The Mathematics Behind MADNESS

G. Fann, R. Harrison, J. Hill, J. Jia, and D. Galindo

Oak Ridge National Laboratory

G. Beylkin (Colorado)

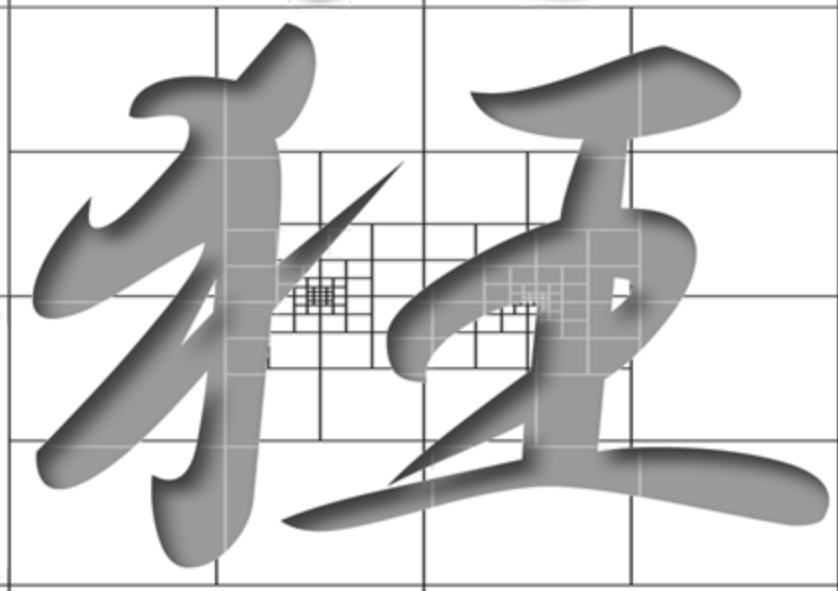
M. Mohlenkamp(Ohio U.),

F. Perez (Berkeley)

Y. Ou (JICS/Tennessee)

SciDAC 2010, Chattanooga, TN

MADNESS



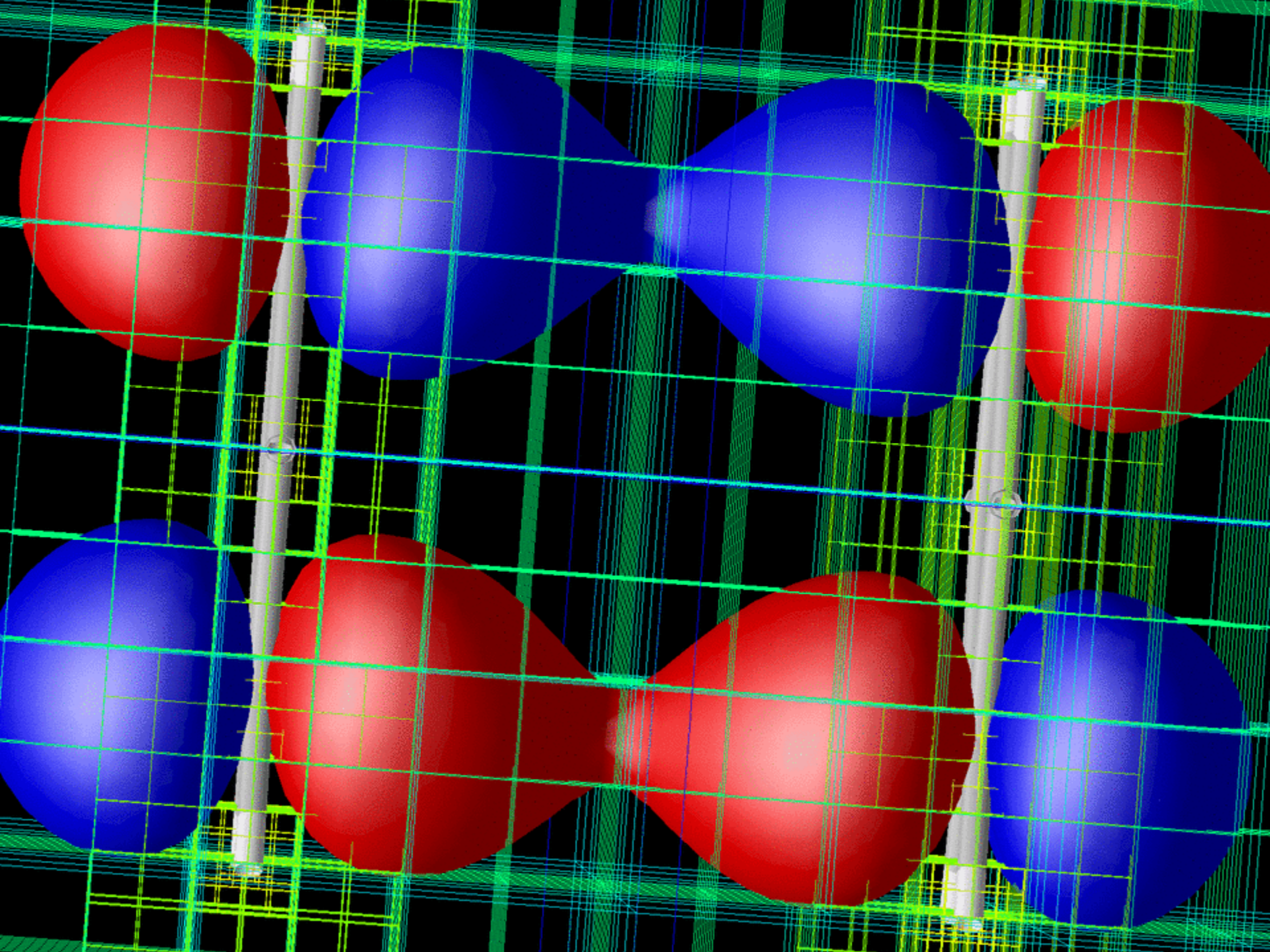
Multiresolution
Adaptive
Numerical
Scientific
Simulation

Office of Science
U.S. DEPARTMENT OF ENERGY

ASCR

SciDAC
Scientific Discovery through Advanced Computing

OAK RIDGE
National Laboratory



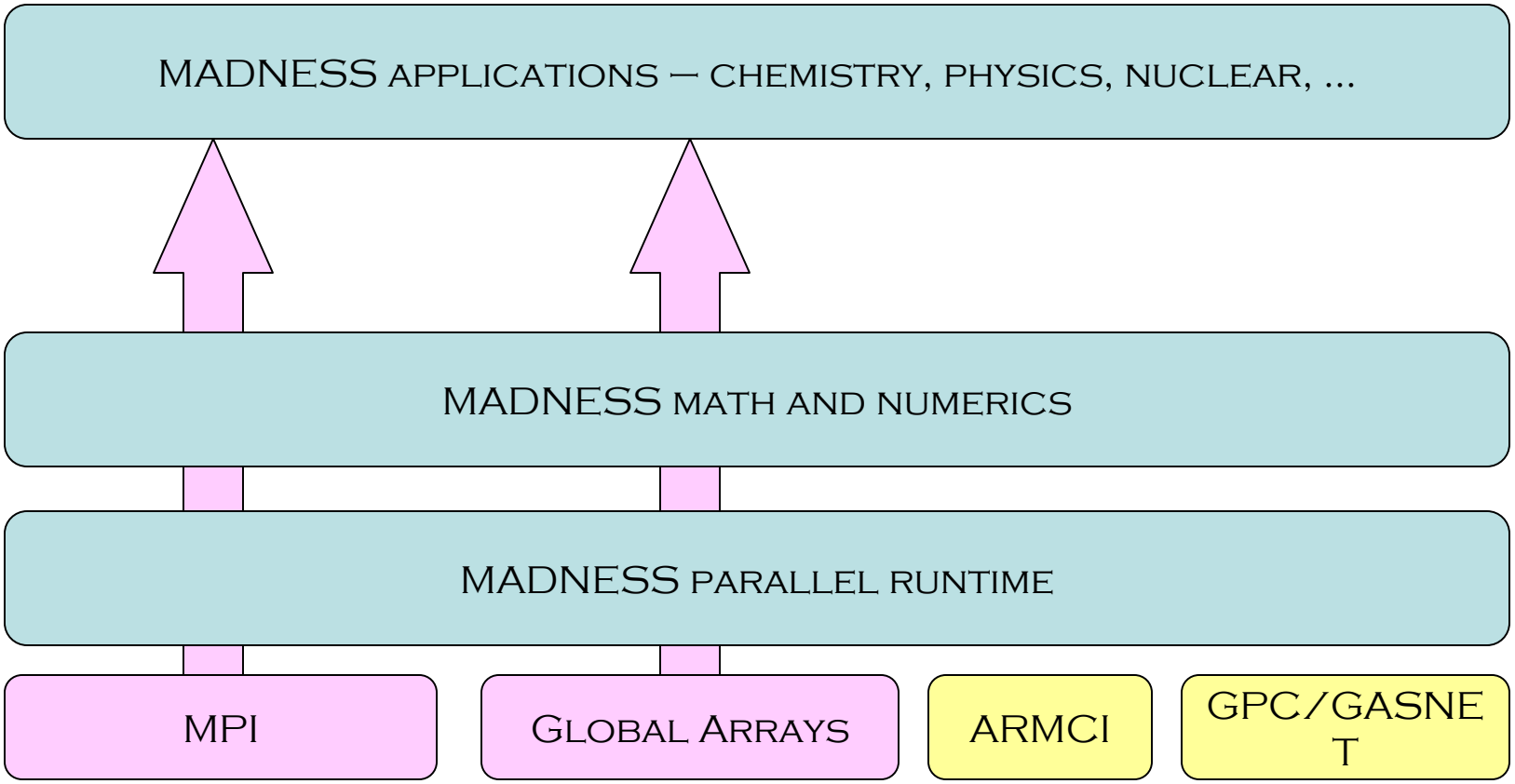
An Introduction to MADNESS

- What is it?
- Example of use
  - Computational Chemistry
  - Nuclear Density Functional Theory
  - Navier-Stokes Equation
- Multiresolution analysis and representation of functions
- Non-standard form,
- Separated representation of operators and functions
- Examples using operators

What is MADNESS?

- Multiresolution Adaptive Numerical Scientific Simulation
  - It is an environment to prototype and develop scientific applications using multiresolution methods in multiwavelet basis from 1-D to 6-D.
  - The important concepts
    - *Multiresolution, multiwavelet, low-separation rank approximation of functions and operators*
    - *Multiresolution discontinuous and singular bases*
    - *Each function and operator has its own adaptive representation*
  - It presently contains non-linear time-independent and time-dependent solvers for 3-D Density Functional Theory (DFT) code and Schrodinger equations for chemistry and nuclear physics, but others are working on nano-charge dynamics, bio-fluid, chemical reactions, …

