Extending the scalability of electronic structure calculations by algorithms re-engineering

Alessandro Curioni
Department Manager, Mathematical & Computational Sciences
IBM Zurich Research Laboratory
Outline of the Presentation

- Introduction
  - Computational Sciences at IBM Zurich Research Laboratory
  - The need of extreme parallelism

- The CPMD code: a research tool for ab-initio MD
  - CPMD parallelization and scaleout
  - Cache/Network Optimized Orthogonalization
  - Performance on tests systems: time scales and system sizes

- The new challenges toward exascale computing
  - Fault tolerant and energy efficient algorithms

- Successful application examples

- Conclusions
IBM Research - Zurich

Since 1956

40 different nationalities

90 Collaborative projects with universities, industrial partners and governments

Two Nobel Prizes (1986 and 1987)

New Nanotech Center opened in 2011
Mathematical & Computational Science

Business Optimization
- Inventory Optimization
- Optimization under uncertainty

Data Analytics
- Operational Risk
- Customer analytics

Computational Sciences
- Simulation of complex systems
- Supercomputing applications
Computational Science at ZRL

- Developing and applying atomistic simulation techniques
  - Ab-initio Molecular Dynamics
  - Large Scale Classical Molecular Dynamics
  - Reactive Force Fields optimization and design
  - Multiscales simulations

- Applications to relevant IBM technology problems such as:
  - High-K materials
  - Defects diffusion in Si/SiO2 systems
  - Metal-CNTs contacts
  - Strain dependent transport properties of Si nanowires
  - Nano-Molecular switches

- Applications to relevant problems of key industries:
  - Enzymatic Reactions and Drug design
  - Accurate materials simulations (e.g. materials aging)
  - Energy production and Energy Storage
  - Micro – Finite Elements simulations
  - DNS Computational Fluid Dynamics

- Algorithm development and scaleout for simulation of complex systems

- Collaborations and Joint Projects with key partners such as:
  - ABB, Egypt government, Novozymes, Nestle Research, Ford Research, Mitsubishi Chemical, Akzo Nobel, Merck, Novartis,…
Real Examples

- Corrosion of Aluminum by Water
  - with Ford Research
    Science 282, 265 (1998)

- Materials for OLED or Organic Electronics

- Materials for Novel Dielectrics

- Degradation of Coffee Aroma
  - with Nestle’s Research Center
Real Examples

- a-Silicon Photovoltaics
  - Physical Review Letters 107, 255502 (2011)

- Li-Air Batteries

- Materials for High Voltage Insulators
  - with ABB Research

- Evolution and Degradation of Aerosols
  - with PMI research
Real Examples

- Quantum Refined Scoring Function for Drug Design
  - with Novartis

- Ligand-Protein interaction with QRFF
  - with Organon, Telethon Institute and S. Raffaele Hospital

- QM/MM Modeling of Enzymatic Reactions
Real Examples

- DNS – CFD Simulations of Aircraft Trailing Vortices
  - with ETH – Prof. Koumoutsakos

- μ-FEM Simulations of Bone Structures
  - with ETH – Prof. Mueller
The Origin of the Problem: End of CMOS Real Scaling

Most of the exponential increase in supercomputer speed is due to massively parallelism:

1 processor in 1990 - ~300'000 in 2010
The Spectrum of Atomistics Simulations

Computational Complexity and the number of Algorithmic Dwarfs playing a key role increase with the accuracy of the methods used.
BlueGene Evolution:

Goals:
- Three orders of magnitude performance in 10 years
- Push state of the art in Power efficiency, scalability, & reliability
- Enable unprecedented application capability

Performance

Blue Gene / P
PPC 450 @850MHz
1+ PF

Blue Gene / Q
In progress
~50PF

Goals:
- Converge HPC Arch. & Development streams
- Lay the ground work for Exascale & usability
- Address many of the power efficiency, reliability and technology challenges

2004 2008 2012 2016 2020

1EF
Outline of the Presentation

- Introduction
  - Computational Sciences at IBM Zurich Research Laboratory
  - The need of extreme parallelism

- The CPMD code: a research tool for ab-initio MD
  - CPMD parallelization and scaleout
  - Cache/Network Optimized Orthogonalization
  - Performance on tests systems: time scales and system sizes

- The new challenges toward exascale computing
  - Fault tolerant and energy efficient algorithms

