9th EuroPVM/MPI

4th Austrian-Hungarian Workshop on Distributed and Parallel Systems (DAPSYS 2002)

Johannes Kepler University Linz, Austria
September, 29th-October 02nd, 2002

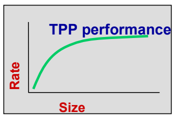
High Performance Computing, Computational Grid, and Numerical Libraries

Jack Dongarra
Innovative Computing Lab
University of Tennessee
http://www.cs.utk.edu/~dongarra/

TOP500 SUPERCOMPUTER

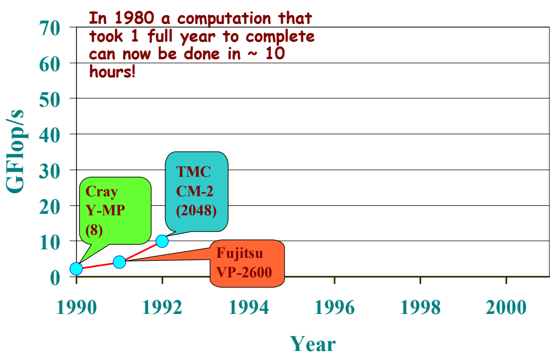
H. Meuer, H. Simon, E. Strohmaier, & JD

- Listing of the 500 most powerful Computers in the World
- Yardstick: Rmax from LINPACK MPP
  Ax=b, dense problem

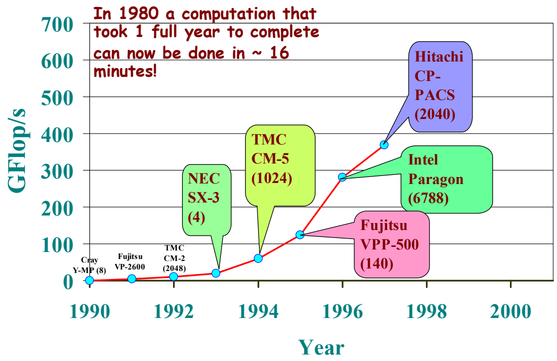


- Updated twice a year
  SC'xy in the States in November
  Meeting in Mannheim, Germany in June
- All data available from **www.top500.org**