MADNESS architecture



Intel Thread Building Blocks being considered for multicore

Example: Integral Formulation for Soln of Bound State Schrödinger Equation

- Can solve by using differential formulation and diagonalize Hamiltonian
  - Issue: high condition number resulting from adaptive representation of unbounded operators, such as Laplacian, limits accuracy and precision
  - Use as preconditioner
- We form and solving the integral equation
  - Eliminates the derivative operator and related "issues"
  - Converges as fixed point iteration *with no preconditioner*
- In dimensionless scattering formulation: with energy E and potential V

$$\left(-\nabla^2 + V\right)\Psi = E\,\Psi$$

$$\Psi = -\left(-\nabla^2 - E\right)^{-1} V\,\Psi$$

$$= -G * \left(V\,\Psi\right)$$

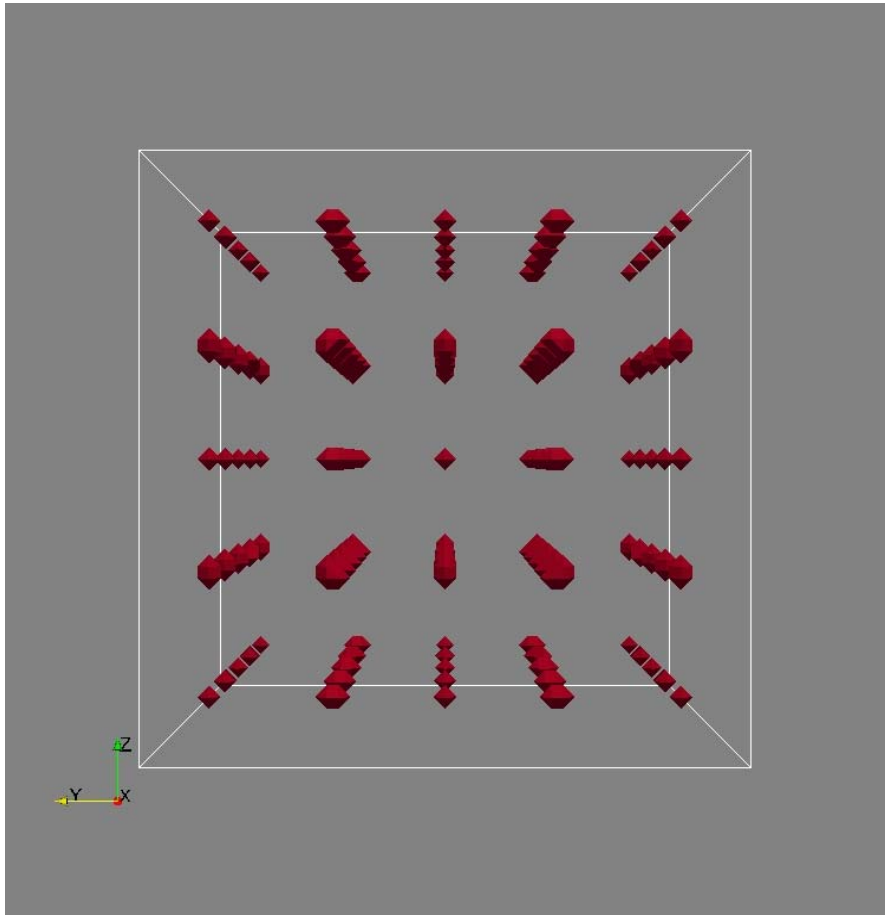
$$\left(G * f\right)(r) = \int ds\, \frac{e^{-k|r-s|}}{4\pi\,|r-s|}\, f(s) \quad \text{in 3D} \;;\; k^2 = -E$$

MADNESS: High-level composition

- Coding composition is close to the physics, example with h=m=1 (chemist notation)
- $$E = \left\langle \psi \left| -\frac{1}{2}\nabla^2 + V \right. \right\rangle \psi + \int \psi^2(x) \frac{1}{|x-y|} \psi^2(y)\,dx\,dy$$

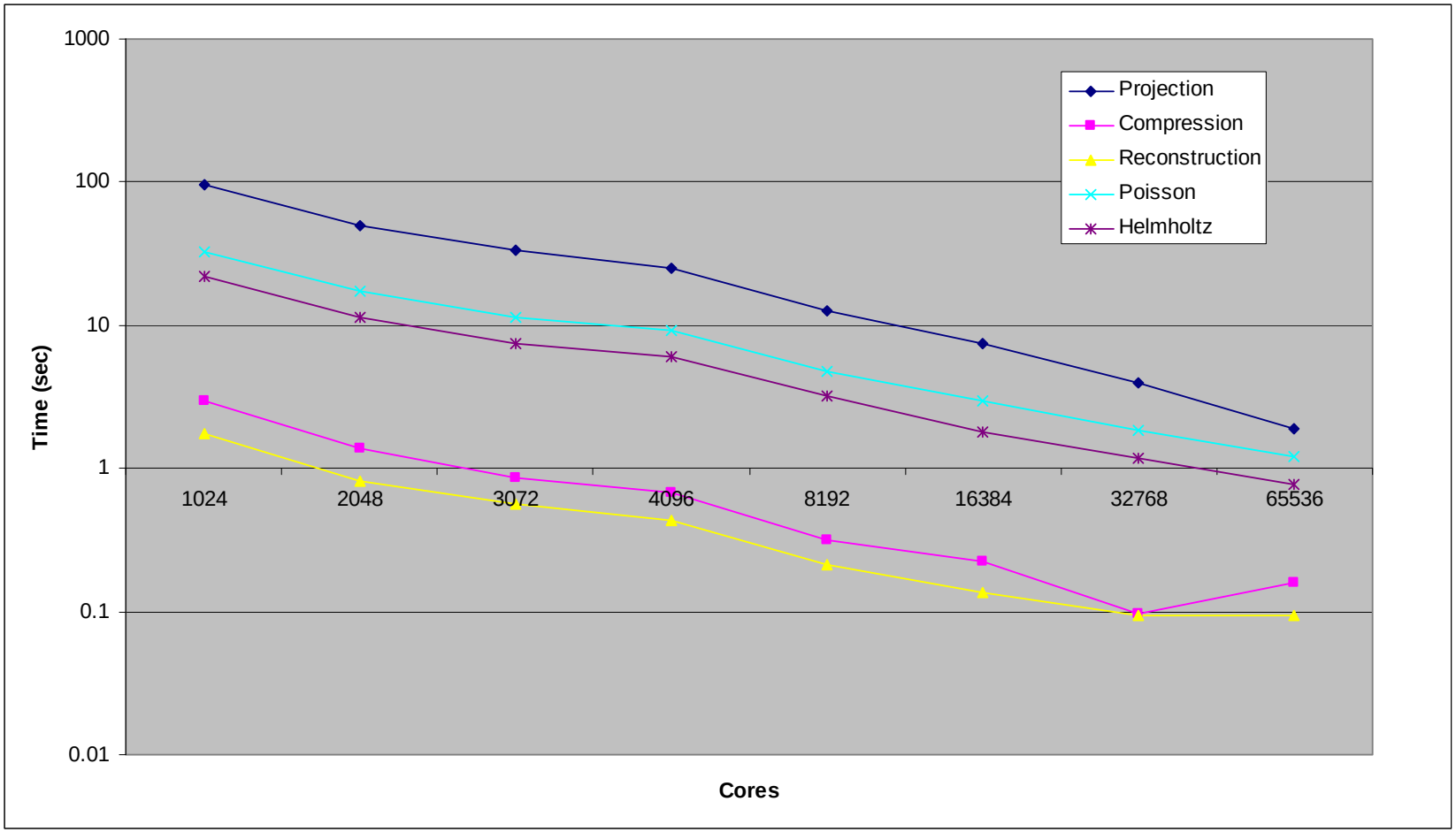
```
operatorT op = CoulombOperator(k, rlo, thresh);
functionT rho = psi*psi;
double twoe = inner(apply(op,rho),rho);
double pe = 2.0*inner(Vnuc*psi,psi);
double ke = 0.0;
for (int axis=0; axis<3; axis++) {
    functionT dpsi = diff(psi,axis);
    ke += inner(dpsi,dpsi);
}
double energy = ke + pe + twoe;
```

Scaling of Poisson and Helmholtz in MADNESS with Lattice of Atom Centered Potential



- Charge density for Poisson Equation
- Scattering for Helmholtz Eqn, bound-state or Yukawa
- Lattice of 1000 atoms
- Potential centered at each atom center
- 10 levels of refinement
- 1.91B equations and variables

Scaling of MADNESS for on Cray XT-5
Poisson and Helmholtz
1.91B eqns, 10 levels of refinement, rel. accuracy 1.e-10, 12/1/2009



Chemistry and Nuclear Physics Application:No Spin-Orbit Integral Formulation of Bound State Schrödinger Equation

- Given H, in multiwavelet basis and guess wavefunctions, diagonalize
- Solving the integral equation
  - Eliminates the derivative operator and related “issues”
  - Converges as fixed point iteration *with no preconditioner*
- In dimensionless form:

$$\left(-\nabla^2 + V\right)\Psi = E\Psi$$

$$\Psi = -\left(-\nabla^2 - E\right)^{-1} V\Psi$$

$$= -G * \left(V\Psi\right)$$

$$\left(G * f\right)(r) = \int ds \frac{e^{-k|r-s|}}{4\pi|r-s|} f(s) \text{ in 3D ; } k^2 = -E$$

E.g., Energy of Hydrogen Atom

```
from mra import *
L = 32.0
Function.k = 7
Function.thresh = 1e-5

def psi(x,y,z):
    fac = 102.12922378276677 #sqrt(L**3/pi)
    x,y,z = (x-0.5)*L,(y-0.5)*L,(z-0.5)*L
    return fac*exp(-sqrt(x*x+y*y+z*z))

def V(x,y,z):
    x, y, z = (x-0.5)*L, (y-0.5)*L, (z-0.5)*L
    return -1.0/sqrt(x*x+y*y+z*z)

psi = Function(function=psi)
V   = Function(function=V)
px, py, pz = Del*psi
S  = psi.inner(psi)
VE = (V*psi).inner(psi)/S
TE = 0.5*(px.inner(px)+py.inner(py)+pz.inner(pz))/L**2/S
print "norm",S,"kinetic",TE,"potential",VE,"energy",TE+VE
```

$$\psi(r) = \left(\frac{L^3}{\pi}\right)^{\frac{1}{2}} e^{-r} \qquad S = \langle \psi | \psi \rangle = 1$$

$$T = \langle \psi | -\tfrac{1}{2}\nabla^2 | \psi \rangle = \tfrac{1}{2}\langle \nabla\psi | \nabla\psi \rangle$$

$$V = \langle \psi | -r^{-1} | \psi \rangle \qquad E = T + V$$

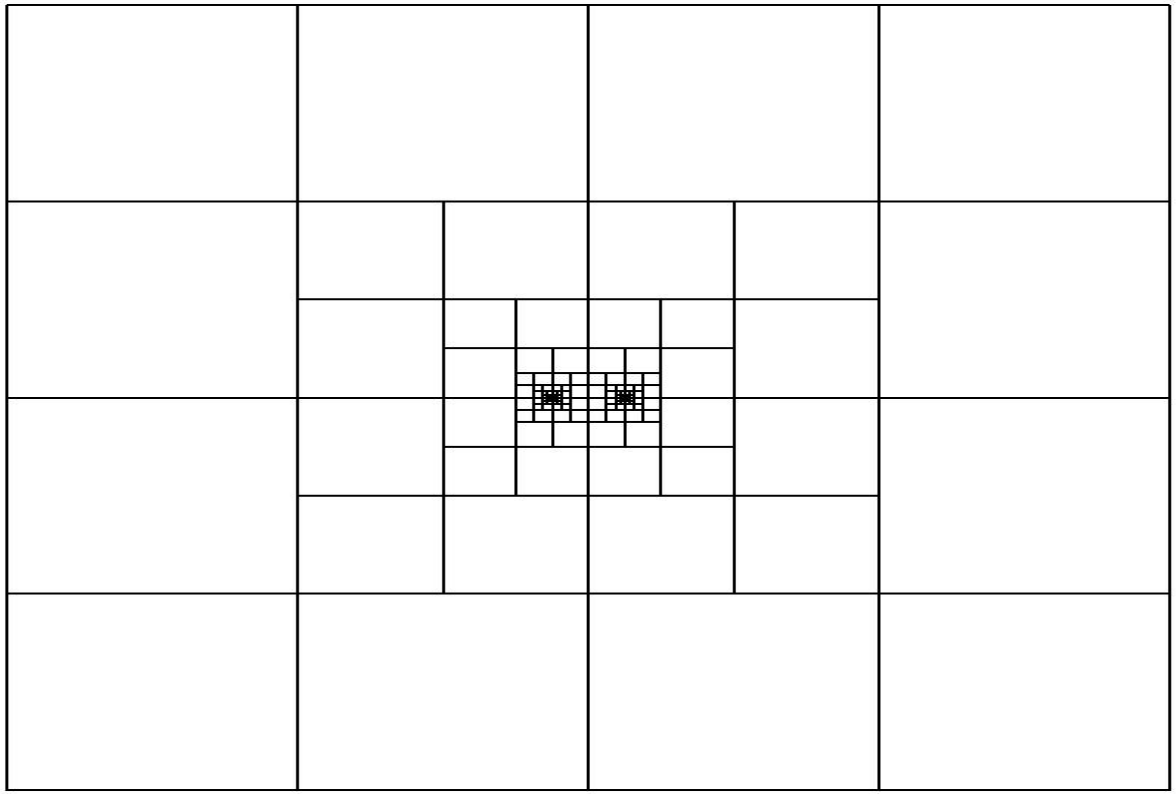
norm 1.00000000014

kinetic 0.500000335672

potential -1.00000013916

energy -0.499999803488 (exact: -0.5)

2-D Slice of the 3-D Potential for H_2



H_2, Schrodinger's Equation

- Chemists need high precision;
  - they need to compute differences of energies.
- Hydrogen Mol, bond length r=1.4 bohrs
- Best variational energy known to us: -1.133629571456 (Mitin)
- k=5, -1.1335567888
- k=7, -1.1336294353
- k=9, -1.1336295698
- k=11, -1.1336295713
- k=13, -1.1336295714
- Truncation 1.e(2-k), with smoothing