- Successful application examples

- Conclusions
CPMD code history

- Born at **IBM Zurich** from the original Car-Parrinello Code in 1993 (J. Hutter) - implement DFT in the plane waves / pseudopotentials framework;

- developed in many other sites during the years; it has many unique features, e.g. path-integral MD, QM/MM interfaces, TD-DFT, LR calculations, metadynamics;

- since 2001 distributed free for academic institutions ([www.cpmd.org](http://www.cpmd.org));

The CPMD code: some facts

- Version 3.15.3:
- 10000+ licenses (www.cpmd.org) in more than 50 nations
- 1500+ members of the cpmd mailing list
- 3000+ publications (since 2001)
- More then 30000 citations

- Scalability up to 1Mthreads nodes
- 1 ns/week on 100 atoms system
- ~2000 molecules/day BG/P Rack
- Largest calculation: ~20000 atoms

- Used widely as benchmark for HPC
- Many of the algorithms innovations/concepts set the standard in the community
### The evolution of ab-initio MD at ZRL

<table>
<thead>
<tr>
<th>Year</th>
<th>System (limit)</th>
<th>Type of calculation</th>
<th>HW</th>
<th>Type of algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1992</td>
<td>one organic molecule of ~50 atoms</td>
<td>dynamics; electronic structure</td>
<td>RISC6000/580 (125 MFlops)</td>
<td>serial</td>
</tr>
<tr>
<td>1994</td>
<td>liquid 100 atoms. organics water</td>
<td>reaction dynamics - free energy</td>
<td>SP1-16 nodes (2 GFlops)</td>
<td>parallel/MPI</td>
</tr>
<tr>
<td>1996</td>
<td>biomolecules 200 atom models and in water</td>
<td>reaction dynamics; electronic structure</td>
<td>SP2/66MHz 16 nodes (4.2 GFlops)</td>
<td>parallel/MPI</td>
</tr>
<tr>
<td>1998</td>
<td>complex interfaces 400 atoms. water/oxide organic/metal</td>
<td>all of the above</td>
<td>SP2/166MHz 32 nodes (20.5 GFlops)</td>
<td>parallel/MPI</td>
</tr>
<tr>
<td>2000</td>
<td>supramolecular systems 1000 atoms. 2D quantum dots arrays</td>
<td>all of the above</td>
<td>SP3/200MHz 64/2 ways nodes (102.4 GFlops)</td>
<td>parallel/MPI+ OpenMP</td>
</tr>
<tr>
<td>2002</td>
<td>small proteins realistic interfaces 2000 atoms</td>
<td>all of the above</td>
<td>p690/1.3GHz 8/32 ways nodes (1.3 TFlops)</td>
<td>parallel/MPI+ OpenMP</td>
</tr>
<tr>
<td>2006</td>
<td>complex systems 5000 atoms</td>
<td>all of the above</td>
<td>2 BG/L Racks 4096 processors (11TFlops)</td>
<td>parallel/MPI+ taskgroup+</td>
</tr>
<tr>
<td>2008</td>
<td>complex systems 8000 atoms</td>
<td>all of the above</td>
<td>2 BG/P Rack 8192 processors (26 TFlops)</td>
<td>parallel/MPI+ taskgroup+ OpenMP</td>
</tr>
</tbody>
</table>
Total Energy of a molecular system

(Khon-Sham formulation of DFT in the BO approximation)

\[ E_{\text{tot}}(R,r) = E_{\text{el}}(r;R) + E_{\text{ion}}(R) \]

\[ E_{\text{el}}(r;R) = E_k + E_{\text{ext}} + E_h + E_{\text{xc}} \]

\[ n_e(r) = \sum_i f_i |\Psi_i|^2 \]

\[ E_k = -\frac{1}{2} \sum_i \langle \Psi_i | \Delta | \Psi_i \rangle \] (Kinetic Energy)

\[ E_{\text{ext}} = \int V_{\text{ext}}(r)n_e(r)dr \] (Nuclei/Electrons interaction Energy)

\[ E_h = \frac{1}{2} \iint n_e(r_1) \frac{1}{r_{12}} n_e(r_2) \, dr_1 \, dr_2 \] (Hartree Energy)

\[ E_{\text{xc}} = \int \varepsilon_{\text{xc}}(r)n_e(r) \, dr \] (Exchange-Correlation Energy (ManyBody Term))
Total Energy of a molecular system with a plane wave basis set