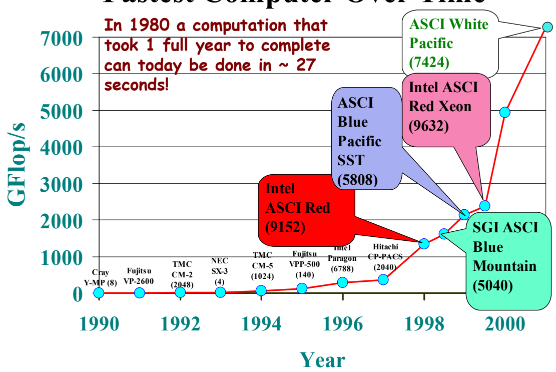
Fastest Computer Over Time



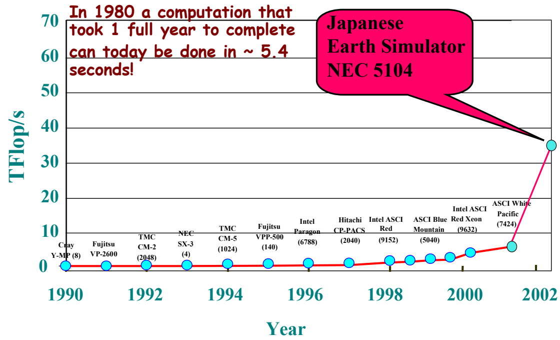
Fastest Computer Over Time



Fastest Computer Over Time



Fastest Computer Over Time

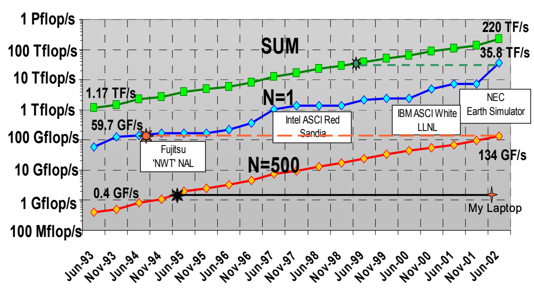


Top10 of the Top500

Rank	Manufacturer	Computer	R_{max} [TF/s]	Installation Site	Country	Year	Area of Installation	# Proc
1	NEC	Earth-Simulator	35.86	Earth Simulator Center	Japan	2002	Research	5120
2	IBM	ASCI White SP Power3	7.23	Lawrence Livermore National Laboratory	USA	2000	Research	8192
3	HP	AlphaServer SC ES45 1 GHz	4.46	Pittsburgh Supercomputing Center	USA	2001	Academic	3016
4	HP	AlphaServer SC ES45 1 GHz	3.98	Commissariat a l'Energie Atomique (CEA)	France	2001	Research	2560
5	IBM	SP Power3 375 MHz	3.05	NERSC/LBNL	USA	2001	Research	3328
6	HP	AlphaServer SC ES45 1 GHz	2.92	Los Alamos National Laboratory	USA	2002	Research	2048
7	Intel	ASCI Red	2.38	Sandia National Laboratory	USA	1999	Research	9632
8	IBM	pSeries 690 1.3 GHz	2.31	Oak Ridge National Laboratory	USA	2002	Research	864
9	IBM	ASCI Blue Pacific SST, IBM SP 604e	2.14	Lawrence Livermore National Laboratory	USA	1999	Research	5808
10	IBM	pSeries 690 1.3 Ghz	2.00	IBM/US Army Reseach Lab (ARL)	USA	2002	Vendor	768

