Considerations on excluded volume, diffusion, drift, reaction...

Franco Bagnoli

April 22, 2013

Let us consider the following microscopic model. We take a regular lattice of size (L_x, L_y). A cell $s(x, y, t)$ located at position x and y at time t can be in a discrete state $(s = 0, 1, 2, .., k)$. For now consider $k = 1$, then $s \in \{0, 1\}$. The dynamics consists in choosing a cell s at random and a neighbor (s'), either vertically with probability $P_{\updownarrow}$, or horizontally with prob$P_{\leftrightarrow} = 1 - P_{\updownarrow}$. If the cells are in a different state $(s \neq s')$ there is a certain probability of a reaction (or exchange). For now we consider only the case of displacement (exchange).

The probability of a vertical exchange is symmetric $\tau(\binom{0}{1}|\binom{1}{0}) = \tau(\binom{1}{0}|\binom{0}{1}) = \tau_{\updownarrow}$ while the horizontal one may be asymmetric $\tau(01|10) = \tau_{\rightarrow}, \tau(10|01) = \tau_{\leftarrow}$ (there is an horizontal field).

Let us consider first the case of a much higher "vertical" probability ($P_{\updownarrow} \simeq 1$), and $\tau_{\updownarrow} = 1$. In this case one can assume that there are no correlations in the vertical sense, and the probability $\omega(x)$ of choosing an occupied cell $(s = 1)$ in column x is simply proportional to the number of occupied cells in the column, $\omega(x) = \sum_y s(x, y)/L_y$ (urn model). The probability of chosing an empty cell is clearly $1 - \omega(x)$.

The evolution equation for $\omega(x, t)$ is

$$\begin{aligned}\omega(x, t+1) = \omega(x, t) &- \omega(x, t)\,(1 - \omega(x+1, t))\,\tau_{\rightarrow} \\ &- \omega(x, t)\,(1 - \omega(x-1, t))\,\tau_{\leftarrow} \\ &+ \omega(x-1, t)\,(1 - \omega(x, t))\,\tau_{\rightarrow} \\ &+ \omega(x+1, t)\,(1 - \omega(x, t))\,\tau_{\leftarrow}.\end{aligned}$$

and developing $\omega(x, t+1) = \omega(x, t) + \partial\omega/\partial t = \omega + \dot{\omega}$ and $\omega(x \pm 1, t) =$

$\omega(x,t) \pm \partial\omega/\partial x + (1/2)\partial^2\omega/\partial x^2 = \omega \pm \omega' + (1/2)\omega''$, we get

$$\dot{\omega} = \frac{1}{2}\left(\tau_{\rightarrow} + \tau_{\leftarrow}\right)\omega'' - \left(\tau_{\rightarrow} - \tau_{\leftarrow}\right)(1-2\omega)\omega', \tag{1}$$

which is a diffusion equation with a nonlinear drift term.

There is a "strange" excluded volume effect for the drift: for $\omega = 1/2$ the drift term disappears independently of the field, while for $1 < \omega < 1/2$ it changes sign. It is simply the "shock wave" effect experienced in highways: once that a patch of high density (and low speed) is formed, it proceeds backward.

In the opposite case $\tau_{\updownarrow} = 0$ the horizontal lines are decoupled and the systems becomes an ensemble of L_y one-dimensional "queues", a model known as *totally asymmetric exclusion process* (ASEP) and sometimes used as a traffic model (exhibiting shock waves).

The case $k = 2$ (two non-interacting chemical species) is even more interesting. Let us study it again in the case $P_{\updownarrow} \simeq 1$, $\tau_{\updownarrow} \simeq 0.5$. Let us indicate with ψ the density of species 1 and with ϕ that of species 2 (same exchange probability). We get

$$\dot{\psi} = \frac{1}{2}\left(\tau_{\rightarrow} + \tau_{\leftarrow}\right)\left(\psi'' + \psi\phi'' - \phi\psi''\right) - \left(\tau_{\rightarrow} - \tau_{\leftarrow}\right)\left(\psi' - 2\psi\psi' - \psi\phi' - \phi\psi'\right),$$

where we can see that the even the single-species diffusion term contains corrections. If we add the equivalent equation for $\dot{\phi}$ and insert $\omega = \phi + \psi$ we get the equation (1).

In the case $\psi = \phi$ (same density for the two species), the diffusive terms becomes "normal" while the drift term changes sign for $\psi = 1/4$ (as it should be since we have seen that in Eq. (1) it changed sign for $\omega = 1/2$ and here $\psi = \omega/2$).

The code for the simulation is located nead the end of the document (it requires gcc and gnuplot; it should work on linux, macosx and cygwin; it does not work on msdos/windows).

Executing it with the following values

```
Lx = 400;
Ly = 100;
rho=0.2;
rhoc=0.8;
```

```
  p1 = 0.9;
  p1c = 0.5;
  pv = 0.9;
  tau1l = tau2l = 0.1;
  tau1r = tau2r = 0.8;
  TMAX = 1000000;
  layer=50;
  tv = 100;
```

one can notice how the peak of the ψ at beginning goes backward, and in any case is blocked in the same position until the density goes below 0.5, while the peak of ϕ moves (and spreds) much more quickly.