\[ \psi \left( \mathbf{r} \right) = \frac{1}{\sqrt{\Omega}} \sum_{\mathbf{G}} c_{i}(\mathbf{G}) e^{i\mathbf{G}\mathbf{r}} \]

\[ n(\mathbf{r}) = \sum_{i} f_{i} \left| \psi_{i}(\mathbf{r}) \right|^{2} \]

\[ E_{\text{kin}} = \frac{1}{2} \sum_{i} f_{i} \sum_{\mathbf{G}} \mathbf{G}^{2} \left| c_{i}(\mathbf{G}) \right|^{2} \]

\[ E_{\text{loc}} = \Omega \sum_{\mathbf{G}} n^{*}(\mathbf{G}) S(\mathbf{G}) V_{\text{loc}}(\mathbf{G}) \]

\[ E_{H} = 2\pi \Omega \sum_{\mathbf{G}} n_{t}^{*}(\mathbf{G}) \frac{1}{\mathbf{G}^{2}} n_{t}(\mathbf{G}) \]

\[ E_{\text{XC}} = \frac{\Omega}{N} \sum_{i} \varepsilon \left( \mathbf{r}_{i} \right) n(\mathbf{r}_{i}) \]

\[ E_{nl} = \sum_{l} \sum_{j} f_{j} \left| F_{j}^{l} \right|^{2} \]
Optimization of Molecular Structure

Optimization of $E_{el}$ $\rightarrow$ Forces on Ions $\rightarrow$ Structure optimization or Molecular Dynamics

$\Psi_1(r) = \sum_j c_{ij} \Phi_j$

Localized basis set (e.g. gaussian functions)

Extended basis set (Plane Waves)

Direct Minimization (Orthogonalization)

Eigensystem (Diagonalization)

Car-Parrinello
The size of a system is determined by the number $M$ of PWs needed for its accurate description, the number $N$ of electrons, and the number $I$ of ions.

Electronic minimization:

*(CPU time)*

- $NM\log M$ (e.g. calculation of the density, calculation of the forces)
- $N^2M\log M$ (e.g. exact exchange)
- $N^2M$ (e.g. orthogonalization)

*(Memory)*

- $NM$ (electronic wavefunction in reciprocal space)
Scaling II

Structure minimization:

(CPU time)

\(-I^3\) (BFGS)

(Memory)

\(-I^2\) (Hessian)

For most systems:

\(M \gg \gg N > I\)

Simulation time dominated by 3D-FFTs for systems <1000 atoms
by Orthogonalization for systems > 1000 atoms
• Distribute plane waves and parallelize 3D-FFT (MPI API)
  • maximum scaling 128 procs - 400 atoms

• Mixed MPI/OpenMP parallelization
  • maximum scaling 1024 procs - 1000 atoms

J. Hutter and A. Curioni, Parallel Computing (31) 1, 2003

• Hierarchical Taskgroup parallelization for BG
  • Extreme scale-out 128k procs – 110 TFlops


• Parallel Linear Algebra and Parallel Initialization
  • >10000 atoms (8K atoms demonstrated on 4 Racks)


• Cache/Network Optimized Orthogonalization
Distributed Memory 3D FFT

For each wave function: Distribute its coefficients over the G-vectors across the z-direction, thus forming “pencils”

Example: 2 processors

- Processor 1
- Processor 2
Distributed Memory Implementation in CPMD

3D FFT: can be computed in 3 steps
1D FFT across Z ⇒ 1D FFT across Y ⇒ 1D FFT across X
...or 3D FFT in two steps
1D FFT across Z ⇒ 2D FFT across X-Y planes
Limited Scalability of Standard 3D FFT

Each processor takes a number of whole planes...

Very good scheme for small – medium sized computational platforms…but

Observe that scalability is limited by the number of planes across the Z-direction! ... Which is in the order of a few hundreds... O(100)...

Thus: not appropriate for extreme scaling
3D FFT Using Task Groups

$$\rho(r) = \sum_{occ} |\psi_i(r)|^2$$

Loop across the number of electrons. Each state requires one 3D FFT.

Hierarchical parallelism*: Assign to each Task Group a number of states.