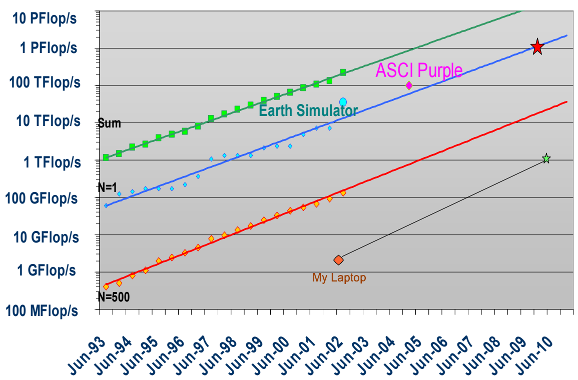
7

TOP500 - Performance



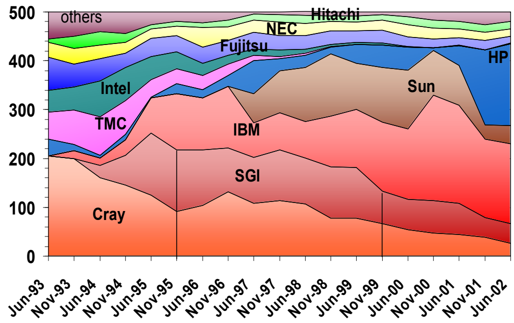
8

Performance Extrapolation



9

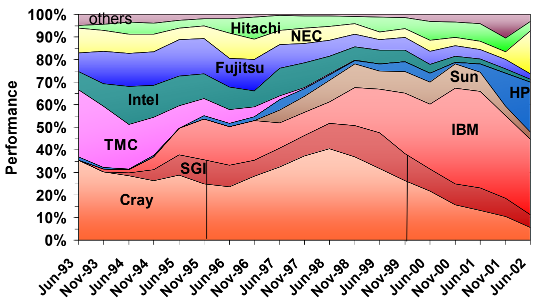
Manufacturers



HP 168, IBM 164

10

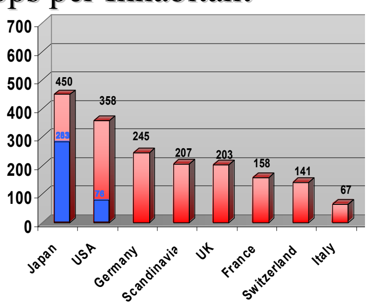
Manufacturers



IBM 33%, HP 22%, NEC 19%

11

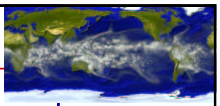
Kflops per Inhabitant



Japan 57 Tf/s US 99 Tf/s

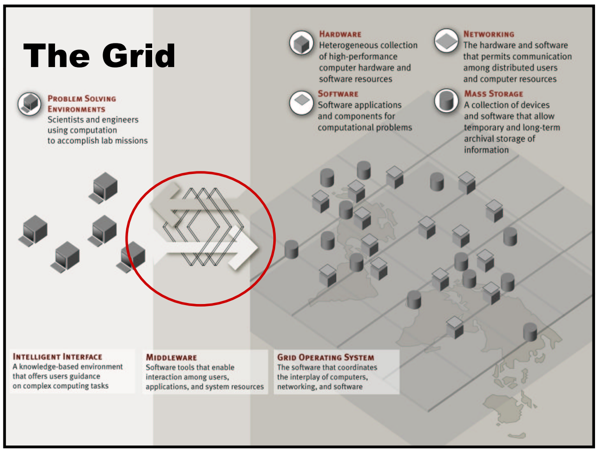
12

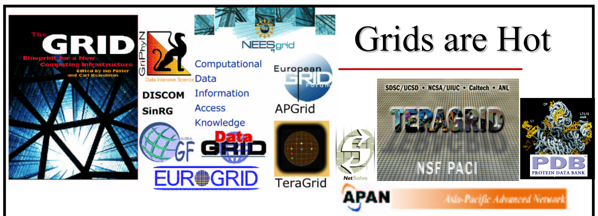
In the past: Isolation
Motivation for Grid Computing



- Today there is a complex interplay and increasing interdependence among the sciences.
- Many science and engineering problems require widely dispersed resources be operated as systems.
- What we do as collaborative infrastructure developers will have profound influence on the future of science.
- Networking, distributed computing, and parallel computation research have matured to make it possible for distributed systems to support high-performance applications, but...
  - Resources are dispersed
  - Connectivity is variable
  - Dedicated access may not be possible

Today: Collaboration

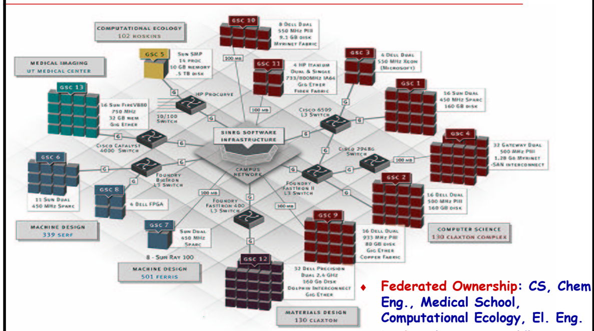




Grids are Hot

- IPG NASA http://nas.nasa.gov/~wej/home/IPG
- Globus http://www.globus.org/
- Legion http://www.cs.virgina.edu/~grimshaw/
- AppLeS http://www-cse.ucsd.edu/groups/hpcl/
- NetSolve http://www.cs.utk.edu/netsolve/
- NINF http://phase.etl.go.jp/ninf/
- Condor http://www.cs.wisc.edu/condor/
- CUMULVS http://www.epm.ornl.gov/cs/
- WebFlow http://www.npac.syr.edu/users/gcf/
- NGC http://www.nordicgrid.net

University of Tennessee Deployment:
Scalable Intracampus Research Grid: SInRG



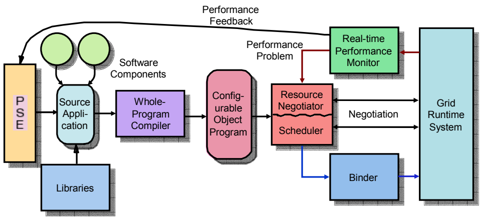
The Knoxville Campus has two DS-3 commodity Internet connections and one DS-3 Internet2/Abilene connection. An OC-3 ATM link routes IP traffic between the Knoxville campus, National Transportation Research Center, and Oak Ridge National Laboratory. UT participates in several national networking initiatives including Internet2 (I2), Abilene, the federal Next Generation Internet (NGI) initiative, Southern Universities Research Association (SURA) Regional Information Infrastructure (RII), and Southern Crossroads (SoX).

