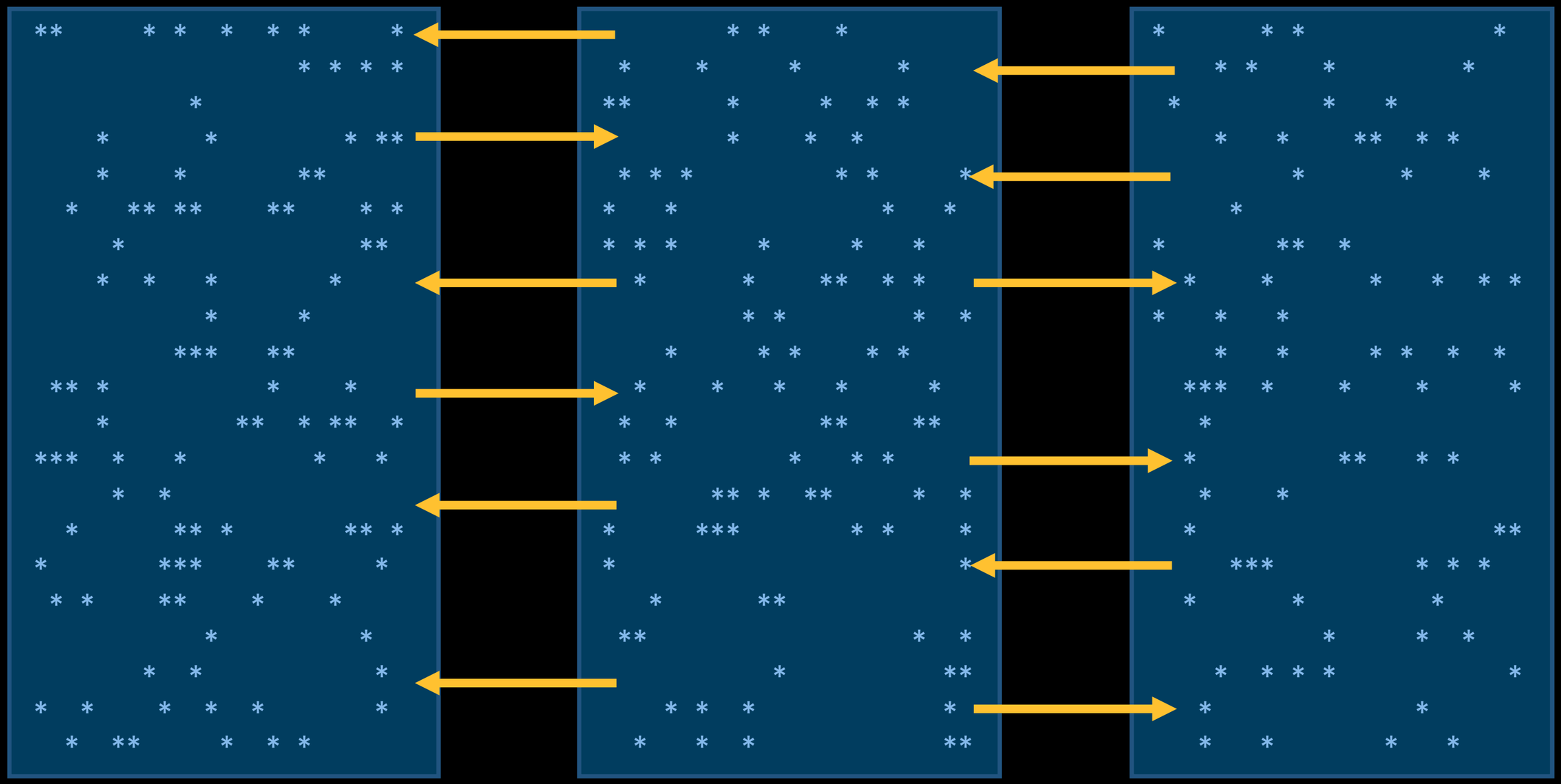
Message-Passing for Plasma Dynamics



V. K. Decyk, C. D. Norton, *Comp. Phys. Communications* **164** (2004) 80-85

Parallel Plasma Simulation

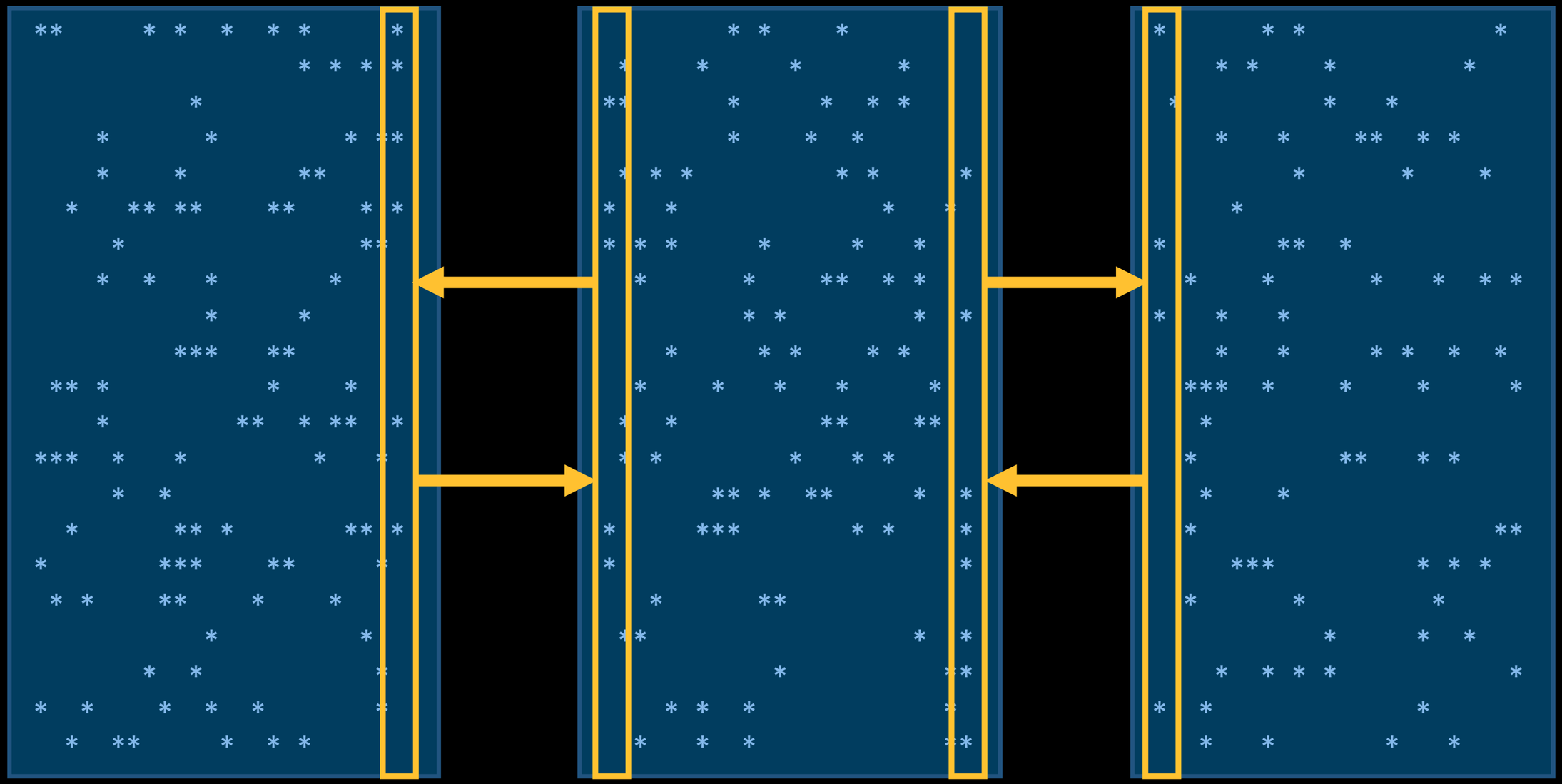
Message-Passing for Plasma Dynamics



Particles propagate from one partition to its neighbor

