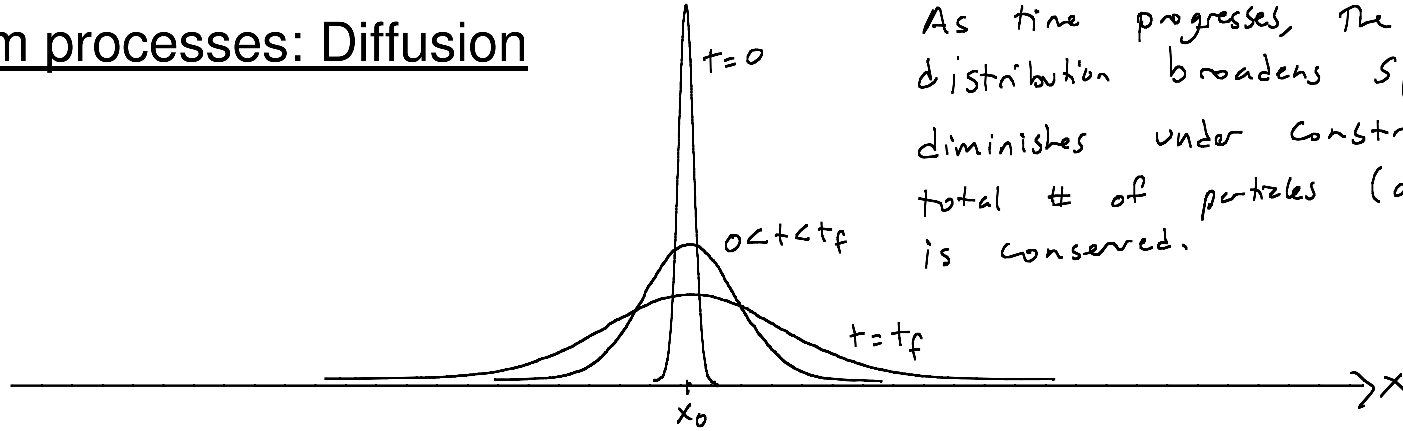
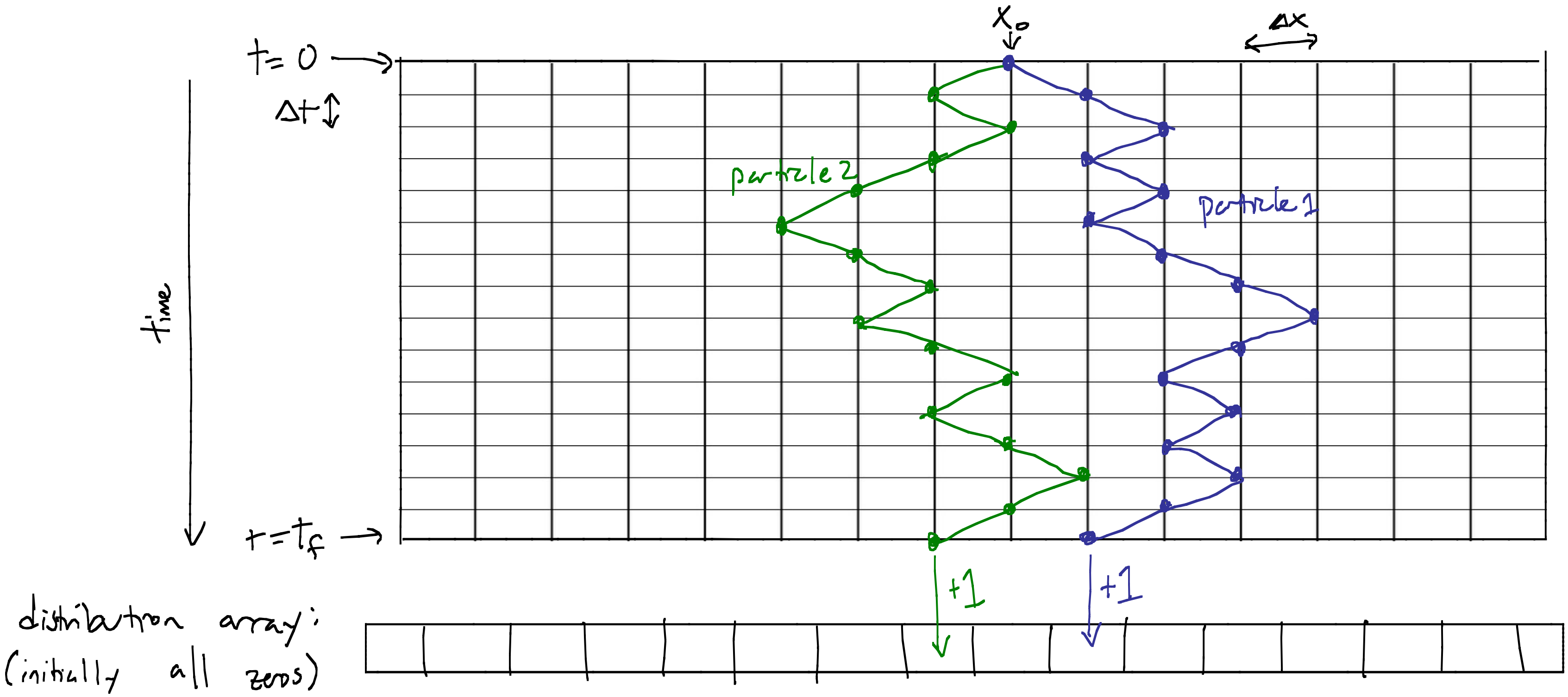
Modeling random processes: Diffusion