The UT campus consists of a meshed ATM OC-12 being migrated over to switched Gigabit by early 2002.

- Federated Ownership: CS, Chem Eng., Medical School, Computational Ecology, El. Eng.
- Real applications, middleware development, logistical networking

NSF/NGS
GrADS - GrADSoft Architecture

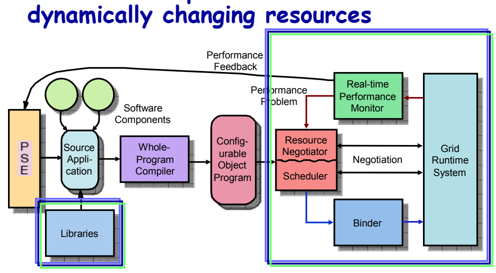
- Goal: reliable performance on dynamically changing resources



PIs: Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, JD, Ian Foster, Lennart Johnsson, Dan Reed, Carl Kesselman, John Mellor-Crummey, Linda Torczon & Rich Wolski

NSF/NGS
GrADS - GrADSoft Architecture

- Goal: reliable performance on dynamically changing resources

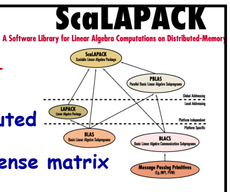


PIs: Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, JD, Ian Foster, Lennart Johnsson, Dan Reed, Carl Kesselman, John Mellor-Crummey, Linda Torczon & Rich Wolski

Major Challenge - Adaptivity

- These characteristics have major implications for applications that require performance guarantees.
- Adaptivity is a key so applications can function appropriately...
  - as resource utilization and availability change,
  - as processors and networks fail,
  - as old components are retired,
  - as new systems are added, and
  - as both software and hardware on existing systems are updated and modified.

ScaLAPACK



- ScaLAPACK is a portable distributed memory numerical library
- Complete numerical library for dense matrix computations
- Designed for distributed parallel computing (MPP & Clusters) using MPI
- One of the first math software packages to do this
- Numerical software that will work on a heterogeneous platform
- Funding from DOE, NSF, and DARPA
- In use today by IBM, HP-Convex, Fujitsu, NEC, Sun, SGI, Cray, NAG, IMSL, …
  - Tailor performance & provide support

To Use ScaLAPACK a User Must:

- Download the package and auxiliary packages (like PBLAS, BLAS, BLACS, & MPI) to the machines.
- Write a SPMD program which
  - Sets up the logical 2-D process grid
  - Places the data on the logical process grid
  - Calls the numerical library routine in a SPMD fashion
  - Collects the solution after the library routine finishes
- The user must allocate the processors and decide the number of processes the application will run on
- The user must start the application
  - "mpirun –np *N* user_app"
    - Note: the number of processors is fixed by the user before the run, if problem size changes dynamically …
- Upon completion, return the processors to the pool of resources

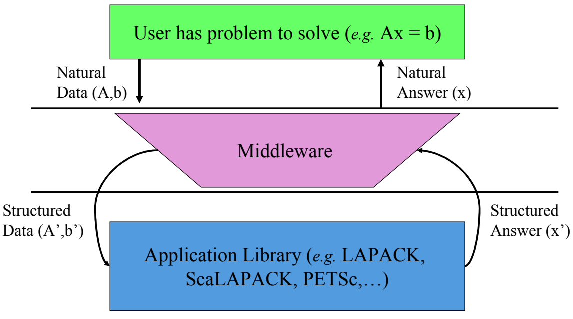
ScaLAPACK Grid Enabled

- Implement a version of a ScaLAPACK library routine that runs on the Grid.
  - Make use of resources at the user's disposal
  - Provide the best time to solution
  - Proceed without the user's involvement
- Make as few changes as possible to the numerical software.
- Assumption is that the user is already "Grid enabled" and runs a program that contacts the execution environment to determine where the execution should take place.

