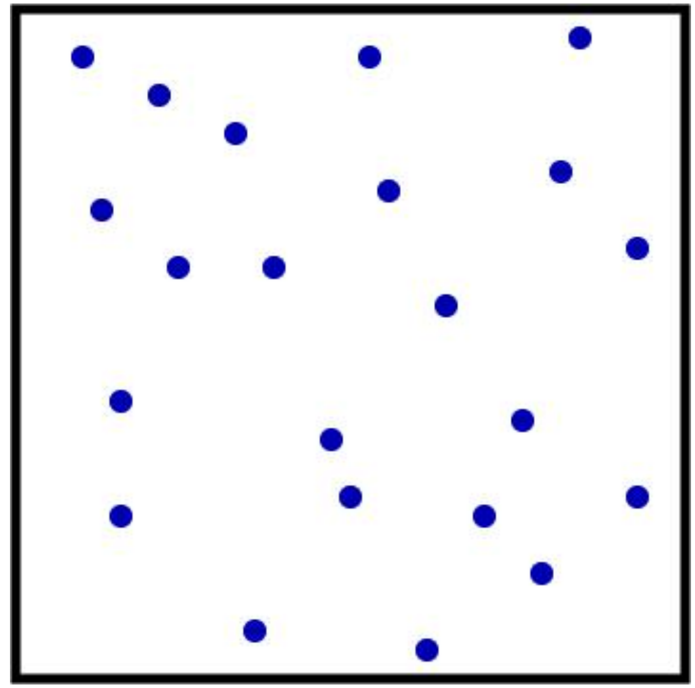
Monte Carlo for particle systems

Interacting molecules in a volume

$$\langle A \rangle = \frac{1}{Z} \int \prod_{i=1}^{N} dx_i^d A(\{x_i\}) \mathrm{e}^{-E(\{x_i\})/k_B T}$$

$$Z = \int \prod_{i=1}^{N} dx_i^d \mathrm{e}^{-E(\{x_i\})/k_B T}$$



We will consider simple spherical particles (atoms)

$$E(\{x_i\}) = \sum_{i=1}^{N} U(\vec{x}_i) + \sum_{i \neq j} V(\vec{x}_i, \vec{x}_j)$$

V=V(r); spherically symmetric. More complicated molecules involve more difficult energy calculations

Long-range interactions (e.g. Coulomb) lead to N^2 scaling of the computational effort (CPU time).

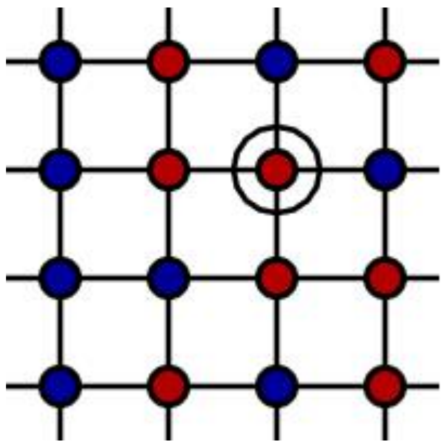
Lennard-Jones (e.g., for noble gases):
- can be truncated

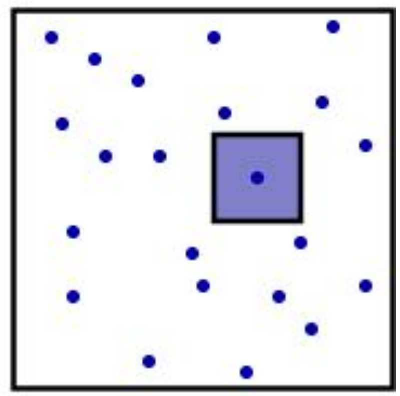
$$V(r) = \frac{a}{r^{12}} - \frac{b}{r^6}$$

Metropolis Monte Carlo scheme

Principle same as in Ising simulation; update involves

- selecting a particle at random
- attempt to move it within a box
- calculate energy change (more complicated than Ising)





Adjust the window so that the acceptance rate is close to 50%

Typically periodic boundary conditions are used

```
x1=x0+delta*rand(-0.5:0.5)
if x1 < 0; x1=x1+ll; end
if y1 > l; x1=x1-ll; end
y1=y0+delta*rand(-0.5:0.5)
if y1 < 0; y1=y1+ll; end
if y1 > l; y1=y1-ll; end
```

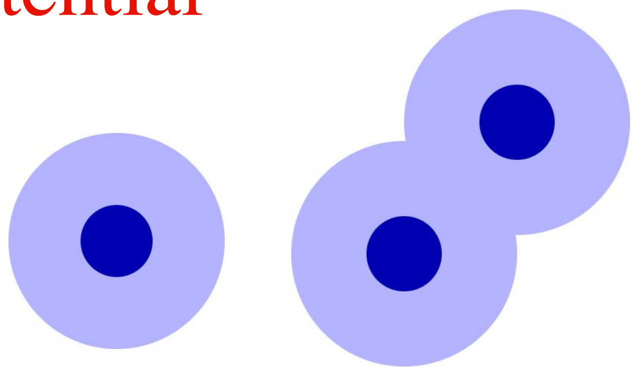
Adjust the window size during equilibration
- e.g., 20 times in total

```
function adjustedelta(steps,accepted,delta,ll)
   arate=accepted/steps
   if arate < 0.4; delta=delta/1.5; end
   if arate > 0.6 && delta < ll/4; delta=delta*1.5; end
   return delta
end
```