* J. Hutter and A. Curioni, Parallel Computing (31) 1, 2005
3D FFT Using Task Groups

- The Task Groups of processors will work on different eigenstates concurrently.
- Number of processors per group: Ideally the one that achieves the best scalability for the original parallel 3D FFT scheme.

EIG 1: ONLY PROC 1  
EIG 2: ONLY PROC 2
Exact Exchange – Gamma Point Parallel Implementation

\[ E_{\text{ex-ex}}(\{\psi\}) = -\frac{1}{2\Omega} \sum_{\sigma = \uparrow, \downarrow} \left( \frac{\Omega}{8\pi^3} \right)^2 \int_{BZ} dk \int_{BZ} dl \]

\[ \times \left[ \sum_{n=1}^{N_{\text{occ}}} \sum_{m=1}^{N_{\text{occ}}} \sum_{G} \frac{4\pi}{|G - k + l|^2} \rho_{mn;k}(G) \rho_{kn;ml}(G) \right] \]


NEW TASKGROUP STRATEGY

Distribute States and Orbital Couples

Exact Exchange: (\(\sim N^2 M \log M\))

• each group computes a subset of the orbital (non redundant) pairs
• cyclic distribution of the pairs (scalapack like) – with dynamic balancing
• the X-energy and the X-contribution to the electronic gradient are summed/redistributed at the end of the computation (inter groups communication)
• possible thresholding via orbital localization and overlap densities estimation
Implementing Exact-Exchange in CPMD
>95% Parallel Efficiency to over 1M threads
CPMD Parallelization & Scale out

• Distribute plane waves and parallelize 3D-FFT (MPI API)
  • maximum scaling 128 procs - 400 atoms

• Mixed MPI/OpenMP parallelization
  • maximum scaling 1024 procs - 1000 atoms

J. Hutter and A. Curioni, Parallel Computing (31) 1, 2003

• Hierarchical Taskgroup parallelization for BG
  • Extreme scale-out 128K procs – 110 TFlops


• Parallel Linear Algebra and Parallel Initialization
  • >10000 atoms (8K atoms demonstrated on 4 Racks)


• Cache/Network Optimized Orthogonalization
Original CPMD Orthogonalization

The Cholesky based orthonormalization

1) Matrix X (Electronic States) is distributed row wise

2) Calculate distributed overlap matrix $S = X^T X$

3) Compute parallel Cholesky decomposition $S = R^T R$

4) Invert triangular matrix $R$. $R^{-1}$ is distributed

5) Compute orthonormal vectors $Y = XR^{-1}$

BLAS 3. Global reduction needed
BLAS 3. Send/Recv
BLAS 3. Send/Recv
BLAS 3. Many Broadcasts
Computational aspects and practical parallel deployment in \textit{ab initio} codes

- BLAS 3. Performance and Cache optimized-
- Numerical stability problems? Not really…

Then what is the catch? …Consider massively parallel deployment! (++)thousands of procs)

+ Matrix X (wavefunctions) is distributed row-wise to all available procs. So, the calculation of the overlap matrix \( S = X^T X \) will scale

- But then the overlap matrix can have less rows (columns) than the available procs!
- Very difficult to scale.
- Not optimized in terms of communication needs and Network Topology
  - Typical overlap matrix sizes: 100 (small), 500 (medium), 2000 (large), 10000 (very large)
  - Typical massively parallel deployment: 10s to 100s of thousands of proc elements

\textbf{SCALAPACK} like scaling stops at hundreds of procs.
How about Gram-Schmidt?

\[ x_1 = x_1/\|x_1\| \]

for \( i = 2, \ldots, k \)

\[ w(1:i-1) = X(:,1:i-1)^T x_i \]

BLAS 2. Global reduction needed

for \( j = 1:i-1 \)

\[ x_i = x_i - w(i) x_j \]

end

BLAS 1. No communication needed

\[ x_i = x_i/\|x_i\| \]

end

BLAS 1. Global reduction needed

Standard Gram-Schmidt (Modified GS rarely needed in \textit{ab initio}) will scale very well on massively parallel platforms with very fast collective communication networks (BG/P).

Scalar per node performance is low though because of BLAS 2 nature! Cost: \( O(2nk^2) \)
Our proposal: Block Gram-Schmidt!

