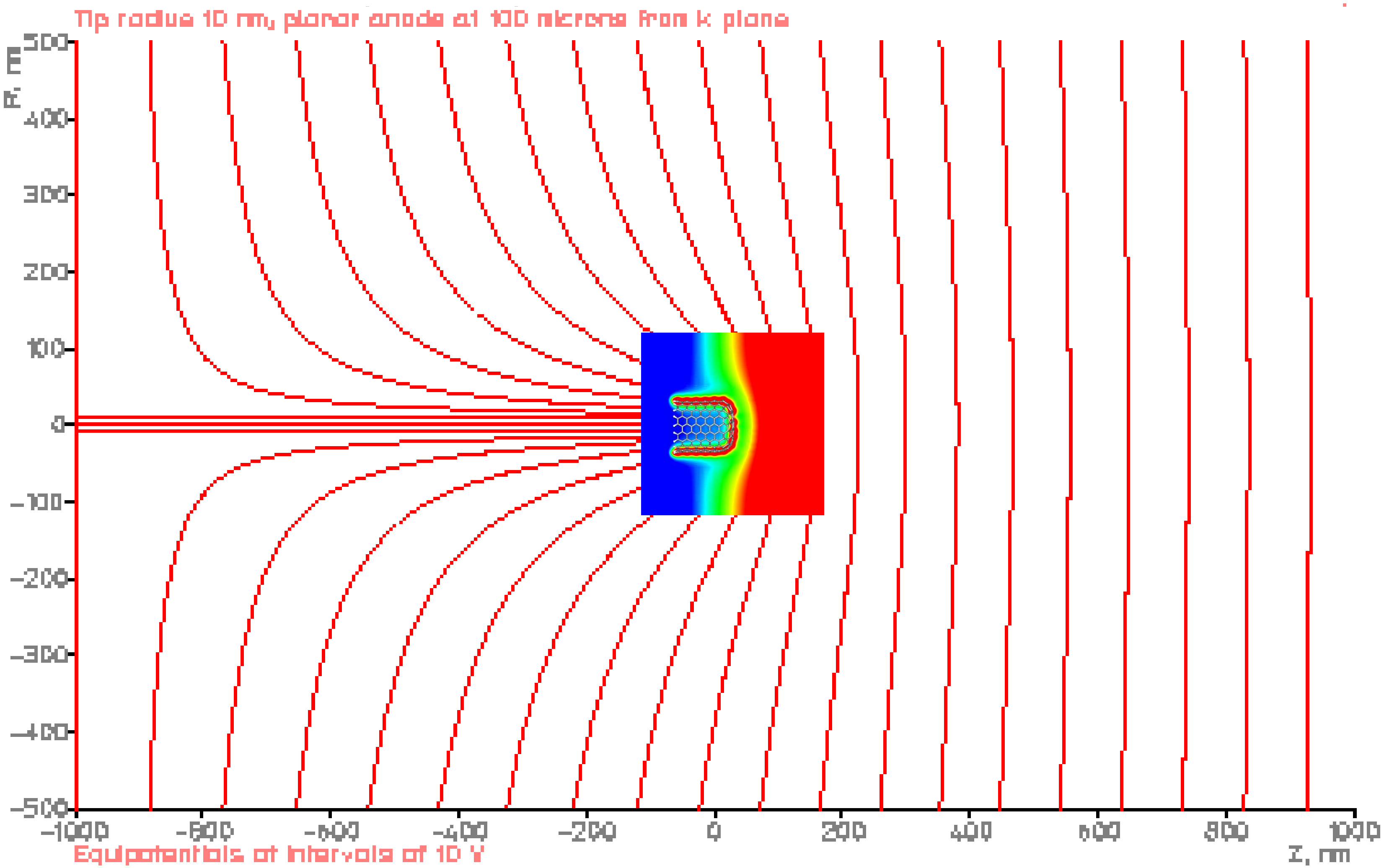
Solving Poisson's equation with non-periodic boundary conditions

Andriy H. Nevidomskyy

ESDG, 9 June 2004

- Motivation
- Traditional Kohn-Sham scheme
- Methods of solving Poisson's equation
- Finite differences scheme
- Results of calculations