Parallel Plasma Simulation

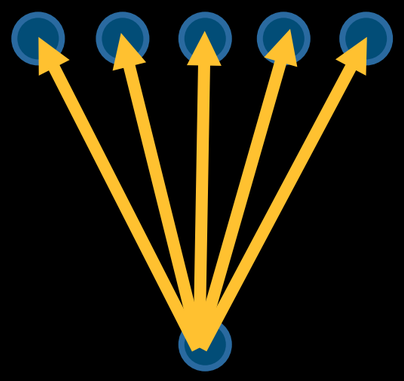
Message-Passing for Plasma Dynamics



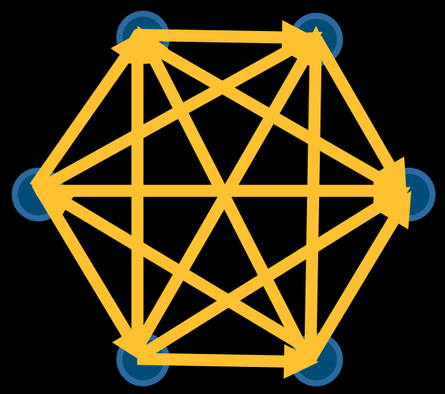
Particles propagate from one partition to its neighbor

Message-Passing Patterns

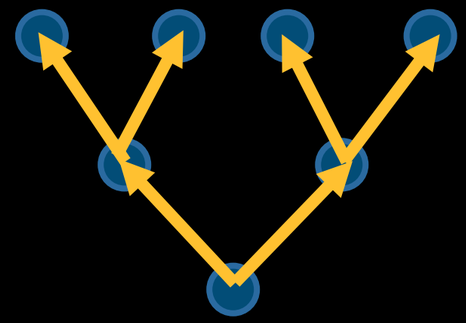
Supported via MPI



“Master-Slave”

