Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

Gerhard Klimeck
Network for Computational Nanotechnology
School of Electrical and Computer Engineering
Purdue University

Purdue Math / SIAM Seminar Series
February 7th, 2008

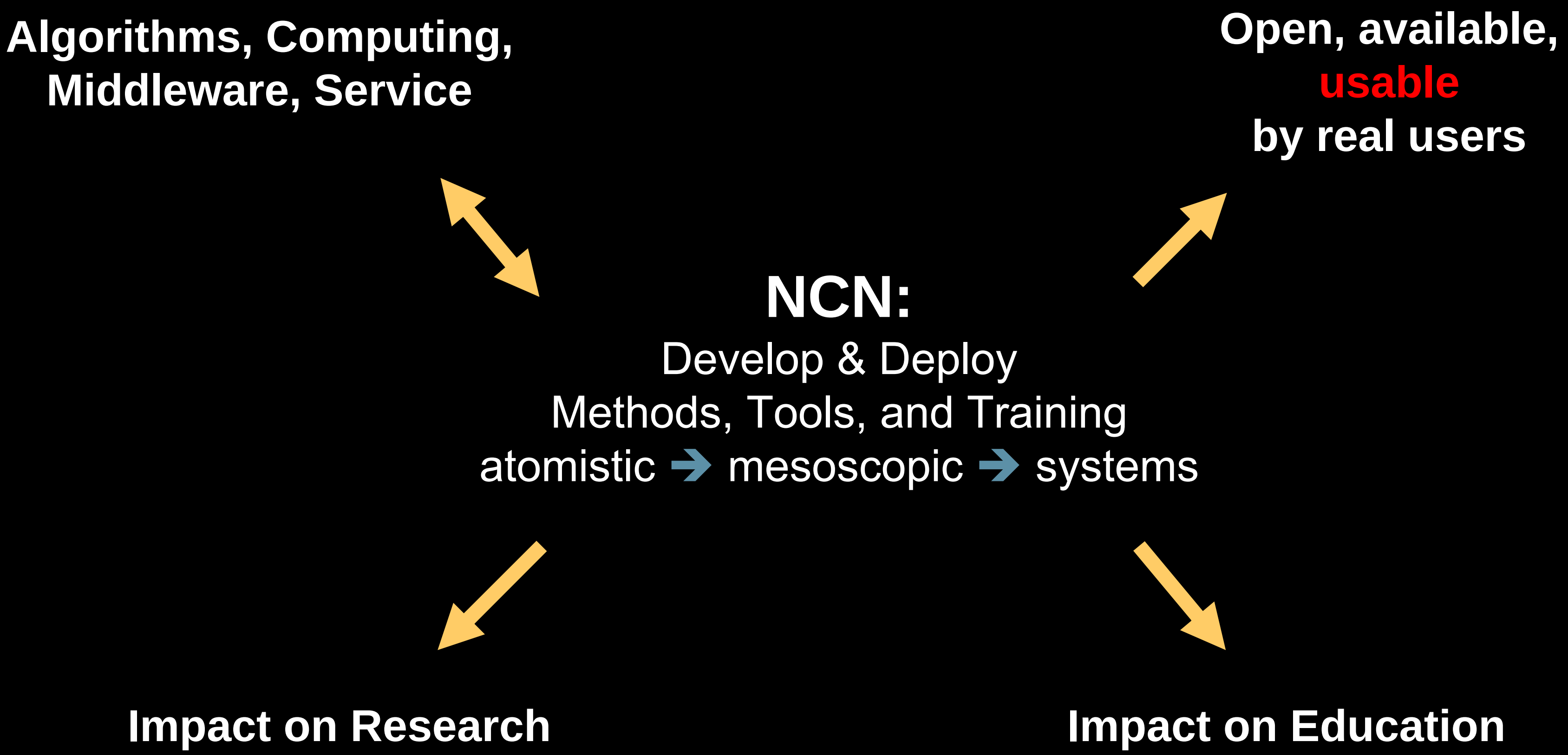
Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

- Network for Computational Nanotechnology (NCN)
- NEMO 3-D
  - »Physics motivation – multimillion atom simulation
  - »Basic algorithms
  - »Scaling to 8,192 processors
- NEMO 1-D
  - »Physics motivation –transport of electrons quantum statistical non-equilibrium mechanics
  - »Basic algorithms
  - »Scaling to 23,000 processors
- OMEN plans
- Algorithm work needed

Acknowledgements

- NEMO 3-D – development at Purdue
  - »Faisal Saied, Sunhee Lee, Hoon Ryu, Maxim Naumov, Hansang Bae, Steve Clark
  - »Rajib Rahman, Neerav Kharche
- NEMO 1-D
  - »Development team at Texas Instruments and JPL
  - »Benchmarking at Purdue: Steve Clark and Hansang Bae
- OMEN
  - »OMEN proposal team
- Alternative Algorithms
  - »Darve – Stanford
  - »Cauley – Purdue
  - »Sameh - Purdue

Overcoming barriers to quantum mechanics simulations in physics, chemistry, biology, and materials to migrate nano-science to nano-technology.



Algorithms, Computing, Middleware, Service

Open, available, **usable** by real users

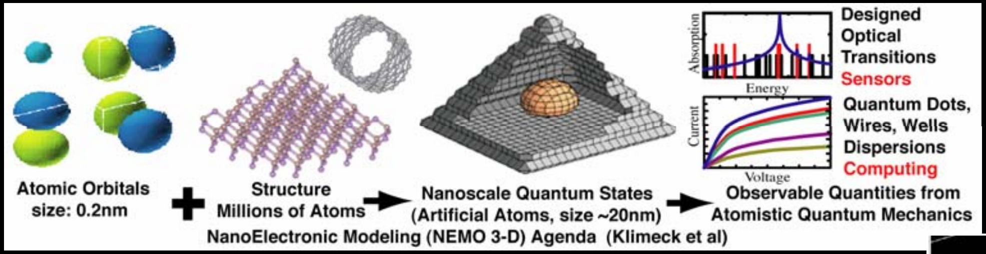
NCN:

Develop & Deploy
Methods, Tools, and Training
atomistic → mesoscopic → systems

Impact on Research

Impact on Education

Example: NEMO3D - Nanoelectronic Modeling



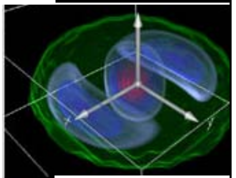
Algorithms Computing,
Middlew ice

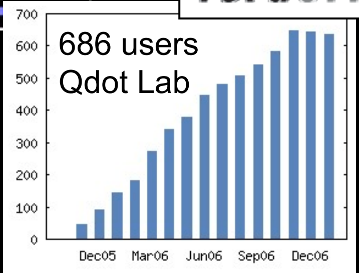
Open Science Grid

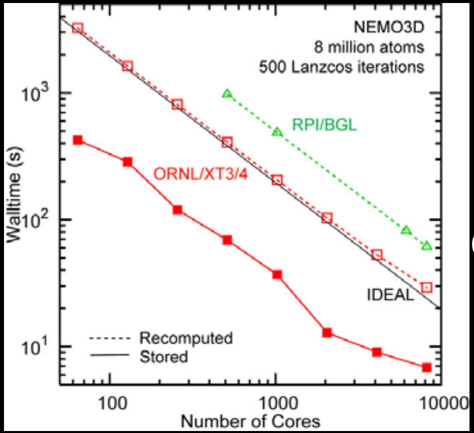
nanoHUB online simulations and more

NCN:

Develop & Deploy
Methods, Tools, and Training
atomistic ➔ mesoscopic ➔ systems







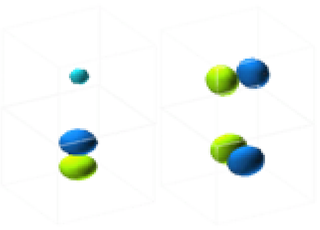
earch

Towards Peta-Apps
- 52M atoms on nanoHUB H/W
- 1,000 CPUs IBM B/G
- 4,096 CPUs Cray XT3

Impact

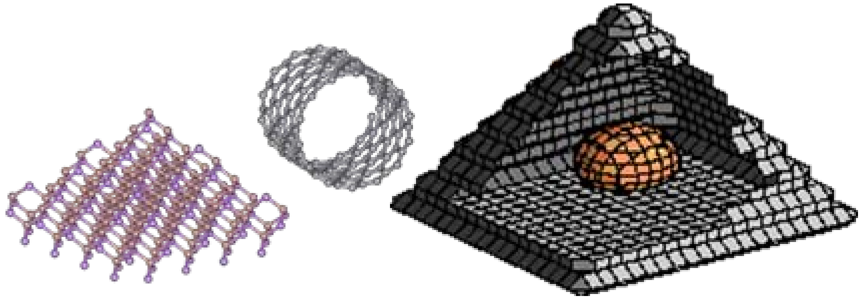
NEMO 3-D at 7 tutorials
SC05, SC06, TG06, TG7,
Nano06, NMDC06, NEMS07,
>130 educators

NEMO 3-D Technical Approach



Atomic Orbitals
size: 0.2nm

+

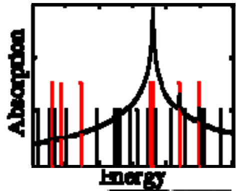


Structure
Millions of atoms

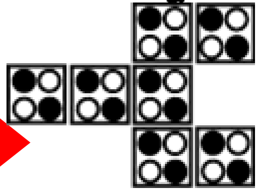
→

Nanoscale Quantum States
(Artificial Atoms, size 20nm)

→



Designed
Optical
Transitions
Sensors



Quantum Dot
Arrays
Computing

Problem:

Nanoscale device simulation requirements:

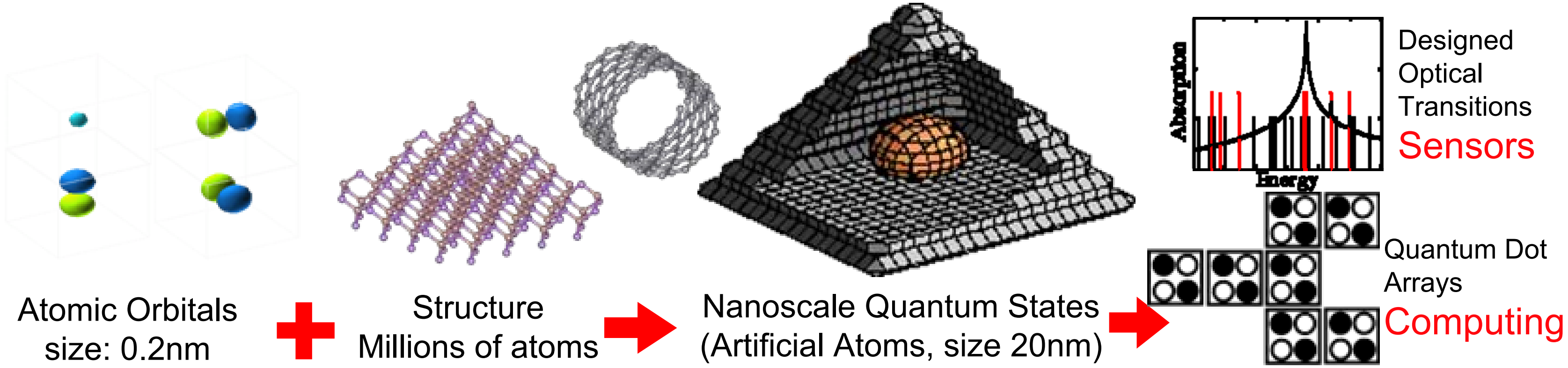
- Cannot use bulk / jellium descriptions, need description of the material atom by atom
=> use pseudo-potential or local orbitals
- Consider finite extend, not infinitely periodic
=> local orbital approach
- Need to include about one million atoms.
=> need massively parallel computers
- The design space is huge: choice of materials, compositions, doping, size, shape.
=> need a design tool

Approach:

- Use local orbital description for individual atoms in arbitrary crystal / bonding configuration
    - Use s, p, and d orbitals.
    - Use genetic algorithm to determine material parameter fitting
- Compute mechanical strain in the system.
- Develop efficient parallel algorithms to generate eigenvalues/vectors of very large matrices
- Develop prototype for a graphical user interface based nanoelectronic modeling tool (NEMO-3D)

NCN Science Impact through HPC

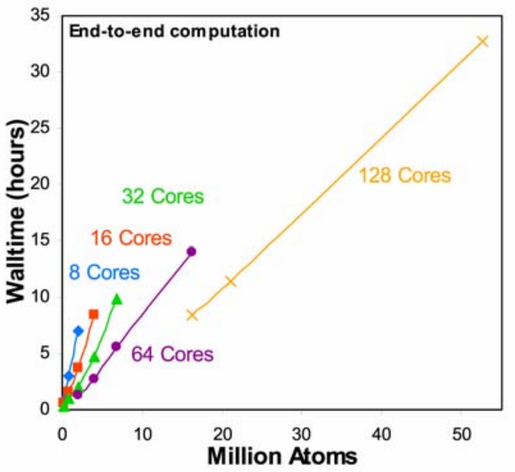
NEMO 3-D



Atomic Orbitals size: 0.2nm + Structure Millions of atoms → Nanoscale Quantum States (Artificial Atoms, size 20nm) →

Designed Optical Transitions
Sensors

Quantum Dot Arrays
Computing



Demonstration / Capability / Impact:

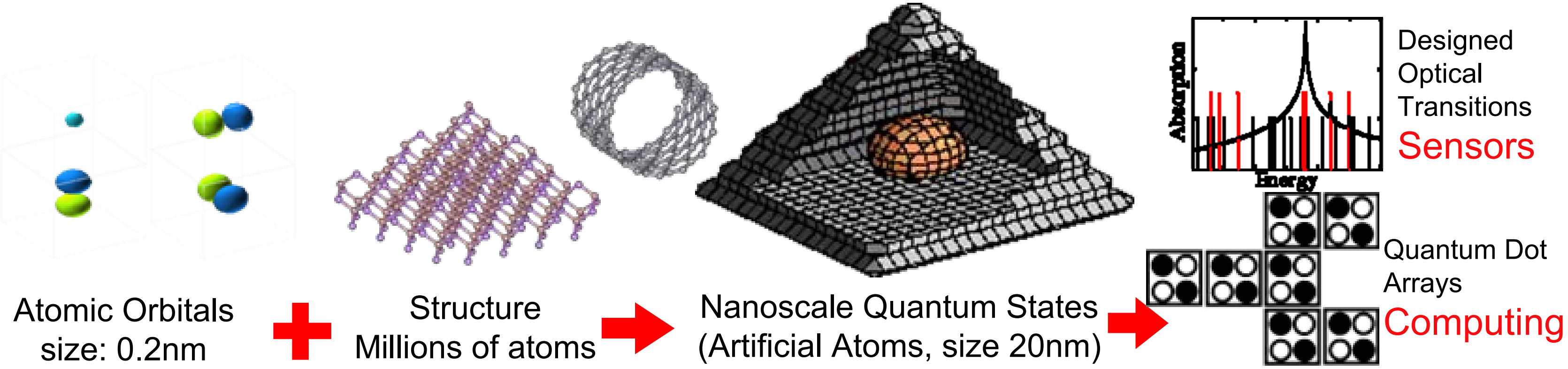
- 52 million atom electronic structure $(101\text{nm})^3$.
- Quantum dots, nanowires, quantum computing…

Approach:

- Use local orbital description for individual atoms in arbitrary crystal / bonding configuration
  - Use s, p, and d orbitals.
  - Use genetic algorithm to determine material parameter fitting
- Compute mechanical strain in the system.
- Develop efficient parallel algorithms to generate eigenvalues/vectors of very large matrices
- Develop prototype for a graphical user interface based nanoelectronic modeling tool (NEMO-3D)

NCN Science Impact through HPC

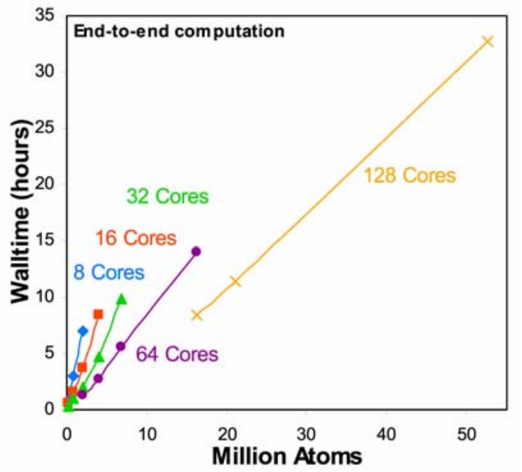
NEMO 3-D



Atomic Orbitals size: 0.2nm + Structure Millions of atoms → Nanoscale Quantum States (Artificial Atoms, size 20nm) →

Designed Optical Transitions **Sensors**

Quantum Dot Arrays **Computing**

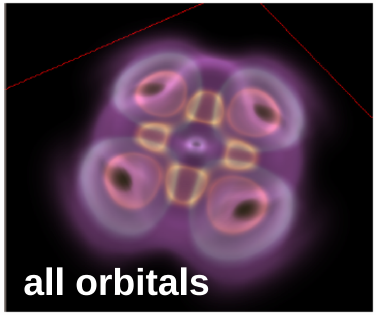


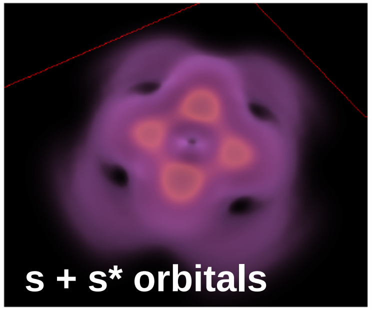
Demonstration / Capability / Impact:

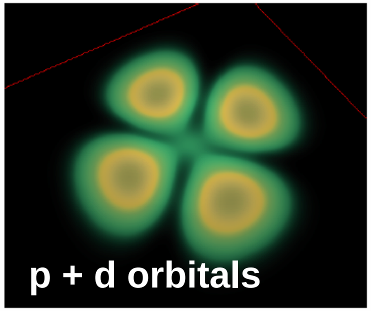
- 52 million atom electronic structure $(101\text{nm})^3$.
- Quantum dots, nanowires, quantum computing…

Visualization on a single GPU:

- How to visualize 21 million atoms on a zincblende lattice?
- Zincblende is not a space-filling mesh!!!
- => VolQD
- Baseline for nanoHUB nanoVIS server

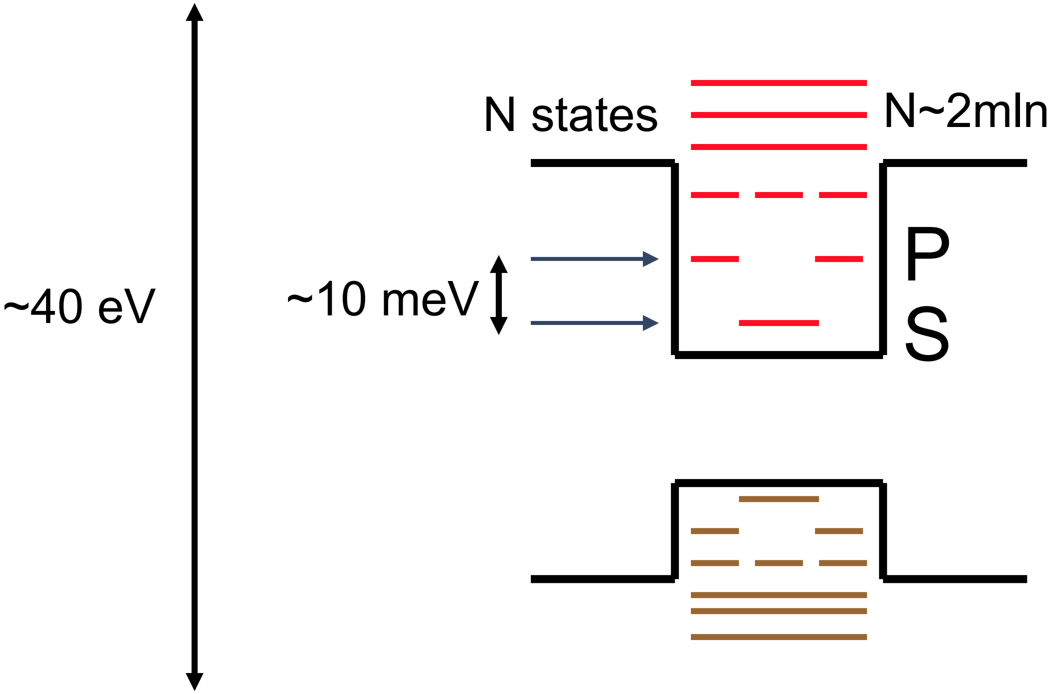

all orbitals


s + s* orbitals


p + d orbitals

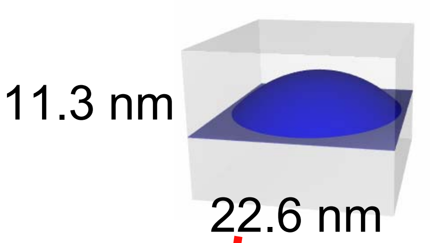
Proof of Concept

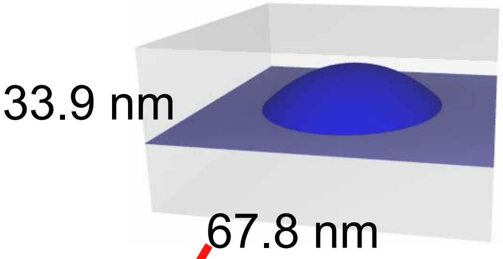
Extraction of Targeted Interior Eigenstates

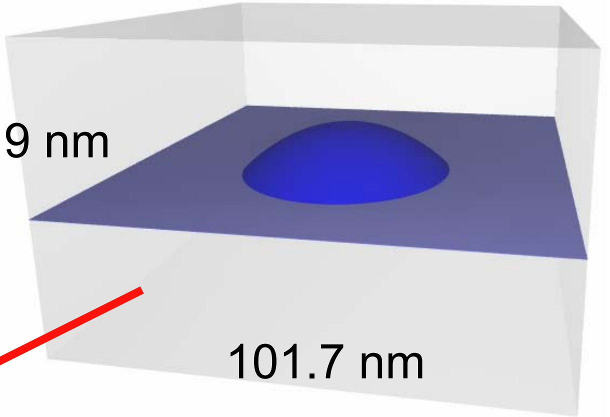


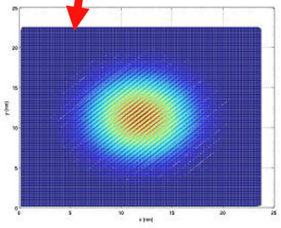
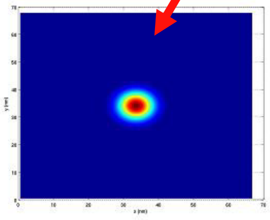
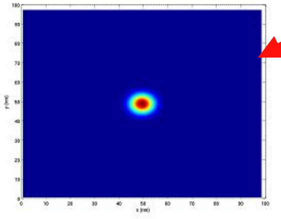
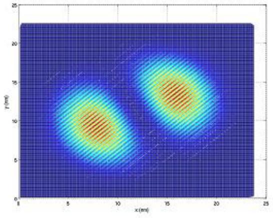
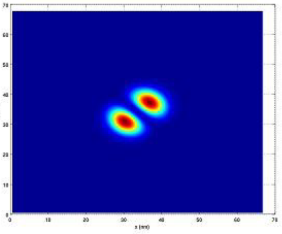
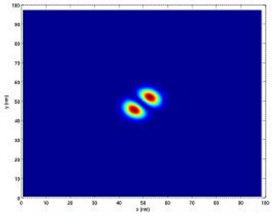
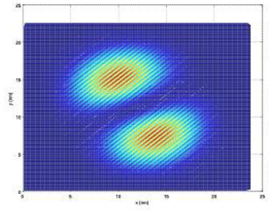
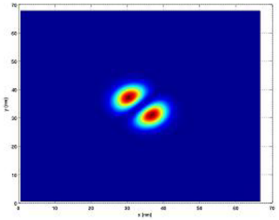
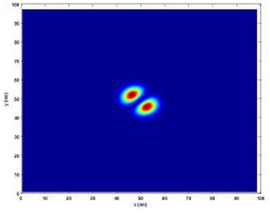
Proof of Concept

