

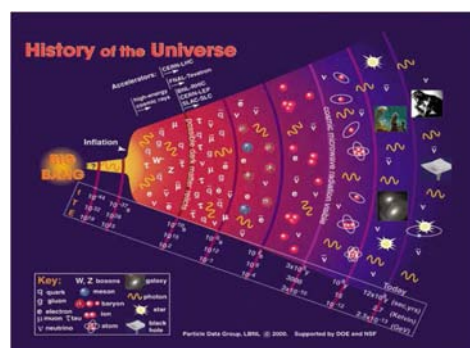
X-Scale-NL : Numerical libraries enabling scientific applications on petascale machines

The future petascale machines, formed by thousands of multi-core processors, will provide a computing power that will allow significant advances for many applications based on numerical simulations. However the efficient usage of a petascale machine depends on the ability of an application to fully utilize it.

Our objectives is to develop numerical libraries algorithms, programming and scheduling techniques that will allow applications to take advantages of the architectural trends of the multi-core processors. We focus on the resolution of sparse linear systems and our algorithm needs to be robust and efficient.

Example of computer-intensif effort in astrophysics in CMB (adapted from J. Borrill, LBNL)

CMB photons are the light produced during the early hot stage of Universe evolution. The quest for more detailed characterization of the primordial CMB signal requires large data volumes, forcing the astrophysical data sizes to follow Moore's-like law.



Estimation of computing needs for experiments in CMB (J. Borrill, LBNL)

Experiment	Date	Time Samples	Sky Pixels	Gflop/Map	Total Cost [Tflop]
COBE	1989	10^9	10^3	1	1
BOOMERanG	2000	10^9	10^5	10^3	10^3
WMAP	2001	10^{10}	10^6	10^4	10^4
Planck	2008	10^{11}	10^7	10^5	10^5
PolarBear	2010	10^{12}	10^6	10^6	10^6
CMBpol	2020+	10^{14}	10^8	10^8	10^8

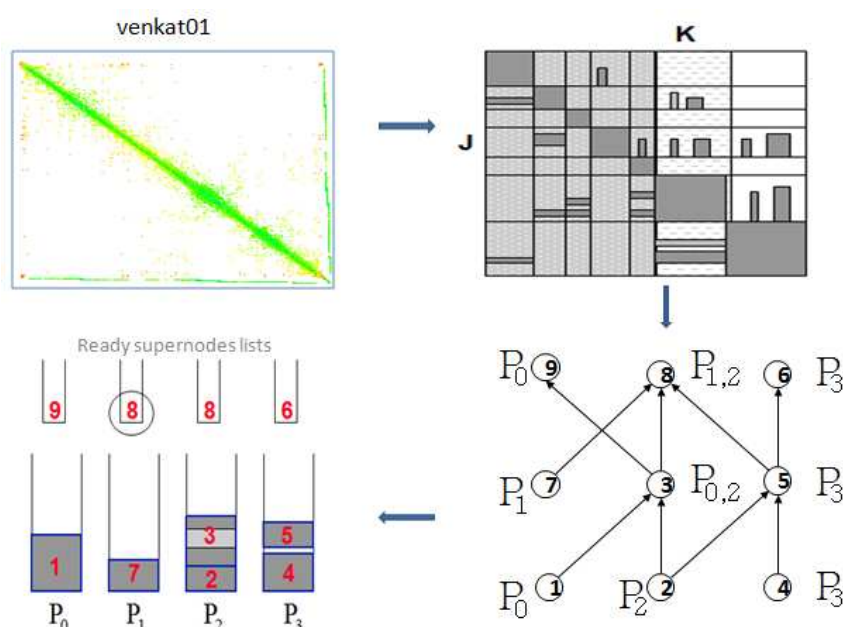
Introduction

- Solving linear system of equations $Ax = b$ lies at the heart of many problem in scientific applications.
- Computers are getting faster, their technology makes current algorithms less and less efficient.
- The recent evolution in computing lead to multi-cores and new algorithms need to be designed to exploit their architectural trends.

Methods

Our method is based on four main steps:

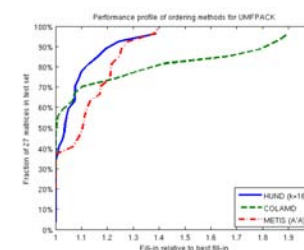
- **Ordering:** The parallel algorithm for the sparse LU factorization is based on HUND, a hypergraph-based unsymmetric nested dissection ordering algorithm for reducing the fill-in.
- **Pivoting:** The strategy to maintain stability is a Communication Avoiding algorithm for the LU factorization (CALU) of dense and sparse matrices distributed in a two-dimensional cyclic layout.
- **Programming techniques:** The parallel algorithms will be developed for the hierarchical models of clusters of SMP multi-core microprocessors.
- **Scheduling techniques:** Multi-core processors introduce several difficulties that have to be taken into account at the software level. To achieve our objectives, scheduling techniques aim at identifying large number of fine grain tasks in the factorization process that should be executed asynchronously and avoid bus contention.



Results of the first study (HUND)

Use combinatorial tools to improve parallelism or exploit asymmetry of the matrix

- Hypergraph based unsymmetric nested dissection ordering (HUND) (with E. Boman - Sandia, T. Davis - U. Florida)
- HUND is close or outperforms the best algorithm among COLAMD, MMD, METIS (tech report to appear).

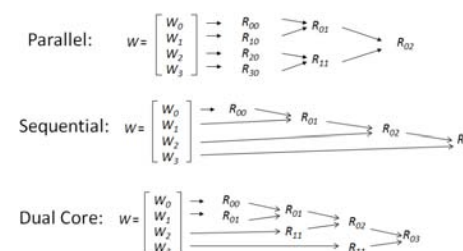


Performances HUND.

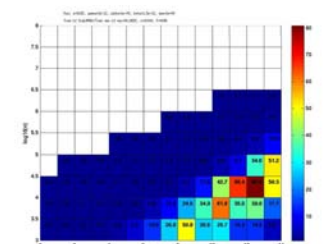
Results of the first study (Pivoting)

Use communication avoiding algorithms (with J. Demmel - Berkeley, J. Langou - Denver)

- New algorithms for LU, QR factorizations of dense matrices, that are communication optimal.
- They lead to an important decrease in the number of messages compared to ScaLAPACK.



	IBM Power 5
TSLU	Up to 4.37x faster (16 procs, 1M x 150)
CALU	Up to 2.29x faster (64 procs, 1000 x 1000)
	Cray XT4
TSLU	Up to 5.52x faster (8 procs, 1M x 150)
CALU	Up to 1.81x faster (64 procs, 1000 x 1000)



Performance vs ScaLAPACK and speedup prediction for a Petascale machine - up to 81x.