Computer Science/Math Challenges Related to Nano-Technology Applications

Stanimire Tomov

Innovative Computing Laboratory (ICL)
The University of Tennessee

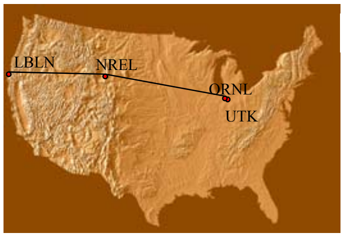
CScADS Workshop: **Libraries and Algorithms for Petascale Applications**
July 30th – August 2nd, 2007
Snowbird, Utah

(DOE-NANO project, supported by **U.S. DOE, Office of Science**)

CS/Math challenges as related to:

- Project: “**Predicting the Electronic Properties of 3D Million-Atom Semiconductor Nanostructure Architectures**”

Supported by: **U.S. DOE, Office of Science**



Materials Science Center, NREL
Alex Zunger, A. Franceschetti, G. Bester
Scientific Computing Center, NREL
W. Jones, Kwiseon Kim, P. Graf

Computational Research Division, LBNL
Lin-Wang Wang, A. Canning, O. Marques, C. Vomel

Dept. of CS, University of Tennessee
Jack Dongarra, Stan Tomov, Julien Langou

Outline

- Background
  - Simulation of nano materials and devices
  - Challenges of future architectures
- Electronic structure calculations
  - Density Functional Theory (DFT)
  - Potentials, Basis selection, etc
- CS/Math Challenges
  - Iterative eigensolvers
  - Preconditioners
  - Kernels optimization
  - Research on new or improved algorithms
- Conclusions

Electronic properties of nano-structures

- Semiconductor Quantum dots (QDs)
  - **Tiny** crystals ranging from a few hundred to few thousand atoms in size; made by humans

  At these small sizes electronic properties critically depend on **shape** and **size**

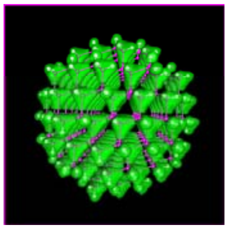
  ⇒ electronic properties can be tuned

  ⇒ enables remarkable applications

  The dependence is quantum mechanical in nature and can be modelled

  - can not be done on macroscopic scales

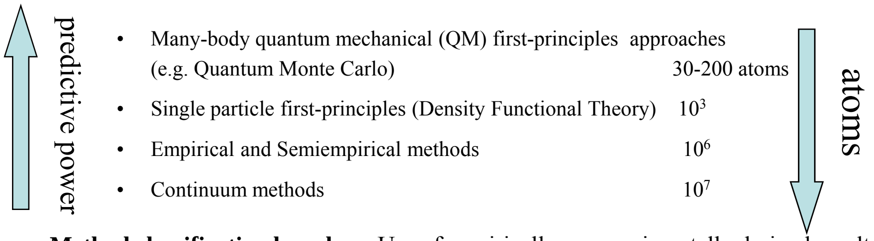
  - has to be at **atomic and subatomic level (nanoscale)**
- Quantum wires (QWs) and devices
  - their conducting properties are affected by build-in nano-materials



Total electron charge density of a quantum dot of gallium arsenide, containing just 465 atoms.

Quantum dots of the same material but different sizes have different band gaps and emit different colors

Nano Materials Simulations



- Many-body quantum mechanical (QM) first-principles approaches (e.g. Quantum Monte Carlo) 30-200 atoms
- Single particle first-principles (Density Functional Theory) 10^3
- Empirical and Semiempirical methods 10^6
- Continuum methods 10^7

- **Method classification based on:** Use of empirically or experimentally derived results
  YES ⇒ empirical or semi-empirical methods
  NO ⇒ ab initio (very accurate; most predictive power; but scales as $O(N^{3-7})$)
- Major petascale computing challenges:
  - Algorithms with reduced scaling; architecture aware (next ...)
  - Highly parallelizable (100s of 1,000s of cores)
    - typical basis functions here (plane-wave basis) have global support

Challenges of Future Architectures

- Parallel computing – not just for HPC architectures but for simple desktops
  - In a few years desktops expected to have 32 cores per multicore processor chip and 128 hardware threads per chip
- Gap between processor and memory speed continue to grow (exponentially)
  - Processor speed improves 59%, memory bandwidth 23%, latency 5.5%

  ⇒ Many familiar and widely used algorithms and libraries have to be rewritten to be able to exploit the power of these new generation architectures
- Petaflop by 2010: DARPA's HPCS program in phase 3, supporting
  - Cray with the Cascade system (with Chapel HPL) / adaptive supercomputing
    - parallelism trough various processor technologies: scalar, vector, multithreading and hardware accelerators (FPGA or ClearSpeed co-processors)
  - IBM with PERCS system (with X10 HPL) / larger SMPs with more memory

Electronic structure calculations

- Density functional theory

Many-body Schrödinger equation (exact but exponential scaling)

$$\{-\sum_i \frac{1}{2}\nabla_i^2 + \sum_{i,j}\frac{1}{|r_i - r_j|} + \sum_{i,I}\frac{Z}{|r_i - R_I|}\}\Psi(r_1,..r_N) = E\Psi(r_1,..r_N)$$

- Nuclei fixed, generating external potential (system dependent, non-trivial)
- N is number of electrons

Kohn Sham Equation: The many body problem of interacting electrons is reduced to non-interacting electrons (single particle problem) with the same electron density and a different effective potential (cubic scaling).