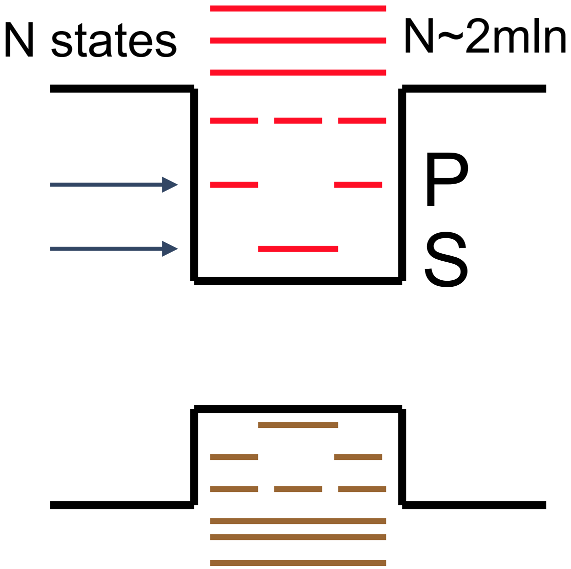
Extraction of Targeted Interior Eigenstates





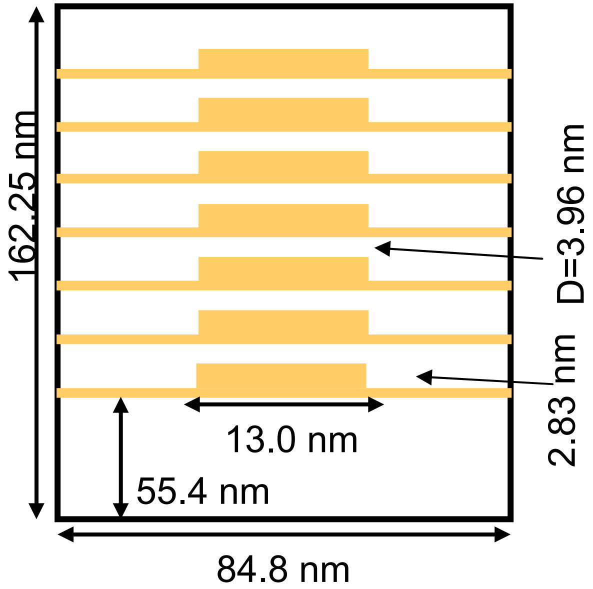


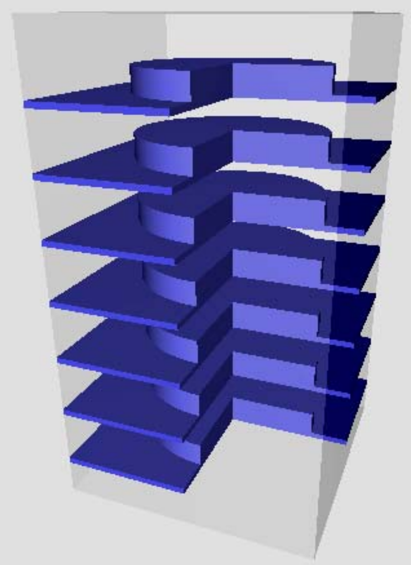
Orbital	300k	6.7M	21M
Electron GS			
Electron 1ES			
Electron 2ES			



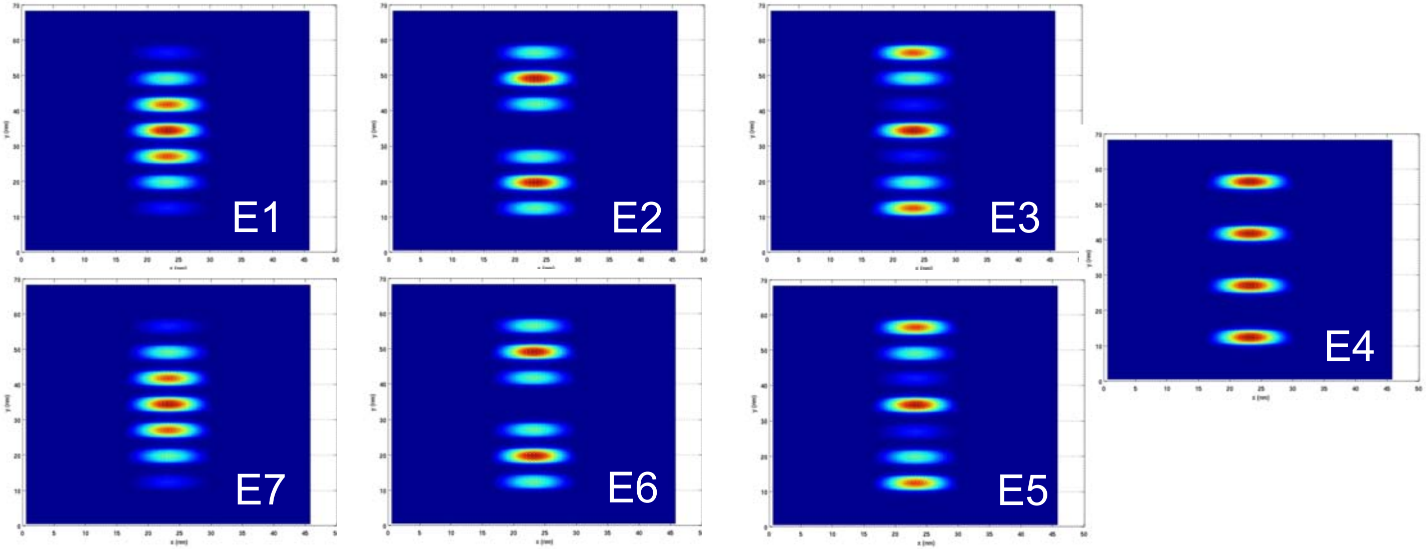
Unique and targeted eigenstates of correct symmetry can be computed in all electronic computational domains

Vertically Coupled Seven-Dot Molecule
Identical dots

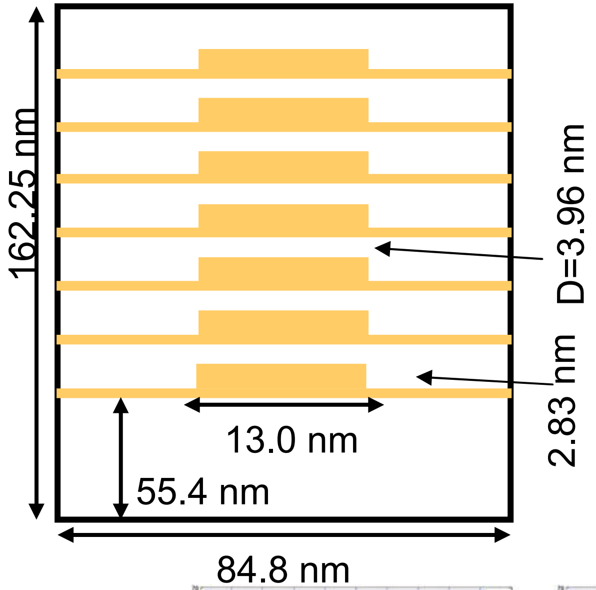


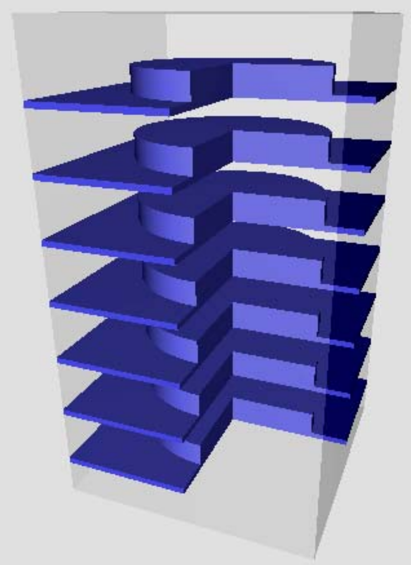


- Application in lasers
- Electronics: 6.1 Ma
- 7 Identical dots, without strain => symmetric miniband with 7 states
- Can derive this analytically from 1 dot simulations



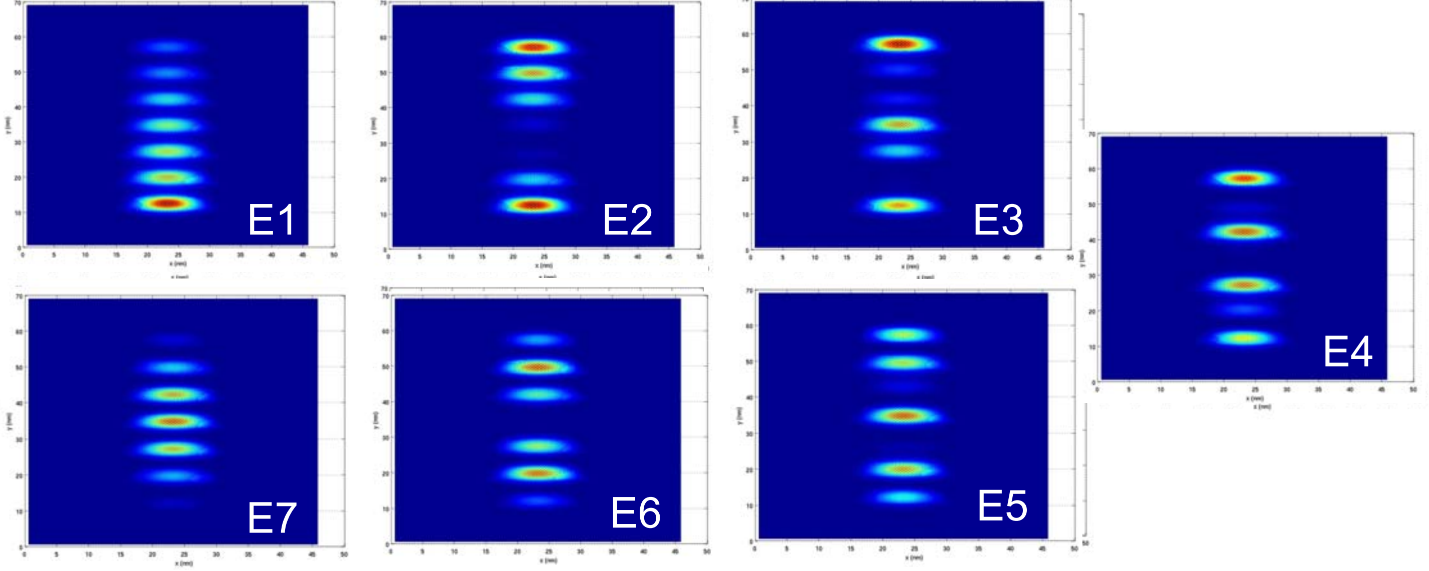
Vertically Coupled Seven-Dot Molecule
Strain asymmetry with identical dots



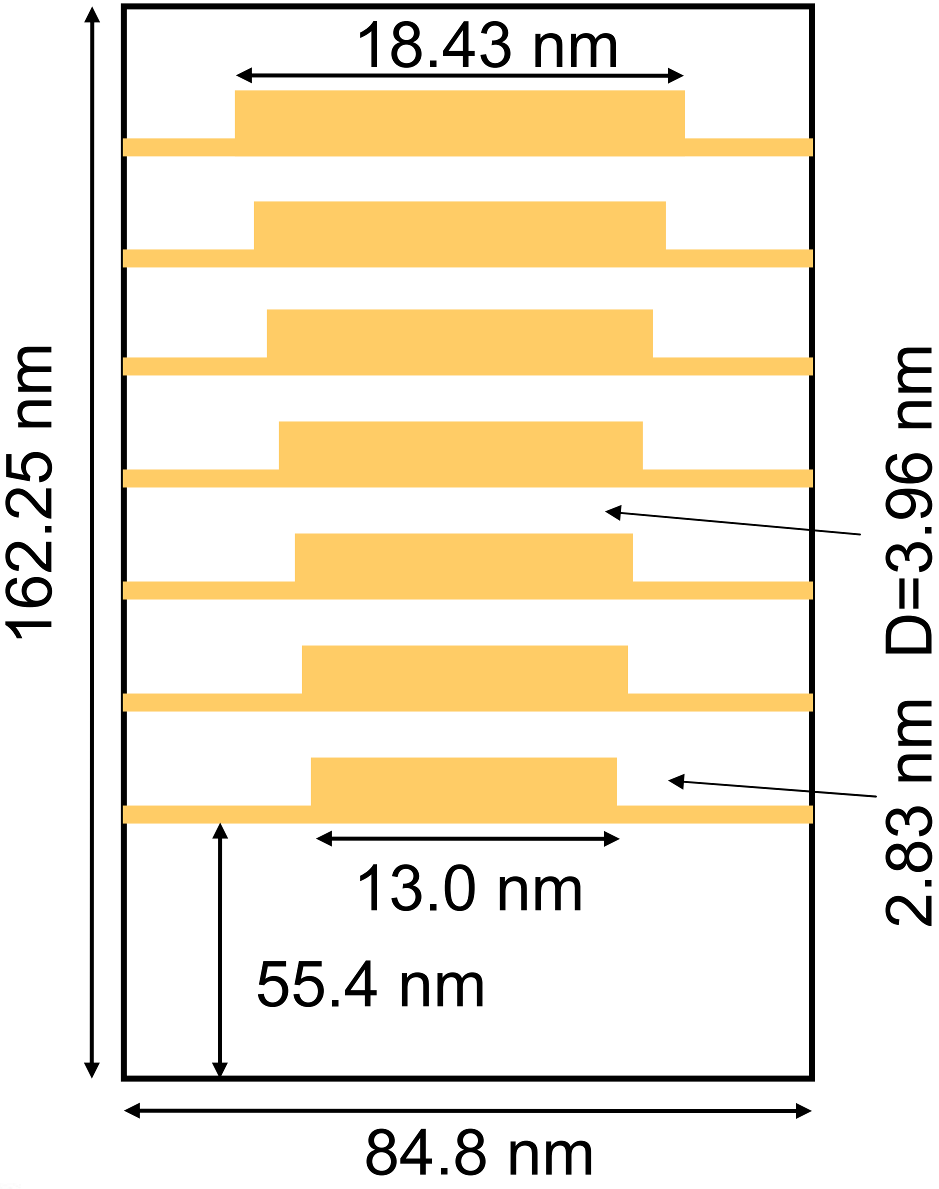


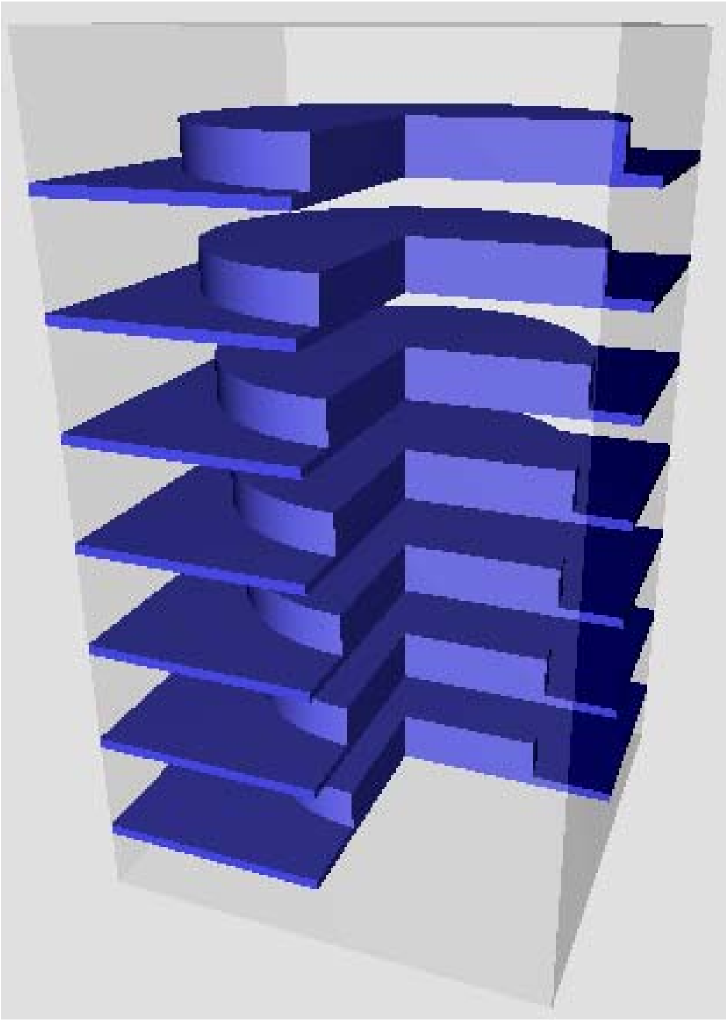
- Application in lasers
- Strain: 44.7 Ma
- Electronics: 6.1 Ma

- 7 Identical dots, with strain => asymmetric miniband
- Ground state in the BOTTOM!
- Cannot derive this analytically!

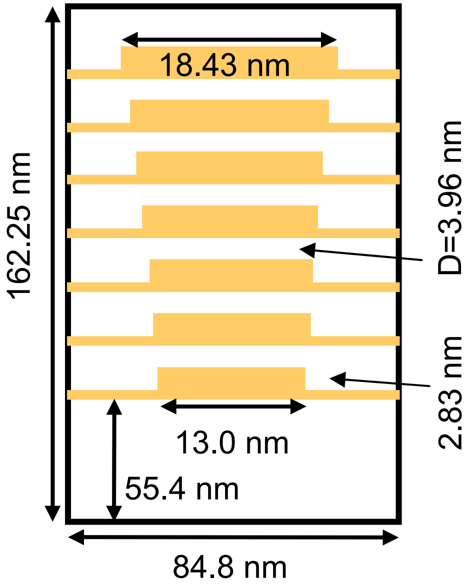


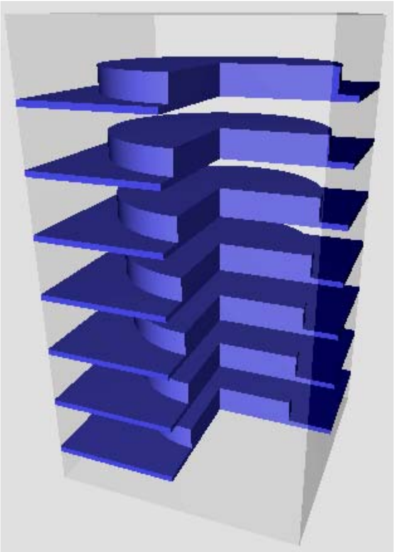
Vertically coupled QD molecule composed of nonidentical dots





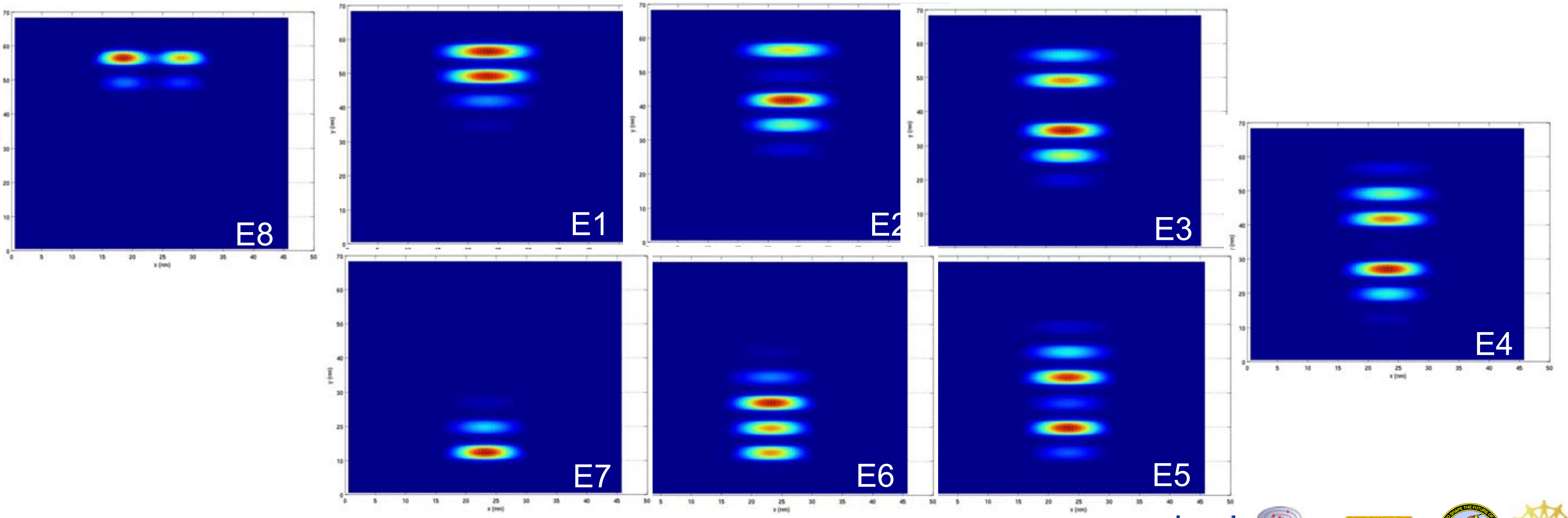
Vertically Coupled Seven-Dot Molecule
Growth asymmetry => non-identical dots



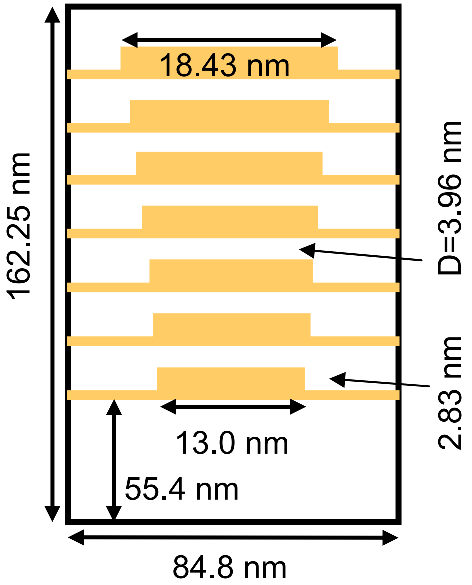


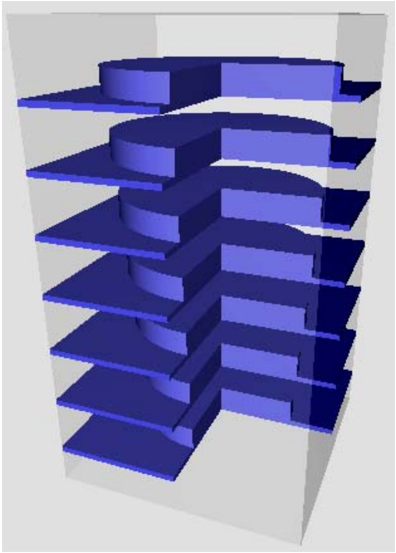
- Application in lasers
- Strain: 44.7 Ma
- Electronics: 6.1 Ma

- 7 non-identical dots, WITHOUT strain => asymmetric miniband
- Ground state at the TOP!
- First 7 states s-like.