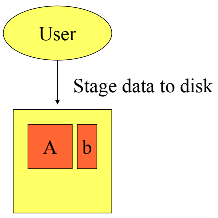
GrADS Numerical Library

- Want to relieve the user of some of the tasks
- Make decisions on which machines to use based on the user's problem and the state of the system
  - Determinate machines that can be used
  - Optimize for the best time to solution
  - Distribute the data on the processors and collections of results
  - Start the SPMD library routine on all the platforms
  - Check to see if the computation is proceeding as planned
    - If not perhaps migrate application

Big Picture…

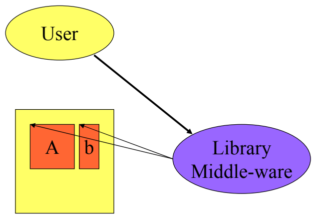


Numerical Libraries for Grids / Clusters



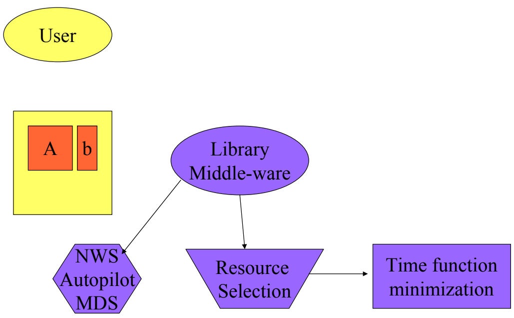
25

Numerical Libraries for Grids / Clusters



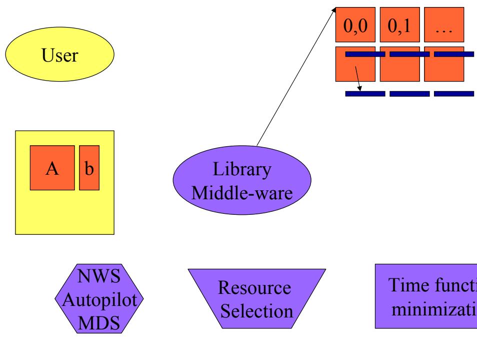
26

Numerical Libraries for Grids / Clusters



27

Numerical Libraries for Grids / Clusters

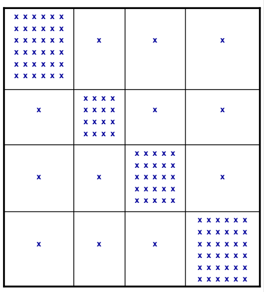


Uses Grid infrastructure, i.e.Globus/NWS/AutoPilot.

28

Resource Selector Input

- Clique based
  - 2 @ UT, UCSD, UIUC
    - Part of the MacroGrid
  - Full at the cluster level and the connections (clique leaders)
  - Bandwidth and Latency information looks like this.
  - Linear arrays for CPU and Memory
- Matrix of values are filled out to generate a complete, dense, matrix of values.
- At this point have a workable coarse grid.
  - Know what is available, the connections, and the power of the machines



29

ScaLAPACK Performance Model

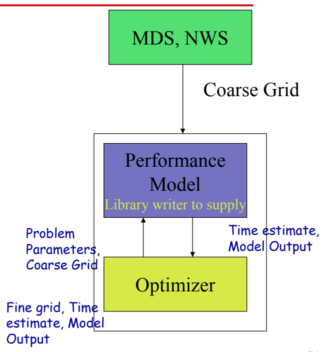
$$T(n,p) = C_f t_f + C_v t_v + C_m t_m$$