Motivation - Field Emission From Carbon Nanotubes



Kohn-Sham scheme

$$E[\psi_i(\mathbf{r})] = -\frac{\hbar^2}{2m}\sum_i f_i \int \mathrm{d}^3r\, \psi_i^*(\mathbf{r})\nabla^2\psi_i(\mathbf{r}) \;+ \int \mathrm{d}^3r\, V_{ion}(\mathbf{r})\, n(\mathbf{r})$$

$$+ \int \mathrm{d}^3r\, \epsilon_{xc}[n(\mathbf{r})]n(\mathbf{r}) + E_H[n(\mathbf{r})]$$

Hartree energy:

$$E_H[n(\mathbf{r})] = \frac{1}{2}\iint \mathrm{d}^3r\, \mathrm{d}^3r'\, \frac{n(\mathbf{r})\, n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \equiv \frac{1}{2}\int \mathrm{d}^3r\, \phi(\mathbf{r})\, n(\mathbf{r})$$

$\boxed{\nabla^2\phi(\mathbf{r}) = 4\pi n(\mathbf{r})}$ – Poisson's equation

Solving Poisson's equation

$$\boxed{\nabla^2 \phi(\mathbf{r}) = 4\pi n(\mathbf{r})}$$

Numerical methods of solving PDEs:

– **finite differencing**	– **simplest by far**
– finite elements	– often used in engineering, cumbersome
– Monte Carlo	– stochastic
– spectral methods	– for regular geometries and smooth fields
– variational methods	

Solving Poisson's equation by finite differences

$$x_i = x_0 + i\Delta_x, \qquad i = 0, 1, ...N_x$$

$$y_j = y_0 + j\Delta_y, \qquad j = 0, 1, ...N_y$$

$$z_k = z_0 + k\Delta_z, \qquad k = 0, 1, ...N_z$$

$$\frac{\partial^2 u}{\partial x^2} \Longrightarrow \frac{u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}}{\Delta_x^2}$$

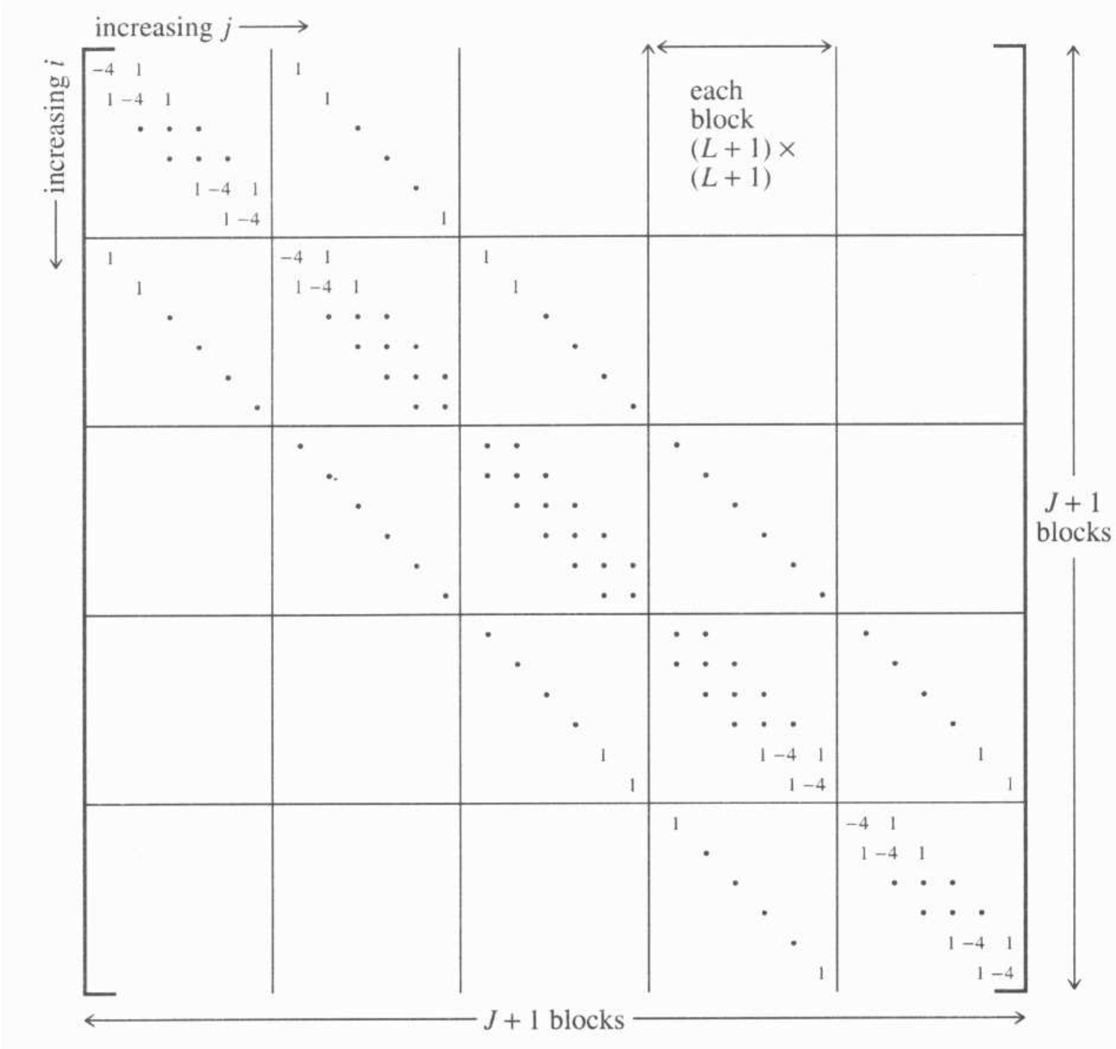
$$\frac{\partial^2 u}{\partial y^2} \Longrightarrow \ldots$$