Consider the $n \times b$ matrices $A$ and $B$ such that

- $A$ is orthonormal: $A^T A = I$
- $B$ is not orthonormal: $B^T B \neq I$

Then, if $W = A^T B$ we can subtract the “overlap” of $B$ on $A$ from $B$ so that $A^T B = I$:

- $B = B - AW$

Observe: this a BLAS3 operation (DGEMM)

We then need to orthonormalize the new $B$:

- We utilize the Cholesky based approach here
- Remember: we keep $b$ small but large enough to yield good BLAS3 performance

Generalizing the procedure yields Block Gram-Schmidt
Our proposal: Block Gram-Schmidt!

\[
M = \frac{k}{b} /* \text{Consider a block size } b: O(100) */
\]

for \(i=1,\ldots,M\)

if \(i>1\)

\[
W(1:b,1:(i-1)*b) = X(:,1:(i-1)*b)^T X(:,(i-1)*b+1:i*b)
\]

\[
X(:,(i-1)*b+1:i*b) = X(:,(i-1)*b+1:i*b) - \ldots X(:,1:(i-1)*b) W(1:b,1:(i-1)*b)
\]

end

\[
S = X(:,(i-1)*b+1:i*b)^T X(:,(i-1)*b+1:i*b)
\]

\[
R = \text{chol}(S)
\]

\[
X(:,(i-1)*b+1:i*b) = X(:,(i-1)*b+1:i*b) R^{-1}
\]

end
Let’s summarize

Matrix size: \( n \times k, k \ll n \)

Cholesky based orthogonalization
- Cost: \( 3nk^2 \)
- BLAS 3
- Difficult to scale on thousands of procs

Gram-Schmidt
- Cost: \( 2nk^2 \)
- BLAS 2
- Easier to scale. Very Low scalar performance

- Cost: \( 2nk^2 \)
- BLAS 3
- \# of messages: \( O(k/b) \) [or \( O(k/b)^2 \) in the modified case]
- Cache optimized AND Network optimized
- Designed to inherit the strong points of preexisting schemes
Tests: Gram-Schmidt v.s. Cholesky based

N=10000 (left) and N=20000 (right). Varying number of vectors to orthogonalize. Times in seconds

The BLAS3 based Cholesky approach although more costly than the BLAS1-2 based GS clearly wins.
Tests: Block GS v.s. Cholesky based ortho

N=20000 (left) and N=40000 (right). Varying number of vectors to orthogonalize.

Comparison of Cholesky based v.s. Block (modified-standard) Gram-Schmidt
(Run times in seconds)
Block GS SMP parallelism

SMP Block GS. 1-4 threads (ESSL), blocksize b=120, left n=20000, right n=40000
Test: Block Gram-Schmidt - Scale-out

Run times: N=1M, k=300, log scale

Run times: N=1M, k=600, log scale
Tests: Block Gram-Schmidt - Scale-out

Run times: N=60M, k=2000, log scale

Run times: N=60M, k=4000, log scale

70% peak on 8 BG/P racks
Tests: Block Gram-Schmidt - BG/P vs BG/Q

- Vector length: 4M, 2K states
  - BG/P: 1MPI, 4T per node
  - BG/Q: 2MPI, 32T per node

Graph showing run time vs number of cores.
Test Cases

- **Test 1**: 32 water molecules  
  Cutoff 70 Ry – norm conserving pseudos
- **Test 2**: 576/1576 atoms Propylene Carbonate/Li$_2$O$_2$  
  Cutoff 100 Ry - norm conserving pseudos
- **Test 3**: 1000-8000 atoms SiC  
  Cutoff 35Ry – norm conserving pseudos
- **Test 4**: ~600 atoms – Li$_2$O$_2$ PC – metadynamics  
  Cutoff 70 Ry – norm conserving pseudos
- **Test 5**: ~500 atoms – aSiH – hybrid functionals  
  Cutoff 70 Ry – norm conserving pseudos
Test 1: 32 water (70Ry)

Best Time per step = 0.09 sec on 2048 nodes (red without taskgroups) ;~650 ps/week

Single processor performance ~2.1 speedup after optimization

PWR7 4.0GHz 1.2 sec/step on 16 processors
Test 2: $\text{Li}_2\text{O}_2$ and Propylen Carbonate

Best Time per step = 12.0 sec ($\text{Li}_2\text{O}_2$) - 2.84 sec (PC)

~ 20 ps / week on 2048 proc
Test 3: Ab-Initio Simulations on large systems (Silicon Carbide supercells)

1000 atoms (4000 electrons) 1BG/P Rack  time/step :  4 sec BG/P
2000 atoms (8000 electrons) 2BG/P Racks  time/step :  25 sec BG/P
4000 atoms (16000 electrons) 2BG/P Racks  time/step:  97 sec BG/P
8000 atoms (32000 electrons) 4BG/P Racks  time/step:  540 sec BG/P
Test 4: Exploiting Intrinsic Parallelism: Multiwalker Metadynamics

Aprotic Solvent Stability in Li-air batteries
Propylen Carbonate + LiPF6 +Li2O2
~600 atoms – 300 K
~10ps 1Week → 5hours
Screening Possible!