- $C_f = \frac{2n^3}{3p}$ Total number of floating-point operations per processor
- $C_v = (3 + \frac{1}{4}\log_2 p)\frac{n^2}{\sqrt{p}}$ Total number of data items communicated per processor
- $C_m = n(6 + \log_2 p)$ Total number of messages
- t_f Time per floating point operation
- t_v Time per data item communicated
- t_m Time per message

30

Resource Selector/Performance Modeler

- Refines the course grid by determining the process set that will provide the best time to solution.
- This is based on dynamic information from the grid and the routines performance model.
- The PM does a simulation of the actual application using the information from the RS.
  - It literally runs the program without doing the computation or data movement.
- There is no backtracking in the Optimizer.
  - This is an area for enhancement and experimentation.
- Simulated annealing used as well



Performance Model Validation

	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
mem(MB)	215	214	227	215	233	479	479
speed	270	270	270	270	330	330	330
load	1	0.99	1	0.99	1	1.04	0.87

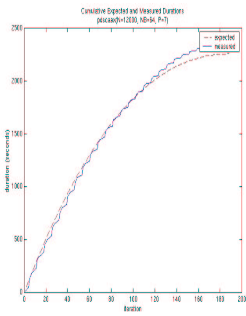
Speed = 60% of the peak

Latency	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
Opus14	-1	0.24	0.29	0.26	83.78	83.78	83.78
Opus13	0.24	-1	0.24	0.23	83.78	83.78	83.78
Opus16	0.29	0.24	-1	0.23	83.78	83.78	83.78
Opus15	0.26	0.23	0.23	-1	83.78	83.78	83.78
Torc4	83.78	83.78	83.78	83.78	-1	0.31	0.31
Torc6	83.78	83.78	83.78	83.78	0.31	-1	0.31
Torc7	83.78	83.78	83.78	83.78	0.31	0.31	-1

Latency in msec

Bandwidth	Opus14	Opus13	Opus16	Opus15	Torc4	Torc6	Torc7
Opus14	-1	248.83	247.31	246.38	2.83	2.83	2.83
Opus13	248.83	-1	244.54	240.94	2.83	2.83	2.83
Opus16	247.31	244.54	-1	247.54	2.83	2.83	2.83
Opus15	246.38	240.94	247.54	-1	2.83	2.83	2.83
Torc4	2.83	2.83	2.83	2.83	-1	81.96	56.47
Torc6	2.83	2.83	2.83	2.83	81.96	-1	50.9
Torc7	2.83	2.83	2.83	2.83	56.47	50.9	-1

Bandwidth in Mb/s



This is for a refined grid

Experimental Hardware / Software Grid

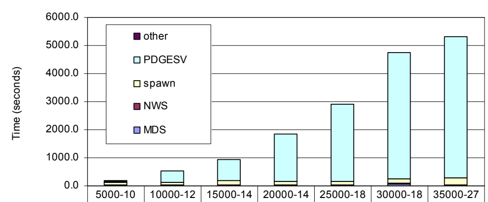
MacroGrid Testbed	TORC	CYPHER	OPUS
Type	Cluster 8 Dual Pentium III	Cluster 16 Dual Pentium III	Cluster 8 Pentium II
OS	Red Hat Linux 2.2.15 SMP	Debian Linux 2.2.17 SMP	Red Hat Linux 2.2.16
Memory	512 MB	512 MB	128 or 256 MB
CPU speed	550 MHz	500 MHz	265 – 448 MHz
Network	Fast Ethernet (100 Mbit/s) (3Com 3C905B) and switch (BayStack 350T) with 16 ports	Gigabit Ethernet (SK-9843) and switch (Foundry FastIron II) with 24 ports	Myrinet (LANai 4.3) with 16 ports each

- Globus version 1.1.3
- Autopilot version 2.3
- NWS version 2.0.pre2
- MPICH-G version 1.1.2
- ScaLAPACK version 1.6
- ATLAS/BLAS version 3.0.2
- BLACS version 1.1
- PAPI version 1.1.5
- GrADS' "Crafted code"