Two-Cosh Test Problem

- Potential

$$V(x,y,z) = \frac{V_1}{1+e^{-R_1/a_1}\cosh(r_+/a_1)} + \frac{V_2}{1+\cosh(r_-/a_2)}$$

$$r_+ = \sqrt{r^2+(z+\varsigma)^2}, r_- = \sqrt{r^2+(z-\varsigma)^2}$$

$$V_{SO} = -\sqrt{-1}\left(\frac{h}{2mc}\right)^2 \left[\nabla V \bullet (\sigma \times \nabla)\right]$$

$$\varsigma = 7.5, h^2/2m = 20.721246..,$$

$$hc = 197.32696..., mc^2 = 939.56535...$$

Chemistry and Nuclear Physics Application
Integral Formulation of Bound State Schrödinger Equation

$$\left(-\nabla^2 + V + V_{SO}\right)\Psi = E\Psi,$$

$$\Psi = (u_r, u_c, v_r, v_c), 2-spinor-wavefunction$$

$$\Psi = -G*\left(V\Psi + V_{SO}.\Psi\right)$$

$$\left(G*f\right)(r) = \int ds \frac{e^{-k|r-s|}}{4\pi|r-s|} f(s) \text{ in 3D}; k^2 = -E$$

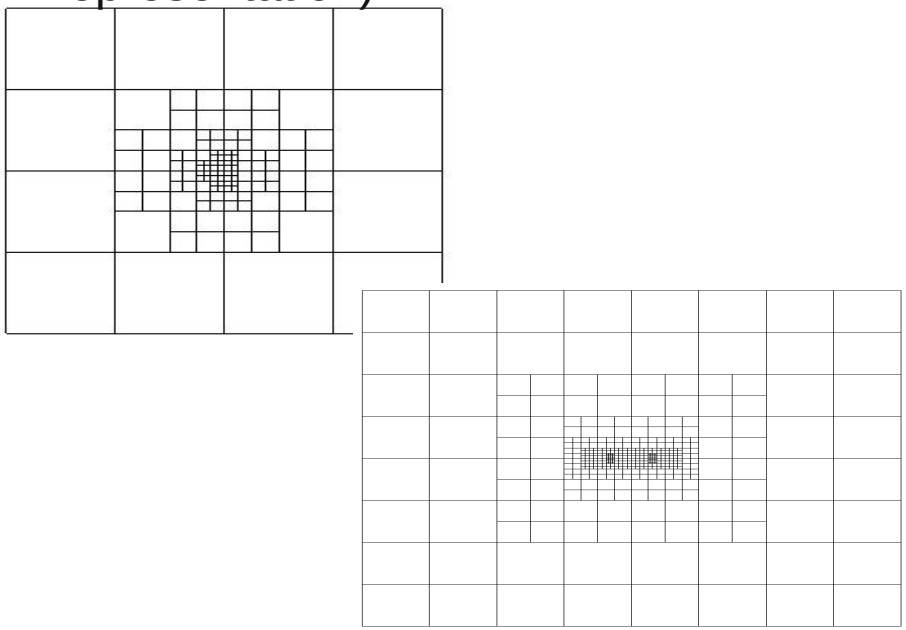
$$V(x,y,z) = \frac{V_1}{1+e^{-R_1/a_1}\cosh(r_+/a_1)} + \frac{V_2}{1+e^{-R_2/a_2}\cosh(r_-/a_2)},$$

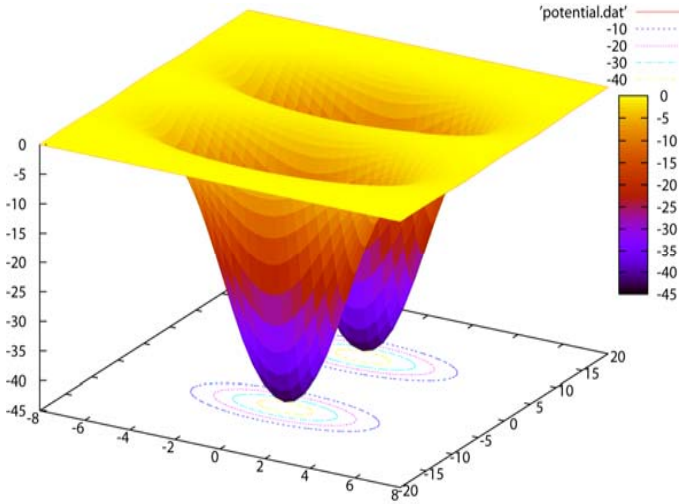
$$r_\pm = \sqrt{r^2 + (z \pm \varsigma)^2},$$

$$V_{SO} = -\sqrt{-1}\lambda_0 \left(\frac{h}{2mc}\right)^2 \nabla V \bullet (\sigma \times \nabla)$$

Representation of Wave-Functions

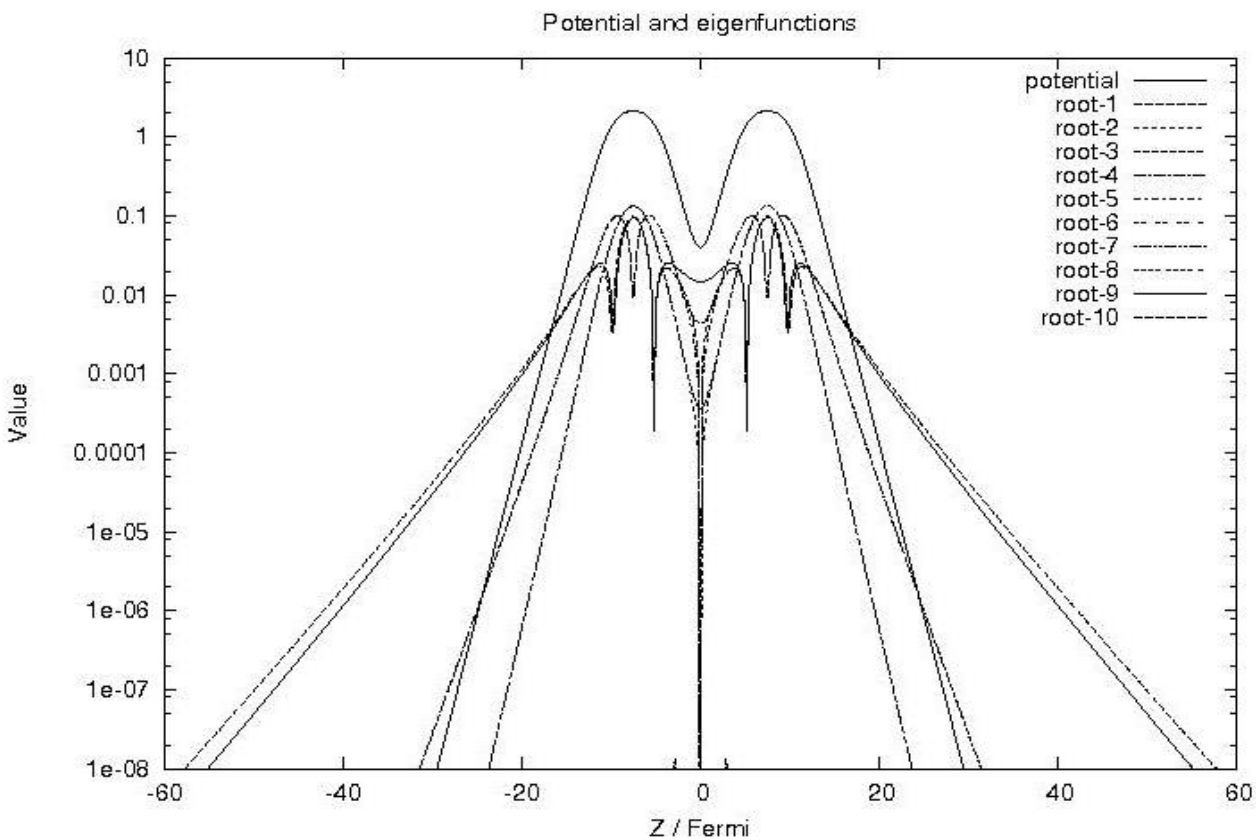
MADNESS (each function and operator has its own adaptive pseudo-spectral expansion representation)





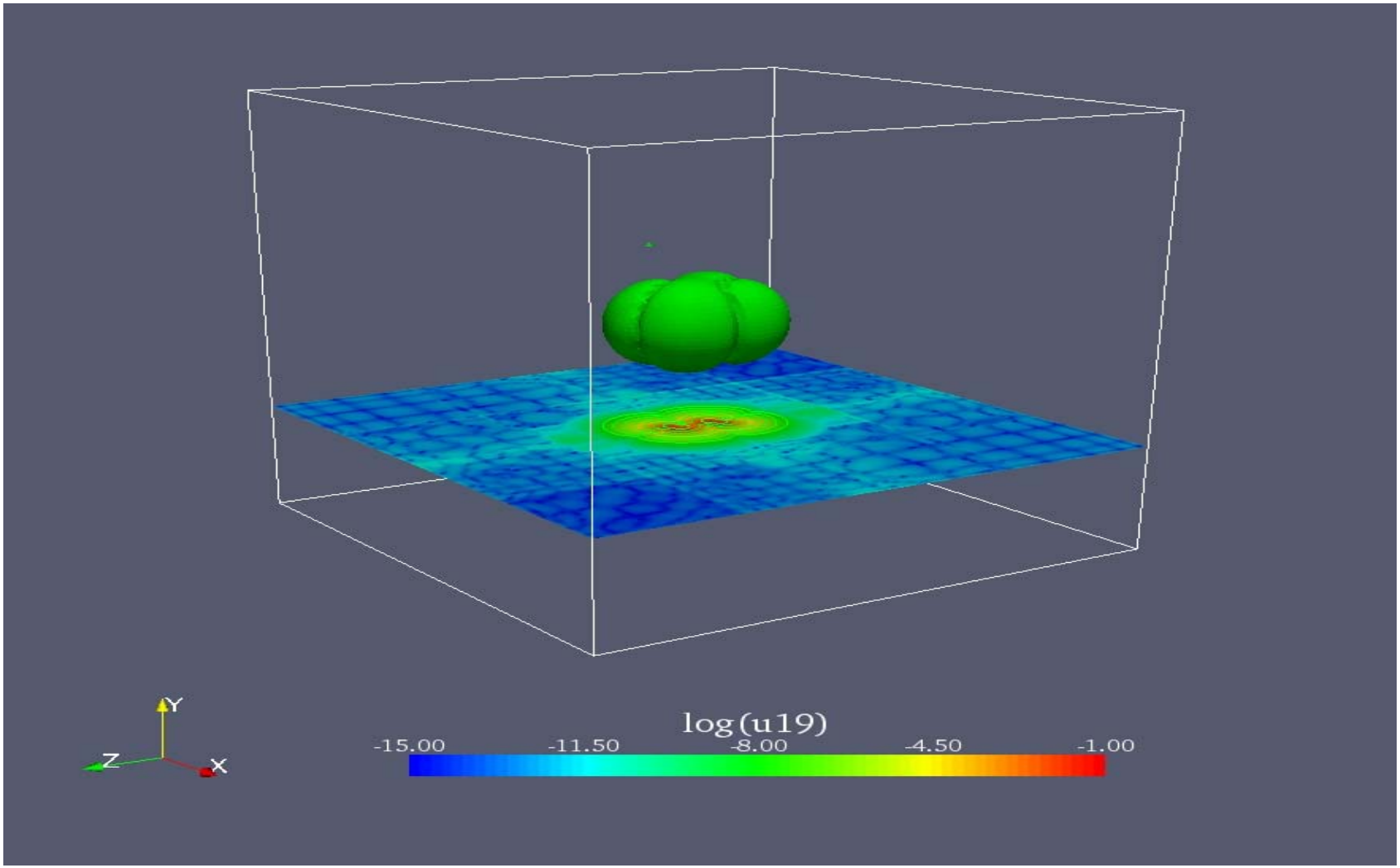
A 2-D slice of the 3-D support of the multiwavelet bases for the 2-cosh potential (left) and one of its wave-functions (right).

