Wavelet-based Poisson Solver for Use in Particle-in-Cell Simulations

Balša Terzić
Northern Illinois University

Ilya Pogorelov
Lawrence Berkeley National Lab

November 6, 2004
Thank you, Henry
Motivation

• Insight into the dynamics of multi-particle systems heavily relies on $N$-body simulations

• It is important for $N$-body codes (Poisson solver) to:
  • account for multiscale dynamics – details do matter!
  • minimize numerical noise due to $N_{\text{simulation}} \ll N_{\text{physical}}$
  • be as efficient as possible, without compromising accuracy
    • for some applications: have a compact representation of system's history

• We present a wavelet-based Poisson solver which addresses all of the requirements listed above
Wavelets

One-Dimensional

(a) Haar wavelet, (b) Morlet wavelet, (c) Daubechies wavelet.

Two-Dimensional

Daubechies’ least symmetric
\( N = 10 \) two-dimensional wavelet.
Wavelet Denoising – An Example


Key Finding: Wavelet denoising in 2D simulation → equivalent of 100-fold more macroparticles.
Wavelet Compression – An Example

- 2D Gaussian distribution
- 1.64×10^6 particles on a 256x256 grid – 65536 coefficients
- Daubechies 12th order wavelets
- thresholding: if |c_{i,j}| < \varepsilon, set to 0
- optimal signal-to-noise ratio (SNR) for a small fraction of coefficients – here < 0.115%
- IDEA: exploit sparsity in wavelet space!
Wavelet-based Poisson Solver

Poisson equation in physical space

\[ \Delta u = f \]

boundary conditions

\[ u_{\text{bnd}} = g \]

discretize Poisson equation on a \( N \times N \times N \) grid

transform discretized Poisson equation to wavelet space

Constrained Preconditioned Conjugate Gradient (CPCG)

Compress by thresholding source \( f \), operator \( L \), solution \( u \)

precondition Laplacian \( L \) with diagonal preconditioner \( P \)

\[ k (L_w) \sim O(N^2) \]

\[ k (PL_w P) \sim O(N) \]
**Constrained Preconditioned Conjugate Gradient (CPCG)**

- iterative solver
- convergence rate:
  \[ |u - u^i|_2 \leq \left( \frac{\sqrt{k - 1}}{\sqrt{k + 1}} \right)^i |u|_2 \]
- initial guess: \( u \) at previous time step
- diagonal (cheap!) preconditioner in wavelet space

**Strengths:**
- removing numerical noise *while* compressing
- hierarchy of scales conserved

**Weaknesses:** boundary conditions!
- need potential on the surface of the grid
  - Green's functions when known
  - open BCs: *coordinate transforms* to map \( \infty \) onto the surface of the grid
3D 'fuzzy cigar' on a 32x32x32 grid

BCs: grounded rectangular pipe ($V=0$ on the sides), open in $z$-direction
3D Plummer sphere on a 32x32x32 grid

BCs: open in all directions (analytically specified)
Conclusions

• Presented an iterative wavelet-based Poisson solver (CPCG)
  - wavelet compression and denoising achieves computational speedup
  - efficient preconditioning and sparsity of operators in wavelet space further reduce computational time

• Conceptually verified and thus far quite promising

• Still to be done:
  - optimization (including finding a better preconditioner) and parallelization
  - plugging the algorithm into existing $N$-body code
  - benchmarking against existing beam simulation codes
  - modeling realistic charged particle beams
  - use it in $N$-body simulations of self-gravitating systems
  - wavelet formalism can be applied to other PDEs
Auxiliary Slides
Wavelets

• What are wavelets?
  - orthogonal basis of functions
  - family of high- and low-pass filters
    • filters can be derived from the requirements on wavelets, without knowing the shape of the wavelet family \emph{a priori}

\[ f(x) = \sum h_i^k \Psi_i^k(x) \]
\[ \Psi_i^k(x) = \Psi(2^k x - i) \]

• Why wavelets?
  - simultaneous \textbf{time} and \textbf{frequency} localization (FFT only frequency)
  - \textbf{data compression}: computational speed-up
  - \textbf{denoising}: removing graininess has the same effect as running the same multiparticle $N$-body simulation with many times more particles
How Do Wavelets Work?

Wavelet analysis (wavelet transform):

- **Approximation** – apply low-pass filter to Signal and down-sample
- **Detail** – apply high-pass filter to Signal and down-sample
- **Wavelet synthesis** (inverse wavelet transform): up-sampling & filtering
- **Complexity**: \(4MN\), \(M\) the size of the wavelet, \(N\) number of cells
  - Recall: FFT complexity \(4N \log_2 N\)
Wavelet Decomposition

The continuous wavelet transform of a function $f(t)$ is

$$\gamma(s,\tau) = \int_{-\infty}^{\infty} f(t) \Psi_{s,\tau}^* (t) dt;$$

$$\Psi_{s,\tau} (t) = \frac{1}{\sqrt{s}} \psi \left( \frac{t - \tau}{s} \right)$$

where $\psi$ is the wavelet function with scale and translation dimensions $s$ and $\tau$, respectively.

![Wavelet Decomposition Illustration](image-url)
2D EXAMPLE: “MICROBUNCHED” DENSITY DISTRIBUTION (with $U=0$ starting guess)

BCs: open in all directions (analytically specified)