(runs ANL – scaling Juelich)
Test 5: aSiH - Hybrid Functional

Best Time per step = ~30 sec
Opportunities

• System Sizes: ~1'000'000 atoms?

• Throughput: ~ 100-1000 ns/week (hundreds atoms)
  1 ns/week (thousand atoms)

• Accuracy (effective meta functionals)

• Accurate Complex Chemistry/Materials Science via Enhanced Sampling (Metadynamics, Path Sampling)

• Possible exploitation of Millions of Threads
Outline of the Presentation

• Introduction
  ▪ Computational Sciences at IBM Zurich Research Laboratory
  ▪ The need of extreme parallelism

• The CPMD code: a research tool for ab-initio MD
  ▪ CPMD parallelization and scaleout
  ▪ Cache/Network Optimized Orthogonalization
  ▪ Performance on tests systems: time scales and system sizes

• The new challenges toward exascale computing
  ▪ Fault tolerant and energy efficient algorithms

• Succesfull application examples

• Conclusions
Deep Computing Research: Exascale: Innovation areas demanded by power, cost and usability

(20M Watts vs 2G Watts)

Cost will put even stronger constraints in memory per threads

Cost and packaging constraints will specialize network connectivity and topology

Very large improvement in programmability and software efficiency, for millions of threads (PGAS, asynchronous)

Current – Best failure rate (BG) 0.01 Failures /Teraflops/Month

1 Failure every 4 Minutes at Exaflop
Algorithm research will play an increasing role in Exascale computing Research.

Focus shift from Sustained Performance to Energy Efficiency and Fault Tolerance.
Energy Efficient and Fault Tolerant Algorithms

• Re-engineering of simulation methods and algorithms using time to solution, energy efficiency and fault tolerance as optimization criteria
  - System constraints: extreme parallelism, low ratio memory/computation, low ratio communication/computation, data locality, (programmable)-accelerators

• Example: Iterative Linear Solver with Mixed Precision (C. Bekas and A. Curioni)

  Quadratic Cost Iterative Refinement:
  ~ 1 sec time to solution
  ~ 12 % sustained performance
  ~ 0.005 kWh energy usage
  Fault Tolerant (~10% cost)

  Standard Iterative Solver:
  ~20 sec time to solution
  ~9 % sustained performance
  ~0.0116 kWh energy usage
  Fault Tolerant (~10% cost)

  Standard Direct Solver (Linpack):
  ~15 sec time to solution
  ~80% sustained performance
  ~0.0441 kWh energy usage
  Non Fault Tolerant (~100% cost)
Quadratic Cost Iterative Refinement

✓ LOW PRECISION: LP
✓ HIGH PRECISION: HP
✓ Let CG(A,y,k) be a procedure implementing k steps of Conj. Gradient in single precision

1. Compute initial solution: \( x_0 = \text{CG}(A,b,k) \)  Cost: \( O(kn^2) \)
2. Compute initial residual: \( r_0 = b - Ax_0 \)  Cost: \( O(n^2) \)
3. \( k = 0 \)
4. REPEAT
6. Solve for residual: \( d_k = \text{CG}(A,r_k,k) \)  Cost: \( O(kn^2) \)
7. Update solution: \( x_{k+1} = x_k + d_k \)  Cost: \( O(n) \)
8. Compute residual: \( r_{k+1} = b - Ax_{k+1} \)  Cost: \( O(n^2) \)
9. \( k = k + 1 \)
10. UNTIL \( ||r_{k+1}|| \cdot tol \)

Key properties:
✓ Dominant cost \( O(kn^2) \). Performed in LOW PRECISION. Cost in HP is \( O(n^2) \)
✓ We can take great advantage of fast single precision hardware!
✓ Even on platforms without fast low prec hardware: benefit (30% or so) from reduced memory traffic
Outline of the Presentation