Plot of Potential and Absolute Value of Wave Functions for the 2-cosh Potential, domain > 120 fermi

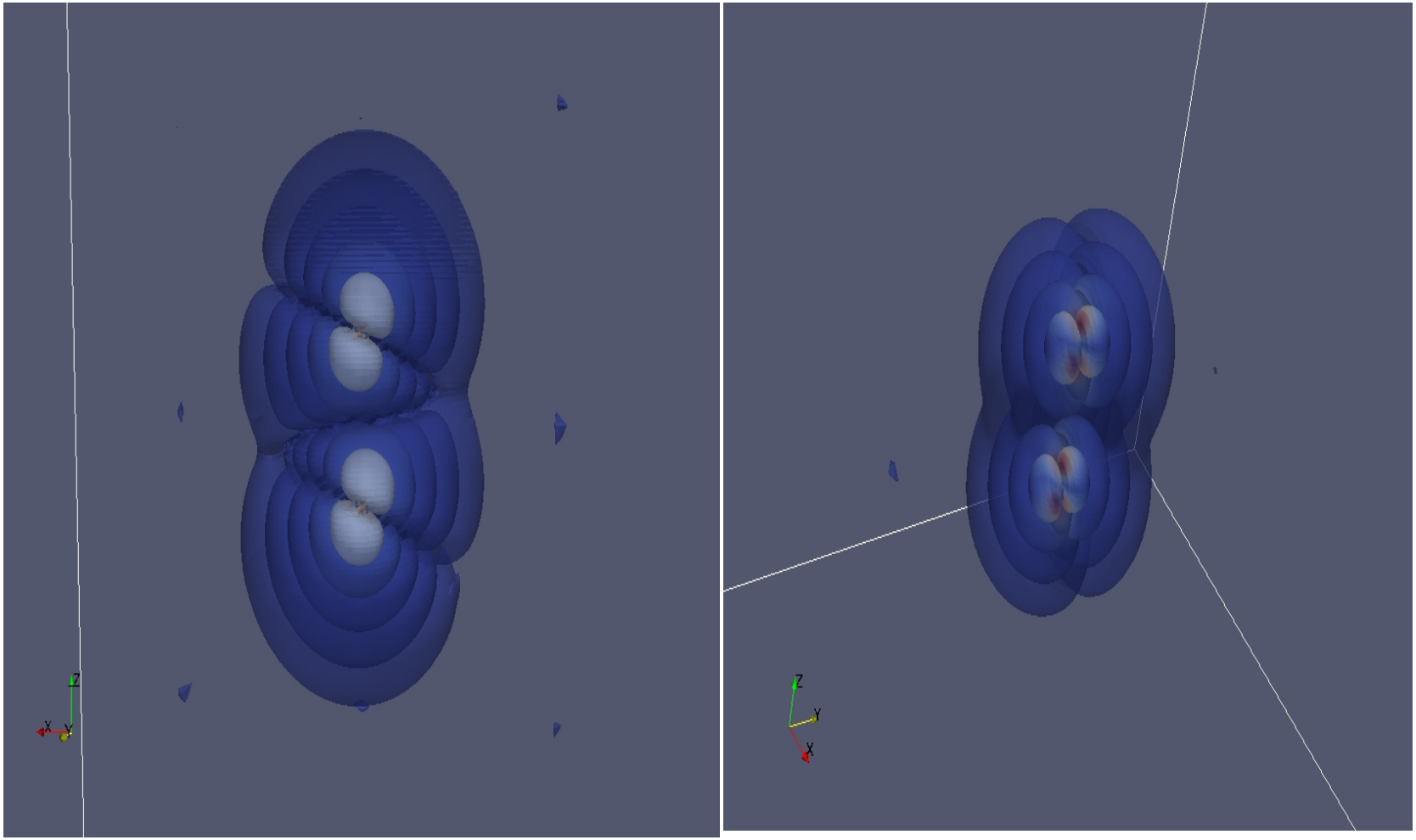


Fann, Harrison, Pei, Ou, Nazarewicz 2009

Log(abs(u_19)) for HFB 2-cosh



Nuclear structure for an anisotropic two well potential (3-D)



Recent Progress of MADNESS

- Solving A.Bulgac's SLDA and ASLDA equations for imbalanced Fermi condensates. [A. Bugac. PRA 76:040502, PRL 101:215301]

$$\begin{bmatrix} h_a(\boldsymbol{r}) - \lambda_a & \Delta(\boldsymbol{r}) \\ \Delta^*(\boldsymbol{r}) & -h_b(\boldsymbol{r}) + \lambda_b \end{bmatrix} \begin{bmatrix} u_i(\boldsymbol{r}) \\ v_i(\boldsymbol{r}) \end{bmatrix} = E_i \begin{bmatrix} u_i(\boldsymbol{r}) \\ v_i(\boldsymbol{r}) \end{bmatrix}$$

$$h_a = -\frac{\hbar^2}{2m}\nabla \cdot (\nabla \alpha_a) + U_a + V_{\text{ext}} \quad h_b = -\frac{\hbar^2}{2m}\nabla \cdot (\nabla \alpha_b) + U_b + V_{\text{ext}}$$

- Pairing regularization to avoid cutoff divergent (Bulgac)

$$\mathcal{E} = \alpha_a(x)\frac{\tau_a}{2} + \alpha_b(x)\frac{\tau_b}{2} + \frac{(3\pi^2(\rho_a + \rho_b))^{5/3}}{10\pi^2}\beta(x) - \Delta\kappa$$

$$\frac{1}{g_{eff}(\mathbf{r})} = \frac{n^{1/3}(\mathbf{r})}{\gamma} + \Lambda_c(\mathbf{r}),$$

$$\Lambda_c(\mathbf{r}) = -\frac{k_c(\mathbf{r})}{2\pi^2\alpha}\left\{1 - \frac{k_0(\mathbf{r})}{2k_c(\mathbf{r})}\ln\frac{k_c(\mathbf{r}) + k_0(\mathbf{r})}{k_c(\mathbf{r}) - k_0(\mathbf{r})}\right\}$$

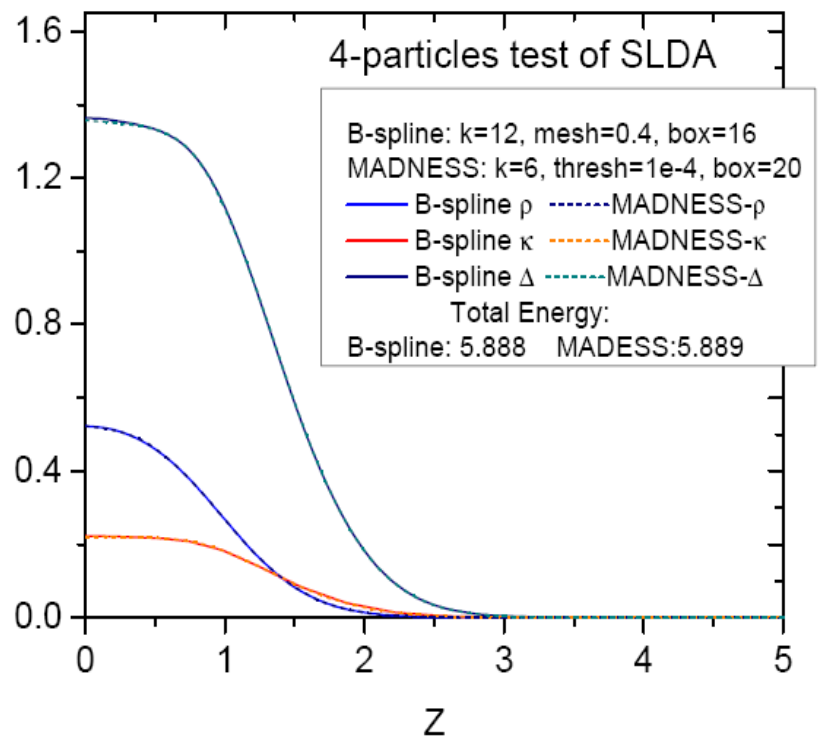
Pei, Fann, Harrison, Ou, Nazarewicz 2010

Recent Progress of MADNESS-HFB

Exterior potential, a deformed

$$V_{ext}(\mathbf{r}) = V_{shift}(1 - \exp(-\frac{(\omega_r^2(x^2+y^2)+\omega_z^2 z^2)}{2 \times V_{shift}}))$$

- Benchmarking is done: densities, eigenvalues, occupation numbers (MADNESS and B-spline methods)
- SLDA-MADNESS has been benchmarked (Feb. 2010)
- ASLDA-MADNESS has been benchmarked (Mar.2010)
- Can treat 100 particles in a deformed trap



Pei, Fann, Harrison, Ou, Nazarewicz 2010

Multiresolution Methods for DFT

- Adaptive 3-D multiresolution pseudo-spectral methods for nuclei using discontinuous basis
  - User defined accuracy gives universal reference and good scalability
  - Consistent description of bound and resonant states
- No assumption on symmetry
- Each wave-fn has its own adaptive structure
- Can handle nearly singular or discontinuous functions, and high gradients
- Work is proportional to spectral accuracy
- Controllable and guaranteed precision

New algorithm for solving non-linear self-consistent HFB I

1. Given Hamiltonian H, and guess wavefunctions $\{\phi_i\}$ form matrix $(H_{i,j}) = <\phi_i, H\phi_j>$
   - With regularization, guess for densities, …
2. Diagonalize to obtain updated and orthogonal wave-functions
3. Update potentials, anomaly density, boundary conditions, correlations…using new eigenvalues and eigenfunctions
4. Form and solve the Lippman-Schwinger integral equation by scattering methods (approximation expansion in u,a, and G's)
   1. Construction of scattering kernel for each eigenvalue for u's via non-linear optimization
   2. Solve for u's
   3. Update potential, densities, BC, correlations, fitting which are functions of u's
   4. Construction of scattering kernel for each eigenvalue for a, via non-linear optimization
   5. Solve for a's
5. Check error estimates

New algorithm for solving non-linear self-consistent HFB II

- Error estimates
  - If converged within tolerance, exit
  - If not converged go to step 1.
  - If further refinement required and not converged within a certain number of iterations
    - *Set new number of multiwavelets, precision, truncation errors, iteration count, …*
    - *Project all variables to new level of subspace*
    - *Goto step 1 on last slide*

Solving Incompressible Navier-Stokes Equation

- Consider the classical incompressible Navier-Stokes equations, in a periodic domain

$$\mathbf{u}_t + \nabla p = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \mathbf{Re}^{-1}\nabla^2\mathbf{u} + f \qquad \nabla\cdot\mathbf{u} = 0$$