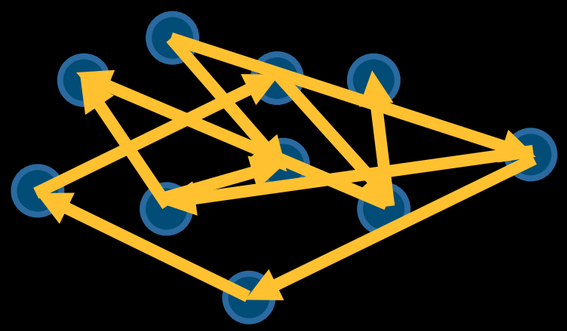


All to All

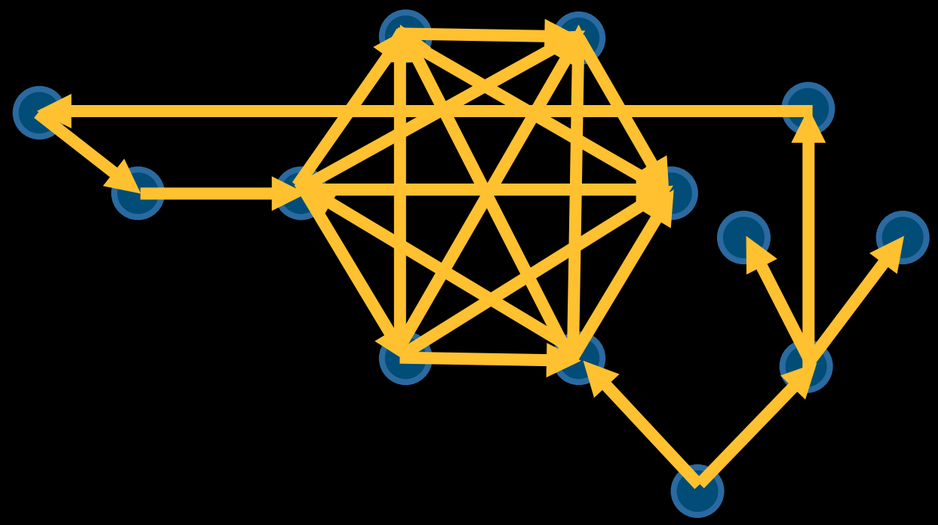


Tree

Nearest Neighbor



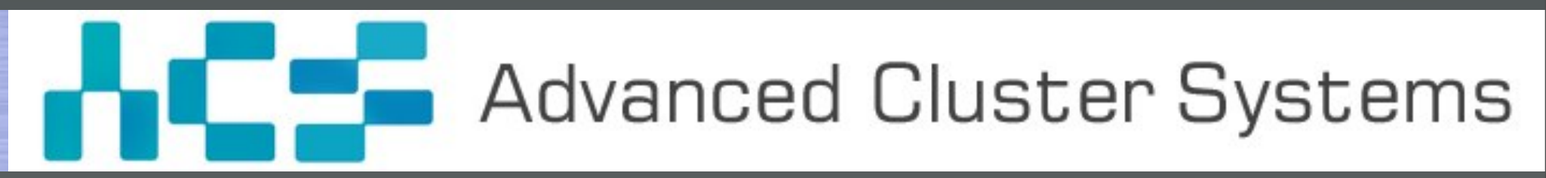
Irregular



Or Combinations

A Demonstration

Dean E. Dauger, Ph. D.
d@daugerresearch.com

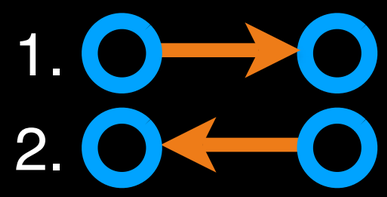


http://daugerresearch.com

Ping-pong Benchmark

gridMathematica and SEM

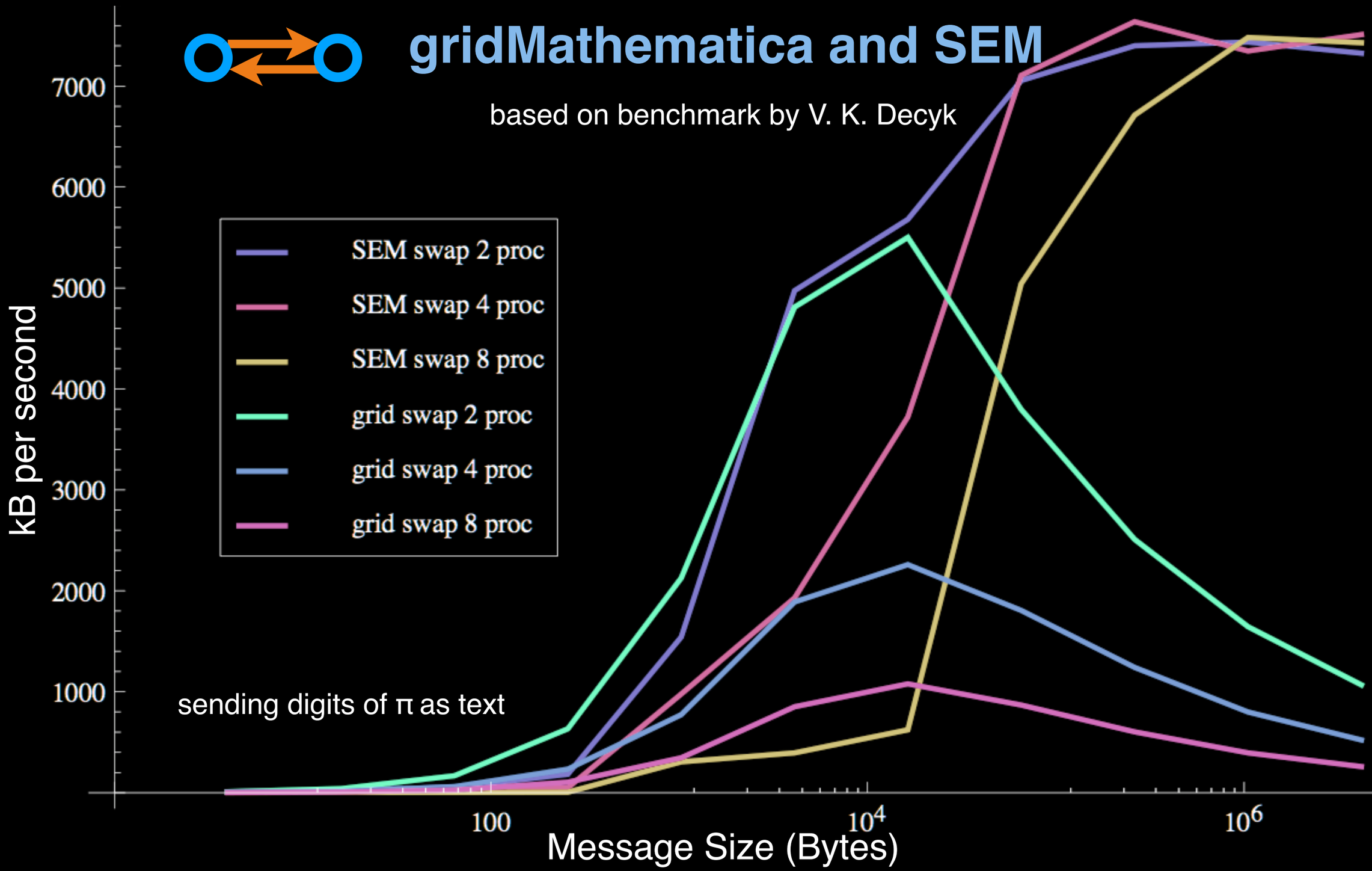
based on benchmark by V. K. Decyk



Swap Benchmark

gridMathematica and SEM

based on benchmark by V. K. Decyk



“We found SEM to be very efficient in terms of stability and use for financial engineers who do not have the time to optimize load balancing issues but want to focus on modeling.”

- Vineer Bhansali, Managing Director, PIMCO