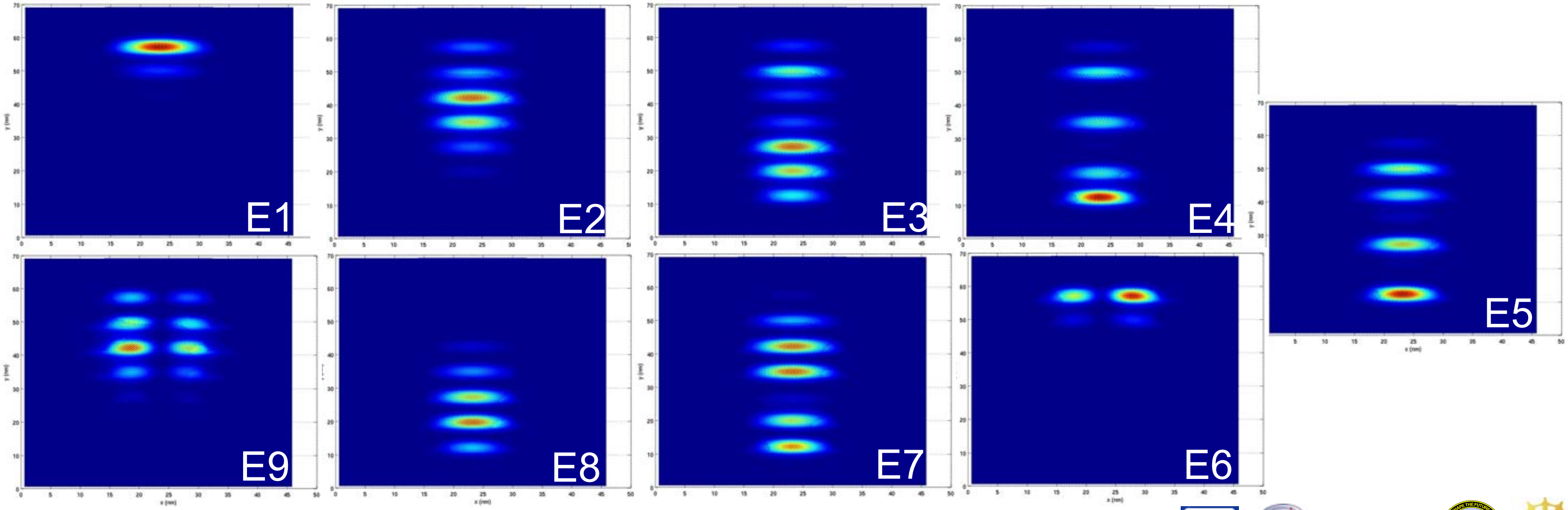
Vertically Coupled Seven-Dot Molecule
Growth asymmetry => non-identical dots





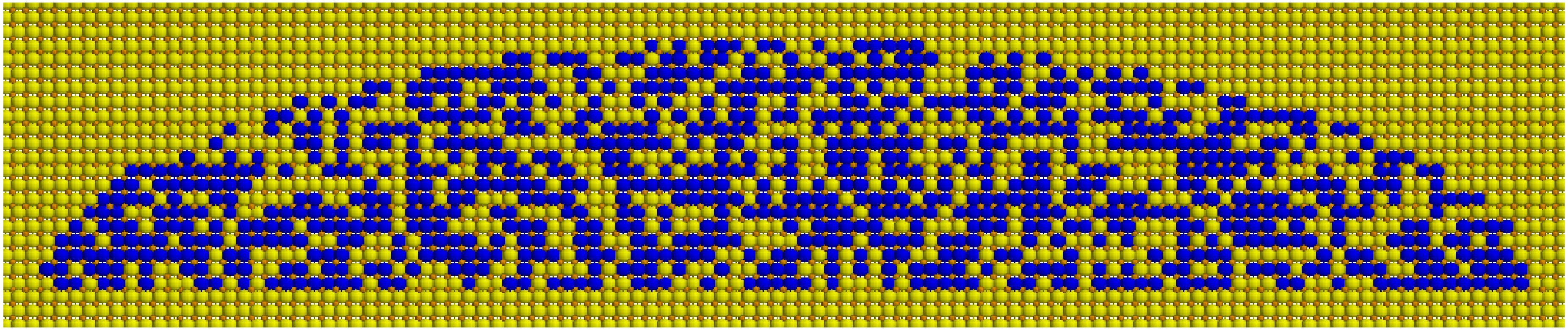
- Application in lasers
- Strain: 44.7 Ma
- Electronics: 6.1 Ma

- 7 non-identical dots, **WITH** strain => competing asymmetries
- Large dot->top, strain->bottom
- p-like states mixed into band.



Gerhard Klimeck

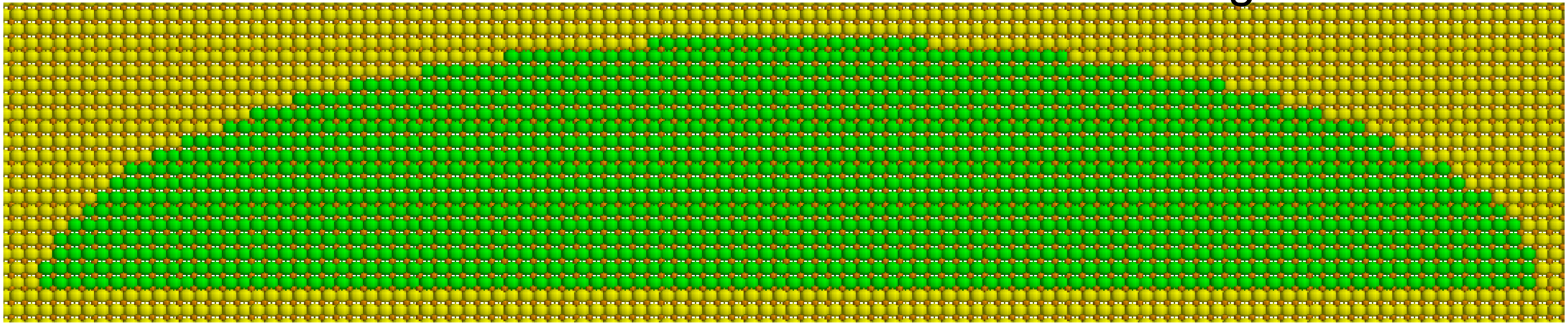
How to Treat Disordered Systems?



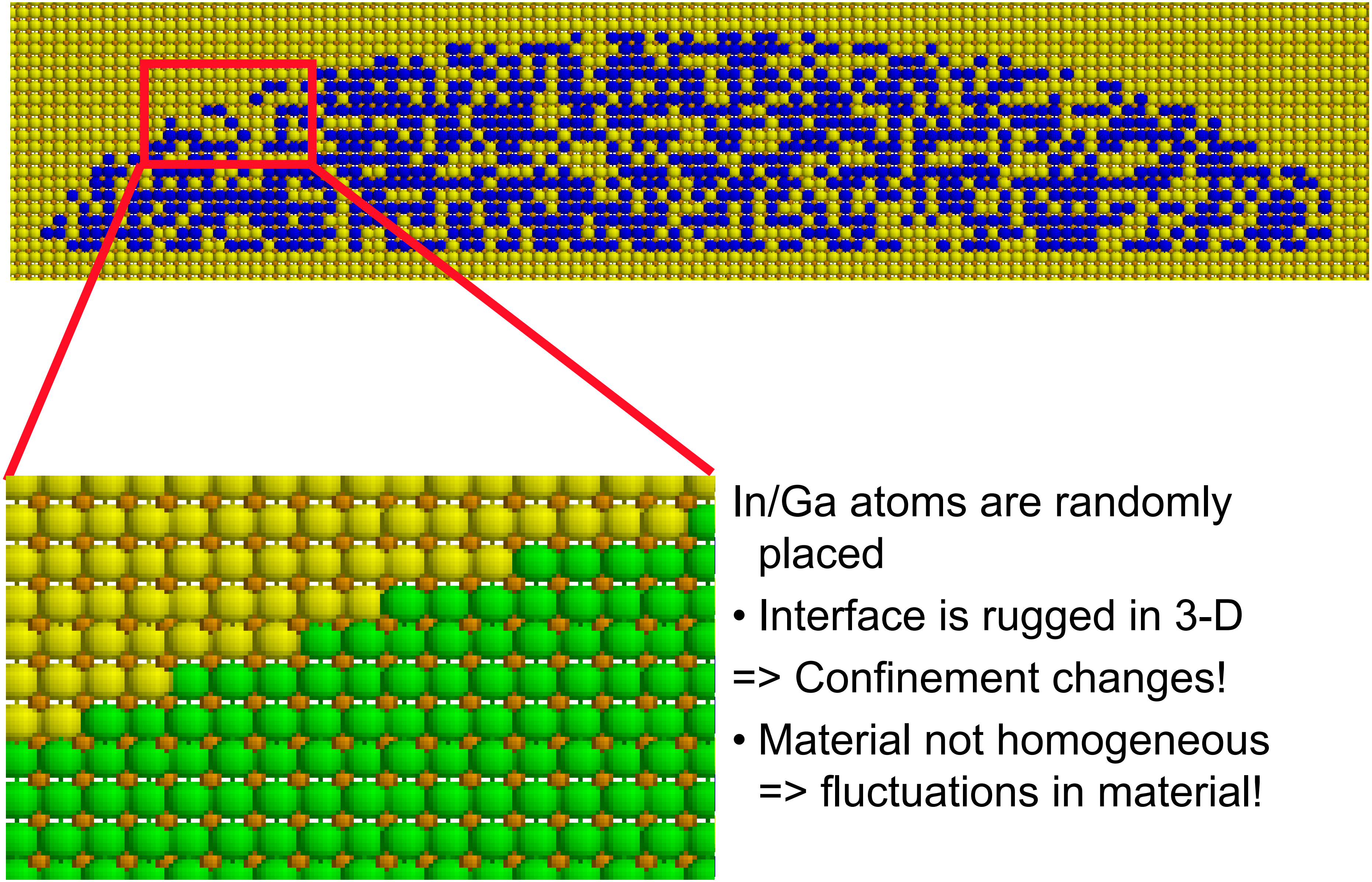
- In and Ga atoms are randomly placed
- Real alloys are spatially disordered
- Different samples “look” different => statistical properties

Typical approach: VCA

- average over In and Ga
- create an “average atom” “InGa”



What do interfaces REALLY mean now?



In/Ga atoms are randomly placed

- Interface is rugged in 3-D

=> Confinement changes!

- Material not homogeneous
  => fluctuations in material!

- VCA ignores all that!

Inhomogeneous Broadening due to Alloy Disorder

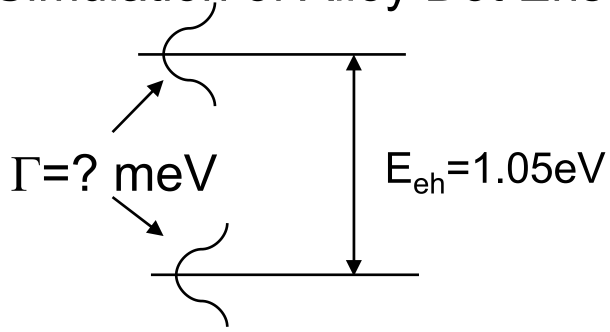
Problem:

- Cations are randomly distributed in alloy.
- Does alloy disorder limit electronic structure uniformity for dot ensembles?
- Requires atomistic simulation tool.

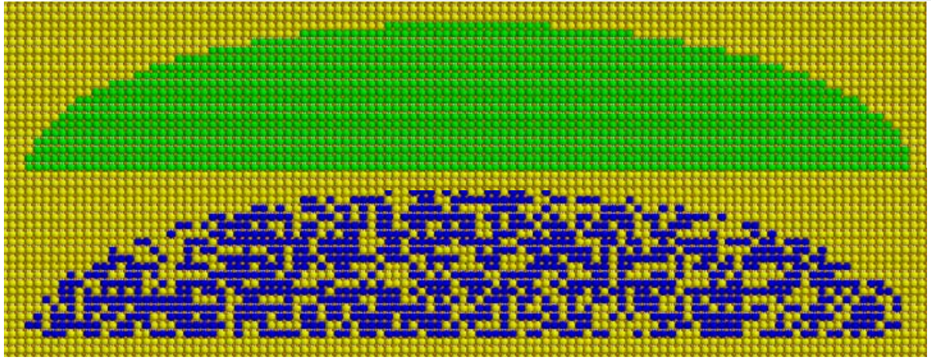
Approach:

- Simulate a statistical ensemble of dots.
  - Identical in size and shape
  - Different only in cation ordering.

Simulation of Alloy Dot Ensemble



Measured Γ=34.6 meV (R. Leon, PRB, **58**, R4262)



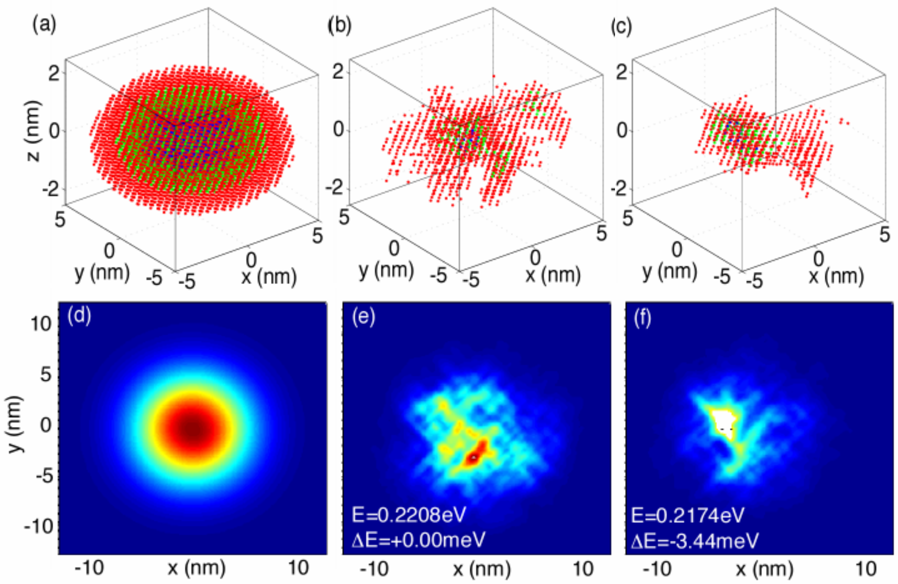
$In_{0.6}Ga_{0.4}As$ Lens Shaped Dot

Diameter=30nm,Height=5nm, GaAs embedded

~1,000,000 Atom Simulation, sp3s* basis

In and Ga atoms are randomly distributed

Inhomogeneous Broadening?



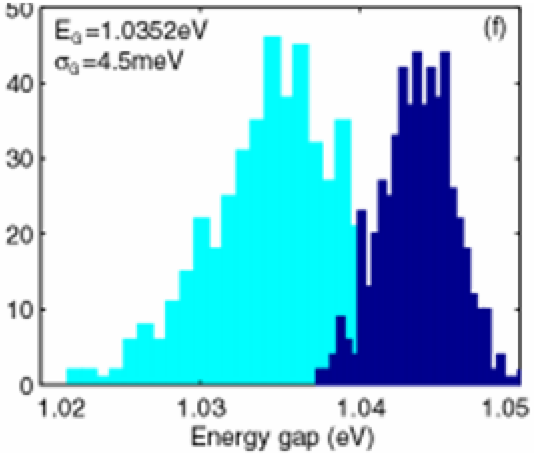
Inhomogeneous Broadening due to Alloy Disorder >1,000 quantum dot ensemble

Problem:

- Cations are randomly distributed in alloy.
- Does alloy disorder limit electronic structure uniformity for dot ensembles?
- Requires atomistic simulation tool.

Approach:

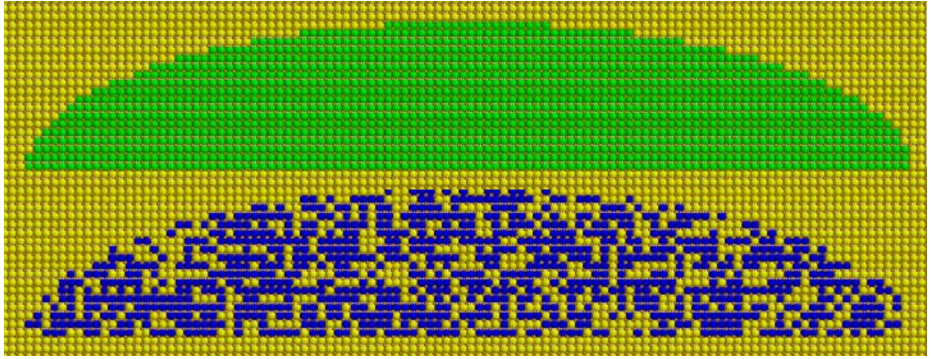
- Simulate a statistical ensemble of dots.
  - Identical in size and shape
  - Different only in cation ordering.



Simulated Γ=1 - 5meV

No size or shape fluctuations

Measured Γ=34.6 meV (R. Leon, PRB, **58**, R4262)



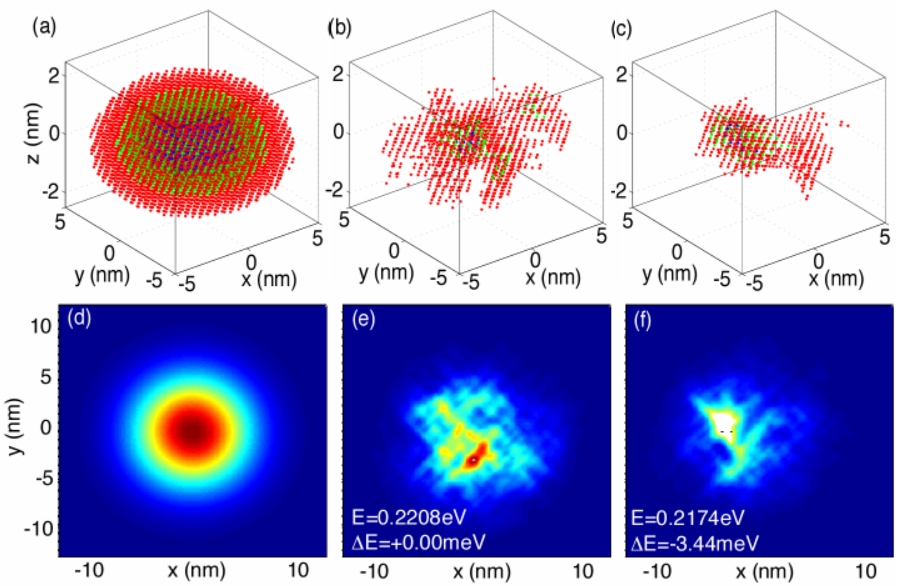
$In_{0.6}Ga_{0.4}As$ Lens Shaped Dot

Diameter=30nm,Height=5nm, GaAs embedded

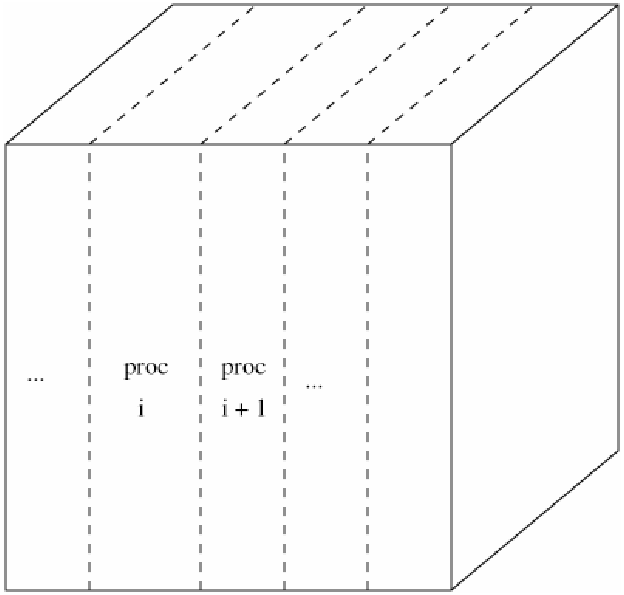
~1,000,000 Atom Simulation, sp3s* basis

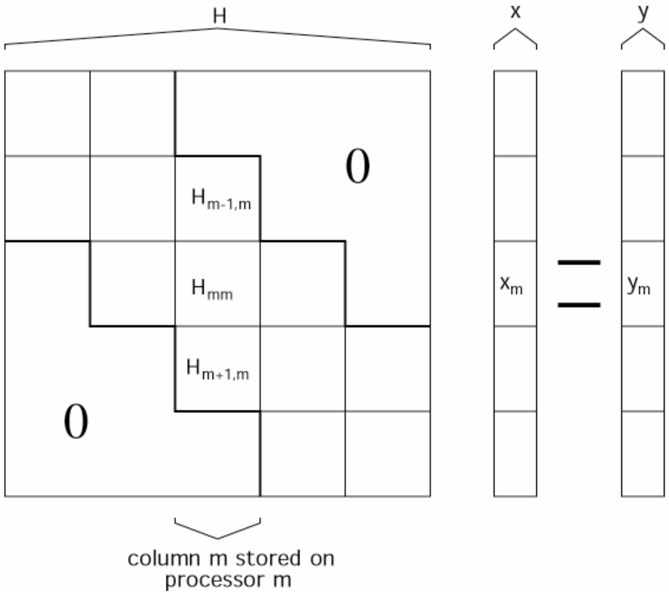
In and Ga atoms are randomly distributed

Inhomogeneous Broadening?



Parallelization and Methods





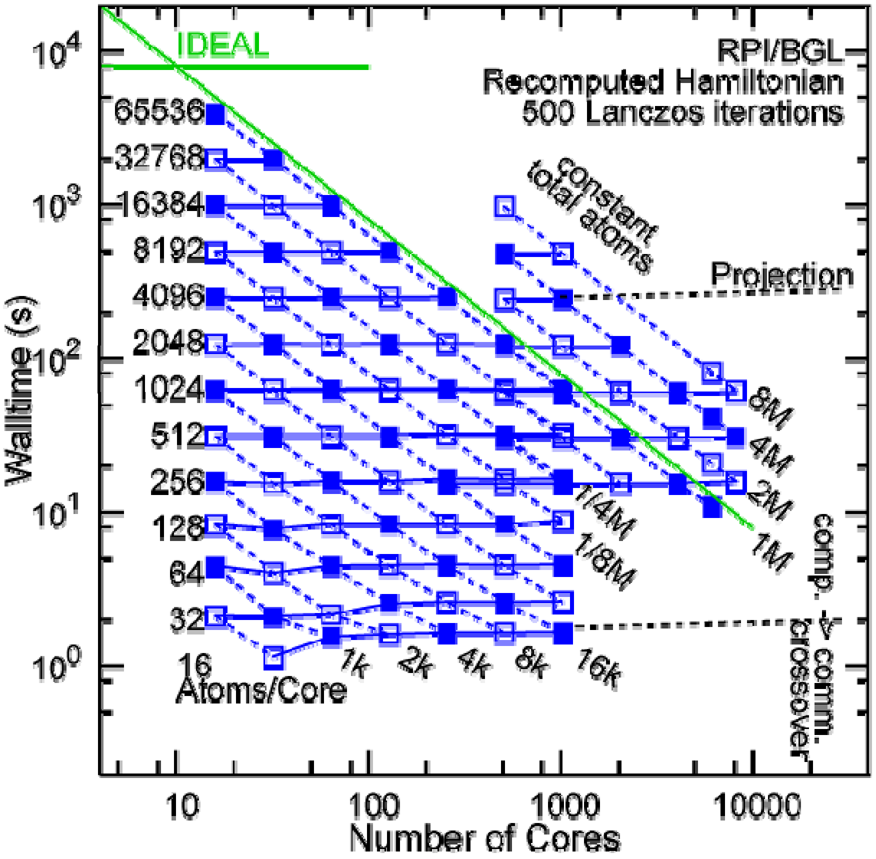
- Divide Simulation domain into slices.
- Communication only from one slice to the next (nearest neighbor)
- Communication overhead across the surfaces of the slices.