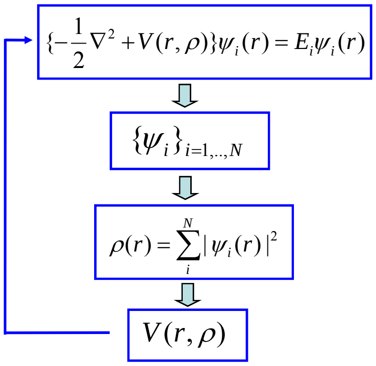
$$\{-\frac{1}{2}\nabla^2 + \int\frac{\rho(r')}{|r-r'|}dr' + \sum_I \frac{Z}{|r-R_I|} + V_{XC}\}\psi_i(r) = E_i\psi_i(r)$$

$$\rho(r) = \sum_i |\psi_i(r)|^2 = |\Psi(r_1,..r_N)|^2$$

- V_{XC} represents effects of the Coulomb interactions between electrons
- ρ is the density (of the original many-body system

V_{XC} is not known except special cases ☹ use approximation, e.g. Local Density Approximation (LDA) where V_{XC} depends only on ρ

Selfconsistent calculation



$$\{-\frac{1}{2}\nabla^2 + V(r,\rho)\}\psi_i(r) = E_i\psi_i(r)$$

$$\{\psi_i\}_{i=1,..,N}$$

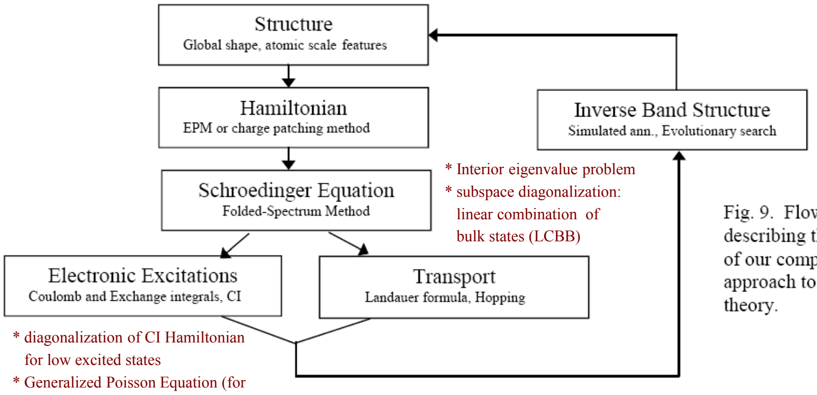
$$\rho(r) = \sum_i^N |\psi_i(r)|^2$$

$$V(r,\rho)$$

N electrons
N wave functions
lowest N eigenfunctions

- Requires diagonalization and/or orthogonalization
- Scales as $O(N^3)$ and may be prohibitively high
- Work on new algorithms with reduced scaling
  (the need to know more physics and interact with physicists)
- There are for example O(N) algorithms to find directly the total energy

Computational framework


Structure
Global shape, atomic scale features
Hamiltonian
EPM or charge patching method
Schroedinger Equation
Folded-Spectrum Method
Electronic Excitations
Coulomb and Exchange integrals, CI
Transport
Landauer formula, Hopping
Inverse Band Structure
Simulated ann., Evolutionary search

* Interior eigenvalue problem

* subspace diagonalization: linear combination of bulk states (LCBB)

* diagonalization of CI Hamiltonian for low excited states

* Generalized Poisson Equation (for electric field needed in CI for the many-body problem

Fig. 9. Flowchart describing the structure of our computational approach to nanostructure theory.

Basis selection

- **Plane-waves, grid functions, or Gaussian orbitals**
- Plane-waves: $\psi_{nk}(r) = \sum_{g,|g|<E_{cut}} C_g^n(k) e^{i(g+k).r}$
  - Good approximation properties
  - Can be preconditioned easily (and efficiently) as the kinetic energy (the laplacian) is diagonal in Fourier space, the potential is diagonal in real space
  - Usually codes are in Fourier space and go back and forth to real with FFTs
  - Concern may be scalability of FFT on 100s of 1,000s of processors as it requires global communication
- Grid functions: e.g. finite elements, grids, or wavelets
  - Domain decomposition techniques can guarantee scalability for large enough problems
  - Interesting as they enable algebraically based preconditioners as well
  - Including multigrid/multiscale
    - e.g. real-space multigrid methods (RMG) by J. Bernholc et al (NCSU)