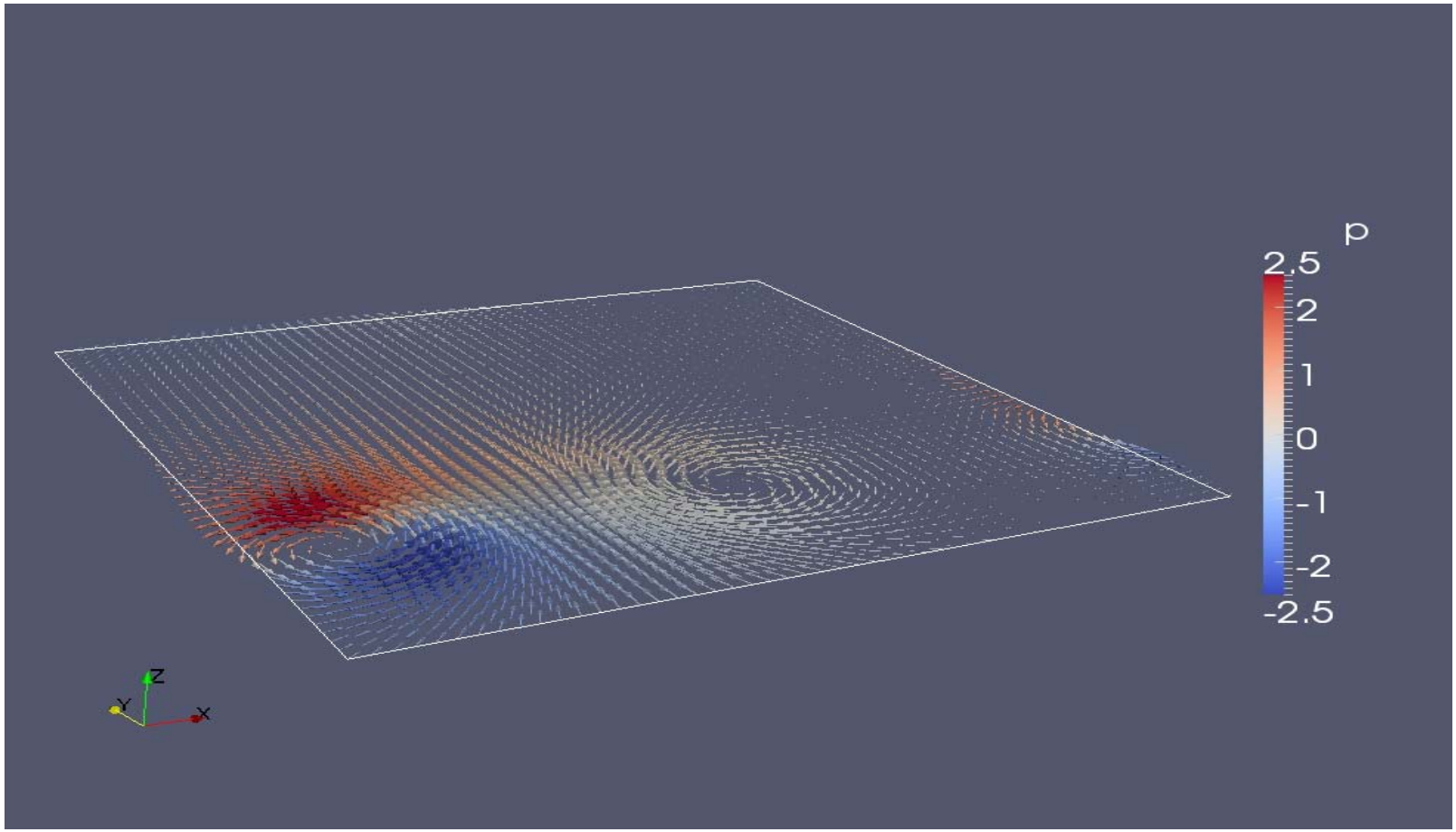
- We demonstrate a solution method using multiresolution based Poisson and Helmholtz operators
  1. Backward Euler in time, implicit (iterative solvers)
  2. Fast integral methods that decompose the system into two equations by a projection method

$$\Delta p = \nabla\cdot f - \nabla\cdot(\mathbf{u}\cdot\nabla)\mathbf{u}$$ (Poisson Problem)

$$\left(\frac{\mathbf{Re}}{\delta t} - \Delta\right)\mathbf{u}^{t+1} = \mathbf{Re}\left(f - (\mathbf{u}^t\cdot\nabla)\mathbf{u}^t - \nabla p + \frac{1}{\delta t}\mathbf{u}^t\right)$$ (Helmholtz Problem)

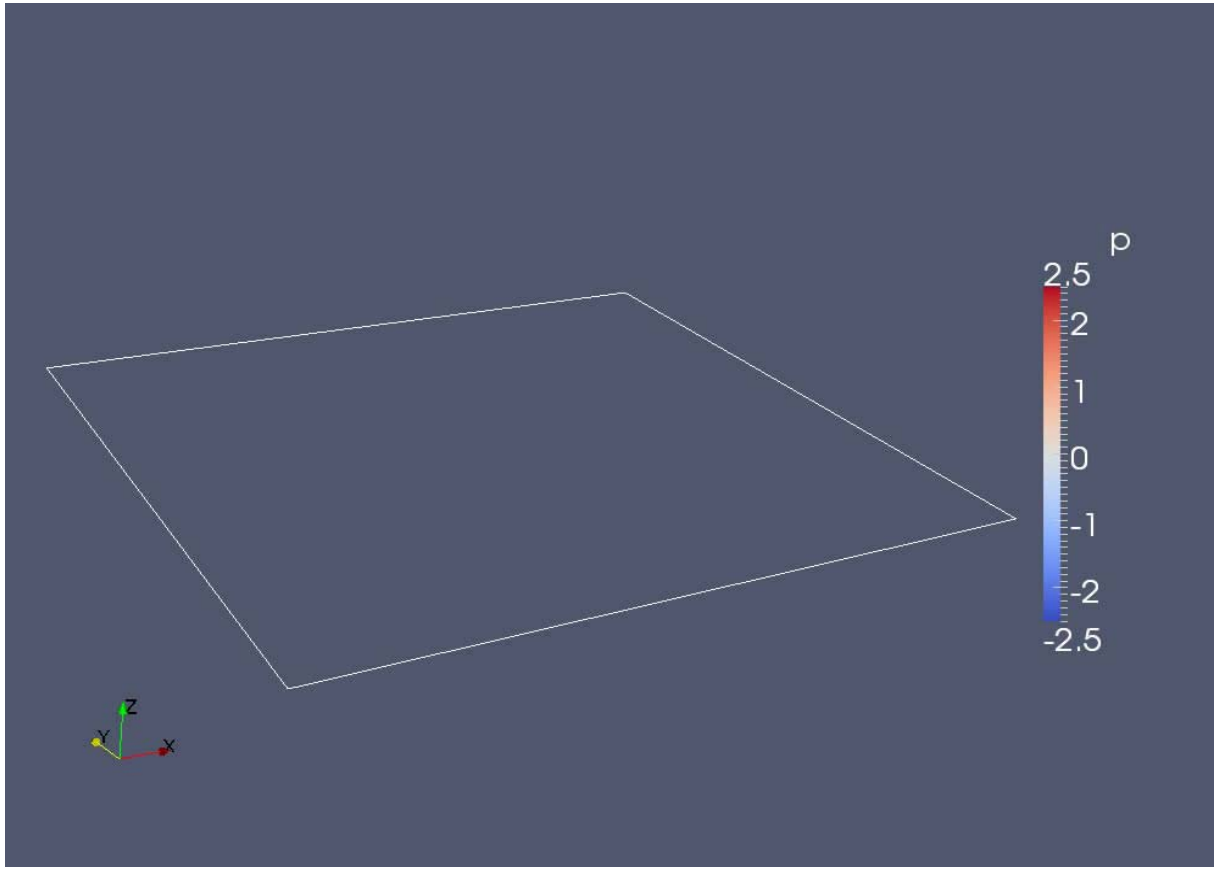
Problem from Hou's 2008 paper, (Jia, Hill, Fann, Harrison, 2009)

3-D Navier-Stokes Simulation using MADNESS periodic-domain, counter-rotating vortices



2-D slice of a 3-D simulation of two vortices rotating in opposite directions

Simulation 2-D slice



Essential techniques for fast computation

- Multiresolution Analysis

$$V_0 \subset V_1 \subset \cdots \subset V_n$$

$$V_n = V_0 + \left(V_1 - V_0\right) + \cdots + \left(V_n - V_{n-1}\right)$$

- Low-separation rank (Beylkin-Mohlenkamp)

$$f(x_1, \ldots, x_d) = \sum_{l=1}^{M} \sigma_l \prod_{i=1}^{d} f_i^{(l)}(x_i) + O(\varepsilon)$$

$$\left\| f_i^{(l)} \right\|_2 = 1 \qquad \sigma_l > 0$$

- Low-operator rank (e.g. SVD, HOSVD)

$$A = \sum_{\mu=1}^{r} \underline{u}_\mu \sigma_\mu \underline{v}_\mu^T + O(\varepsilon)$$

$$\sigma_\mu > 0 \qquad \underline{v}_\mu^T . \underline{v}_\lambda = \underline{u}_\mu^T . \underline{u}_\lambda = \delta_{\mu\lambda}$$

Another Key Component

- Trade precision for speed –everything
  - Don't do anything exactly
  - Perform everything in $O(\varepsilon)$
  - Require
    - *Robustness*
    - *Speed*
    - *Guaranteed, arbitrary, finite precision*

Multiwavelets and Fast Methods

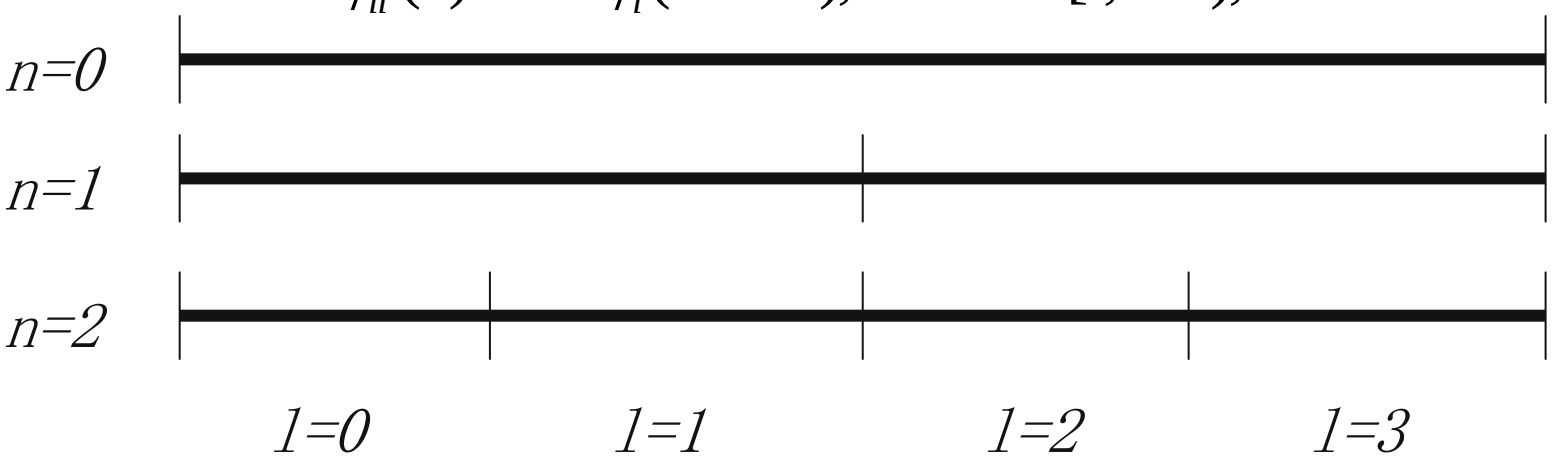
- Multiple representations permit each function and operator to have its own resolution, or representation depending on precision
- Automatic adaptivity for spectral expansion and order of accuracy (h & p)
  - Integral and differential operators (compatible)
  - Functions, projections between different level of expansions (depending on accuracy)
  - Compatible between function and operator calculus
- Accurate treatment of singularities
- Accurate treatment of higher order derivatives
  - Improved treatment of Gibbs type phenomena
- Green's function (Poisson, Helmholtz, etc.) via Low-Separation Rank
  - Fast real analysis based O(N log ε) method
  - Accuracy is proportional to
    - *Degrees of freedom*
    - *Work (flops)*
- DFT, Lippmann-Schwinger, Hartree-Fock, …
- Time-dependent DFT for molecular HF and DFT, Navier-Stokes

1-D Scaling Function Basis: Alpert's Multiwavelets

- Divide domain into 2^n pieces (level n)
  - Adaptive sub-division (local refinement)
  - l^{th} sub-interval $[l*2^{-n},(l+1)*2^{-n}]$ $l=0,\ldots,n-1$
- In each sub-interval define a polynomial basis
  - First k Legendre polynomials, P_i, rescale and normalized on [0,1)
  - Orthonormal, disjoint support