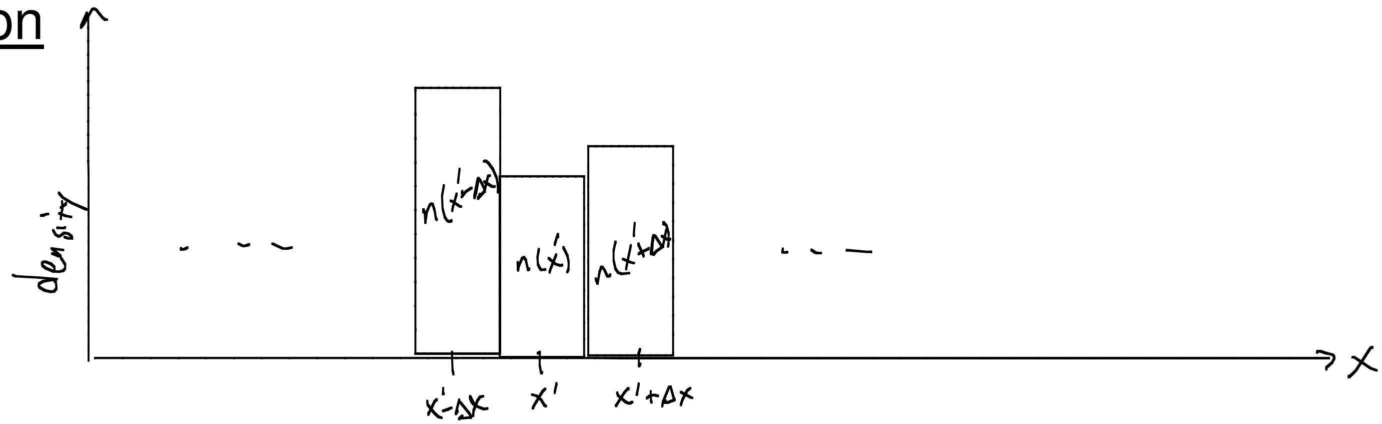


As time progresses, the initially narrow distribution broadens spatially and diminishes under constraint that total # of particles (area under curve) is conserved.

We can simulate this process by following the "random walk" of a large number of particles that start @ $X=X_0$ and move left or right at every timestep until $t=t_f$.



Analytic solution



Rate equation: During every element of time Δt, half the particles within range Δx will go left and half go right. The number/density rate of change at $x=x'$ is

$$\frac{\partial n(x',t)}{\partial t} = \frac{1}{2}\left(\frac{n(x'-\Delta x,t) - n(x',t)}{\Delta t}\right) + \frac{1}{2}\left(\frac{n(x'+\Delta x,t) - n(x',t)}{\Delta t}\right)$$

"diffusion coef" or "diffusivity"

$$= \left(\frac{\Delta x^2}{2\Delta t}\right) \frac{\frac{n(x'+\Delta x,t) - n(x',t)}{\Delta x} - \frac{n(x',t) - n(x'-\Delta x,t)}{\Delta x}}{\Delta x} \xrightarrow{\lim_{\Delta x \to 0}} \left(\frac{\Delta x^2}{2\Delta t}\right) \frac{\frac{\partial n(x'+\Delta x,t)}{\partial x} - \frac{\partial n(x',t)}{\partial x}}{\Delta x} = D\,\frac{\partial^2 n(x',t)}{\partial x^2}$$

So we have: $\boxed{\frac{\partial n}{\partial t} = D\,\frac{\partial^2 n}{\partial x^2}}$

Solution to this PDE for initial conditions w/ distribution $n(x, t=0)$ concentrated @ $x = x_0$ given by "gaussian" $n(x,t) = \frac{1}{\sqrt{4Dt\pi}}\, e^{-\frac{(x-x_0)^2}{4Dt}}$

Matlab implementation

```
function distrib=diffusion(Npart,tf)
% Monte-Carlo simulation of 1-dimensional diffusion due to random motion.
% Npart : number of particles
% tf : final time to evaluate distribution
% distrib: output spatial distribution
% Ian Appelbaum, Sept 24, 2014

%define increments
dx=1;
dt=1;

Nx=1e3; % number of points along x-axis

distrib=zeros(1,Nx); % initialize distribution

x0=dx*round(Nx/2); % choose initial position of particle

for n=1:Npart % loop through all particles
  x=x0; % initialize particle position
  for t=0:dt:tf   % increment thru time

    %choose right or left motion randomly
    if round(rand)
      x=x+dx; %go right
    else
      x=x-dx; %go left
    end

  end %end loop thru time

  %increment element of distrib corresponding to final particle position
  distrib(x/dx)=distrib(x/dx)+1;

end % end loop thru particles

plot(distrib,'o') %plot the resulting distribution

%compare to analytic solution of partial-differential rate equation
hold on;
D=dx^2/(2*dt); fplot(@(x) 2*Npart./sqrt(4*pi*D*tf)*exp(-((x-x0).^2)./(4*D*tf)),[0, 2*x0],'r')
xlabel('distance'); ylabel('frequency'); legend('Monte-Carlo','analytic'); hold off

end %end function
```

any nonzero value is logical "true"

"nested for loop"

execute when round(rand)=0

Invoke at command prompt:

```
>> diffusion(1e4,1e2);
>> hold on; diffusion(1e4,1e3);
>> hold on; diffusion(1e4,1e4);
```