$$\frac{\partial^2 u}{\partial z^2} \Longrightarrow \ldots$$

Matrix equations

$$\mathbf{A} \cdot \mathbf{u} = \mathbf{b}$$

Finite difference matrix



Solving finite differences: Relaxation methods

$$\mathcal{L}u = f$$

$$\Downarrow$$

$$\frac{\partial u}{\partial t} = \mathcal{L}u - f$$

$$\Downarrow$$

In finite differences – *Jacobi's method*:

$$u^{n+1}_{i,j,k} = u^{n}_{i,j,k} + \left(\mathcal{L}u - f_{i,j,k}\right)\Delta t$$

Improve: **Successive overrelaxation method (SOR)**

$$\boxed{\mathbf{u}^{n+1} = \mathbf{u}^{n} + \omega\,\left(\mathcal{L}\mathbf{u}^{n} - \mathbf{f}\right)\Delta t, \qquad 1 < \omega < 2}$$

Remarks on numerical efficiency

Matrix $J \times J \times J$, where J is a fine grid dimension in ONETEP ($J \sim$ number of "psync" functions in each direction).

Successive overrelaxation (SOR) method:
Number of iterations to reduce initial error by a factor of 10^{-p}:

$$N_{\text{iter}} \simeq \frac{1}{3} pJ \sim J$$

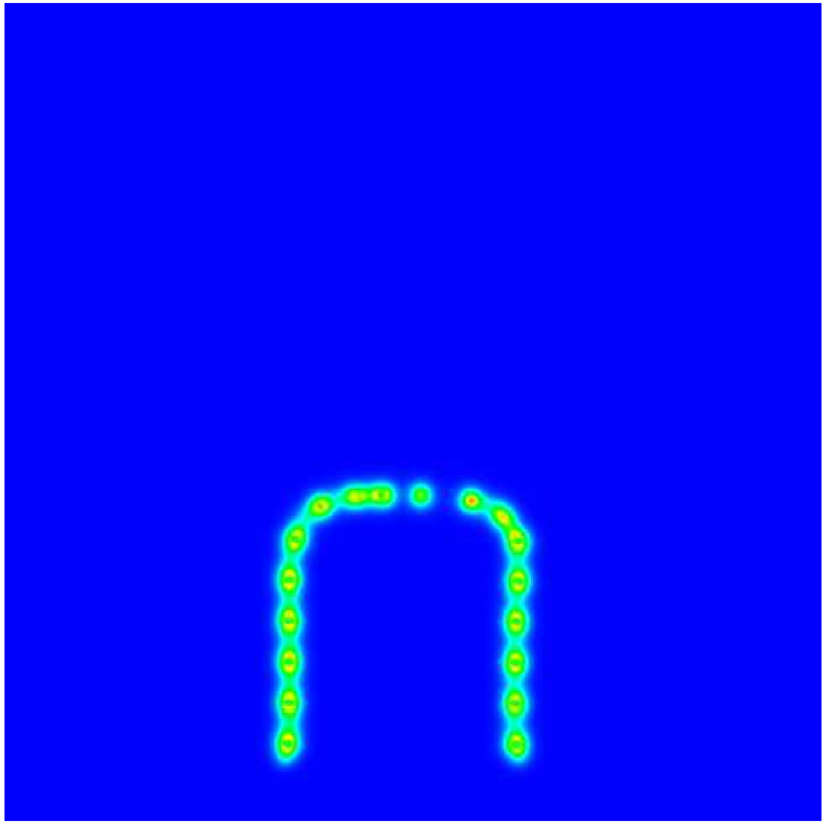
Each iteration involves J^3 operations $\Longrightarrow$ SOR is $O(J^4)$ method, i.e. $O(N \times N^{1/3})$, where $N = J^3$.

C.f.:	FFT transform	–	$O(N \log(N))$ method
	multigrid methods	–	$O(N)$ method

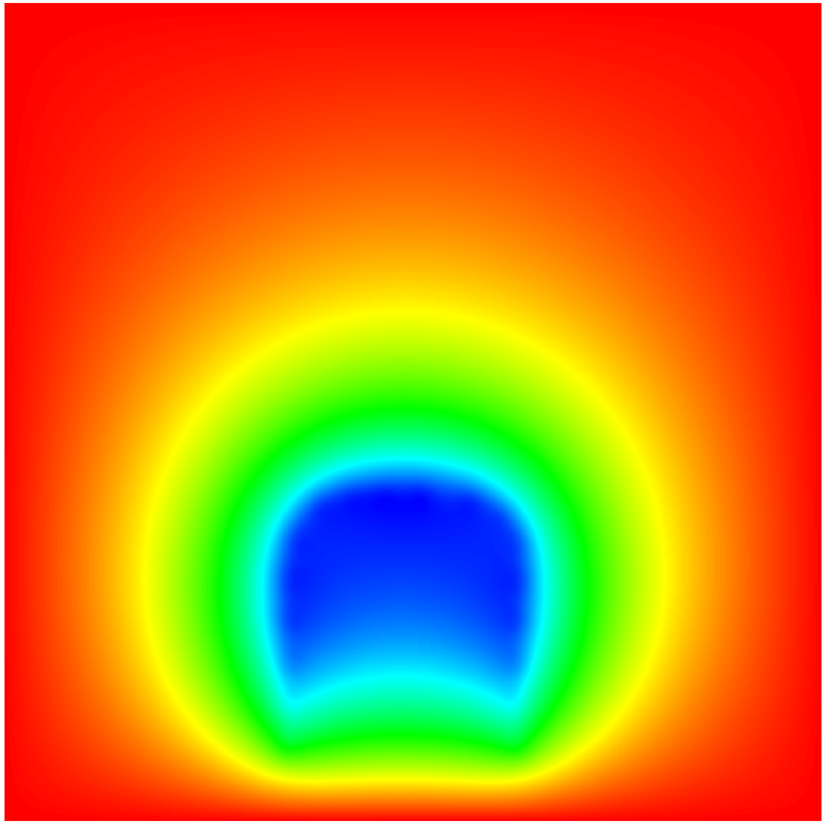
N	N_{iter}	time
10	26	0.001 sec on Pentium-4 (2.6GHz)
310	1075	1.5 hours on Pentium-4 (2.6GHz)

Results

Charge



Hartree potential



Dirichlet boundary conditions $\phi(\mathbf{r})|_\Gamma = 0$ $\nearrow$