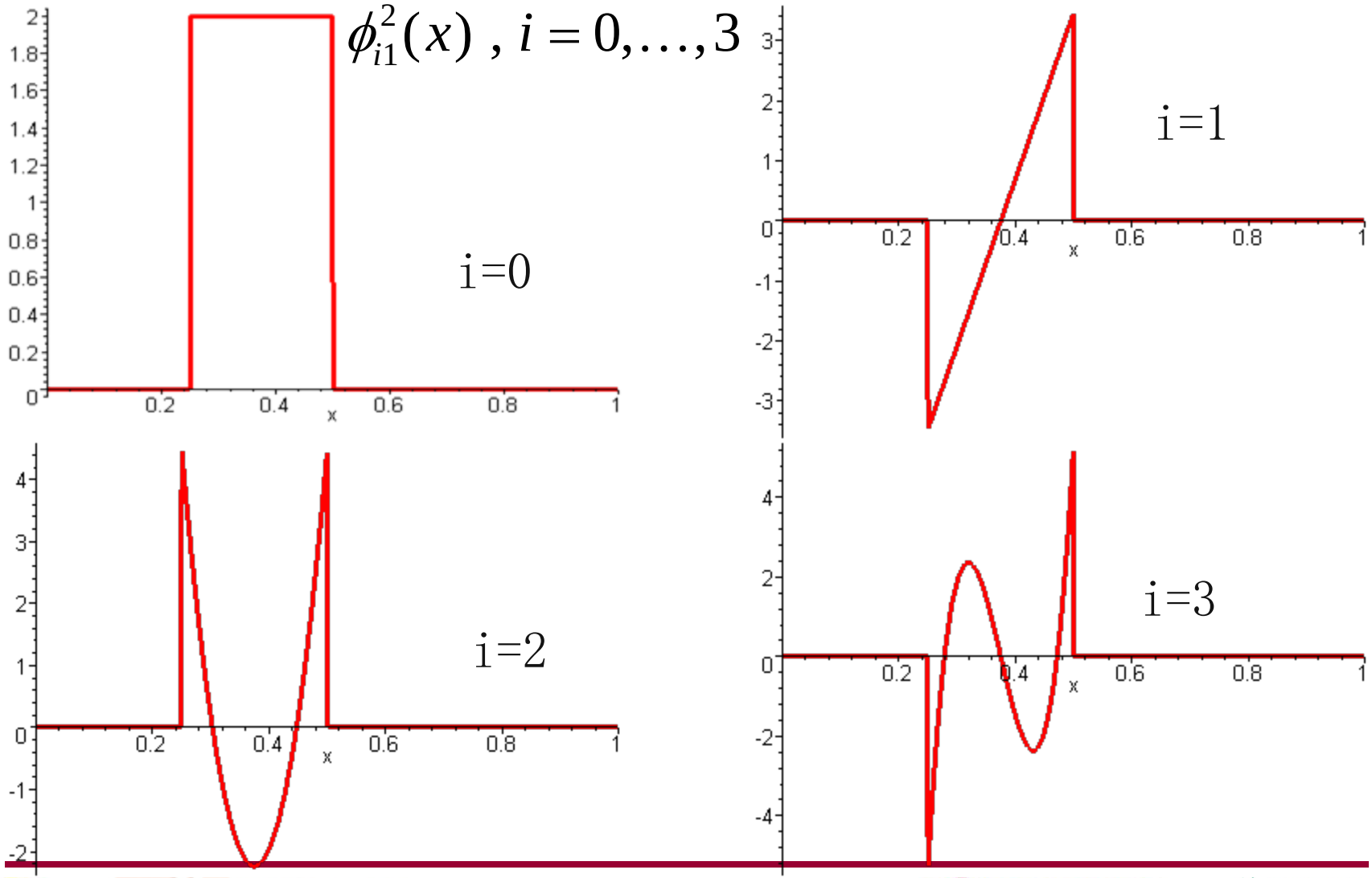
$$\phi_i(x) = \sqrt{2i+1}P_i(2x-1),\ldots x \in [0,1)$$

$$\phi_{il}^n(x) = 2^{n/2}\phi_i(2^n x - l),\ldots x \in 2^{-n}[l,l+1),\ldots 0 \le l \le n-1$$



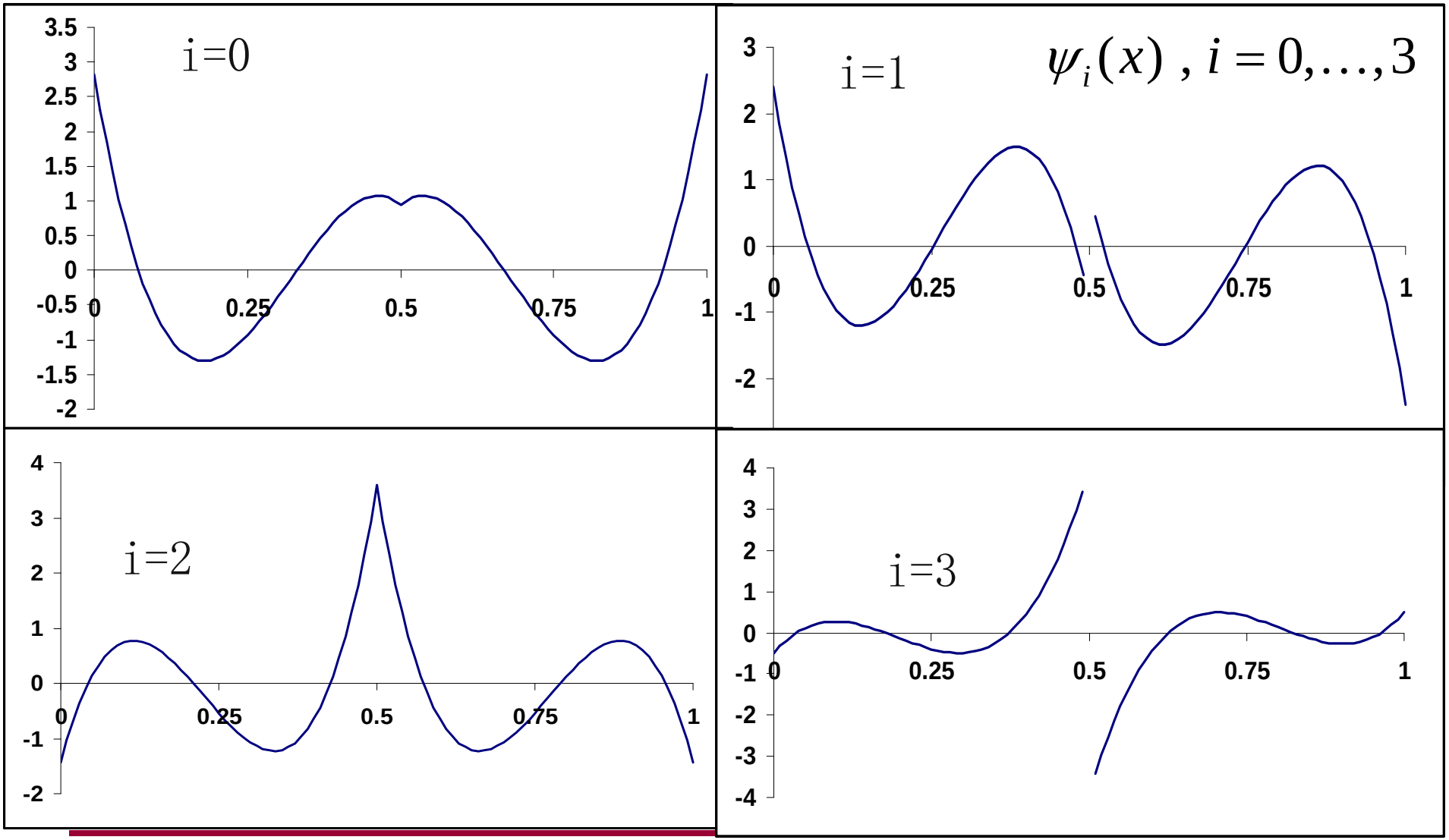
Scaling Function Basis:Alpert's, level, n=2

$$\phi_{i1}^{2}(x)\ ,\ i = 0,\ldots,3$$



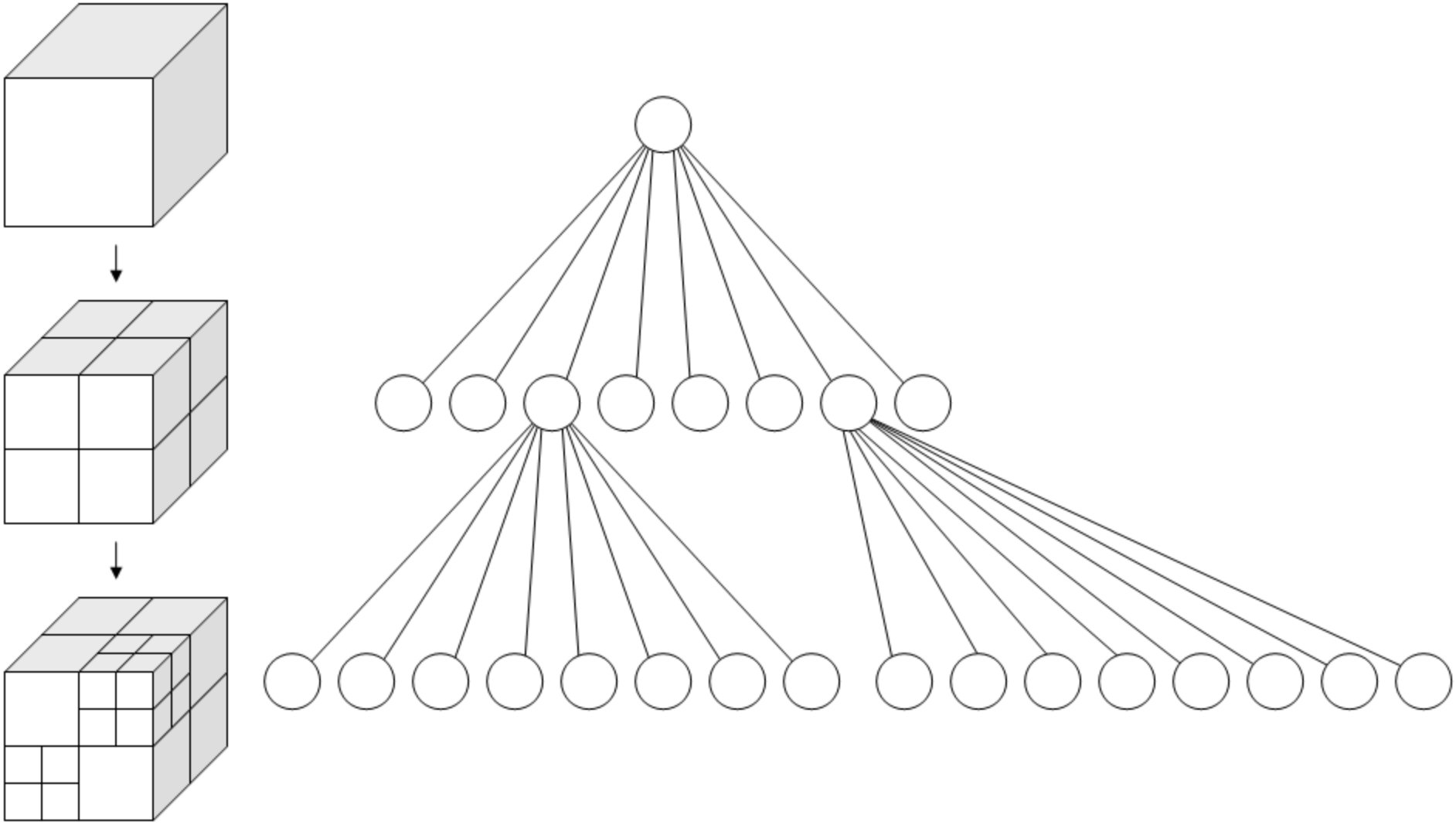
Multiwavelet Basis

$k=4, \quad n=0$



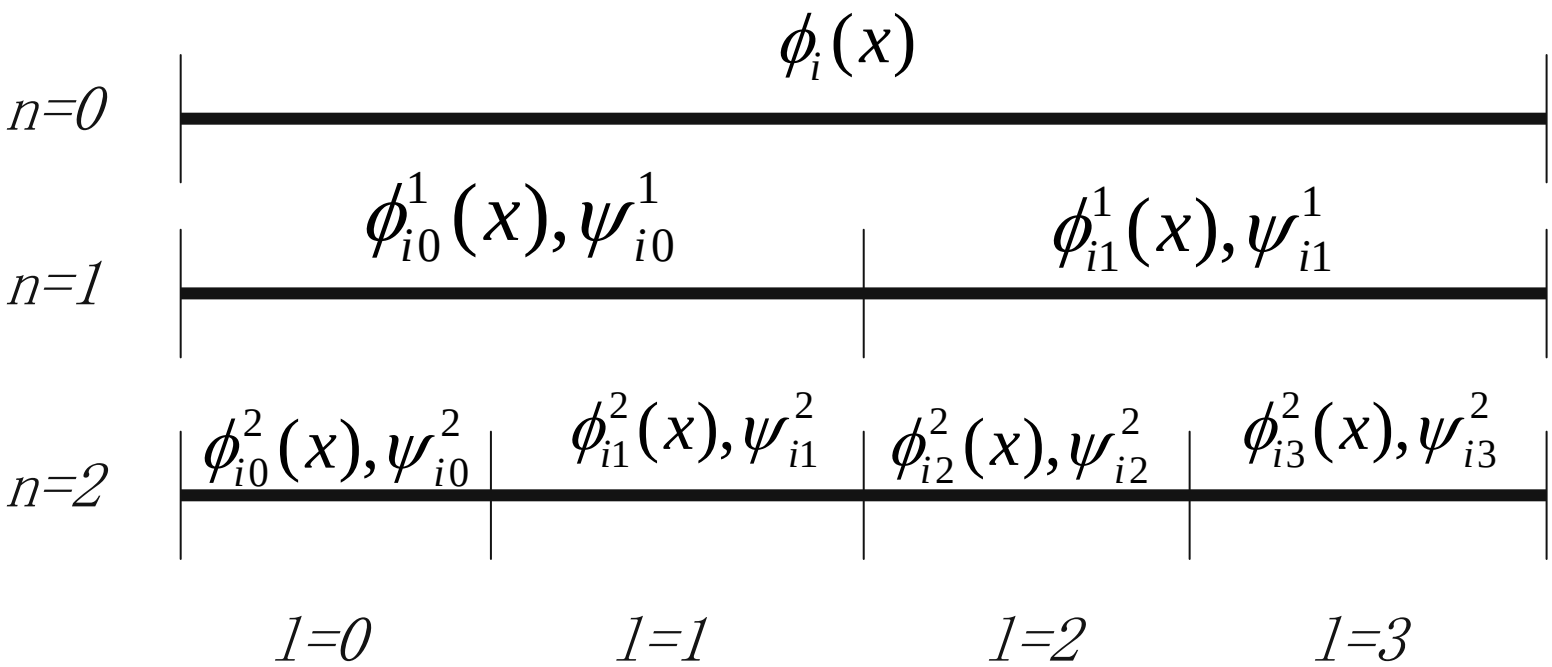
Office of Science
U.S. DEPARTMENT OF ENERGY
ASCR
SciDAC
Scientific Discovery through Advanced Computing
OAK RIDGE
National Laboratory