Libraries

- Use state-of-the-art libraries whenever possible, extend if our particular problems present opportunities for improvement
- We use the Nanoscience Problem Solving Environment (**NanoPSE**) package
  - Integrate various nano-codes (developed over ~12 years)
  - Its design goal: provide a software context for collaboration
    - Features easy install; runs on many platforms, etc.
- LAPACK, ScaLAPACK, BLAS
- PRIMME package (A. Stathopoulos and J. McCombs)
- P_ARPACK (R. Lehoucq, K. Maschhoff, D. Sorensen, C. Yang)

ICL UT INNOVATIVE COMPUTING LABORATORY THE UNIVERSITY of TENNESSEE Computer Science Department

FFT

Problem	P	NERSC (Power3)		Jacquard (Opteron)		Thunder (Itanium2)		ORNLCray (X1)		NEC ES (SX6*)		NEC SX8	
		Gflops/P	%peak	Gflops/P	%peak	Gflops/P	%peak	Gflops/P	%peak	Gflops/P	%peak	Gflops/P	%peak
488 Atom CdSe Quantum Dot	128	0.93	62%			2.8	51%	3.2	25%	5.1	64%	7.5	47%
	256	0.85	57%	1.98	45%	2.6	47%	3.0	24%	5.0	62%	6.8	43%
	512	0.73	49%	0.95	21%	2.4	44%	▨	▨	4.4	55%	▨	▨
	1024	0.60	40%			1.8	32%	▨	▨	3.6	46%	▨	▨

– *** Load Balance Sphere by giving columns to different procs.**
*** 3D FFT done via 3 sets of 1D FFTs and 2 transposes**
*** Flops/Comms ~ logN**
*** Many FFTs done at the same time to avoid latency issues**
*** Only non-zero elements communicated/calculated**
*** Much faster than vendor supplied 3D-FFT**

(from A. Canning (LBNL), work on PARATEC)

ICL UT INNOVATIVE COMPUTING LABORATORY THE UNIVERSITY of TENNESSEE Computer Science Department

Interior Eigenvalue Problem Formulation

- Solve a single particle Schrödinger-type equation

(E) $\mathbf{H\ \Psi_i}$ ♋ $\mathbf{[-0.5\ \Delta + V]\ \Psi_i = \varepsilon_i\ \Psi_i}$

with periodic boundary conditions

- Physical interpretation
  - The Hamiltonian H represents the total energy
    - Laplacian Δ corresponds to kinetic energy of the electrons
    - V is the potential energy; describes the atomic configuration of the systems; precomputed or from experiment
  - Real eigenvalue ε_i is discrete energy level of electron (occupied or not)
  - Complex eigenvector Ψ_i is probability distribution for spacial location of electron

Interior Eigenvalue Problem Formulation

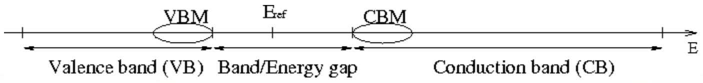
- Basis functions (**Bloch theorem** about the eigenstates of Hamiltonian H with periodic potential V)

$$\psi_{nk}(r) = \sum_{g,|g|<E_{cut}} C_g^n(k) e^{i(g+k).r}$$

- Leads to a discrete eigenvalue problem

$\mathbf{H\ \Psi_i = E_i\ \Psi_i}$, where H is Hermitian

- Properties of H
  - Complex Hermitian indefinite
  - Implicitly defined by M-V product (uses 3D FFT)
  - Eigenvalues with higher multiplicities (to be expected of up to 4)
- Find a few (4-10) interior eigenvalues closest to a given point E_{ref}



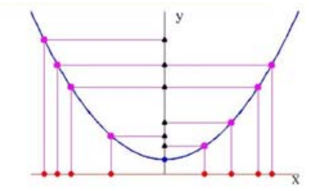
Iterative eigensolvers

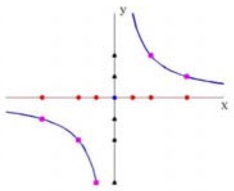
- Based on **local projections**, e.g.
  Solving **Ax = λx** in R^n iteratively:
  * at iteration **i** extract an approximate $\mathbf{x_i}$ from a subspace **V = span[$\mathbf{v_1}$, ..., $\mathbf{v_m}$]** of R^n
  * impose Galerkin constraints:
  $λx – Ax ⊥$ subspace $W = span[w_1,...,w_m]$ of R^n, i.e.
  $w^* A x_i = λ w^* x_i,$ for ∀ w ∈W= $span[w_1,...,w_m]$
- This procedure is also known as **Rayleigh-Ritz**
- In Matrix notations: Let $V = [v_1, ..., v_m]$, $W = [w_1,...,w_m]$
  * Find y ∈R^m s.t. $\mathbf{x_i = V\ y}$ solves
  $\mathbf{W^TA\ V\ y = λW^T Vy}$ (with LAPACK)