Dauger Research

Advanced Cluster Systems

"I CAN endorse SEM with pride. I would emphasize the flexibility and scalability of SEM. And SEM can make writing *Mathematica* programs even more flexible than before."

- Yuko Matsuda, Professor,
Tokyo Institute of Technology

Who to Contact

Dean E. Dauger, Ph. D.
Dauger Research, Inc.
d@daugerresearch.com

Zvi Tannenbaum
Advanced Cluster Systems, LLC
zvi@advclustersys.com

http://daugerresearch.com & http://advclustersys.com

Roadmap

Wolfram Research Booth	**Booth #163** **Wed 3:00 pm**
Exhibitor Forum Session	**Room 19AB** **Thurs 10:30 am**
Advanced Cluster Systems Booth	**Booth #162** **Mon-Thurs**
Machine Evaluation Workshop 19	**Daresbury, UK** **2 Dec 2008**

For More Information

Reference Library

Documentation

Supercomputing Engine for *Mathematica* Site
http://daugerresearch.com/pooch/mathematica/

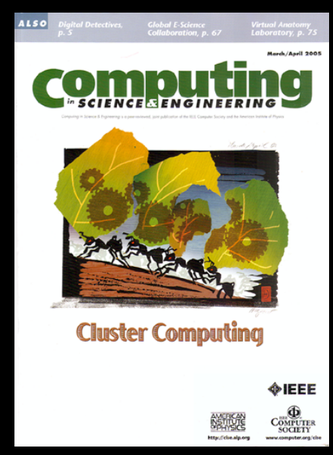
Advanced Cluster Systems Site
http://advclustersys.com/

Tutorials on Writing Parallel Code
http://daugerresearch.com/vault/tutorials/

Mac Clustering on National Television
http://daugerresearch.com/awards/KeepingAmericaStrong/

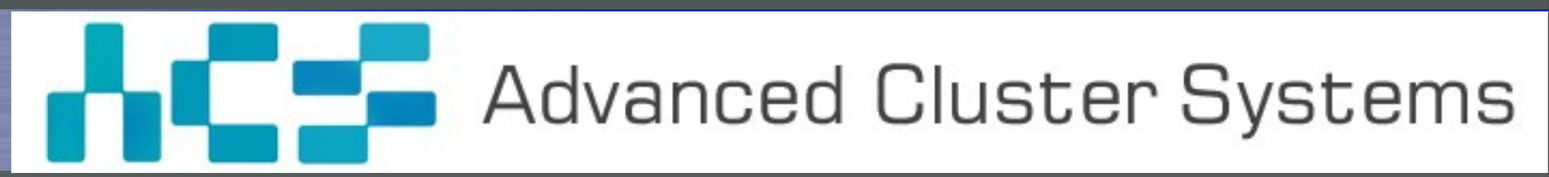
Related Publications
http://daugerresearch.com/vault/publication/

Multicore Eroding Moore's Law
http://macresearch.org/multicore_eroding_moores_law



Q&A

Dean E. Dauger, Ph. D.
President, Dauger Research, Inc.
d@daugerresearch.com



http://daugerresearch.com & http://advclustersys.com/