Oct-tree Adaptivity



Adaptivity by Spectral Expansion
Support of Basis Functions

$$f^{n}(x) = \sum_{i=0}^{k-1} s_{i0}^{0}\phi_{i0}^{0}(x) + \sum_{n'=0}^{n-1}\sum_{l=0}^{2^{n'}-1}\sum_{i=0}^{k-1} d_{il}^{n'}\psi_{il}^{n'}(x)$$



Two-scale relationship - I

$$\begin{pmatrix} \phi(x) \\ \psi(x) \end{pmatrix} = \sqrt{2} \begin{pmatrix} H^{(0)} & H^{(1)} \\ G^{(0)} & G^{(1)} \end{pmatrix} \begin{pmatrix} \phi(2x) \\ \phi(2x-1) \end{pmatrix}$$

Filter coefficients for k=1 (Haar)

$$\begin{pmatrix} H^{(0)} & H^{(1)} \\ G^{(0)} & G^{(1)} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\begin{pmatrix} s_0^0 \\ d_0^0 \end{pmatrix} = \sqrt{2} \begin{pmatrix} H^{(0)} & H^{(1)} \\ G^{(0)} & G^{(1)} \end{pmatrix} \begin{pmatrix} s_0^1 \\ s_1^1 \end{pmatrix} = \begin{pmatrix} s_0^1 + s_1^1 \\ s_0^1 - s_1^1 \end{pmatrix}$$

Hence, coefficients of scaling functions and wavelets are often referred to as <u>s</u>um and <u>d</u>ifference coefficients.

Compression of a function

$$V_n = V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_{n-1}$$

- Recursively apply the two-scale relation
- The basis is the scaling functions at level 0 and the multiwavelets at all levels

$$f^{n}(x) = \sum_{i=0}^{k-1} s_{i0}^{0} \phi_{i0}^{0}(x) + \sum_{n'=0}^{n-1} \sum_{l=0}^{2^{n'}-1} \sum_{i=0}^{k-1} d_{il}^{n'} \psi_{il}^{n'}(x)$$

- Haar basis (*k=1*) gives rise to calling coefficients of the scaling functions and wavelets as sums and differences respectively (hence notation *s* and *d*)
- Compression & reconstruction are *O(N)* operations

Two equivalent representations

- Scaling function basis (reconstructed)

$$f^{n}(x) = \sum_{l=0}^{2^{n}-1} \sum_{i=0}^{k-1} s_{il}^{n} \phi_{il}^{n}(x)$$

- Multi-wavelet basis (compressed)

$$f^{n}(x) = \sum_{i=0}^{k-1} s_{i0}^{0} \phi_{i0}^{0}(x) + \sum_{n'=0}^{n-1} \sum_{l=0}^{2^{n'}-1} \sum_{i=0}^{k-1} d_{il}^{n'} \psi_{il}^{n'}(x)$$

- Rapid compression/reconstruction
  - Asymptotically faster than the FFT
  - Use most appropriate basis for a given operation

A Third Equivalent Representation

- The function tabulated at the Gauss-Legendre quadrature points in each of the adaptively refined boxes
  - Enables rapid multiplication of functions and application of local functions (e.g., V_{xc})
  - Diagonal transformation from interpolating polynomials

Truncation Error

- Sparsity is from truncation.
- To satisfy the global error condition

$$\left\| f - f^{n} \right\|_2 \le \varepsilon \left\| f \right\|_2$$

- Truncate according to

- This is rather conservative – usually use

$$\left\| d_l^n \right\|_2 \le 2^{-n/2} \varepsilon \left\| f \right\|_2$$

$$\left\| d_l^n \right\|_2 \le \varepsilon$$

Summary I

- Scaling functions
  - Easy to manipulate, evaluate, integrate, …
- Multi-resolution analysis
  - Separates behavior between length scales
  - Local truncation while preserving global error bound
  - Vanishing moments
- Multi-wavelets
  - High-order convergence with adaptive representation
  - Disjoint support – efficient description of singularities if locacated at faces/edges/corners (more efficient than smooth wavelets since they do not have disjoint support)
- Fast compression and reconstruction
  - Orthogonal transformations – numerically stable

Extension to higher dimensions

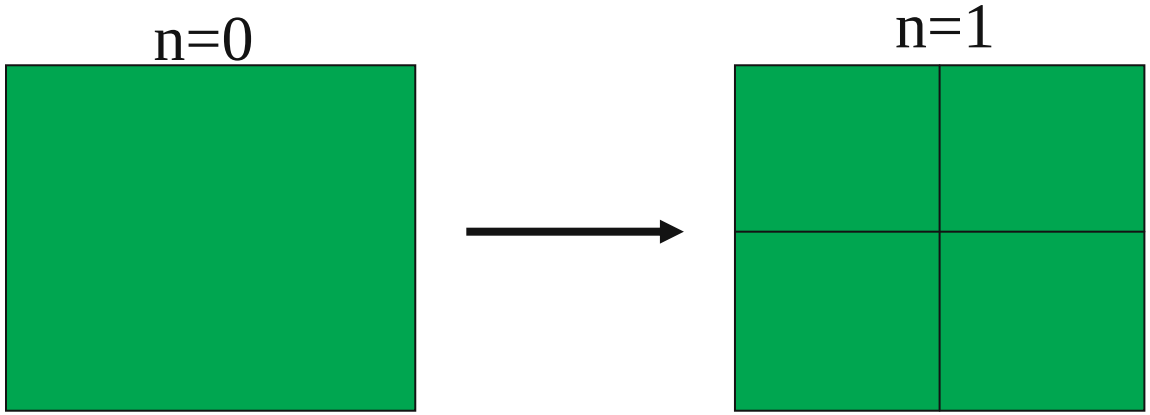
- Scaling function basis is tensor product

- Wavelet basis – tensor product is one choice
  - Standard form – compress each dimension just as for a matrix
  - But cannot refine strictly locally since length scales are mixed between dimensions
- To refine locally need the non-standard form

$$\phi^n_{ii'll'}(x,y) = \phi^n_{il}(x)\phi^n_{i'l'}(y)$$

Non-standard form of functions

- Construct local basis for $W_{n-1}=V_n - V_{n-1}$



$$V_0 = \phi(x)\phi(y)$$

$$V_1 = \sqrt{2}\left\{\begin{array}{l}\phi(2x)\phi(2y), \phi(2x)\phi(2y-1), \phi(2x-1)\phi(2y), \\ \phi(2x-1)\phi(2y-1)\end{array}\right\}$$

$$= \{\phi(x)\phi(y), \phi(x)\psi(y), \psi(x)\phi(y), \psi(x)\psi(y)\}$$

$$W_0 = V_1 - V_0 = \{\phi(x)\psi(y), \psi(x)\phi(y), \psi(x)\psi(y)\}$$

Non-standard form of operators

- Standard form
  - Matrix elements between different length scales
  - Not very efficient on modern computers with deep memory hierarchies
  - Potentially *O(Nlog N)* non-zero terms
- Non-standard form
  - No matrix elements between lengths scales
  - *O(N)* terms
  - Act on (modified) non-standard form of functions
  - Derivation is instructive

Non-standard form of operators - II

- P_n an orthogonal projection into V_n
- Q_n an orthogonal projection into $W_n = V_{n+1} - V_n$

  $P_n + Q_n = P_{n+1}$
- Consider the projection of an operator T

$$
\begin{aligned}
T_n &= P_n T P_n \\
&= \left(P_{n-1} + Q_{n-1}\right) T \left(P_{n-1} + Q_{n-1}\right) \\
&= P_{n-1} T P_{n-1} + Q_{n-1} T Q_{n-1} + Q_{n-1} T P_{n-1} + P_{n-1} T Q_{n-1} \\
&= T_{n-1} + A_{n-1} + B_{n-1} + C_{n-1} \\
&= T_0 + \sum_{\eta=0}^{n-1} A_\eta + B_\eta + C_\eta
\end{aligned}
$$

NS Form of Operators III

- Matrix elements in the scaling function basis

- Matrix elements of the NS form

$$\left[r_{lm}^{n} \right]_{ij} = \int \phi_{il}^{n}(x) \left(T \phi_{jm}^{n}(x) \right) dx$$

$$\begin{pmatrix} r_{l,m}^{n-1} & \gamma_{l,m}^{n-1} \\ \beta_{l,m}^{n-1} & \alpha_{l,m}^{n-1} \end{pmatrix} = \begin{pmatrix} H^{(0)} & H^{(1)} \\ G^{(0)} & G^{(1)} \end{pmatrix} \begin{pmatrix} r_{2l,2m}^{n} & r_{2l,2m+1}^{n} \\ r_{2l+1,2m}^{n} & r_{2l+1,2m+1}^{n} \end{pmatrix} \begin{pmatrix} H^{(0)} & H^{(1)} \\ G^{(0)} & G^{(1)} \end{pmatrix}^{T}$$

Integral operators - I

- Consider

$$Tf(x) = \mathrm{pv}\int dy K(x-y) f(y)$$

$$\left[r_{lm}^{n}\right]_{ij} = \int dx \int dy K(x-y)\phi_{il}^{n}(x)\phi_{jm}^{n}(y)$$

$$= 2^{-n}\int dx \int dy K(2^{-n}\left(x-y+l-m\right))\phi_i(x)\phi_j(y)$$

$$= 2^{-n}\int dz K(2^{-n}\left(z+l-m\right))\Phi_{ij}(z)$$

$$\Phi_{ij}(z) = \int dx \phi_i(x)\phi_j(x-z)$$

$$= \sum_{p=0}^{2k-1}\left(c_{ijp}^{(-)}\phi_p(z+1) + c_{ijp}^{(+)}\phi_p(z)\right)$$

$$\left[r_{l}^{n}\right]_{p} = 2^{-n}\int dz K(2^{-n}\left(z+l\right))\phi_p(z)$$

Integral operators - II

- The autocorrelation and cross-correlation functions also have a 3 term 2-scale relations.
- Matrix elements easy to evaluate from compressed form of kernel *K(x)*
- Application in 1-d is fairly efficient
  - $O(k^2)$ operations
- In 3-d seems to need $O(N_{box}k^6)$ operations
  - Prohibitively expensive
  - Cross-correlations reduces to $O(8N_{box}k^4)$
  - LSR does more
- More intelligent approach
  - $O(N_{box}k^4)$ operations for many "physical" kernels
  - Even better is known to be possible