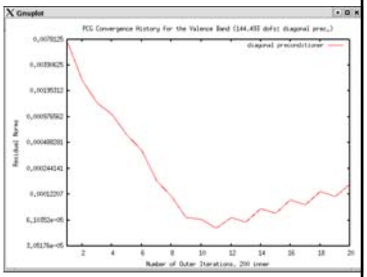
Need special attention on petascale architectures as it has "sequential" part

- The choice for V and W is crucial and determines various methods
  – Setting various parameters is non trivial

Spectral transformations







- Folded spectrum
  Ax = λx →
  $(A-E_{ref}I)^2\ x = \mu\ x$
  ***clustering* of eigenvalues**

- Shift and invert
  Ax = λx →
  $(A-E_{ref}I)^{-1}\ x = \mu\ x,\ \ E_{ref} \neq \lambda$
  need to *invert (inner iteration)*

Convergence **stagnation** comp. the valence band on a 1,523 atoms CdSe QD (with folded spectrum)

- Convergence of ith smallest eigenstate of CG depends on the ratio $\dfrac{x_{i+1} - x_i}{x_{\max} - x_{\min}}$

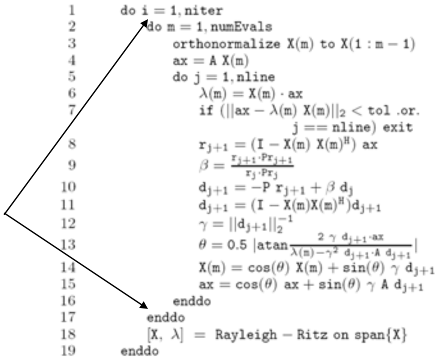
Iterative eigensolvers

- We studied several eigensolvers on our problems
  - Preconditioned conjugate gradient (PCG) from PESCAN, part of **NanoPSE**
  - **Block PCG (BEPCG)**
  - Implicitly restarted Arnoldi/Lanczos from P_ARPACK
  - Generalized Davidson (GD) with restart and Jacobi-Davidson with QMR as inner solver (JDQMR) from PRIMME
  - Locally optimal block preconditioned conjugate gradient (LOBPCG); own implementation

PCG eigensolver

- Have been successfully used in the field
- PCG extended to a subspace method
  - Band-by-band inner-outer iteration

do i=1,niter
[X] = state by state CG-type minimization of the Rayleigh functional (with deflation)
[X, λ] = Rayleigh-Ritz on span{X}
enddo



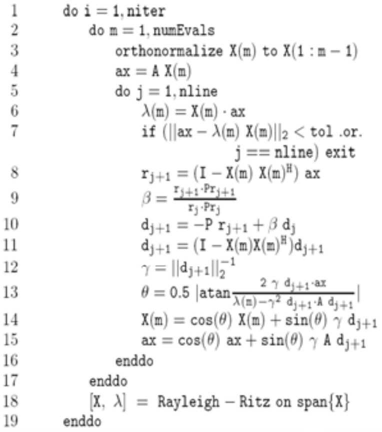
```
do $i = 1, niter$
    do $m = 1, numEvals$
        orthonormalize $X(m)$ to $X(1:m-1)$
        $ax = A\ X(m)$
        do $j = 1, nline$
            $\lambda(m) = X(m) \cdot ax$
            if $(||ax - \lambda(m)\ X(m)||_2 < tol$ .or.
                        $j == nline)$ exit
            $r_{j+1} = (I - X(m)\ X(m)^H)\ ax$
            $\beta = \frac{r_{j+1} \cdot P r_{j+1}}{r_j \cdot P r_j}$
            $d_{j+1} = -P\ r_{j+1} + \beta\ d_j$
            $d_{j+1} = (I - X(m) X(m)^H) d_{j+1}$
            $\gamma = ||d_{j+1}||_2^{-1}$
            $\theta = 0.5\ |atan \frac{2\ \gamma\ d_{j+1} \cdot ax}{\lambda(m) - \gamma^2\ d_{j+1} \cdot A\ d_{j+1}}|$
            $X(m) = \cos(\theta)\ X(m) + \sin(\theta)\ \gamma\ d_{j+1}$
            $ax = \cos(\theta)\ ax + \sin(\theta)\ \gamma\ A\ d_{j+1}$
        enddo
    enddo
    $[X,\ \lambda]\ =\ Rayleigh - Ritz$ on $span\{X\}$
enddo
```