- Limiting operation:
  complex sparse matrix-vector multiplication
- Enable Hamiltonian storage or
  re-computation on the fly.

- Strain computed in open source CG algorithm
- Electronic structure needs eigenvalues and eigenvectors. Matrix is Hermitian
- Released NEMO 3-D methods:
  - Standard 2-pass Lanczos
  - PARPACK about 10x slower
  - Folded Spectrum Method (Zunger), also typically slower than Lanczos
  - Block Lanczos
  - Tracemin

Detailed Scaling Benchmarks Recomputed System Matrix

- IBM Blue Gene L – RPI
- June 2007 - #7 / TOP500

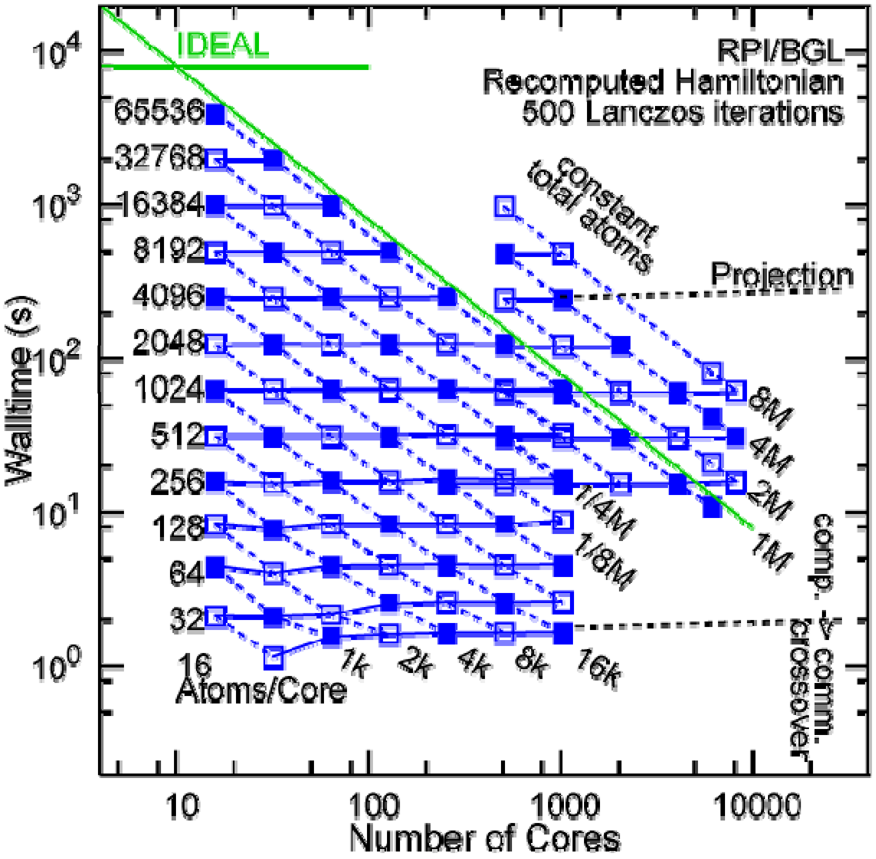


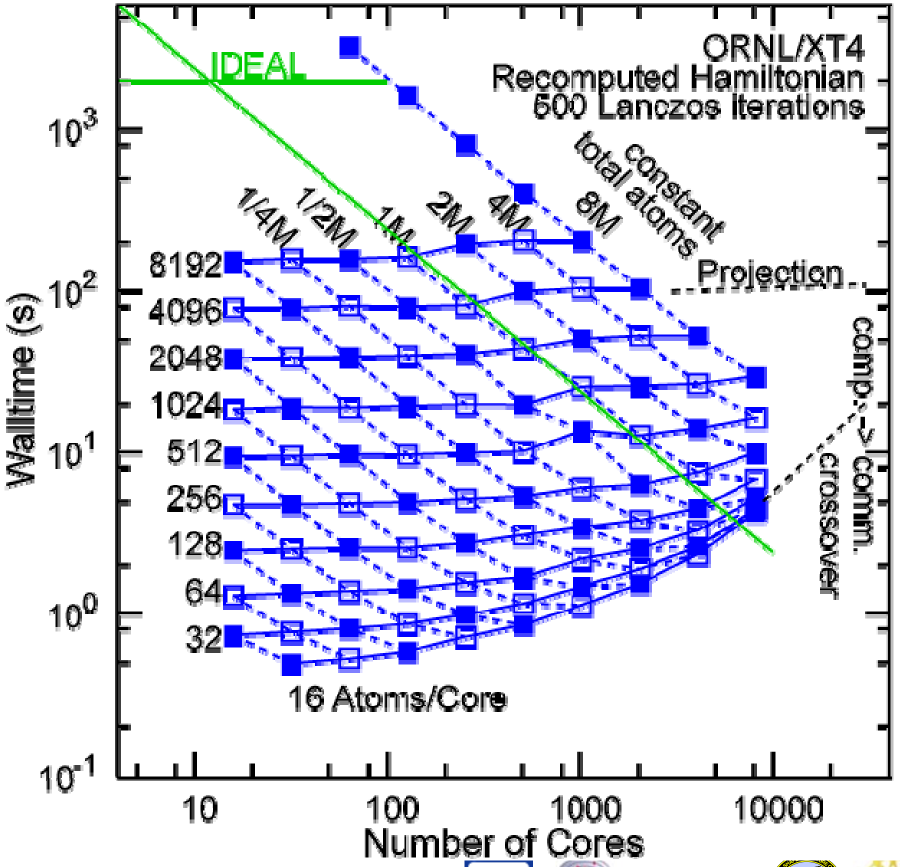
- Almost perfect scaling !

Detailed Scaling Benchmarks Recomputed System Matrix

- IBM Blue Gene L – RPI
- June 2007 - #7 / TOP500

- Cray XT3/4 – Oak Ridge Natl. Lab
- June 2007 - #2 / TOP500

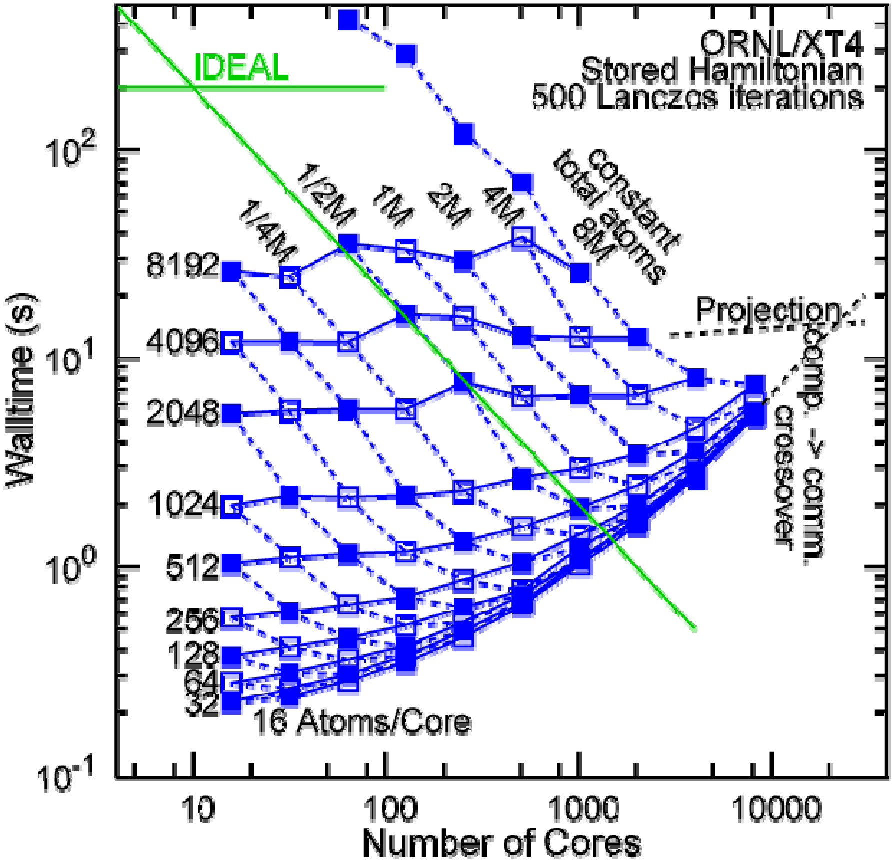


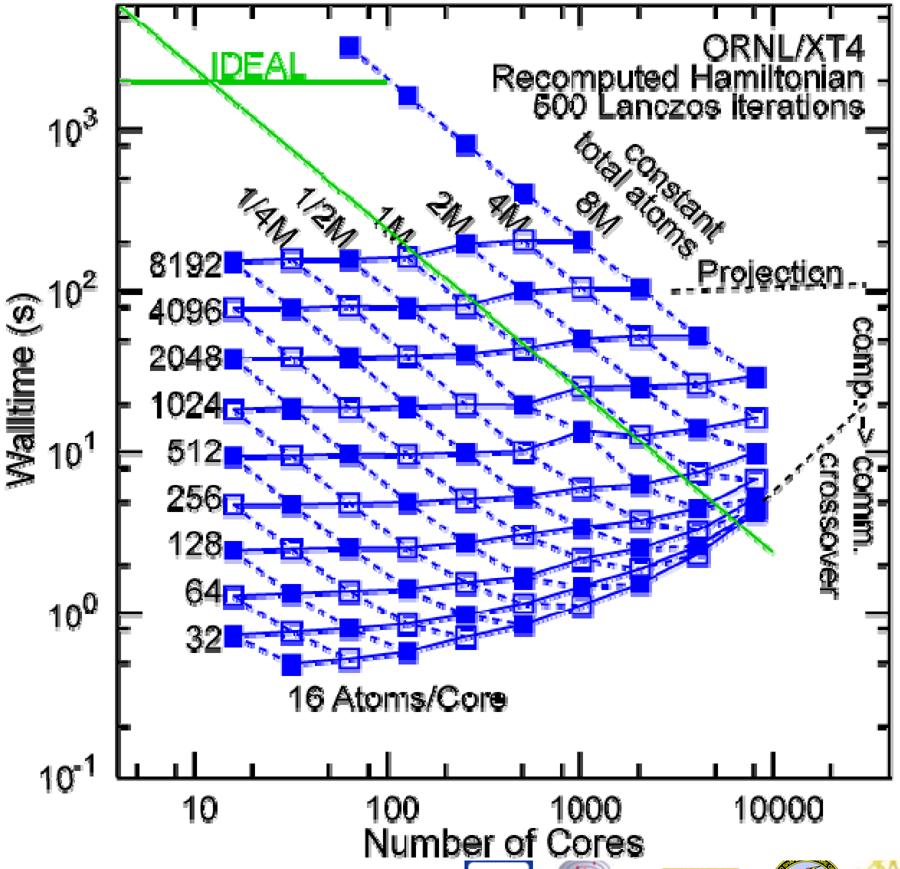


Detailed Scaling Benchmarks Cray XT3/4
Recomputed System Matrix vs Stored System Matrix

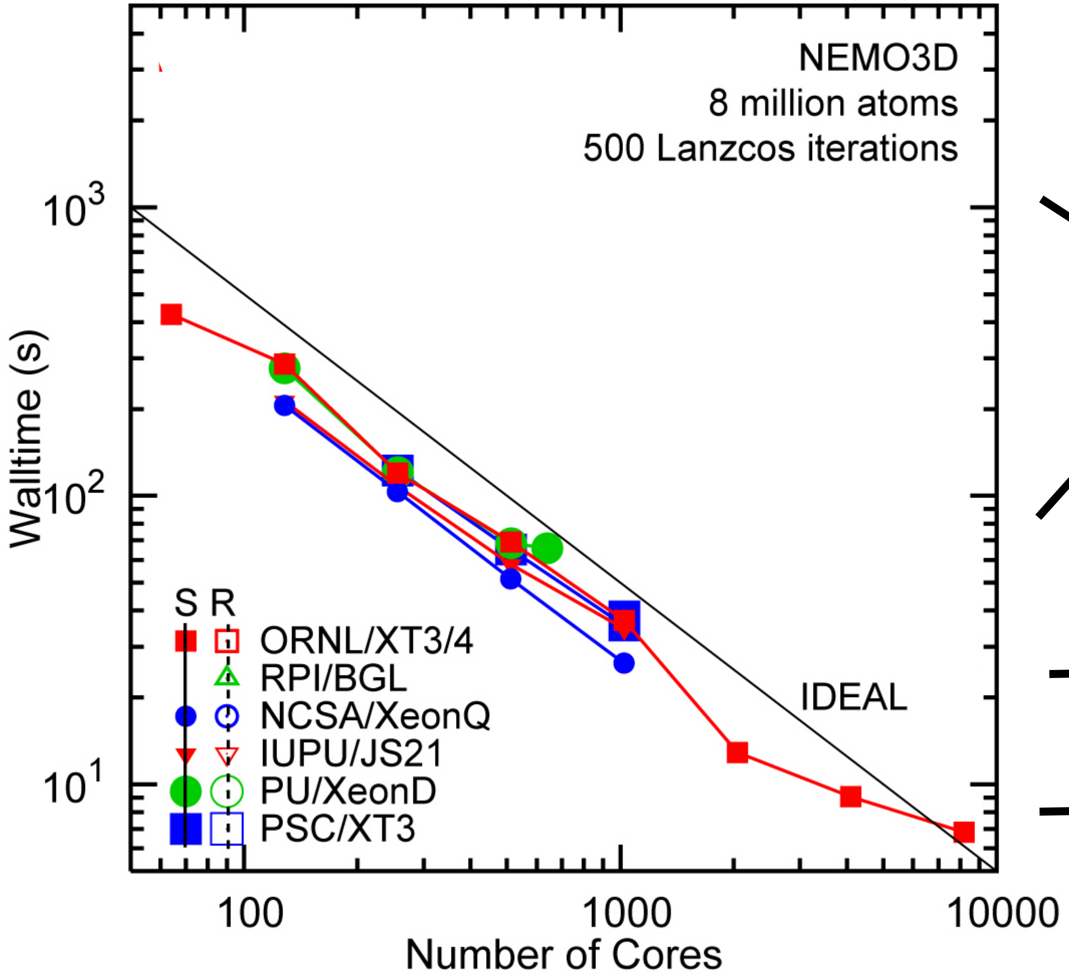
- Store the Sparse Matrix H for Hx=y

- Recompute the Sparse Matrix H for Hx=y





8 million atoms - Scaling up to 8,192 cores
TOP500: #2, #7, #8, #30, #46



Recompute Matrix

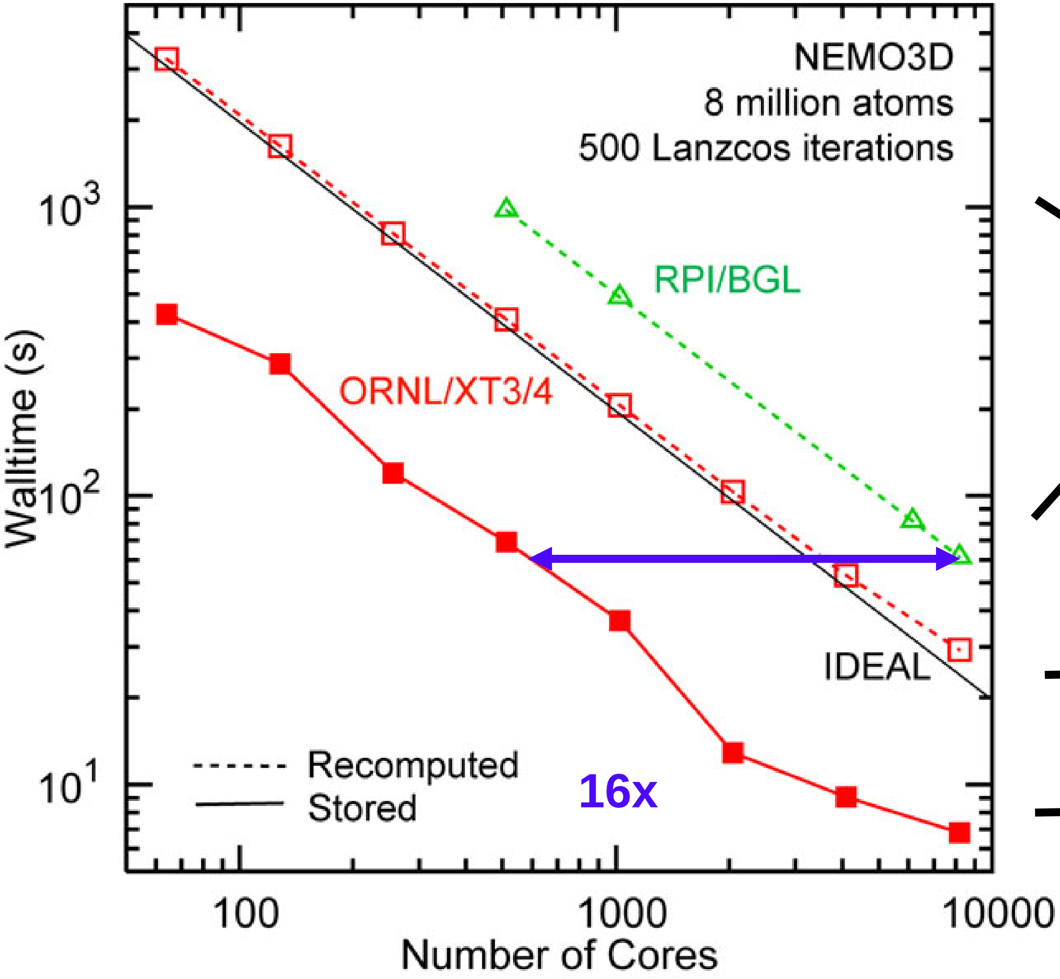
Store Matrix

- Clusters have enough memory – can store the Hamiltonian => speed
- Blue Gene small memory – need to recompute matrix on the fly
- XT3/4 best for NEMO3D

8 million atoms - Scaling up to 8,192 cores TOP500: #2, #7

Should I ask
my CIO or NSF
to buy:

1,000 XT4 cores?
or
16,000 BG/L cores



Recompute Matrix

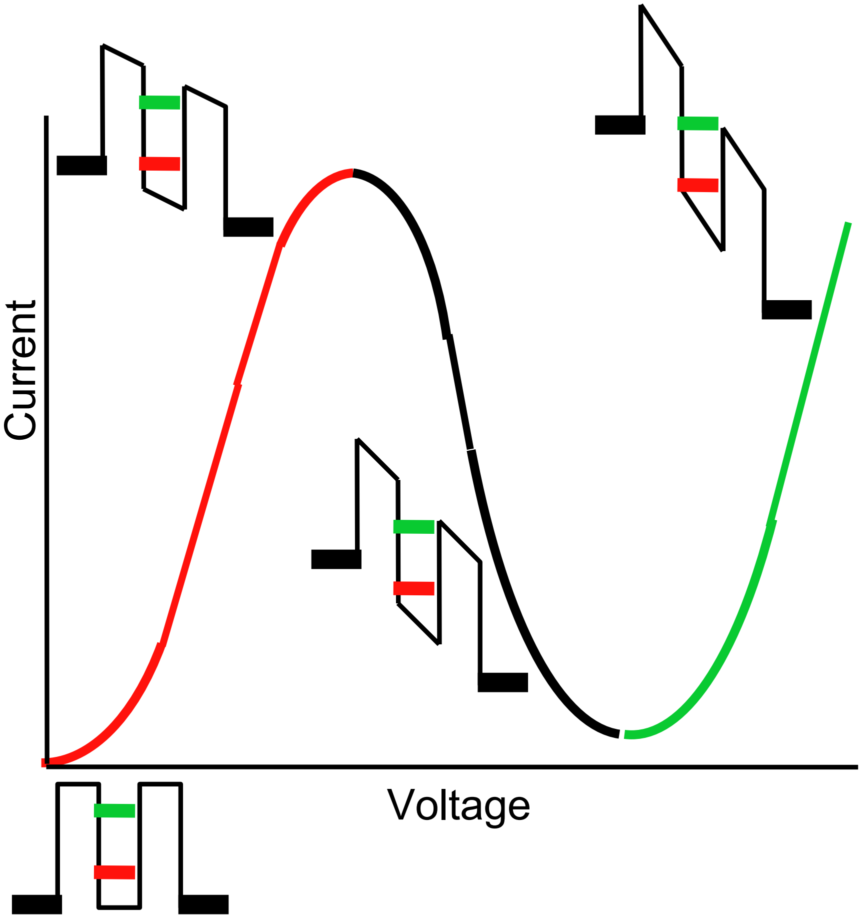
Store Matrix

- Clusters have enough memory – can store the Hamiltonian => speed
- Blue Gene small memory – need to recompute matrix on the fly
- XT3/4 best for NEMO3D

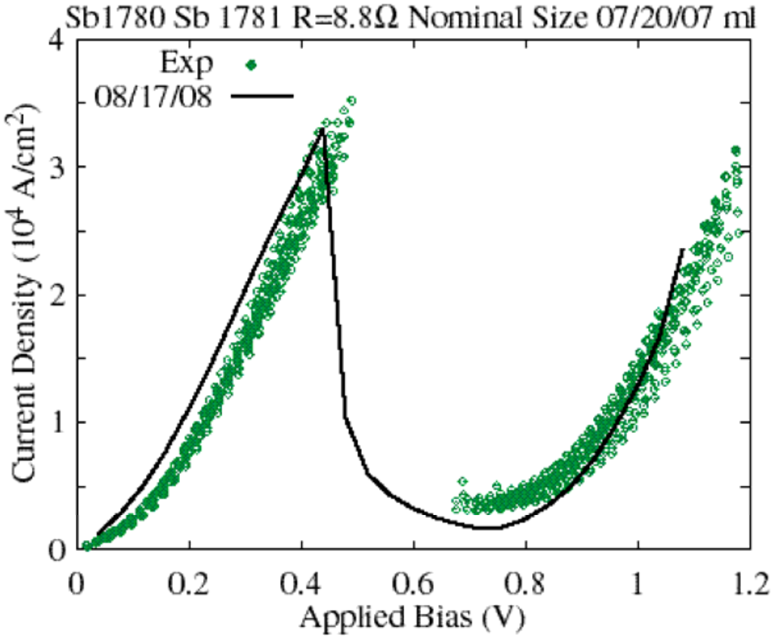
Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

- Network for Computational Nanotechnology (NCN)
- NEMO 3-D
  - »Physics motivation – multimillion atom simulation
  - »Basic algorithms
  - »Scaling to 8,192 processors
- NEMO 1-D
  - »Physics motivation –transport of electrons quantum statistical non-equilibrium mechanics
  - »Basic algorithms
  - »Scaling to 23,000 processors
- OMEN plans
- Algorithm work needed

Resonant Tunneling Diode



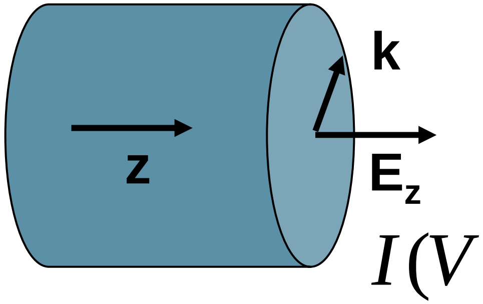
Conduction band diagrams for different voltages and the resulting current flow.