Independent components being put together and interacting

PDGESV Time Breakdown

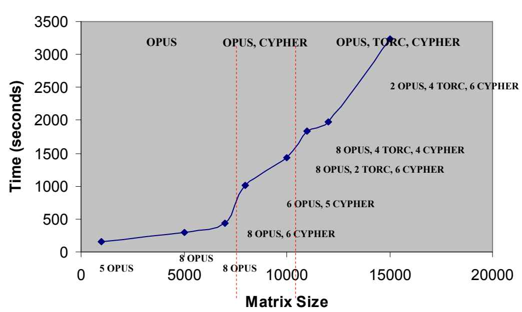
ScaLAPACK - PDGESV - Using collapsed NWS query from UCSB
42 machine available, using mainly torc, cypher, msc clusters at UTK [Jan 2002]



	5000-10	10000-12	15000-14	20000-14	25000-18	30000-18	35000-27
other	4.7	5.3	1.0	2.3	7.6	1.1	4.7
PDGESV	58.7	394.7	749.4	1686.7	2747.1	4472.7	5020.4
spawn	92.2	105.9	154.1	124.7	135.6	181.0	264.5
NWS	5.5	7.4	12.3	12.0	4.2	10.2	8.5
MDS	13.0	11.0	10.0	11.0	14.0	73.0	12.0

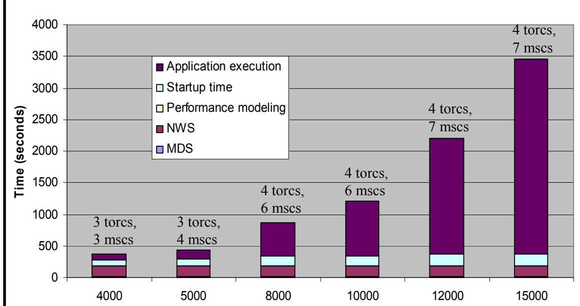
Matrix Size - Nproc

ScaLAPACK across 3 Clusters

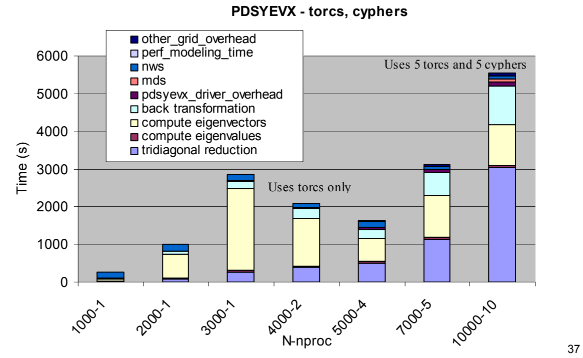


QR – Timing Breakdown

Execution of QR factorization over the grid



PDSYEVX – Timing Breakdown



Check Point / Restart Library

- **SRS (Stop-ReStart) – A user-level check pointing library that allows reconfiguration of applications.**
- **Reconfiguration of number of processors and / or data distribution.**

Two kinds of rescheduling.

- Rescheduling on request –migrate on increase in the load on executing machines (violates performance contract).
- Opportunistic rescheduling – migrate to utilize newly available machines.

Related Work

- ***CUMULVS from ORNL.***
- ***Involves checkpointing.***
- ***API looks very similar to SRS.***

CUMULVS	*SRS*
♦*A focus on computational steering and remote visualization.* ♦*Designed for PVM applications, but can be used with MPI.* ♦*Does not deal with general reconfiguration of application.*	♦*Simple motivation to be able to stop and continue parallel applications.* ♦*Designed for MPI applications in a Grid setting.* ♦*Deals with reconfiguration of application.*