Blocking

- PCG extended to a subspace method
  - Band-by-band inner-outer iteration
    - Of concern here is that the band-by-band computation uses only a fraction of the peak performance of current computer architectures
  - It is possible instead of the band-by-band updates for the eigenstates to organize the computation so that a block of eigenstates is 'simultaneously' updated (next)
    - Results in performing **Rayleigh-Ritz (RR) on larger subspaces**
      - **Can be implemented in terms of BLAS 3 operations**
      - **Can block communications and reduce latency overhead in distributed computing**
      - **Larger subspaces lead to accelerated convergence (in terms of RR iterations)**

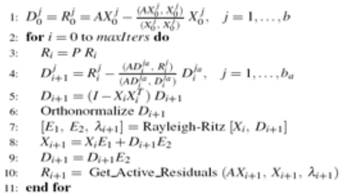
Block PCG: BEPCG and LOBPCG

Band-by-band PCG



```
do i = 1, niter
    do m = 1, numEvals
        orthonormalize X(m) to X(1 : m − 1)
        ax = A X(m)
        do j = 1, nline
            $\lambda(m) = X(m) \cdot ax$
            if ($||ax - \lambda(m)\ X(m)||_2 < tol$ .or.
                                 j == nline) exit
            $r_{j+1} = (I - X(m)\ X(m)^H)\ ax$
            $\beta = \frac{r_{j+1} \cdot P r_{j+1}}{r_j \cdot P r_j}$
            $d_{j+1} = -P\ r_{j+1} + \beta\ d_j$
            $d_{j+1} = (I - X(m)X(m)^H) d_{j+1}$
            $\gamma = ||d_{j+1}||_2^{-1}$
            $\theta = 0.5\ |atan \frac{2\ \gamma\ d_{j+1} \cdot ax}{\lambda(m) - \gamma^2\ d_{j+1} \cdot A\ d_{j+1}}|$
            $X(m) = \cos(\theta)\ X(m) + \sin(\theta)\ \gamma\ d_{j+1}$
            $ax = \cos(\theta)\ ax + \sin(\theta)\ \gamma\ A\ d_{j+1}$
        enddo
    enddo
    $[X,\ \lambda]$ = Rayleigh − Ritz on span{X}
enddo
```

BEPCG



```
$D_0^j = R_0^j = AX_0^j - \frac{(AX_0^j, X_0^j)}{(X_0^j, X_0^j)} X_0^j, \quad j = 1, \ldots, b$
for $i = 0$ to $maxIters$ do
    $R_i = P\ R_i$
    $D_{i+1}^j = R_i^j - \frac{(AD_i^{ja}, R_i^j)}{(AD_i^{ja}, D_i^{ja})} D_i^{ja}, \quad j = 1, \ldots, b_a$
    $D_{i+1} = (I - X_i X_i^T)\ D_{i+1}$
    Orthonormalize $D_{i+1}$
    $[E_1,\ E_2,\ \lambda_{i+1}]$ = Rayleigh-Ritz $[X_i,\ D_{i+1}]$
    $X_{i+1} = X_i E_1 + D_{i+1} E_2$
    $D_{i+1} = D_{i+1} E_2$
    $R_{i+1}$ = Get_Active_Residuals $(AX_{i+1},\ X_{i+1},\ \lambda_{i+1})$
end for
```

LOBPCG

```
do i=1,niter
    [R] = P (AX -  λX)
    [X, λ] = Rayleigh-Ritz on span{X, $X_{i-1}$, R}
Enddo
```

Of interest is

* if the 3rd vector in LOBPCG improve convergence

vs using 2 (current approximate and search direction) as in BEPCG

* if not, will BEPCG yield improved reliability and performance

Some results/conclusions on eigensolvers

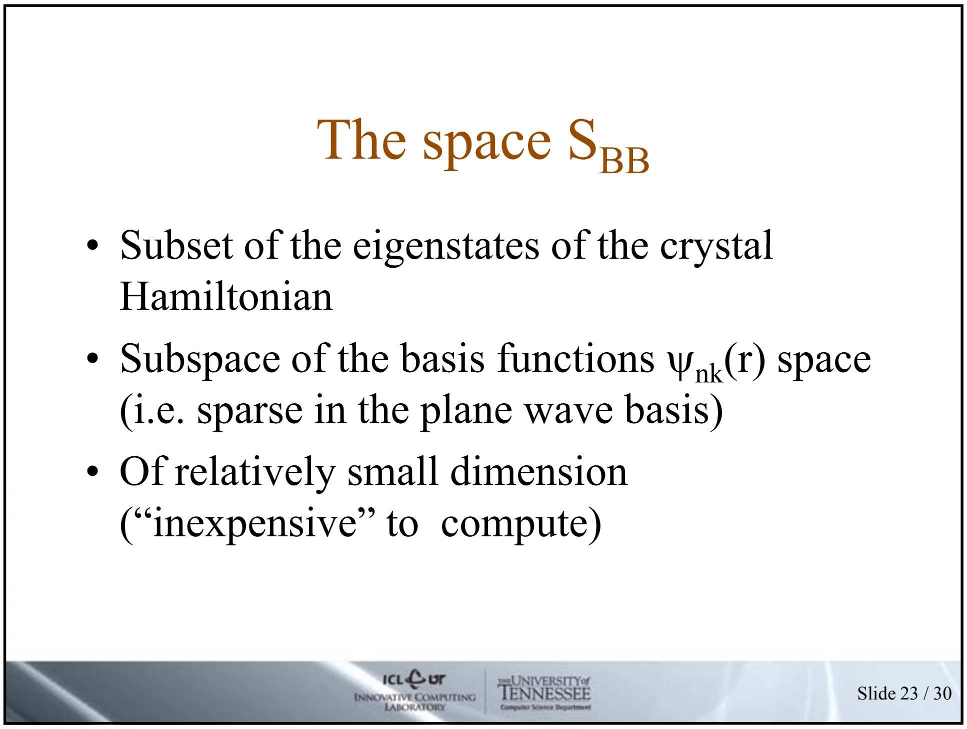
- GD+k (Olsen) turned to be very reliable and at the same time up to 5 times faster than the commonly used PCG
- PCG still useful as it requires very small amount of memory and is robust
- LOBPCG wasn't competitive with the preconditioner used (competitive without preconditioning)
- IRL was very fast for some problems but in general unreliable when used with memory comparable with the others (improved filtering may help, blocking); does not support multiple start vectors and preconditioning
- Need to explore other spectral transformations, e.g. Harmonic Ritz values
- For more substantial speedups, improved reliability, and robustness we need better preconditioners