12 different I-V curves: 2 wafers, 3 mesa sizes, 2 bias directions

50nm	1e18	InGaAs
7 ml	nid	InGaAs
7 ml	nid	AlAs
20 ml	nid	InGaAs
7 ml	nid	AlAs
7 ml	nid	InGaAs
50 nm	1e18	InGaAs

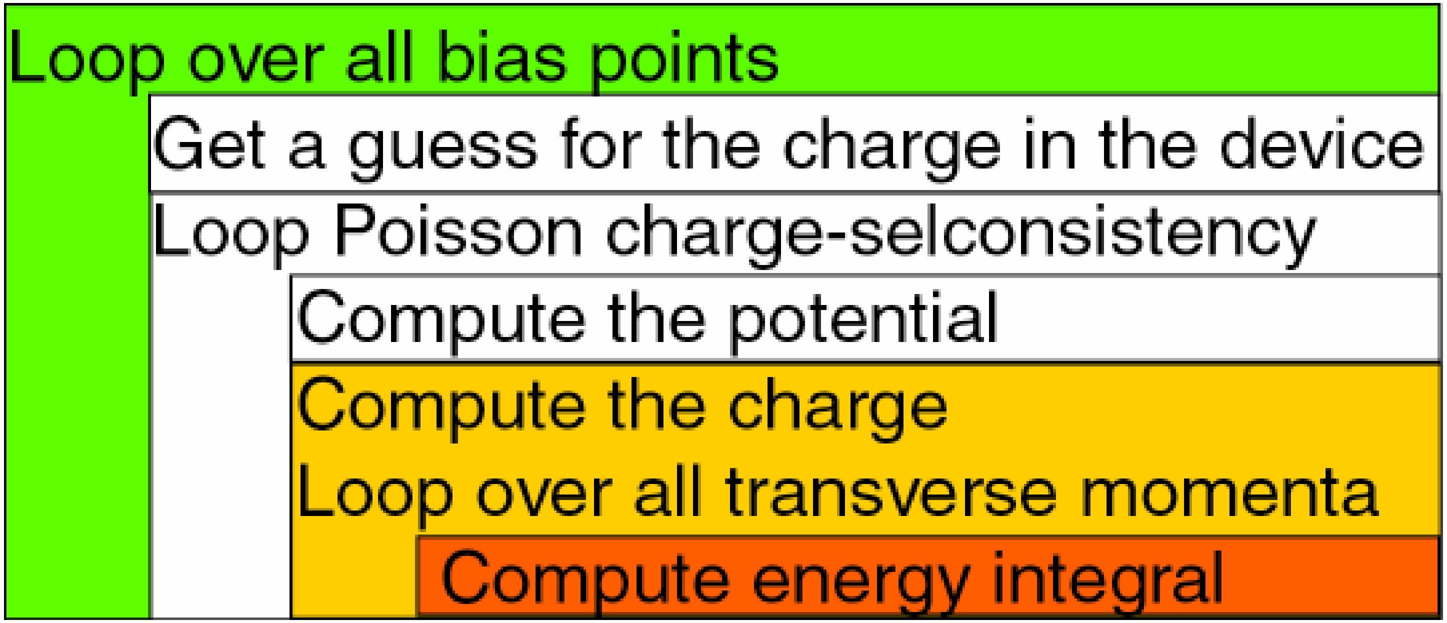
Algorithm Overview



2D integral for many bias points!

Total energy "E"
Momentum "k" accounts for transverse freedom

$$I(V) \propto 2\pi \int k dk \int dE T(E,k)\left(f_L(E,V) - f_R(E,V)\right)$$

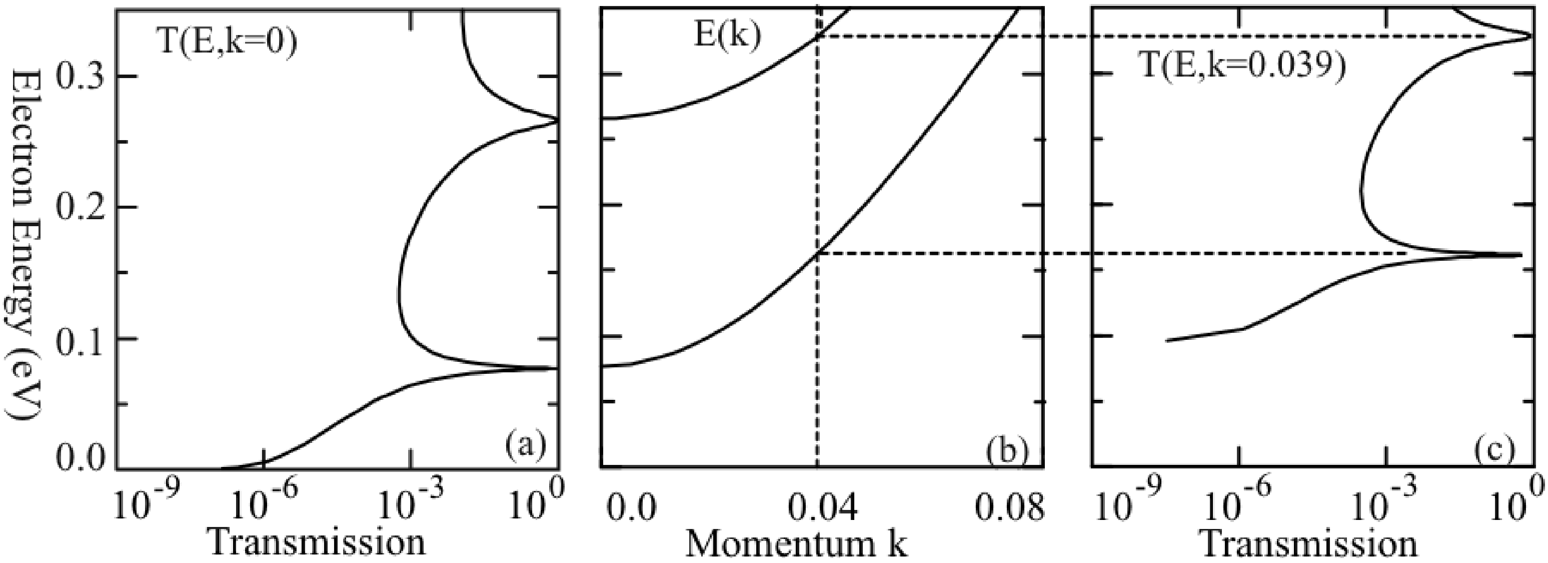


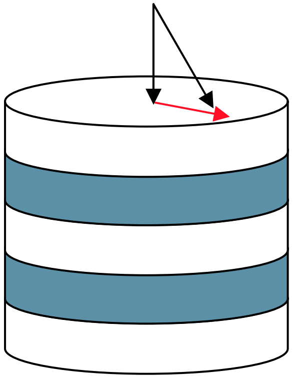
```
Loop over all bias points
    Get a guess for the charge in the device
    Loop Poisson charge-selconsistency
        Compute the potential
        Compute the charge
        Loop over all transverse momenta
            Compute energy integral
```

Have three levels of possible parallelization:

• **Loop over bias points**	**coarse grain**
• **Loop over momentum points**	**medium grain**
• **Loop over energy points**	**fine grain**

Dispersion in the Transverse Direction
Electron vs. Hole Subbands

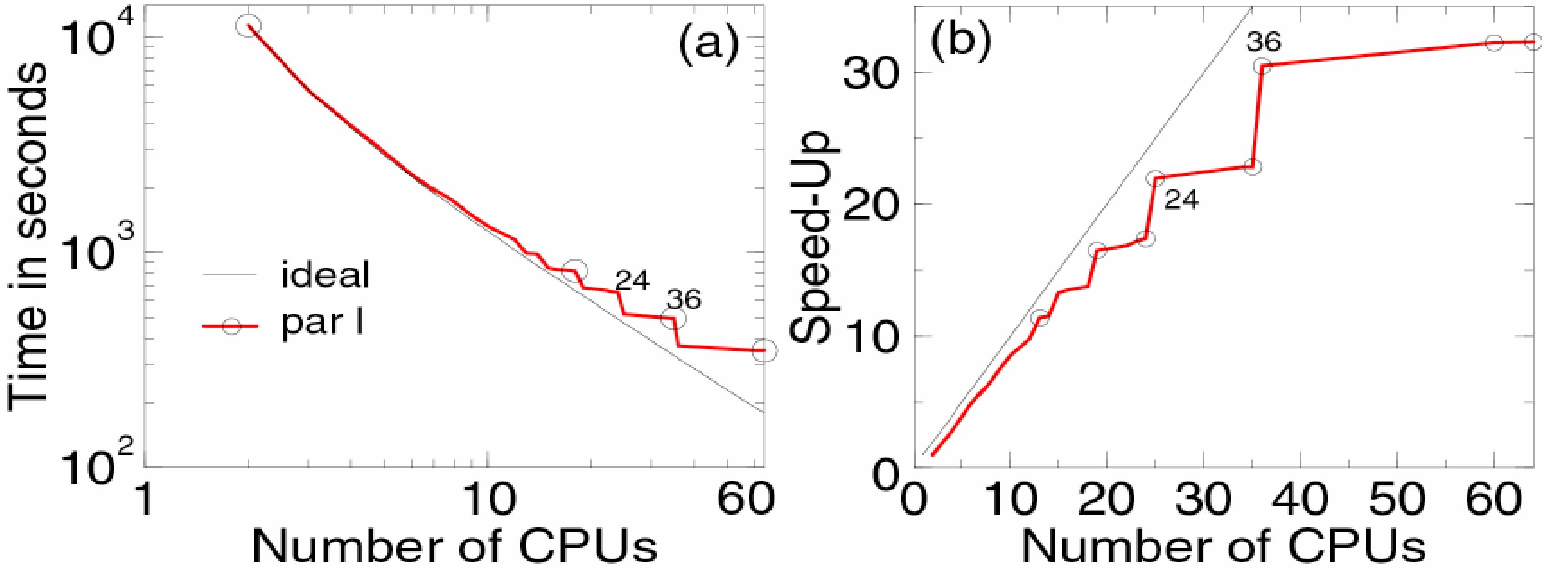




Electron:

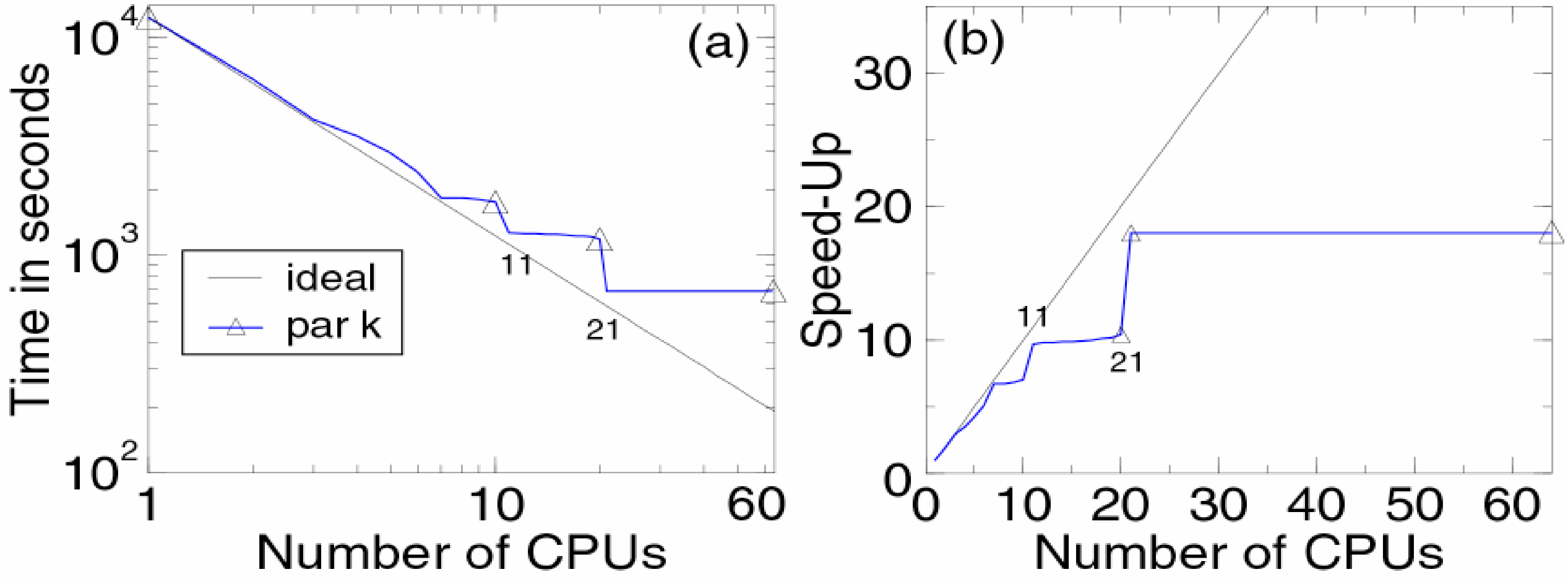
- Dispersion "simple", almost parabolic
- Transmission simple "replica"

Parallelization of Bias Points



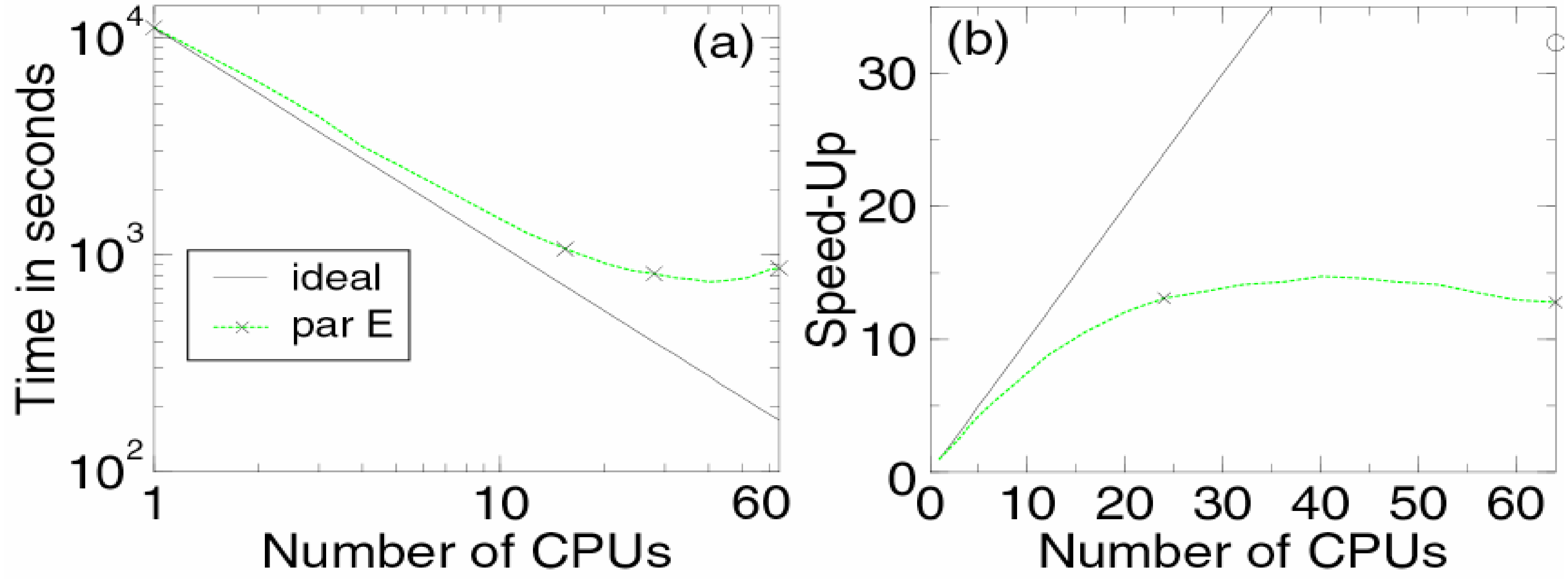
- Master-Slave approach
  - » master only distributes and gathers individual bias points
- This I-V has 70 bias points:
  - » Good scaling up to 15 CPUs, Strong steps at 36 and 24 CPUs / Load imbalance
    - ✓ Max speed-up: `21 at 24 CPU 87% efficiency`
      `30 at 36 CPU 83%`
      `efficiency`
    - ✓ Large CPU speed-up: `32 at 64 CPU 50% efficiency`

Parallelization of Momentum Integral

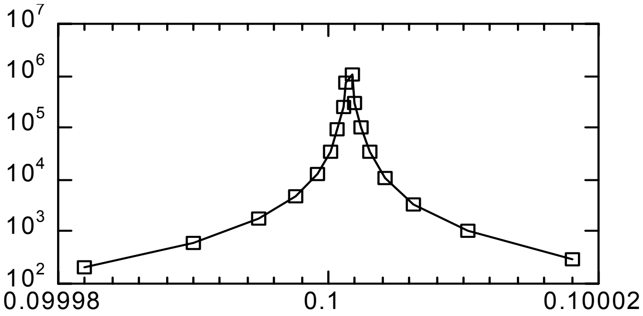


- K-grid has 21 points:
- Even bin distribution in one communication step.
- Results:
  - »Good scaling up to 8 CPUs, Strong steps at 11 and 21 CPUs
    - ✓Load imbalance
    - ✓Max speed-up: `10 at 11 CPU 91% efficiency` `18 at 21 CPU 86% efficiency`
    - ✓Large CPU Speed-up: `18 at 64 CPU 28% efficiency`

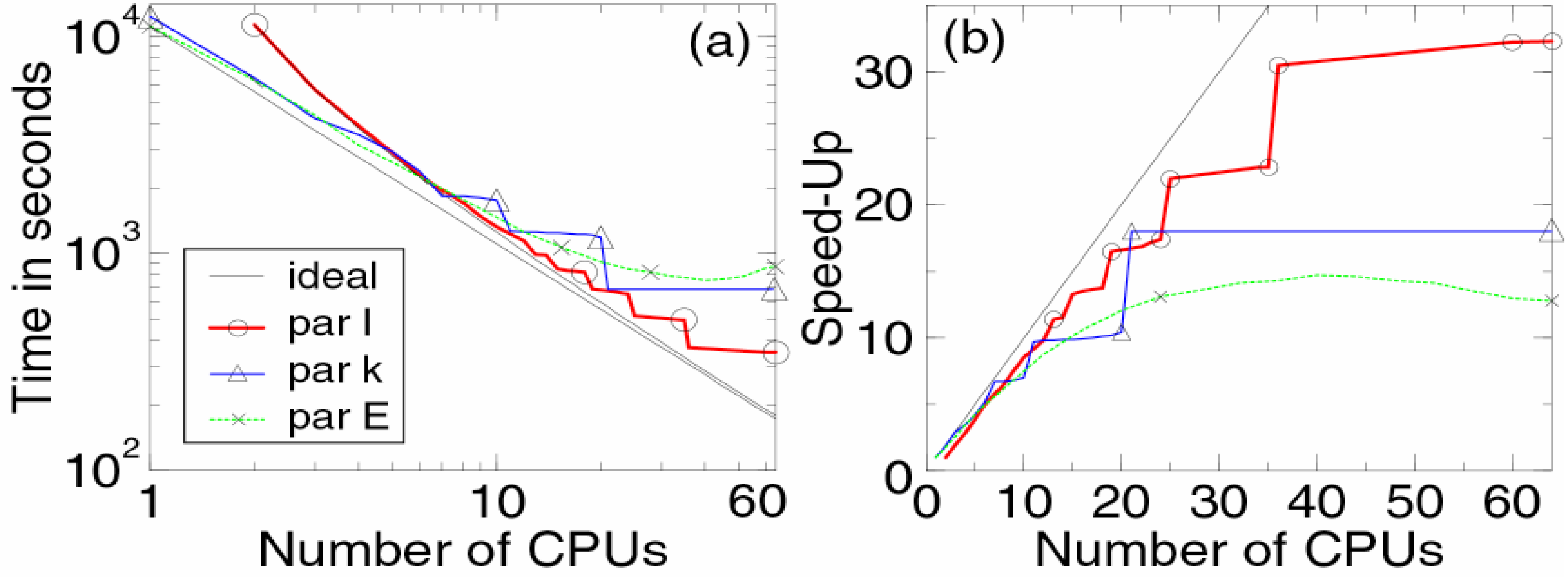
Parallelization of Energy Integral



- Adaptive mesh refinement starting from 50-200 nodes
- Refinement ends on 2 nodes at a time for 1 resonance
- Energy grid is at the lowest level
  -> communication overhead
- Maximum speed-up: `7.5 at 10 CPU   75% efficiency`
  `15 at  40 CPU   38% efficiency`
- Large CPU speed-up: `13 at 64 CPU   20%`



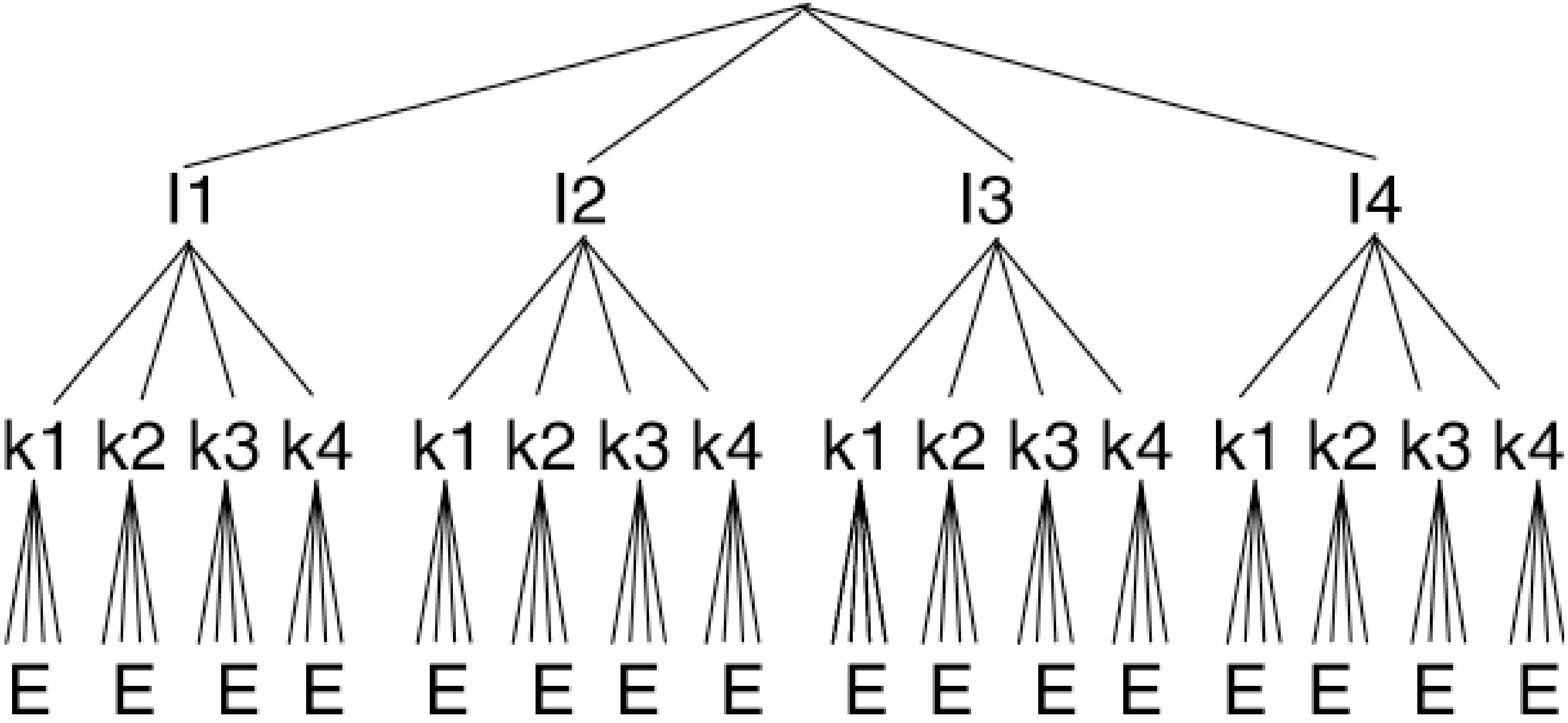
Parallelization of One Level



- Parallelization on 64 CPUs is unsatisfactory in all one level algorithms!
  - » For this particular benchmark - typical for electron RTD computation

- How about parallelization on multiple levels?