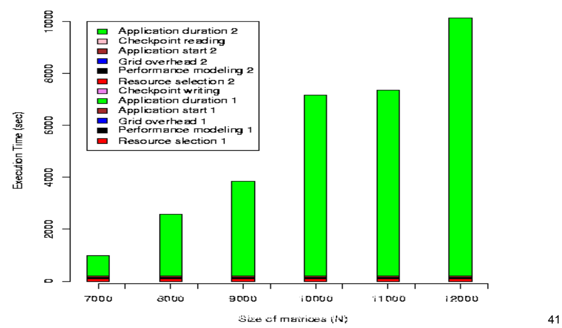
Experimental Setup

Machine name	Location	Processor	Speed (MHz)	Memory (MByte)	Network
msc	U. of Tennessee	4 proc Pentium III	933	512	100 Mb switched Ethernet
opus	U. of Illinois	8 proc Pentium III	450	256	1.28 Gb full duplex myrinet

- Problem used – ScaLAPACK QR factorization.
- Mechanisms to predict the execution time of the application on a given set of resources using a performance model.
- Also to predict the remaining execution time of an executing application.
- During execution if run violates the predicted execution time consider migrations.
- All under program control.

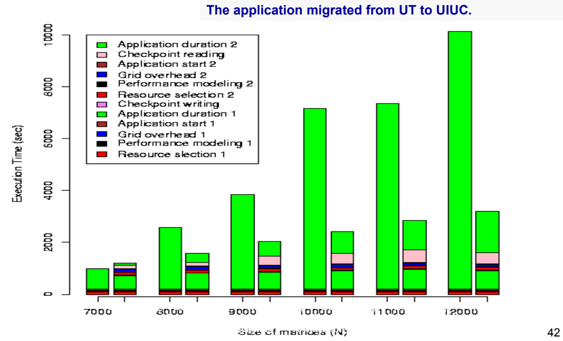
Rescheduling Results for QR Factorization

Performance model forced chose 4 UT machines. 5 minutes into the run, artificial load is introduced on 4 UT machines.



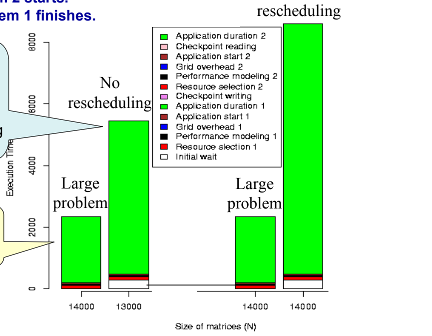
Rescheduling Results for QR Factorization

Performance model forced chose 4 UT machines. 5 minutes into the run, artificial load is introduced on 4 UT machines. Performance model violation occurs. The application migrated from UT to UIUC.

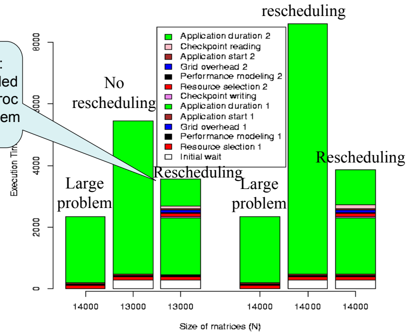


Opportunistic Rescheduling Results

Problem 1 started and 2 minutes into the run Problem 2 starts.
Then Problem 1 finishes.



Opportunistic Rescheduling Results



Research Directions

- Parameterizable libraries
- Fault tolerant algorithms
- Annotated libraries
- Hierarchical algorithm libraries
- "Grid" (network) enabled strategies

A new division of labor between compiler writers, library writers, and algorithm developers and application developers will emerge.

Conclusion

- Exciting time to be in scientific computing
- Grid computing is here
- The Grid offers tremendous opportunities for collaboration
- Important to develop algorithms and software that will work effectively in this environment

Collaborators

- TOP500
  - H. Mauer, Mannheim U
  - H. Simon, NERSC
  - E. Strohmaier, NERSC
- GrADS
  - Sathish Vadhiyar, UTK
  - Asim YarKhan, UTK
  - Ken Kennedy, Fran Berman, Andrew Chein, Keith Cooper, Ian Foster, Carl Kesselman, Lennart Johnsson, Dan Reed, Linda Torczon, & Rich Wolski

Many opportunities within the ICL at Tennessee