A bulk band (BB) preconditioner

- A preconditioner based on physical intuition, example of collaboration with physicists
- Use a subset of the eigenstates of the crystal Hamiltonian (denoted as bulk band space S_{BB})
- A numerical motivation:

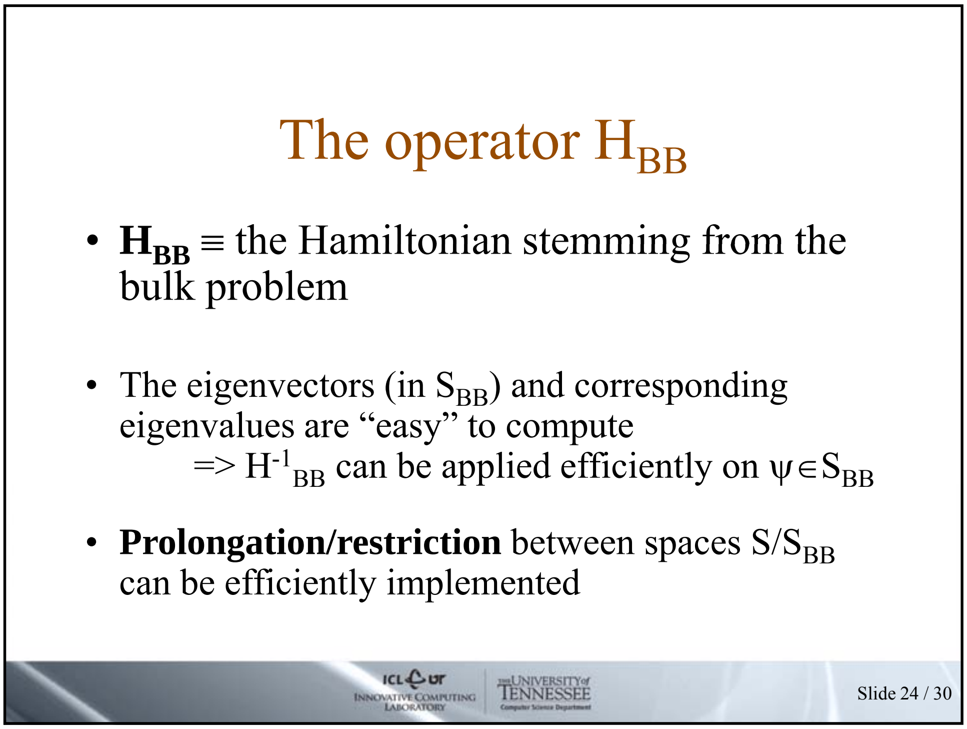
$$\psi_i = \psi_i^{S_{BB}} + \psi_i^{S_{BB}^{\perp}}$$

Angle $\angle(\psi_i, \psi_i^{S_{BB}})$ is small ($\approx 2° - 3°$)



The space S_{BB}

- Subset of the eigenstates of the crystal Hamiltonian
- Subspace of the basis functions $\psi_{nk}(r)$ space (i.e. sparse in the plane wave basis)
- Of relatively small dimension ("inexpensive" to compute)



The operator H_{BB}

- $\mathbf{H_{BB}} \equiv$ the Hamiltonian stemming from the bulk problem

- The eigenvectors (in S_{BB}) and corresponding eigenvalues are "easy" to compute
  => H^{-1}_{BB} can be applied efficiently on $\psi \in S_{BB}$