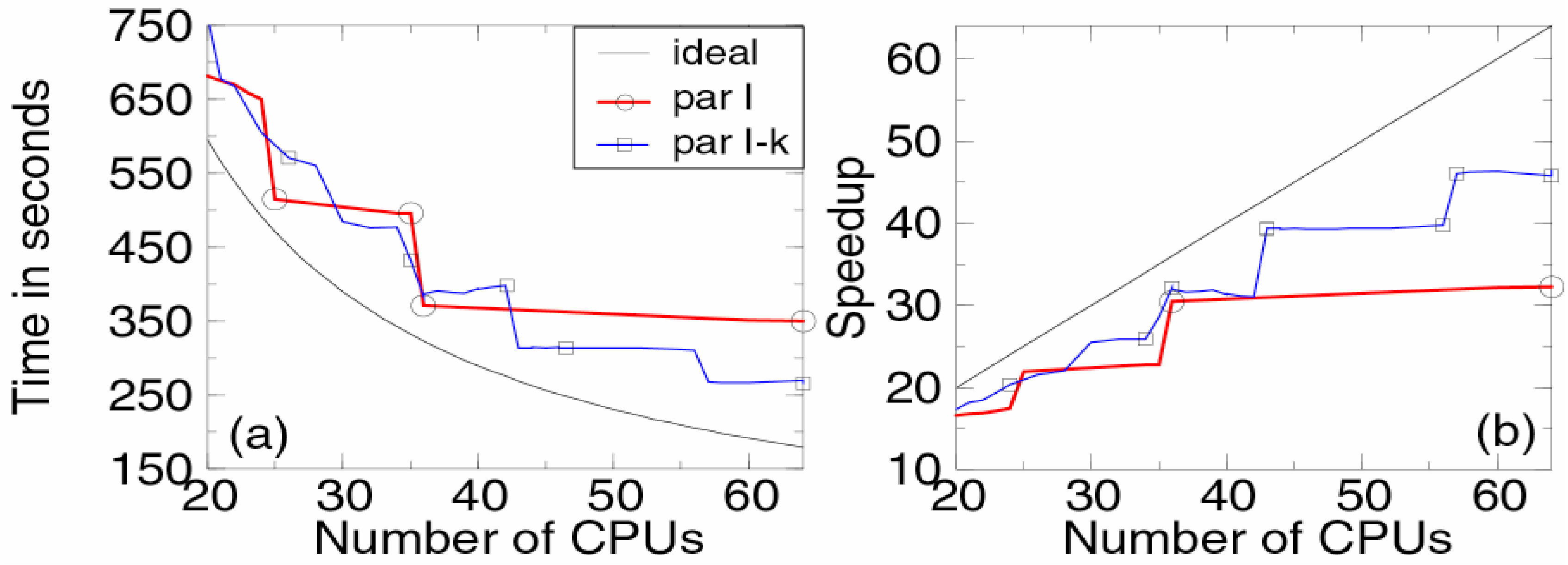
Parallelization of Multiple Levels


l1
l2
l3
l4
k1 k2 k3 k4 k1 k2 k3 k4 k1 k2 k3 k4 k1 k2 k3 k4
E E E E E E E E E E E E E E E E

- Parallelization on 64 CPUs is unsatisfactory in all one level algorithms!

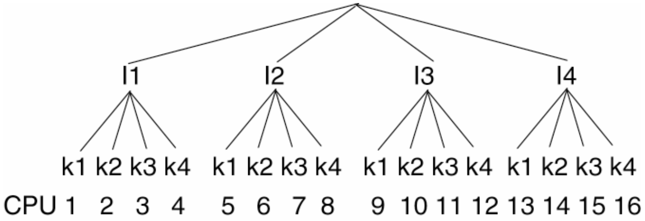
- How about parallelization on multiple levels?
  - » 4 possibilities: I-K, I-E, K-E, I-K-E all of them implemented
  - » Try to maximize number of CPUs in the coarser grids

Parallelization of Bias and Momentum



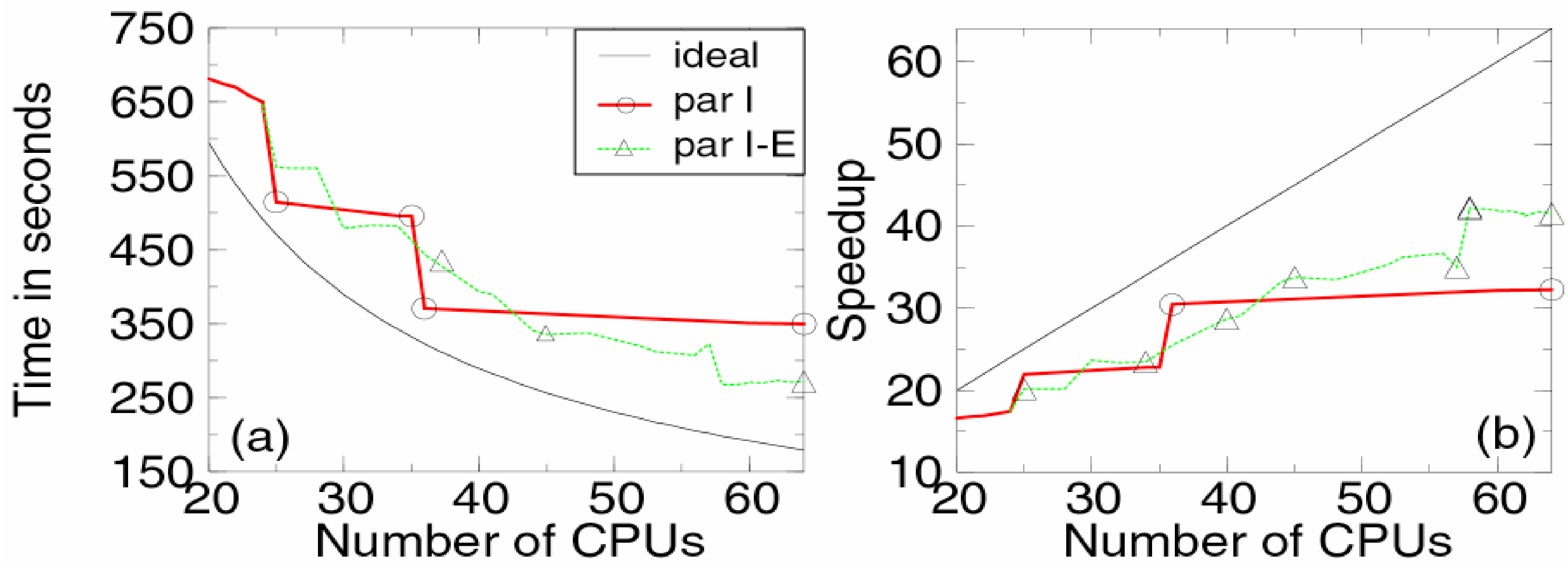
- Still see some load imbalance problems

- Maximum speed-up: `40 at 43 CPU 93% efficiency`
  `46 at 57 CPU 81% efficiency`
- Large CPU speed-up: `45 at 64 CPU 70% efficiency`



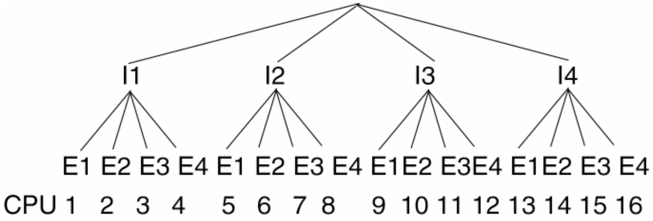
- Compared to l parallelism only:

Parallelization of Bias and Energy



- Much fewer load imbalance problems

- Maximum speed-up: 43 at 58 CPU 74% efficiency
- Large CPU speed-up: 42 at 64 CPU 65% efficiency



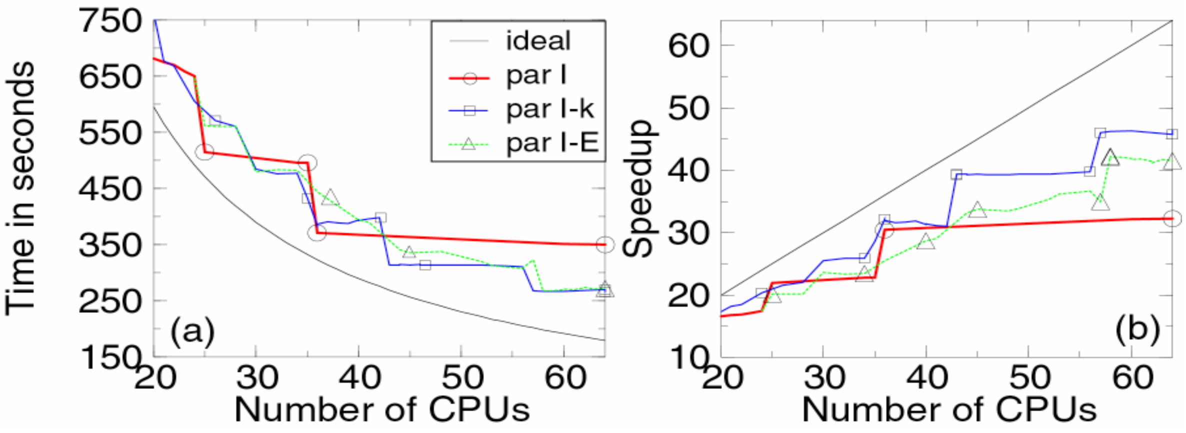
I1 I2 I3 I4

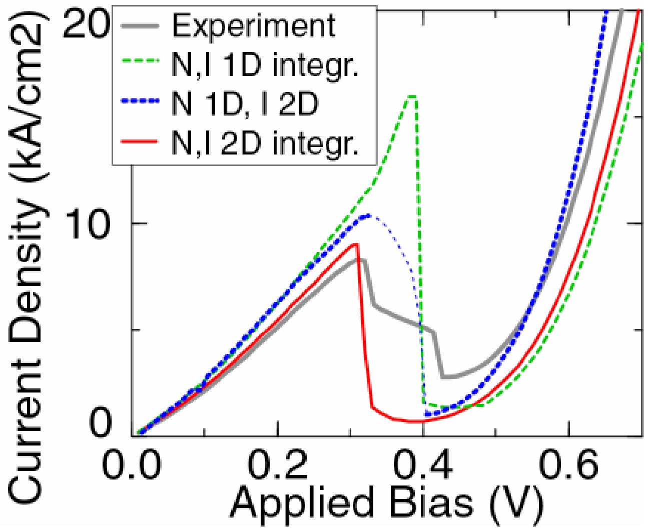
E1 E2 E3 E4 E1 E2 E3 E4 E1E2 E3E4 E1E2 E3 E4

CPU 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

- Compared to I parallelism only:
- Large CPU speed-up: 32 at 64 CPU 50%

NEMO 1-D Parallelization Conclusions year 2002



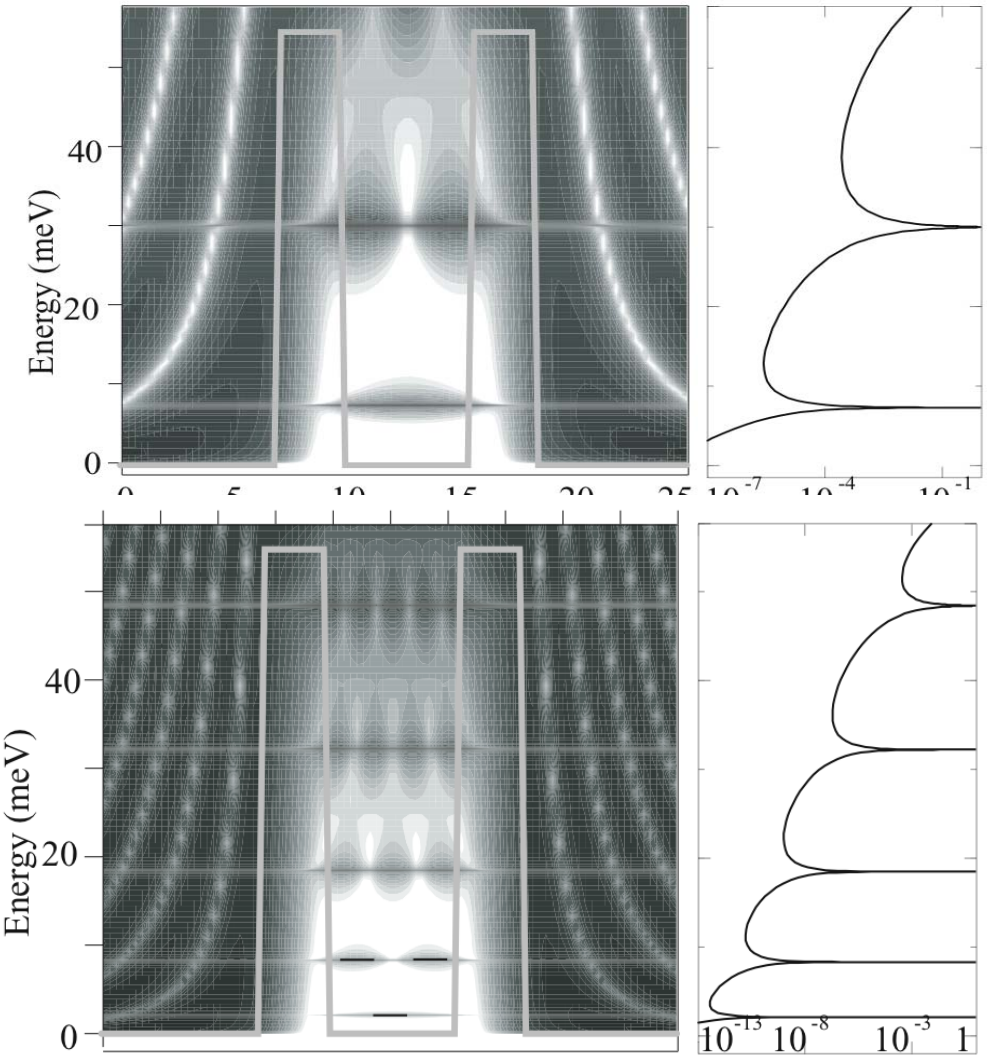


- Efficient parallelization for realistic RTD simulations is non-trivial.
- Parallelization on multiple levels:
  - » Flexibility to tackle different problems
  - » Enabled simulation of fully charge self-consistent simulations

    ✓Reduced compute time from 14 days to 6 hours.

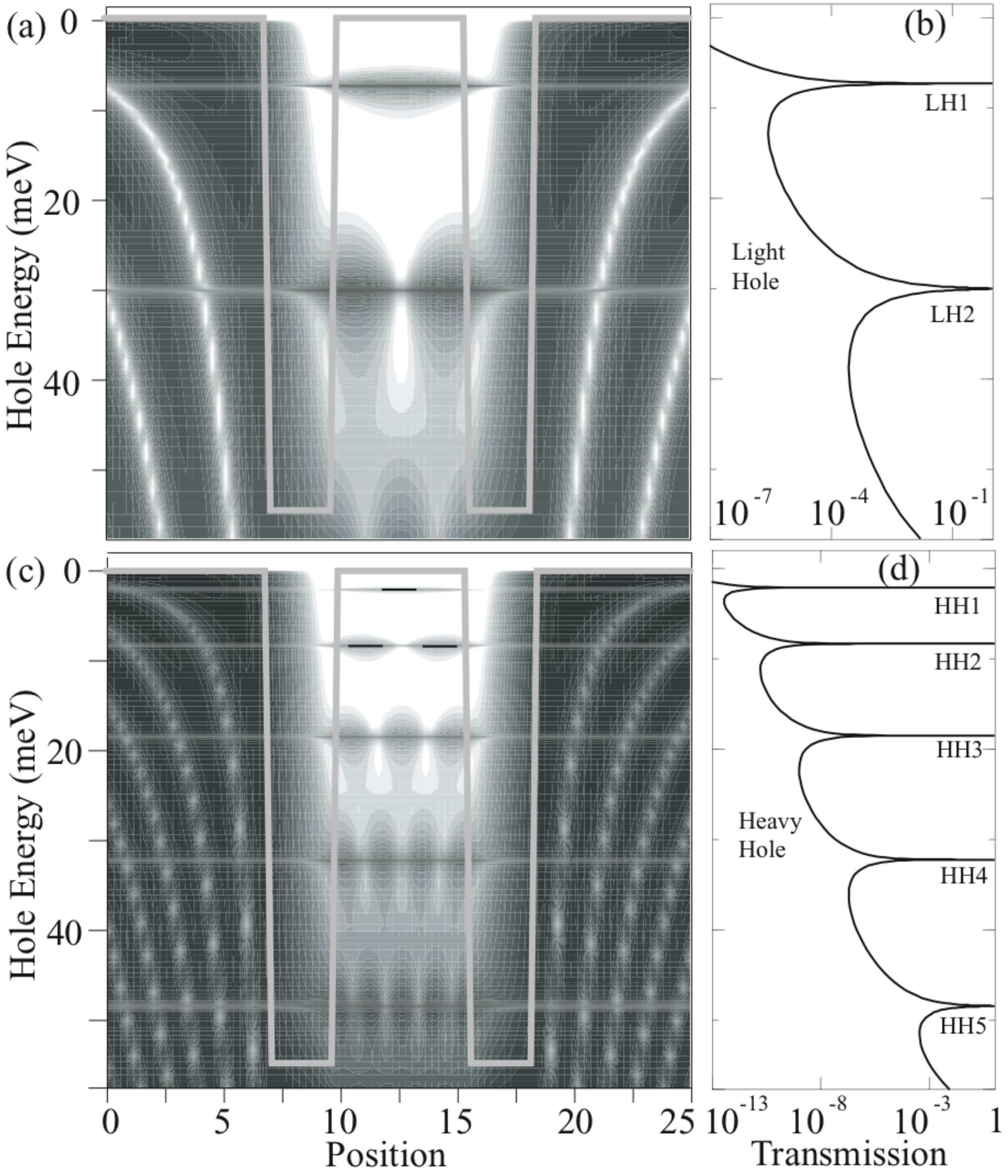
    YEAR 2002 – 64 node cluster
- Beowulf clusters are affordable and useful for computational electronics.

Electron transport in RTDs: Density of States and Transmission



- Density of States: Shows the spatial and energetic "location" of possible states
- Transmission: shows spikes where the DOS is strong in the central RTD
- Small effective mass: large state separation
- Large effective mass: "heavy" electrons
  - » small state separation
  - » Sharp peaks - strong confinement weak coupling to outside
  - » Deep background/peak ratio: 10^{13} strong confinement weak coupling to outside

Hole transport in RTDs: Simplified Density of States and Transmission



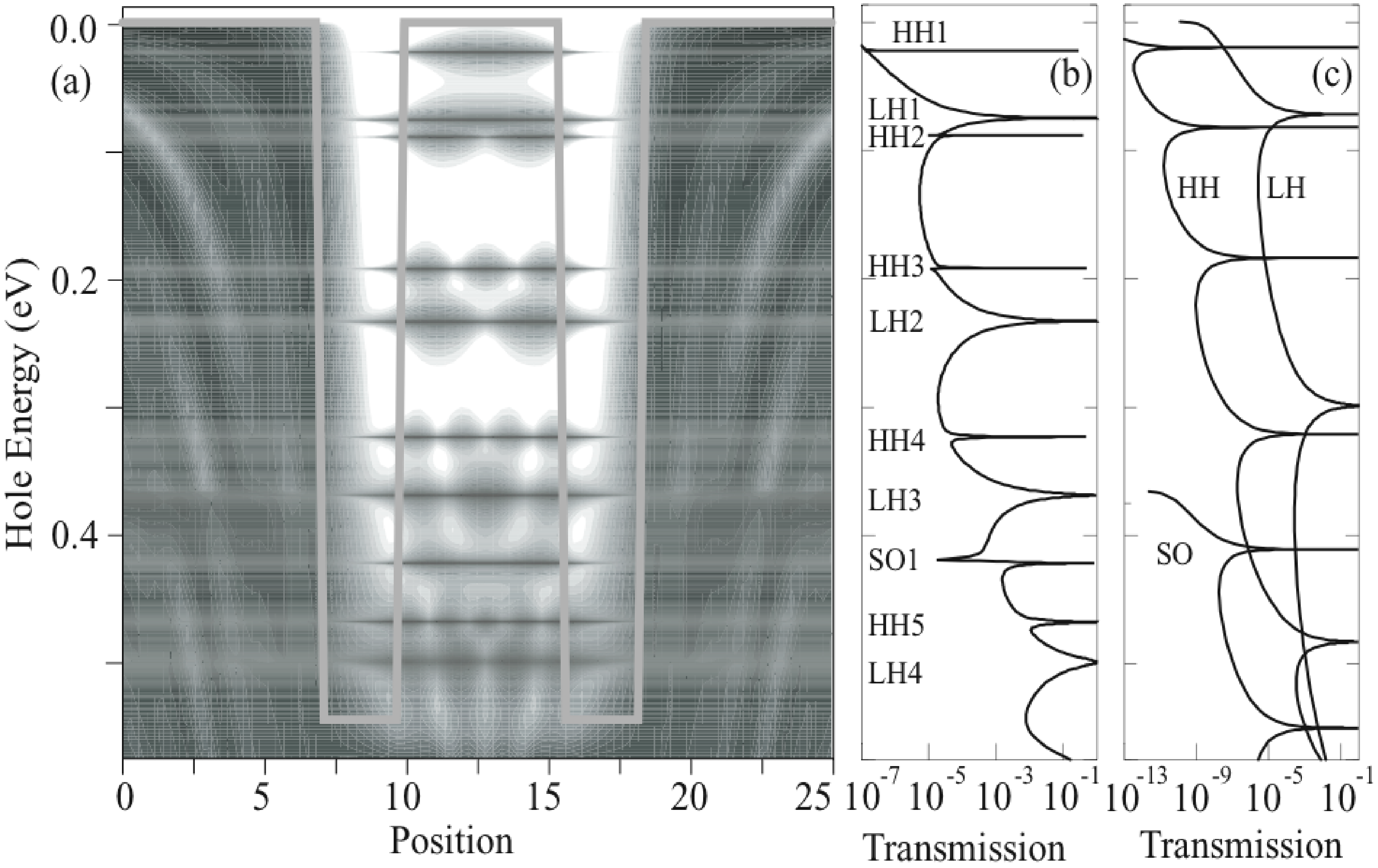
- Holes

  Are just electrons

  upside-down????

- Not quite!
  - » LH and HH are coupled
  - » Highly non-parabolic dispersion
  - » Highly anisotropic dispersion
- Very unintuitive transport behavior!

