# Solving Laplace's Equation

- Laplace's equation, $\nabla^2 V = -\rho$, is fundamental equation of electrostatics. Also steady-state heat flow, soap bubbles, fluid dynamics, gravity…
- How can we solve this when boundary conditions are given?
- $\nabla^2 V = d^2V/dx^2 + d^2V/dy^2 + \ldots$ $dV/dx = (f(x+dx)-f(x))/dx$, $d(dV/dx) = (f'(x)-f'(x-dx))/dx = (f(x+dx)-2f(x)+f(x-dx))/dx^2$. In 2D, $\nabla^2 f = f(r+dx)+f(r-dx)+f(r+dy)-f(r-dy)-4f(r)$.
- Rewrite: $f(r) = (\rho + \sum f_{neighbors})/2n_{dim}$.

# Method of Relaxation:

- $f(r)=(\rho+\sum f\text{neighbors})/2n_{dim}$ suggests one possible way to solve.
- Take current f(r), and replace by suitably averaged neighbors (plus ρ if charge is non-zero).
- This works, and is called method of relaxation (since we relax to solution).
- It does not, however, work very well. Convergence is slow.
- Another way to write: $f(x,t+dt)=f(x,t)+\alpha(f_{neighb}(x,t)-f(f,t))$.
- Relaxation iterations can be thought of as time evolution of this setup. We can increase $\alpha$ to speed convergence. Converges faster, but still slow. Need $0<\alpha<2$ (why?)

# Conjugate Gradient

- We can also write as matrix equation
- In interior [-4 1 1 1 1][$f_{00}$ $f_{10}$ $f_{-10}$ $f_{01}$ $f_{0-1}$=$\rho_{00}$
- If potential fixed on boundaries, we can just keep those function values fixed.
- This lets us use matrix solvers to solve Ax=b
- Standard tool is (preconditioned) conjugate gradient, works most naturally for positive-definite A. Happily, for Lapace/Poisson, A is already positive-definite.
- Conjugate gradient can be plugged in, converges much faster than standard relaxation - in particular gets to exact solution in finite # of iterations.
- Boundary conditions can be put on right, since we aren't solving for them.

# CG 2

- In general, some cells in domain have fixed values. Rather than list all these cells, we'll use a mask. Mask=1 where we define boundary, 0 otherwise.
- Rather than keeping track, we can take x -> (1-mask)x to zero out boundaries. Then Ax doesn't need to know about mask.
- RHS we can get from using mask * boundary conditions.
- With this, we can do matrix operations by simple sums over domain.
- Let's write a few examples…

# Multigrid

- Relaxation is fast on small scales, slow on large
- If grid is n cells across, takes n timesteps to propagate information.
- Trick: solve low-resolution problem.  Gets large scales right quickly
- Gradually increase resolution - then only need a few steps to solve for corrections introduced by higher-res
- This can be very fast - low-res parts are extremely fast, so can solve full problem for the price of a few full-res iterations.

# Fourier techniques

- For Laplace, we know Green's function is 1/r.
- If we know charge, we could work out potential everywhere via convolution with 1/r.
- We can again use CG to solve for rho on the boundaries that gives V on the boundaries.
- We get V everywhere else through convolution.
- 