- **Prolongation/restriction** between spaces S/S_{BB} can be efficiently implemented

BB preconditioner

- Let Q the prolongation (basis embedding) from S_{BB} to S and Q^T the corresponding restriction (projection) from S to S_{BB}

- **The BB preconditioner**

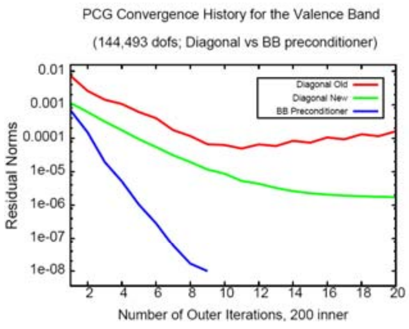
$$\mathbf{P\,R \;\equiv\; w\,Q\,H_{BB}^{-1}\,Q^T\,R + D^{-1}\,R}$$

$$w = \lambda_{max}^{-1}\,(QH_{BB}^{-1}\,Q^T\,H)$$

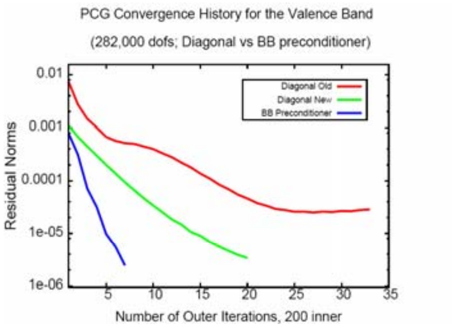
$$d_i^{-1} = \frac{E_k^2}{(0.5q_i^2 + V_0 - E_{ref})^2 + E_k^2}$$

(q_i is diagonal term for the Laplacian, V_0 the average potential, and E_k the average kinetic energy of ψ_i)

Numerical results


PCG Convergence History for the Valence Band

(144,493 dofs; Diagonal vs BB preconditioner)


PCG Convergence History for the Valence Band

(282,000 dofs; Diagonal vs BB preconditioner)

Real space methods

- Grid functions: e.g. finite elements, grids, or wavelets
  - Domain decomposition techniques can guarantee scalability for large enough problems
  - Interesting as they enable algebraically based preconditioners as well
  - Including multigrid/multiscale
    - e.g. real-space multigrid methods (RMG) by **J. Bernholc** et al (NCSU)

Concerns/challenges regarding scalability on petascale machines

- Tuning 'coarse' level operations as they have reduced computation-to-communication ratio
  * in multiscale methods and in additive Schwarz type preconditioners
- Load balancing in additive Schwarz type preconditioners

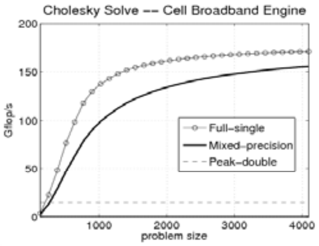
Mixed precision iterative refinement

- We have demonstrated (in a couple of papers) computational speedup in solving Ax = b (with DP accuracy) by

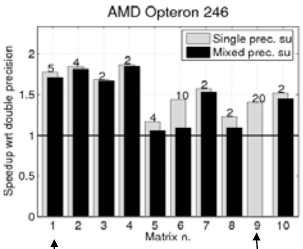
$$x_{i+1} = x_i + \boxed{P}\ (b-Ax_i)$$

Computed and applied in SP, the rest in DP

where P can be the triangular inverses of the LU factorization of A or another iterative solver (e.g. GMRES)