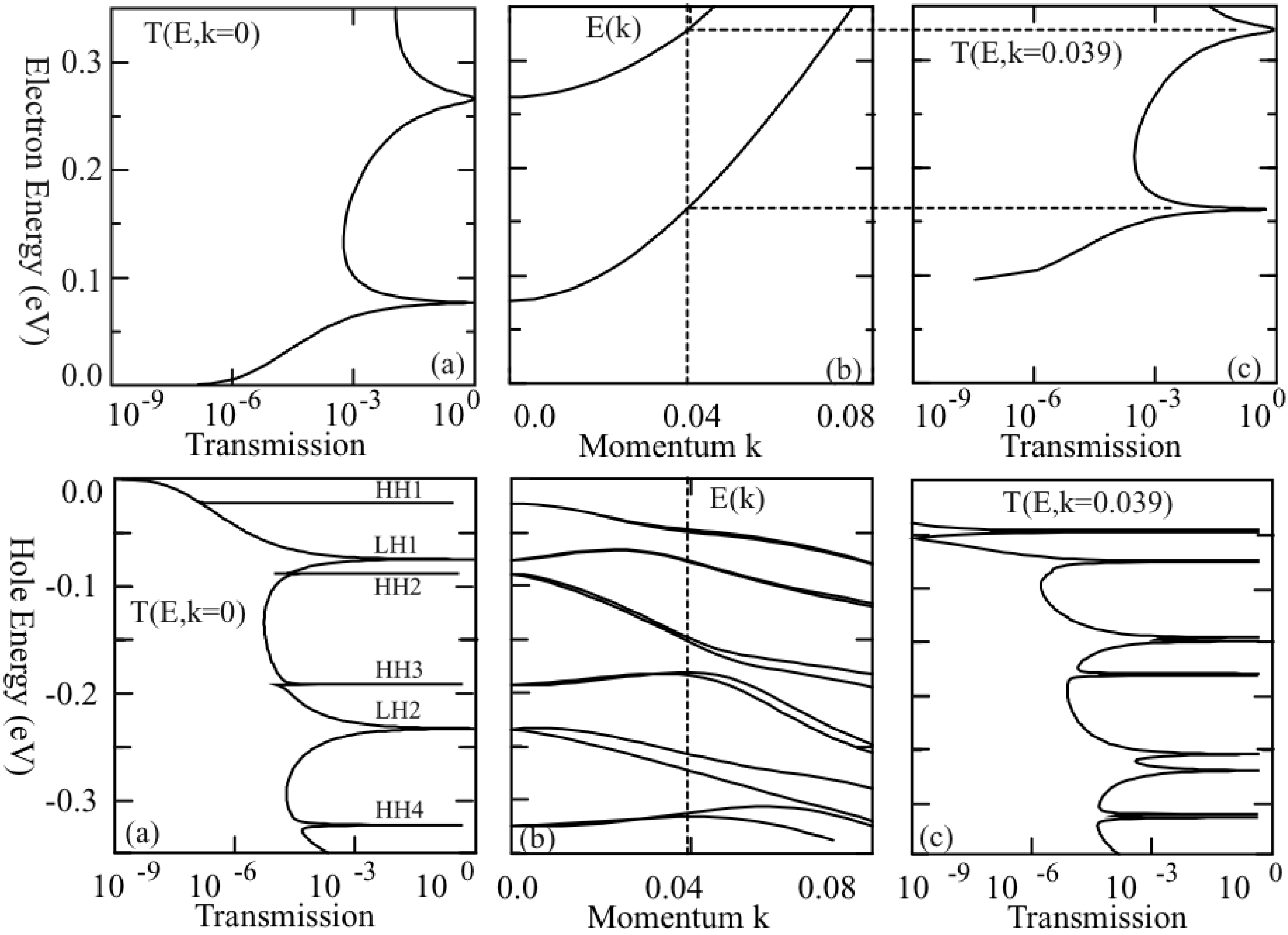
Hole Transport in RTDs sp3s* full band model

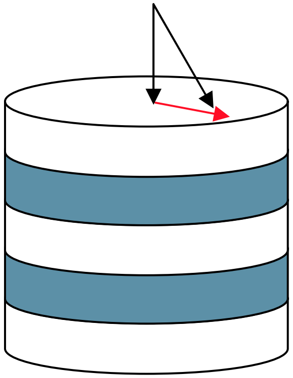


- Transmission coefficient at k_x=0
- sp^3s^* represents all bands simultaneously. Can identify LH, HH, and SO features

Gerhard Klimeck

Dispersion in the Transverse Direction Electron vs. Hole Subbands





Electron:

- Dispersion "simple", almost parabolic
- Transmission simple "replica"

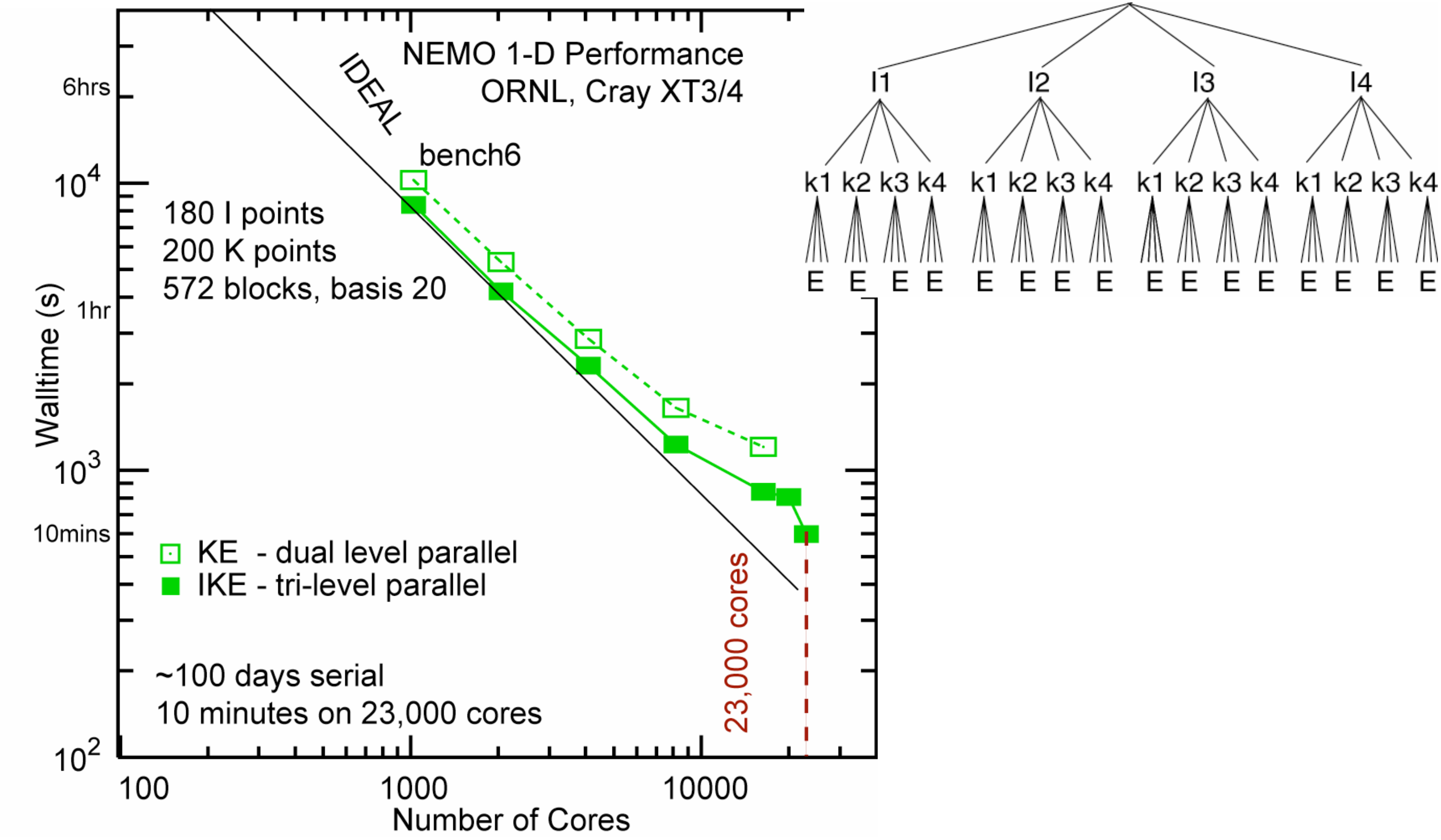
Holes:

- Dispersion "complicated"
- Transmission dramatically altered

Gerhard Klimeck

NEMO 1-D on ORLN Cray XT3/4
Almost Perfect scaling to 23,000 cores!

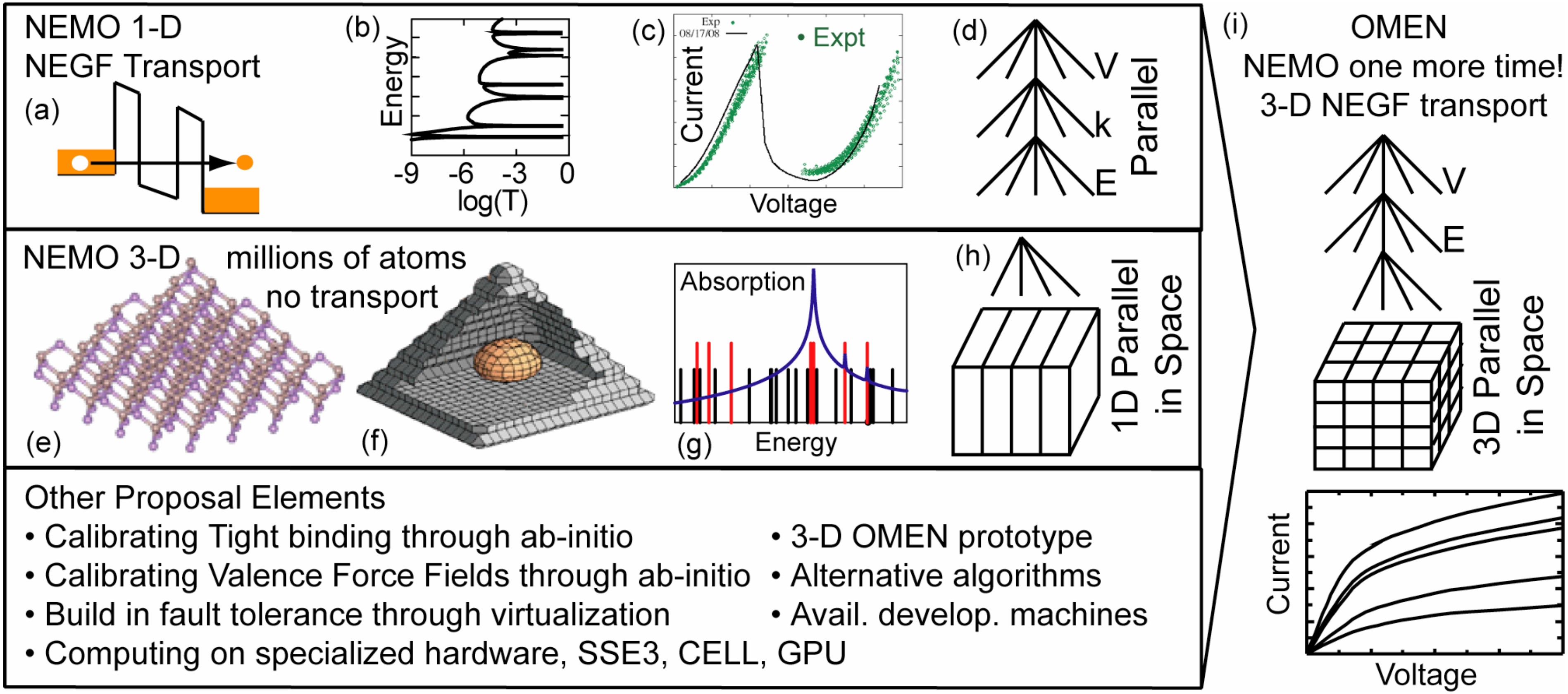
$$I(V) \propto 2\pi \int k dk \int dE T(E,k)\left(f_L(E,V) - f_R(E,V)\right)$$



Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

- Network for Computational Nanotechnology (NCN)
- NEMO 3-D
  - »Physics motivation – multimillion atom simulation
  - »Basic algorithms
  - »Scaling to 8,192 processors
- NEMO 1-D
  - »Physics motivation –transport of electrons quantum statistical non-equilibrium mechanics
  - »Basic algorithms
  - »Scaling to 23,000 processors
- OMEN plans
- Algorithm work needed

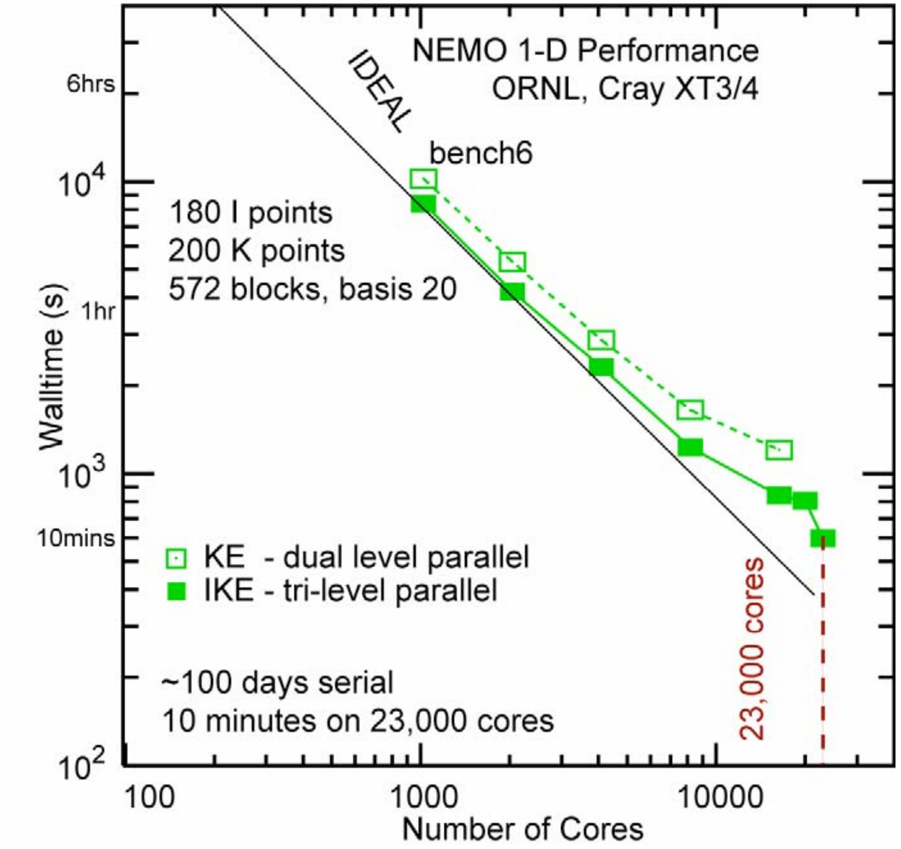
OMEN

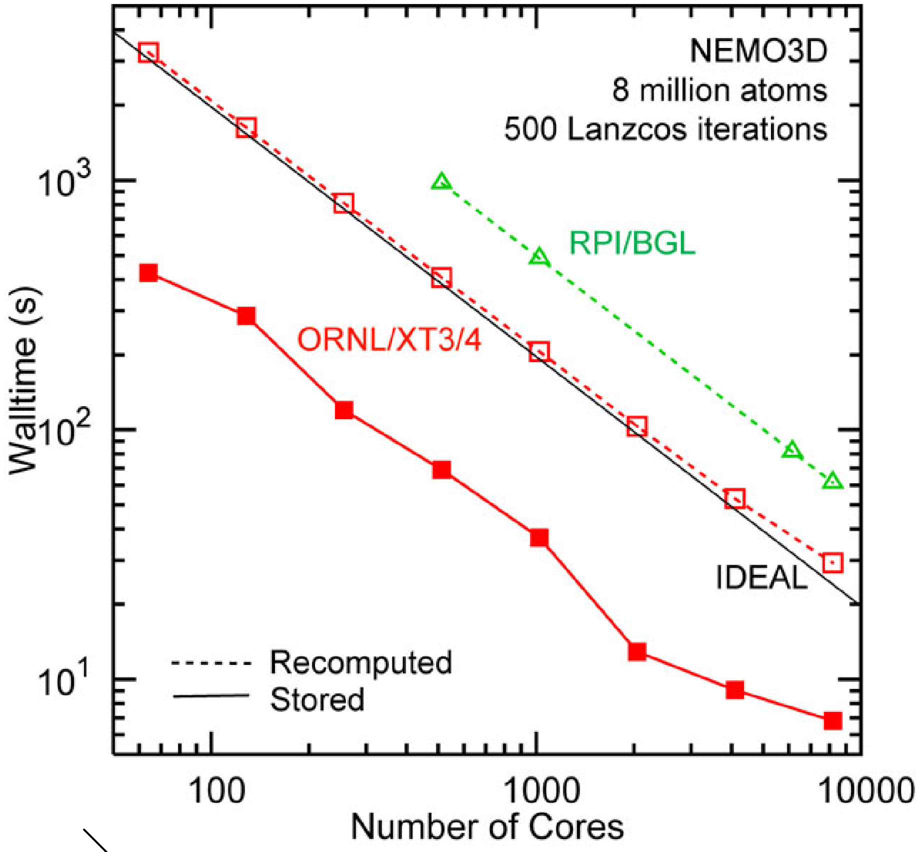


Other Proposal Elements

- Calibrating Tight binding through ab-initio
- Calibrating Valence Force Fields through ab-initio
- Build in fault tolerance through virtualization
- Computing on specialized hardware, SSE3, CELL, GPU
- 3-D OMEN prototype
- Alternative algorithms
- Avail. develop. machines

Moving towards the peta-scale





NEMO 1-D with NEGF

NEMO 3-D with millions of atoms

OMEN

Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

- Network for Computational Nanotechnology (NCN)
- NEMO 3-D
  - »Physics motivation – multimillion atom simulation
  - »Basic algorithms
  - »Scaling to 8,192 processors
- NEMO 1-D
  - »Physics motivation –transport of electrons quantum statistical non-equilibrium mechanics
  - »Basic algorithms
  - »Scaling to 23,000 processors
- OMEN plans
- Algorithm work needed

What is the NEGF method?

- Non Equilibrium Green’s Function method
- 1960s: Developed by the theoretical physicists
- 1990: Vision to use the NEGF as a quantum mechanical method to model nanodevices
- 2000s: Considered state-of-the-art modeling method to predict performance of emerging nanodevices
- Why work on Computational Algorithms?

Typical matrix equation

$$\begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b \end{bmatrix}$$

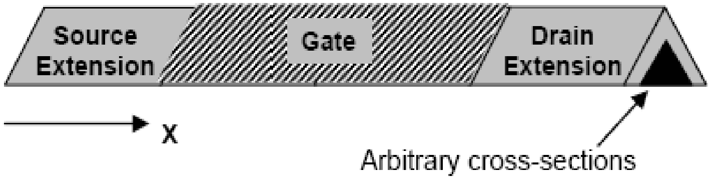
Find column vector x

NEGF matrix equation

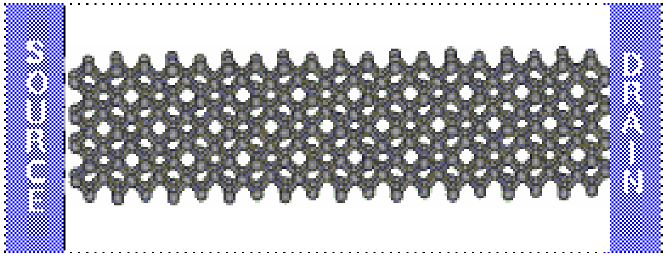
$$\begin{bmatrix} A \end{bmatrix} \begin{bmatrix} G \end{bmatrix} = \begin{bmatrix} I \end{bmatrix}$$

Find diagonal elements of G

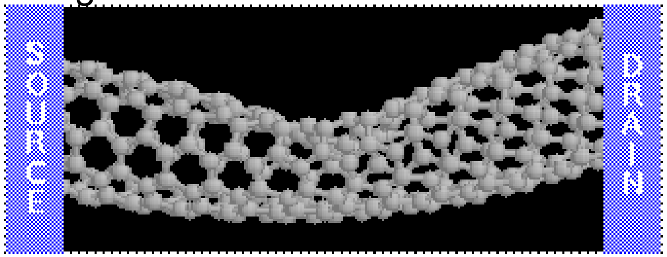
Quasi 1D Transport Devices



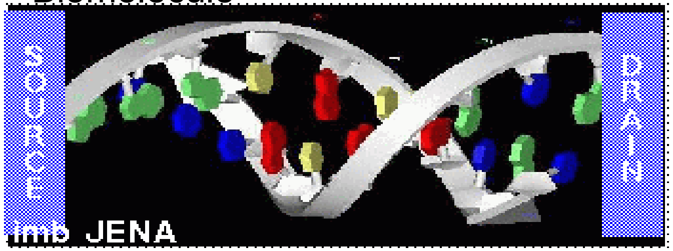
Nanowire



Organic Molecules / Carbon Nanotube



Biomolecule



- Most devices can be decomposed into a predominant transport direction
- Device structures are very different but underlying equations and computational algorithms are very similar

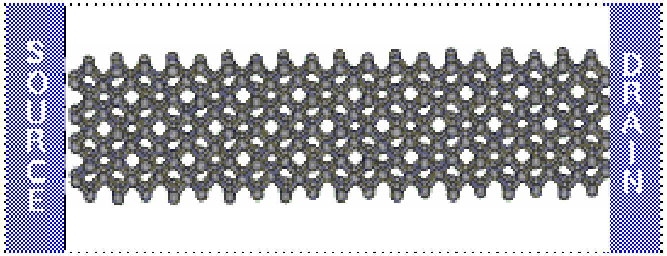
=> Algorithms can be generalized for many configurations and materials

