# Lattice Gas Simulations

Lattice gases are simple cellular automata models of gases and fluid dynamics.

The basic model is as follows:

- particles move on a 2D grid
- particles can move in one of four directions
- if exactly two particles meet head-on, they bounce off in perpendicular dierctions
- all other particle meetings result in no changes

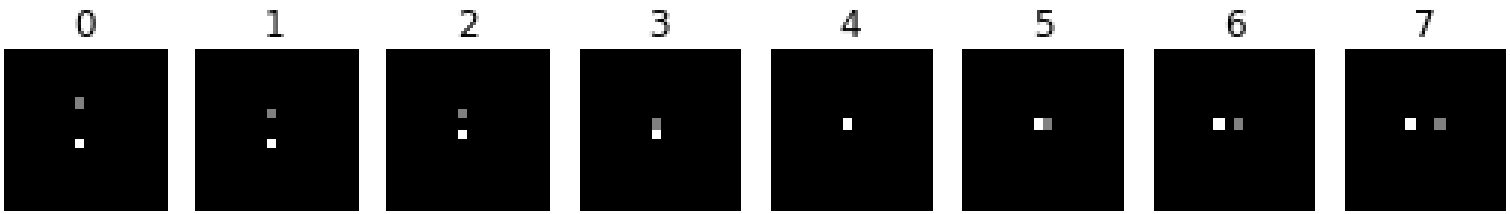
```
1 particles = array([rand(w,h)<0.01 for i in range(4)])
```

```
1 dirs = [(-1,0),(1,0),(-1,1),(1,1)]
```

```
def loop(n,interact=0):
    for t in range(n):
        for i,d in enumerate(dirs):
            if i%2==t%2:
                particles[i] = roll(particles[i],*d)
        if interact:
            swap = OR(AND(AND(particles[0],particles[1]),NOT(OR(
                particles[2],particles[3]))),
                      AND(AND(particles[2],particles[3]),NOT(OR(
                          particles[0],particles[1]))))
            particles[:,:,:] = where(swap,particles[array
                ([2,3,0,1])],particles)
        yield 0.25*sum(particles,axis=0)
```

Let's look at the interaction of two particles. The particles move along the vertical, collide, and change direction. As a result, we have *conservation* of momentum and mass.

```
particles = zeros((4,16,16))
particles[0,10,7] = 1
particles[1,5,7] = 1
for k,image in enumerate(loop(8,1)):
    subplot(1,8,k+1); imshow(8*particles[0]+4*particles[1]+2*
        particles[2]+particles[3]); title(str(k)); axis("off")
```



We can now simulate a large collection of particles.

```
particles = array([rand(w,h)<0.2 for i in range(4)])
particles[:,77:177,11:111] = 0
_=animate(loop(1000,1))
```

Generally, in fluid dynamics, we look at densities and other average quantities.

We can obtain these by smoothing the particle density in our cellular automata simulation.

```
from scipy.ndimage import filters
def density(n,interact=1,hi=0.2,lo=0.0):
    for t,image in enumerate(loop(n,interact)):
        result = filters.gaussian_filter(image,10.0)
        if t%10000==0: print t,amin(result),amax(result)
        yield clip((result-lo)/(hi-lo),0.0,1.0)
```

```
particles = array([rand(w,h)<0.2 for i in range(4)])
particles[:,77:177,11:111] = 0
_=animate(density(5000,1,hi=0.32))
```

```
0 0.0 0.215240833478
```

# More on Lattice Gases

- simulations on 2D square grids are anisotropic
- simulations on 2D hexagonal grids are isotropic at large scales
- simulations on 3D cubic grids are anisotropic
- more complex rules possible for more realistic simulations
- turbulence, other phenomena observable