- Introduction
  - Computational Sciences at IBM Zurich Research Laboratory
  - The need of extreme parallelism

- The CPMD code: a research tool for ab-initio MD
  - CPMD parallelization and scaleout
  - Cache/Network Optimized Orthogonalization
  - Performance on tests systems: time scales and system sizes

- The new challenges toward exascale computing
  - Fault tolerant and energy efficient algorithms

- Successful application examples

- Conclusions
• Activities and Results
  – electronic bandstructures of Li\textsubscript{x}O\textsubscript{y} → conductivity
  – overvoltages – or lack thereof
  – extensive study and forecasts of electrolyte solvent stability
  – electron transport in Li\textsubscript{2}O\textsubscript{2}

• Mostly done on a very large IBM Blue Gene at DOE Argonne National Lab
  – by the Computational Sciences team at IBM Research – Zurich
  – Recipient of INCITE multiple INCITE awards
  – Instrumental to close collaboration with external partners

Simulations of Li\textsubscript{2}O\textsubscript{2} in Propylenecarbonate, T. Laino, A. Curioni, A New Piece in the Puzzle of Lithium/Air Batteries, Chemistry, DOI 10.1002/chem.201103057 (22 February 2012)
Lithium/Air Batteries: screening new solvents

- Suggest a new class of solvents with strong chemical resilience to Li$_2$O$_2$ degradation.

<table>
<thead>
<tr>
<th>Solvent</th>
<th>Energy Barrier (kcal/mol)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC</td>
<td>– (*)</td>
</tr>
<tr>
<td>NMP</td>
<td>24</td>
</tr>
<tr>
<td>NMP-tBut</td>
<td>33</td>
</tr>
<tr>
<td>NMP-F3</td>
<td>– (*)</td>
</tr>
<tr>
<td>NMP-F6</td>
<td>– (*)</td>
</tr>
<tr>
<td>2Met-NMP</td>
<td>51</td>
</tr>
<tr>
<td>PEG-5</td>
<td>16</td>
</tr>
<tr>
<td>Met-PEG-5</td>
<td>23</td>
</tr>
<tr>
<td>F-PEG-5</td>
<td>40</td>
</tr>
<tr>
<td>CH3CN</td>
<td>35</td>
</tr>
</tbody>
</table>

(*) spontaneously decompose
Supercomputing for Energy: Batt500 project
Example: IBM Technology – CMOS - Scaling

Dielectric constant: $\varepsilon \sim 10 - 40$,
Band gap $> 6$ eV
Non-reactive with Si
Small electrical thickness ($<1$ nm); $(\varepsilon_{SiO2}/\varepsilon)t$
Electrical properties $\sim$ Si/SiO$_2$
(low interface defect density, high electron mobilities, low charge trapping)
Computer aided design of materials with tailored properties

Need to calculate structural, electronic and dielectric properties of many candidate materials on realistic environments.
(system sizes ~1000 atoms)

BG/P, allowing to simulate more complex systems for longer timescales, makes computer aided materials design a reality.

~20.000 Atoms from First Principles based Molecular Dynamics –
20 Millions Atoms per BG/P rack with Classical (Empirical) Molecular Dynamics
Hf$_x$Si$_{1-x}$O$_2$ : Gate materials optimization

“Odd” observed behavior explained!


- First Principles Calculations of structures, chemical/physical stability, electronic and electrical properties (dielectric constants) as a function of Hafnium concentration.

- More than 50 virtual samples of Hafnium silicates were simulated in our in-silico study.

- Blue Gene, its scalability and flexibility plus the optimal remapping of our algorithms have been instrumental for the success of our study.

- A single simulation took ~ 5 days on 2 BG/L racks— it would have taken more than 3 months on 8 Racks p690 with Federation switch.

2006 ‘Interface Engineering for Enhanced Electron Mobilities in W/HfO2 Gate’ Stacks’ (US7115959)’
High-K materials

Understand/Develop:
- Structure of the Si/SiO2/HighK interface
- Dependence of the K on chemical composition in SiOxNy
- Odd behavior of the K in Hf/Zr Silicates
- Accurate model for vacancy diffusion in LaxHfyOz systems
- Issues of integration of Ultra-High-K materials

THANK YOU!