=> choose basis wisely

=> compute Poisson smartly

=> design NEGF algorithms independent of geometry and basis

Underlying equations & interpretation



$$\begin{array}{rcl}\text{Density of States} & = & -\frac{1}{\pi} Im[G^r(x,x,E)] \\ \text{Electron Density} & = & -iG^<(x,x,E) \\ \text{Current Density} & \propto & G^<(x,x+1,E)\end{array}$$

$$(E-H)\Psi = 0 \rightarrow (EI - H - \Sigma^r)G^r = \delta \quad \longleftarrow \text{Kinetics}$$

$$(EI - H - \Sigma^r)G^< = \Sigma^< G^a$$

$$(EI - H - \Sigma^r)G^> = \Sigma^> G^a$$

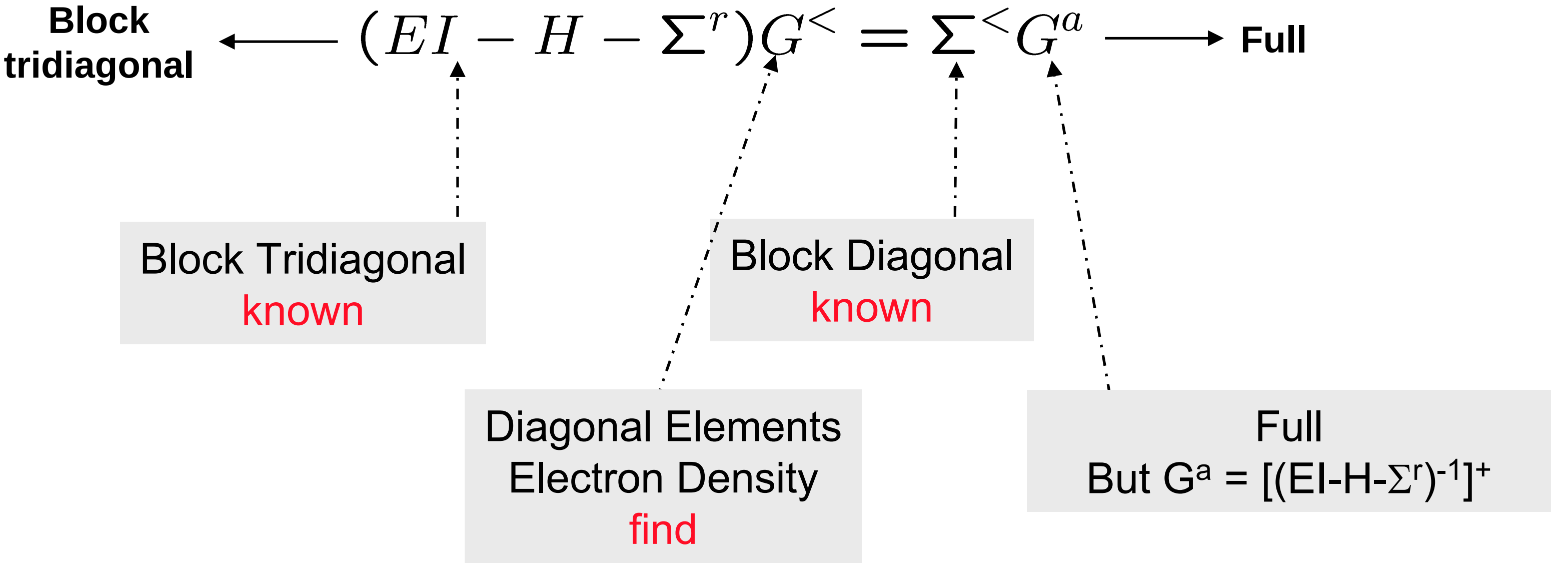
Statistics

Hamiltonian of Device
Atomistic: DFT, Tight-binding, …

Self-energies → Scattering rate
electron-phonon
contacts

scattering in & out of energy E
Statistical mechanics

Mathematical problem & matrix structure



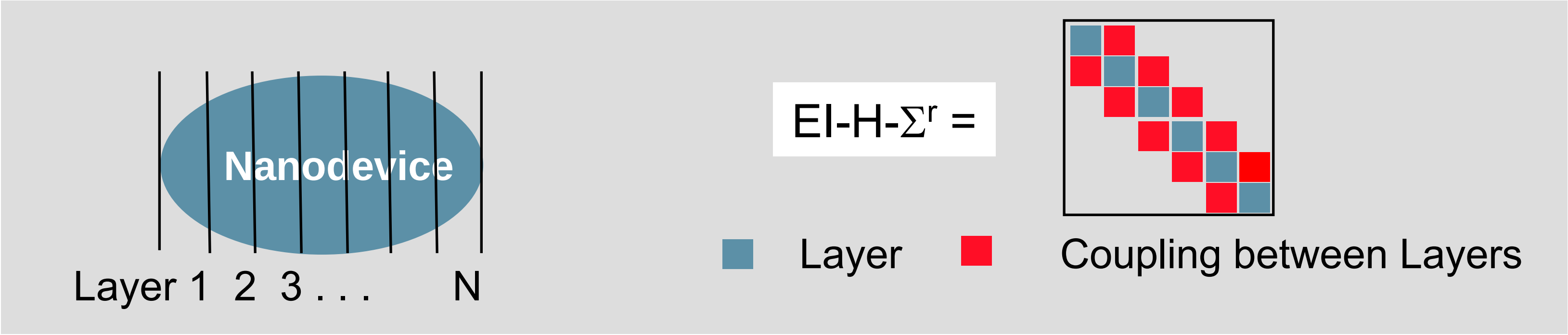
Block tridiagonal ← $(EI - H - \Sigma^r)G^< = \Sigma^< G^a$ → Full

Block Tridiagonal
known

Block Diagonal
known

Diagonal Elements
Electron Density
find

Full
But $G^a = [(EI-H-\Sigma^r)^{-1}]^+$

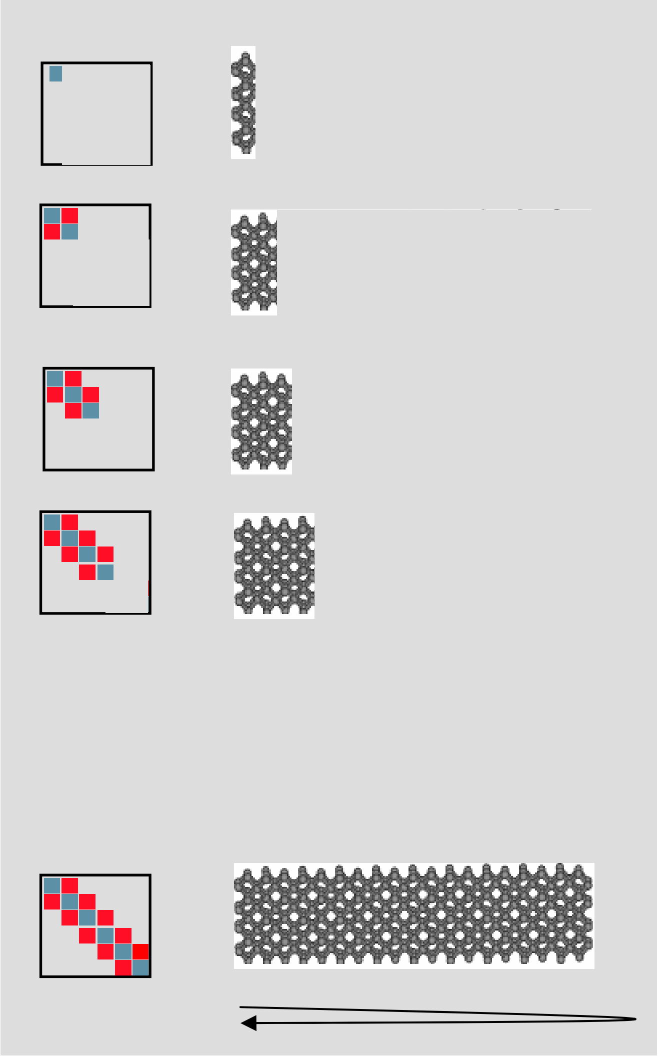


🟦 Layer 🟥 Coupling between Layers

- The **block tridiagonal system** is solved at each energy to obtain electron density (diagonal elements of $G^<$)

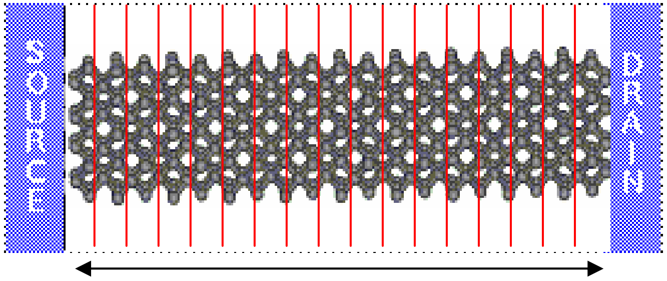
Current approach: Recursive algorithm

# of grid points = $N_x N_y$

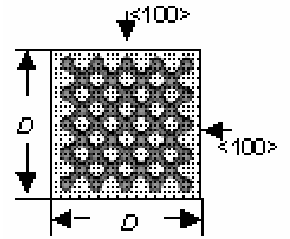


- What are we solving for?
  Electron Density – Diagonal elements of $G^<$
- Algorithmic flow based on Dyson's equation
  - » Recursive Green's function approach
  - » S[illegible]al algorithm at each energy
  - » Highly stable
- Block Size = N_x ; Number of blocks = N_y
- Each step involves a matrix inversion
- Operation Count ~ 6 $N_x^3 N_y$
- Memory ~ 4 $N_x^2 N_y$
- We solve for the diagonal blocks of $G^<$ but we need only the diagonal elements
- Quantum Mechanical Modeling of …,
  J. of Appl. Phys., v. 91, p. 2343-2354 (2002)

Grand Challenges: Nanowire and FinFETs



40 - 100 nm

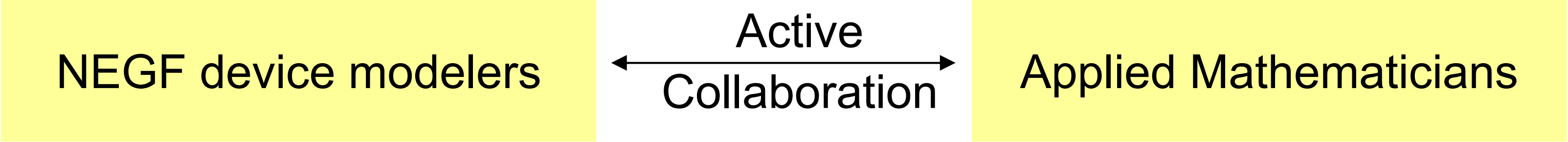


D = 1.5 - 3 nm Nanowire

D = 10 - 30 nm FinFET

- Matrix size of each layer (# of orbitals) ~ 2000+ $(N_x>2000)$ $N_x>200{,}000$
- Number of layers ~ 700 $(N_y>700)$ $(N_y>700)$
- Best Available method - RGF:
  - » Operation Count ~ $6\,N_x^3\,N_y$
  - » Memory ~ $4\,N_x^2\,N_y$
- **Memory ~ 100 GB Computational Time ~ 60,000+ hours**
- **Are there better ways to solve these systems?**

Algorithms development at NCN


NEGF device modelers

Active Collaboration

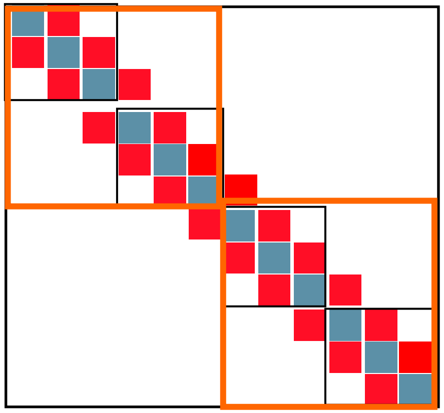
Applied Mathematicians

- Benchmark algorithm - RGF
  - » Operation count ~ $O(N_x^3 N_y)$
  - » Memory ~ $O(N_x^2 N_y)$
  - » Fundamentally serial

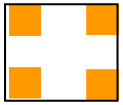
- New algorithms to solve the NEGF equations have been developed:
  - » PDIV - Parallel Divide and Conquer Approach (SRC funded), Stephen Cauley & Jitesh Jain, Balakrishnan-Koh group, Purdue
    - ✓ Operation count ~ $O(N_x^3 N_y /D + N_x^3 \log_2 D)$ reduced by D, with factor
    - ✓ Memory ~ $O(N_x^2 N_y /D + N_x^2 D)$ reduced by D, with factor
    - ✓ Parallel, scales almost linearly with CPUs
  - » FIND - Nested Dissection method (NCN funded), Song Li, Darve group, Stanford
    - ✓ Operation count ~ $O(N_x^2 N_y)$ reduced by N_x
    - ✓ Memory ~ $O(N_x N_y \log(N_x))$ reduced for large N_x,N_y

Divide and Conquer Approach

- Novel divide and conquer approach based on the matrix inversion lemma



D = 4



Corner Blocks

- Divide the original matrix into D sub problems (D sub blocks)
- Calculate corner blocks of inverse in each sub problem independently

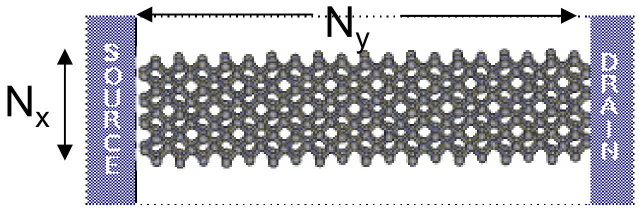
  (This step can be efficiently parallelized)
- Form and update matrix maps that capture concatenation of neighboring blocks in a radix 2 fashion
- The matrix map is used to calculate the diagonal elements of inverse

  (This step can be efficiently parallelized)

Operation count ~ $O(N_x^3 N_y / D + N_x^3 \log_2 D)$ Memory ~ $O(N_x^2 N_y / D + N_x^2 D)$

Better operation count – Better memory usage

Nested dissection method (FIND)



- $N = N_xN_y$ is the number of grid points

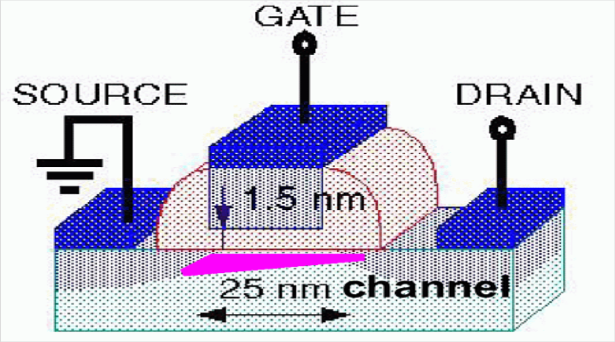
- LU decomposition of sparse matrices scales as O(N)

$$A\,G^r = I \rightarrow LU\,G^r = I \rightarrow G^r(N,N) = 1/U(N,N)$$

- Innovation: Partition the spatial grid in clever ways and keep information about intermediate LU steps to make operation count $O(N_x^2\,N_y)$

Operation count ~ $O(N_x^2\,N_y)$ Memory ~ $O(N_xN_y \log(N_x\,N_y))$*1kB

nanoFET - a NEGF based simulation tool

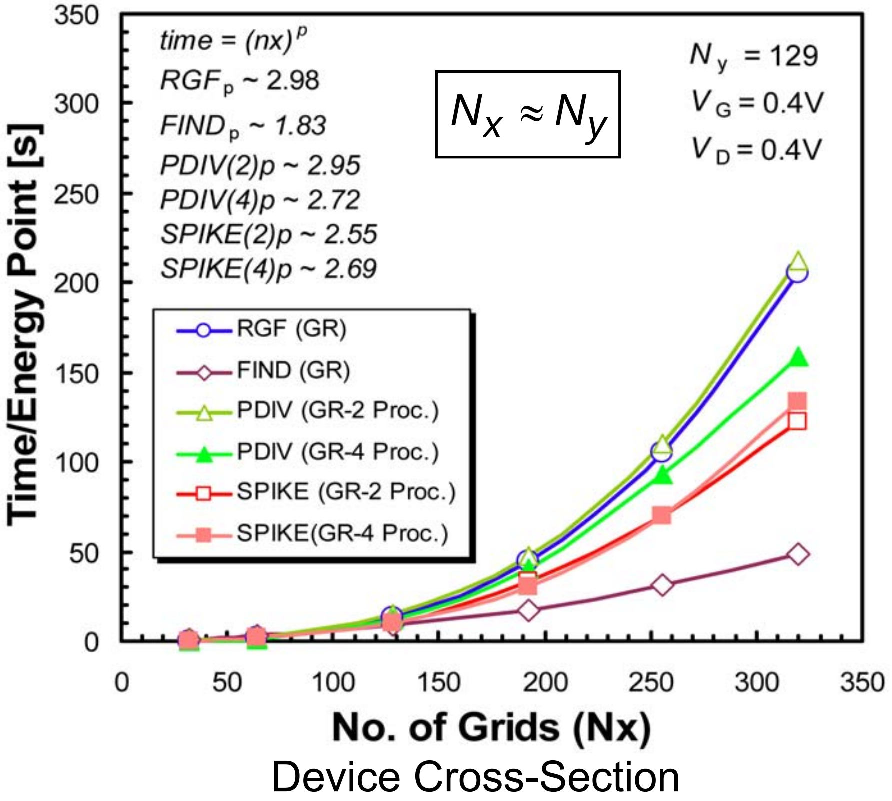


MIT 25 nm device

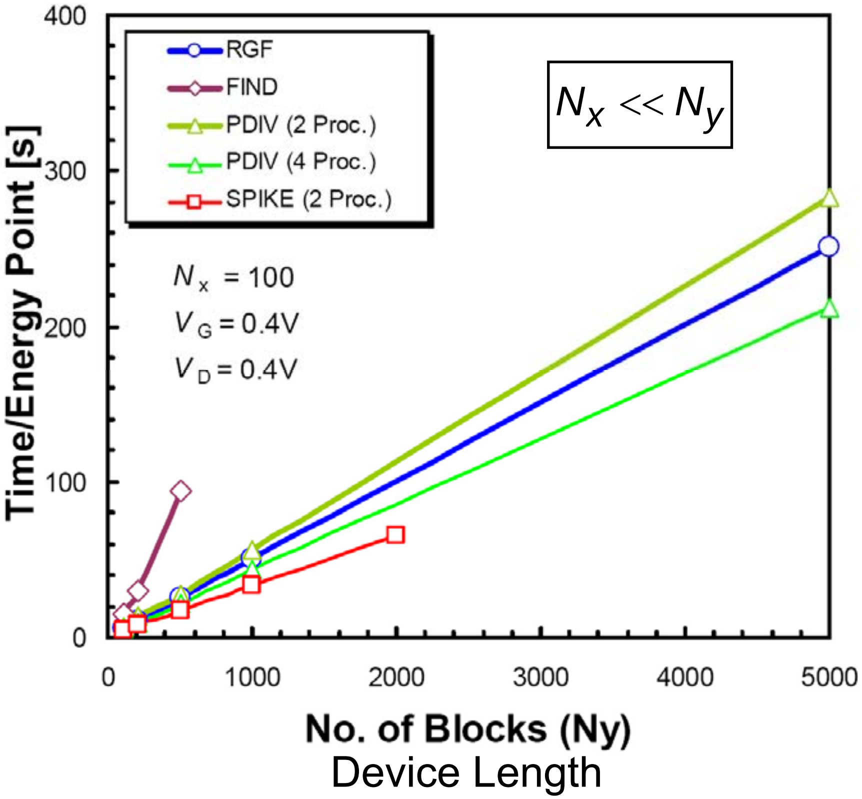
“standard” nano transistor designed by Prof. Dmitri Antoniadis

- Gate Tunneling, Source-drain tunneling
- Computational time for current-voltage characterisitcs in the ballistic transport limit ~ **6 hours**
  - » 64 processor SGI Origin 2000
  - » Nx = 100 and Ny = 350
  - » Energy grid points = 512 - 1024
- Accurate models should include scattering mechanisms - phonons, rough interfaces **Modeling of scattering using current algorithms is prohibitively expensive.**

Comparison inside a real Simulator



Device Cross-Section



Device Length

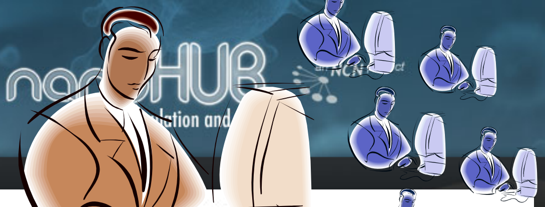
Evident tradeoffs between simulation time, memory, and computation:

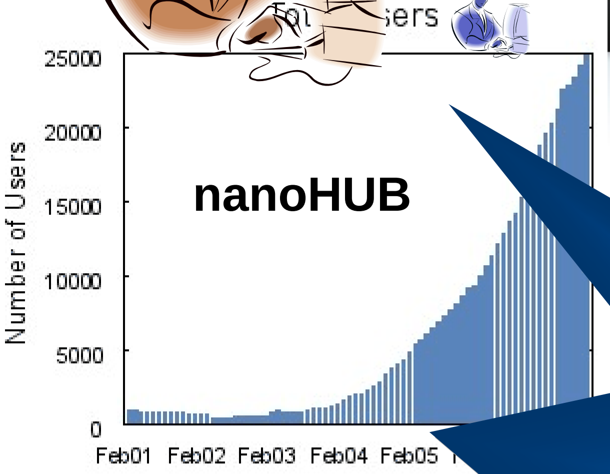
- FIND: fastest on a single processor for Ny ≈ 130, Nx ≤ 325.
- PDIV: parallelizes RGF style method with little overhead,
  memory and computation scales gracefully for Nx ≈ 100, Ny >> Nx.
- SPIKE: fastest for two processors, medium sized problems (Nx = 100 , Ny < 2000).

Algorithms and Parallelization Approaches for Scaling to 23,000 Processors in Nanoelectronic Modeling (NEMO)

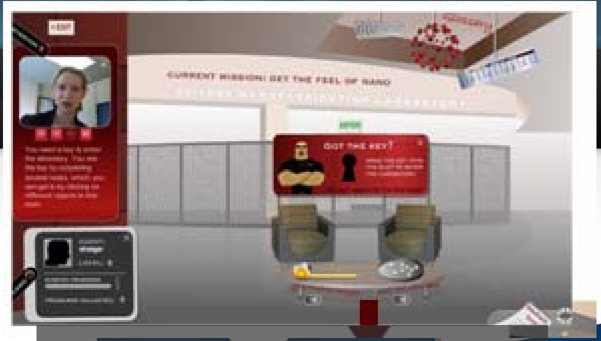
- Network for Computational Nanotechnology (NCN)
- NEMO 3-D
  - »Physics motivation – multimillion atom simulation
  - »Basic algorithms
  - »Scaling to 8,192 processors
- NEMO 1-D
  - »Physics motivation –transport of electrons quantum statistical non-equilibrium mechanics
  - »Basic algorithms
  - »Scaling to 23,000 processors
- OMEN plans
- Algorithm work needed

Network for Computational Nanotechnology



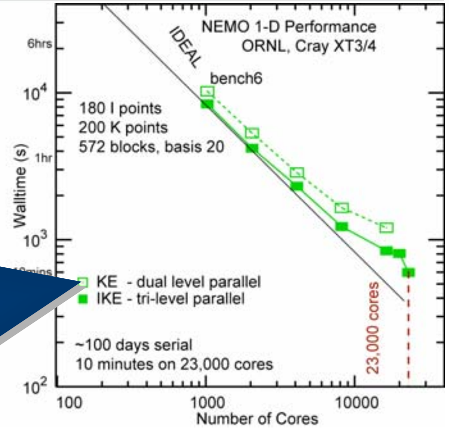


> 26,000 total users
> 5,900 simulation users
> 225,000 simulations



K-12 Education Game

HPC 23,000 cores



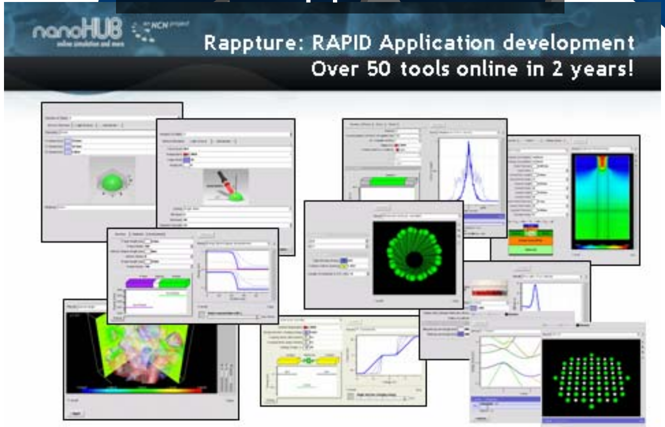
NCN

PURDUE UNIVERSITY

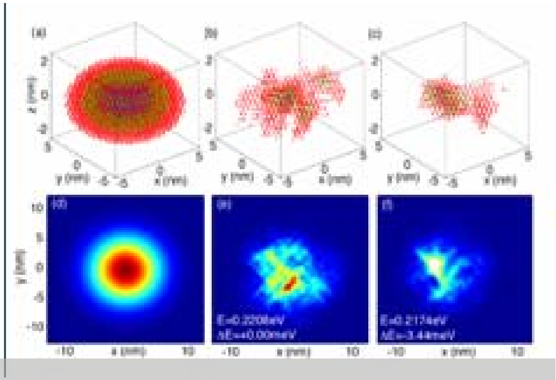
HUB0

www.hubzero.org

Rappture



Rappture: RAPID Application development
Over 50 tools online in 2 years!



Million Atom Simulations

PURDUE UNIVERSITY ITaP NCN

SRC FCRP ARDA NSF