Example: Poisson Kernel

- Example: Poisson Kernel

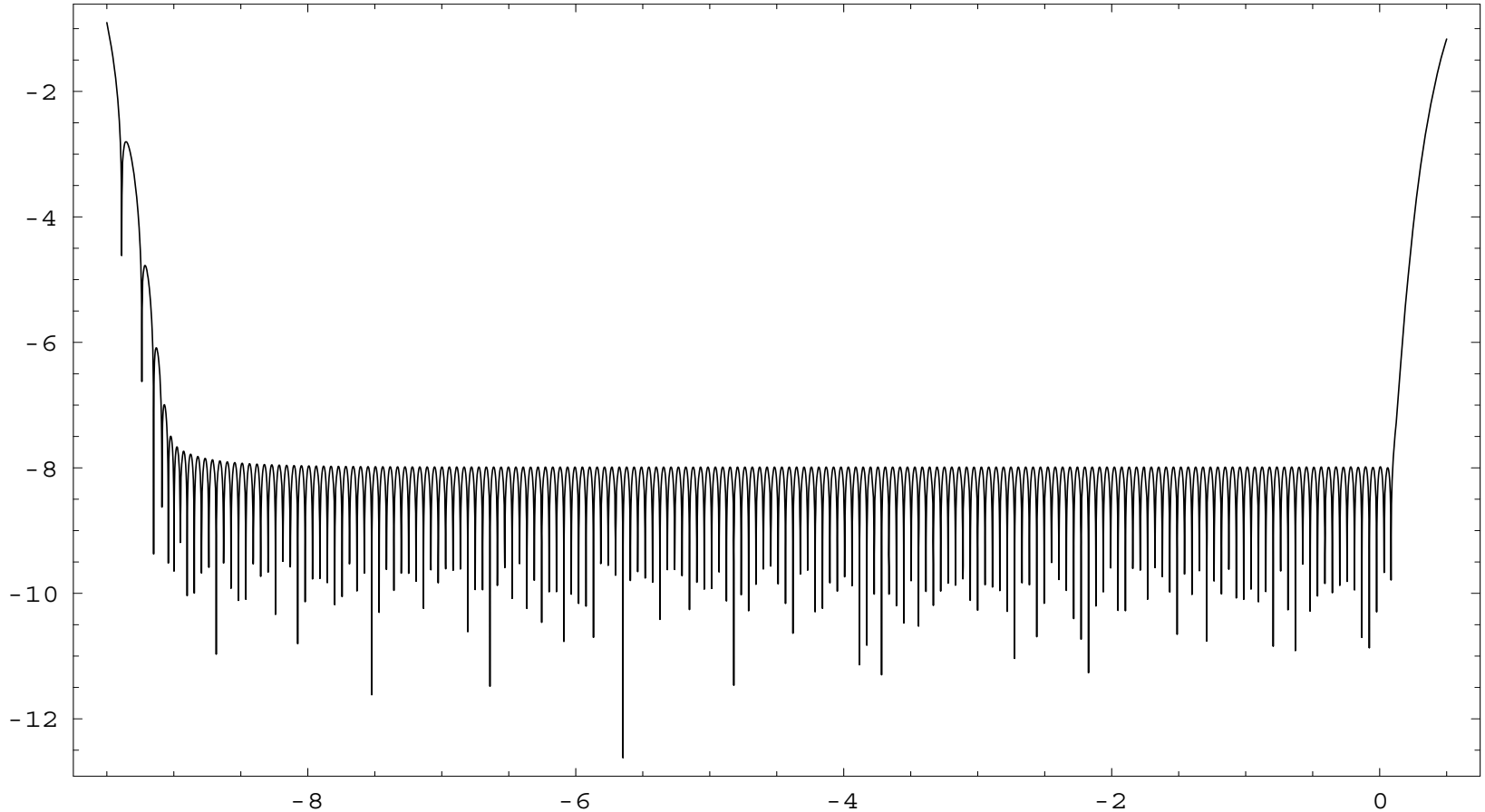
$$\left| \frac{1}{r} - \sum_{m=1}^{M} w_m e^{-p_m r^2} \right| \le \frac{\varepsilon}{r}$$

$$\delta \le r \le 1,$$

$$0 < p_m, w_m$$

$$M = O(-\log \delta)$$

Approximation of Poisson Kernel



Error ($\log_{10}$-scale) of approximating the Poisson kernel from [1e-9, 1] with M=89 terms.

Low Separation Rank Approximation, Poisson Example

Proposition 2. *For any $\alpha > 0$, $0 < \delta \leqslant 1$, and $0 < \epsilon \leqslant \min\{\frac{1}{2}, \frac{8}{\alpha}\}$, there exist positive numbers p_m and w_m such that*

$$\left| r^{-\alpha} - \sum_{m=1}^{M} w_m e^{-p_m r^2} \right| \leqslant \epsilon_0 r^{-2} \quad \text{for all } \delta \leqslant r \leqslant 1, \tag{3}$$

where

$$M = \log \epsilon^{-1} \left[c_0 + c_1 \log \epsilon^{-1} + c_2 \log \delta^{-1} \right]. \tag{4}$$

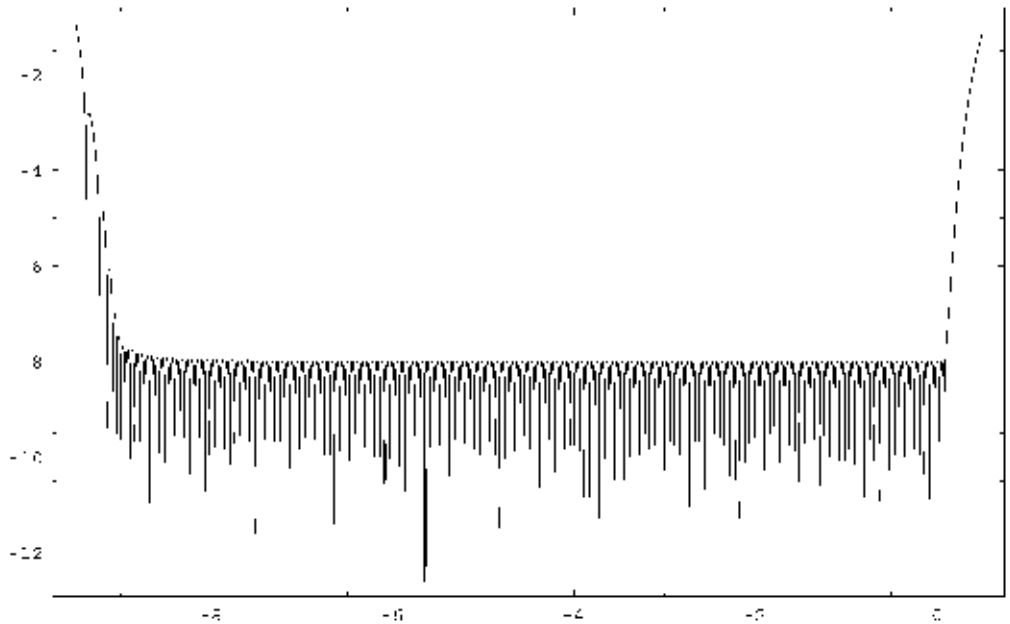
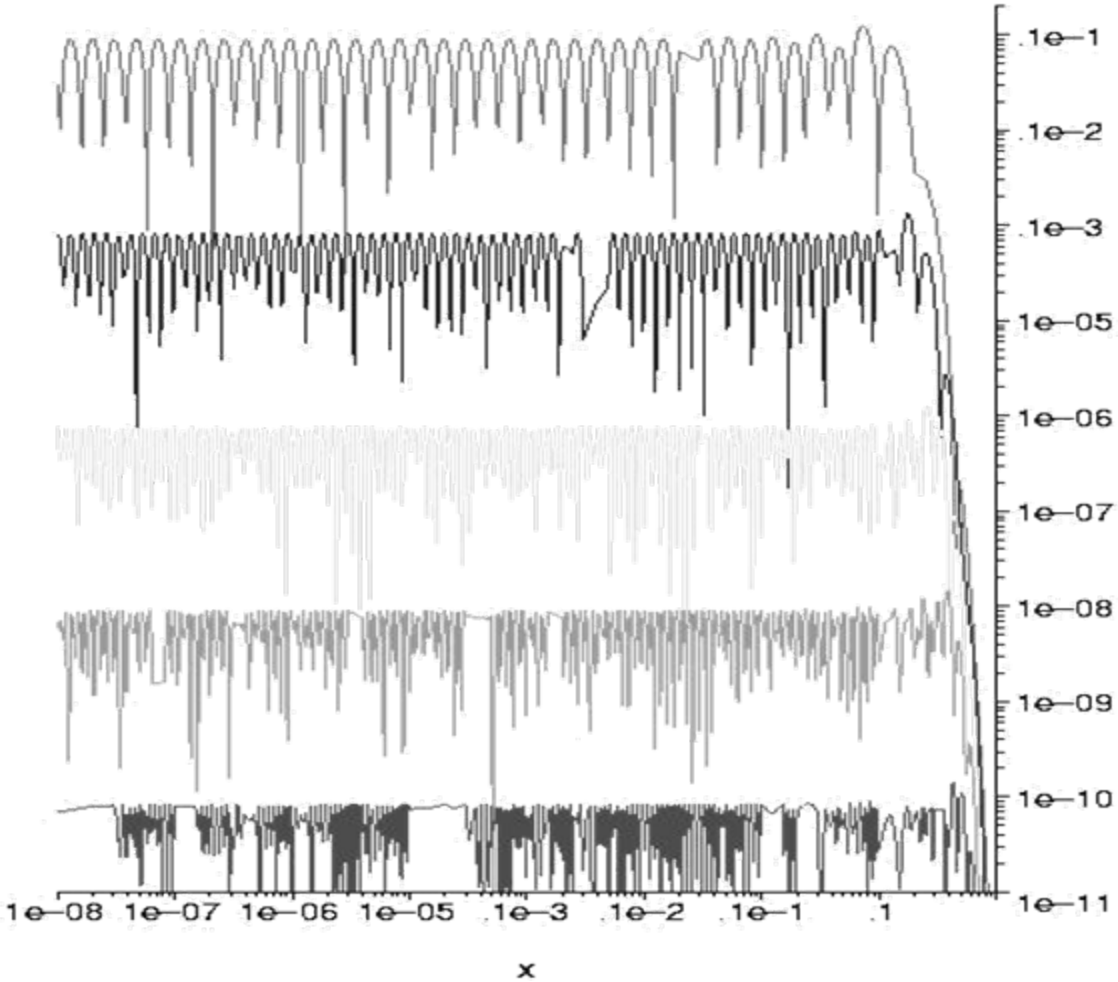


Fig. 1. Error ($\log_{10}$) of approximating the Poisson kernel using Proposition 1 with consequent optimization, where $\epsilon \approx 10^{-8}$, $10^{-9} \leqslant \|\mathbf{x}\| \leqslant 1$ and $M = 89$.

B-C-F-H, Achä, 2007



Automatically generated representations of *exp(-30r)/r* accurate to 1e-10, 1e-8, 1e-6, 1e-4, 1e-2 (relative error) for *r* in [1e-8,1] (92, 74, 57, 39 and 21 terms, respectively).

Low-energy *scattering* states also possible (but stronger dependence on range)

Summary of Computational Setup

- Formulate using integral equation, e.g. Lippman-Schwinger Equation
- Use multiresolution analysis and low separation rank approximation
  - E.g.: representation by multiwavelets and separated representations using Gaussians
- Solve using iterative methods or combination of direct and iterative methods
- Discontinuous spectral element with multi-resolution and separated representations for fast computation with guaranteed precision
- Now relax the tree structure—map tree to hash table

- *Download, FAQ and wiki*

http://code.google.com/p/m-a-d-n-e-s-s/

--will require mpicc, mpicxx, mpif90,.., lapack, blas, gfortran

Example code in m-a-d-n-e-s-s/src/apps/examples

The funding

- The development of MADNESS is funded by the U.S. Department of Energy, Office of Advanced Scientific Computing Research (OASCR) and the division of Basic Energy Science, Office of Science, under contract DE-AC05-00OR22725 with Oak Ridge National Laboratory, by the Math Base program (PI: Fann), SciDAC SAP in computational chemistry (PI: Fann) and SciDAC BES (PI:Harrison) and IAA Algorithm's Project (PI: Geist).
- The application of MADNESS-NDFT to nuclear physics is funded through SciDAC UNEDF (PIs: R. Lusk and W. Nazarewicz) by DOE-OASCR.
- This research was performed in part using
  - resources of the National Energy Scientific Computing Center which is supported by the Office of Energy Research of the U.S. Department of Energy under contract DE-AC03-76SF0098,
  - the National Center for Computational Sciences at Oak Ridge National Laboratory under contract DE-AC05-00OR22725 .