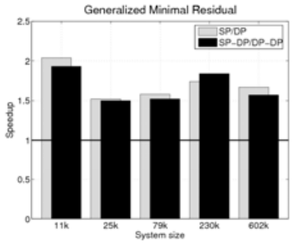
A random dense



Matrix market, sparse

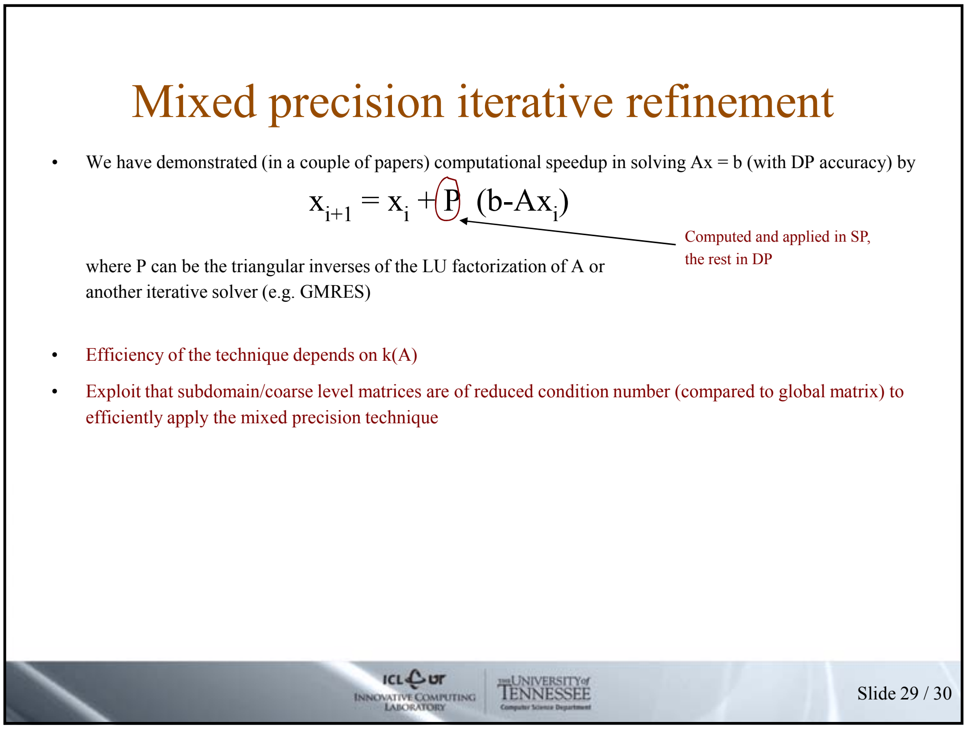
O(10^5)

O(10^9)



Elasticity, adaptive, K(A) = 10^3 .. 10^5

Dongarra / Buttari / Kurzak / Luszczek / Langou / Langou / Tomov



Mixed precision iterative refinement

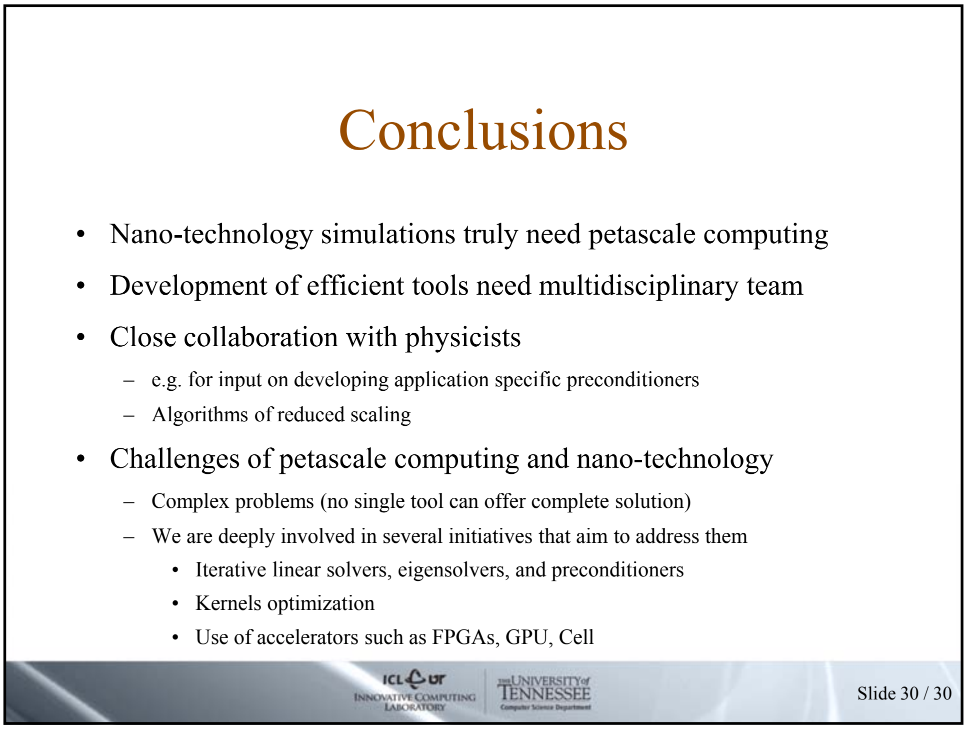
- We have demonstrated (in a couple of papers) computational speedup in solving Ax = b (with DP accuracy) by

$$x_{i+1} = x_i + P\ (b-Ax_i)$$

Computed and applied in SP, the rest in DP

where P can be the triangular inverses of the LU factorization of A or another iterative solver (e.g. GMRES)

- Efficiency of the technique depends on k(A)
- Exploit that subdomain/coarse level matrices are of reduced condition number (compared to global matrix) to efficiently apply the mixed precision technique



Conclusions

- Nano-technology simulations truly need petascale computing
- Development of efficient tools need multidisciplinary team
- Close collaboration with physicists
  - e.g. for input on developing application specific preconditioners
  - Algorithms of reduced scaling
- Challenges of petascale computing and nano-technology
  - Complex problems (no single tool can offer complete solution)
  - We are deeply involved in several initiatives that aim to address them
    - Iterative linear solvers, eigensolvers, and preconditioners
    - Kernels optimization
    - Use of accelerators such as FPGAs, GPU, Cell