(for the reaction part, tomorrow...)

diff.c

```
/* simple diffusion with excluded volume */

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define site(x,y) _site[(y)*(Lx)+(x)]

#define swap(z) (((z)/10)+ ((z)%10)*10)

/*
 * States:
 *
 *  0: empty
 *  1: species A (initial density rhoc*p1c in the central layer,
 *    rho*p1 otherwise)
 *  2: species B (initial density rhoc*(1-p1c) in the central layer,
 *    rho*(1-p1) otherwise)
 *  (rho, rhoc: overall density)
 *
 *  Rules: a cell is chosen at random. With prob pv we take the neighbor
 *   in vertical, otherwise in horizontal
 *  In vertical the exchange probability is one
 *
 *  In horizontal
```

```
 *  1-0 -> 0-1 with prob tau1r
 *  0-1 -> 1-0 with prob tau1l
 *  2-0 -> 0-2 with prob tau2r
 *  0-2 -> 2-0 with prob tau2l
 *
*/

int main() {
  int Lx, Ly;
  int * _site;
  int x, y, x1, y1, z;
  double pv,tau1l,tau1r,tau2r,tau2l;
  double rho, rhoc, p1, p1c;
  int t, TMAX;
  int i, n;
  int layer;
  int **histo, tot[4];
  FILE *gp, *gp1;
  int tv;
  double pp, pp1;

  Lx = 400;
  Ly = 100;
  rho=0.2;
  rhoc=0.8;
  p1 = 0.9;
  p1c = 0.5;
  pv = 0.9;
  tau1l = tau2l = 0.1;
  tau1r = tau2r = 0.8;
  TMAX = 1000000;
  layer=50;
  tv = 10;

  _site = (int *) calloc(Lx*Ly, sizeof(int));
  histo = (int **) calloc(3, sizeof(int*));
  for (i=0; i<3; i++) {
               histo[i] = (int *) calloc(Lx, sizeof(int));
  }
```

```
srand48(time(NULL));

gp = popen("gnuplot", "w");
fprintf(gp, "set xrange [0:%d]\n",Lx);
fprintf(gp, "set yrange [0:%d]\n",Ly);
fprintf(gp, "set cbrange [0:4]\n");
gp1 = popen("gnuplot", "w");
fprintf(gp1, "set xrange [0:%d]\n",Lx);
fprintf(gp1, "set yrange [0:1]\n");

for (x=0; x<Lx; x++) {
  pp = (abs(2*x-Lx) < layer ? rhoc : rho);
  pp1 = (abs(2*x-Lx) < layer ? p1c : p1);
  for (y=0; y<Ly; y++) {
    if (drand48() < pp) {
      site(x,y) = (drand48() < pp1 ? 1 : 2);
    }
  }
}

for (t=0; t<TMAX; t++) {
  for (n=0; n<Lx*Ly; n++) {
    x = (int) (drand48() * Lx);
    y = (int) (drand48() * Ly);

    if (drand48() < pv) {
      x1 = x;
      y1 = (y + 1) % Ly;
      // always exchange in vertical...
      z = site(x,y);
      site(x,y) = site(x1,y1);
      site (x1,y1) = z;
    } else {
      x1 = (x+1)%Lx;
      y1 = y;
      z = site(x,y)*10+site(x1,y1);
      switch(z) {
        case 01:
          pp = tau1l;
          break;
```

```
      case 10:
        pp = tau1r;
        break;
      case 02:
        pp = tau2l;
        break;
      case 20:
        pp = tau2r;
        break;
      default:
        break;
    }
    if (drand48() < pp) z=swap(z);
    site(x,y)=z/10;
    site(x1,y1)=z%10;
  }
}
if (t%tv==0) {
  fprintf(gp,"set title 't=%d'\n",t);
  fprintf(gp,"plot '-' binary array=%dx%d \
    format='%%int' w image\n",Lx, Ly);
  fwrite(_site,sizeof(int), Lx*Ly, gp);
  fflush(gp);
  for (i=0; i<3; i++) {
                              tot[i]=0;
                    }

  for (x=0; x<Lx; x++) {
    for (y=0; y<Ly; y++) {
      if(site(x,y)>0) {
       histo[site(x,y)][x]++;
       histo[0][x]++;
       tot[site(x,y)]++;
       tot[0]++;
      }
    }
  }
  fprintf(gp1,"set title 't=%d'\n",t);
  fprintf(gp1,"plot '-' t 'omega' w l,  \
    '-' t 'psi' w l, '-' t 'phi' w l\n");
```

```
      for (i=0; i<3; i++) {
        for (x=0; x<Lx; x++) {
          fprintf(gp1, "%d %f\n", x, (double)histo[i][x]/Ly);
          histo[i][x]=0;
        }
                                  fprintf(gp1, "end\n");
                        }
      fflush(gp);
    }
  }
  printf("premi un tasto per uscire\n");
  getchar();
  return(0);
}
```