Example: ”atoms” with a simplified potential

N particles in periodic 3D box of length L

$$V(r) = \begin{cases} \infty, & r <\leq r_1 \quad \text{“collision”} \\ -V, & r_1 < r \leq r_2 \\ 0, r > r_2 & \end{cases}$$

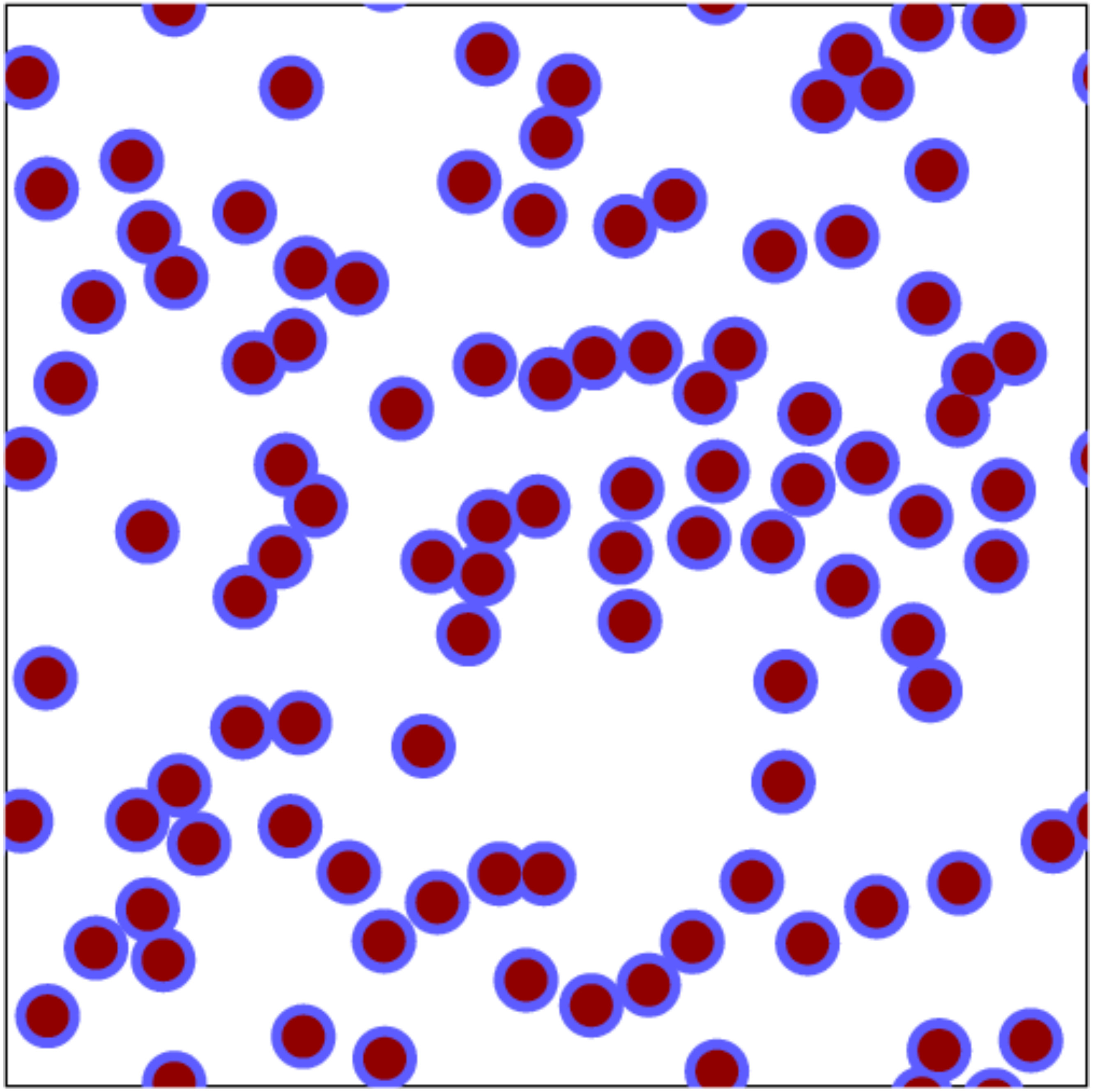


Energy = -V times number of “overlapping” particle pairs
-counting overlaps is expensive, ~N
-some tricks with list of “neighbors” to reduce time

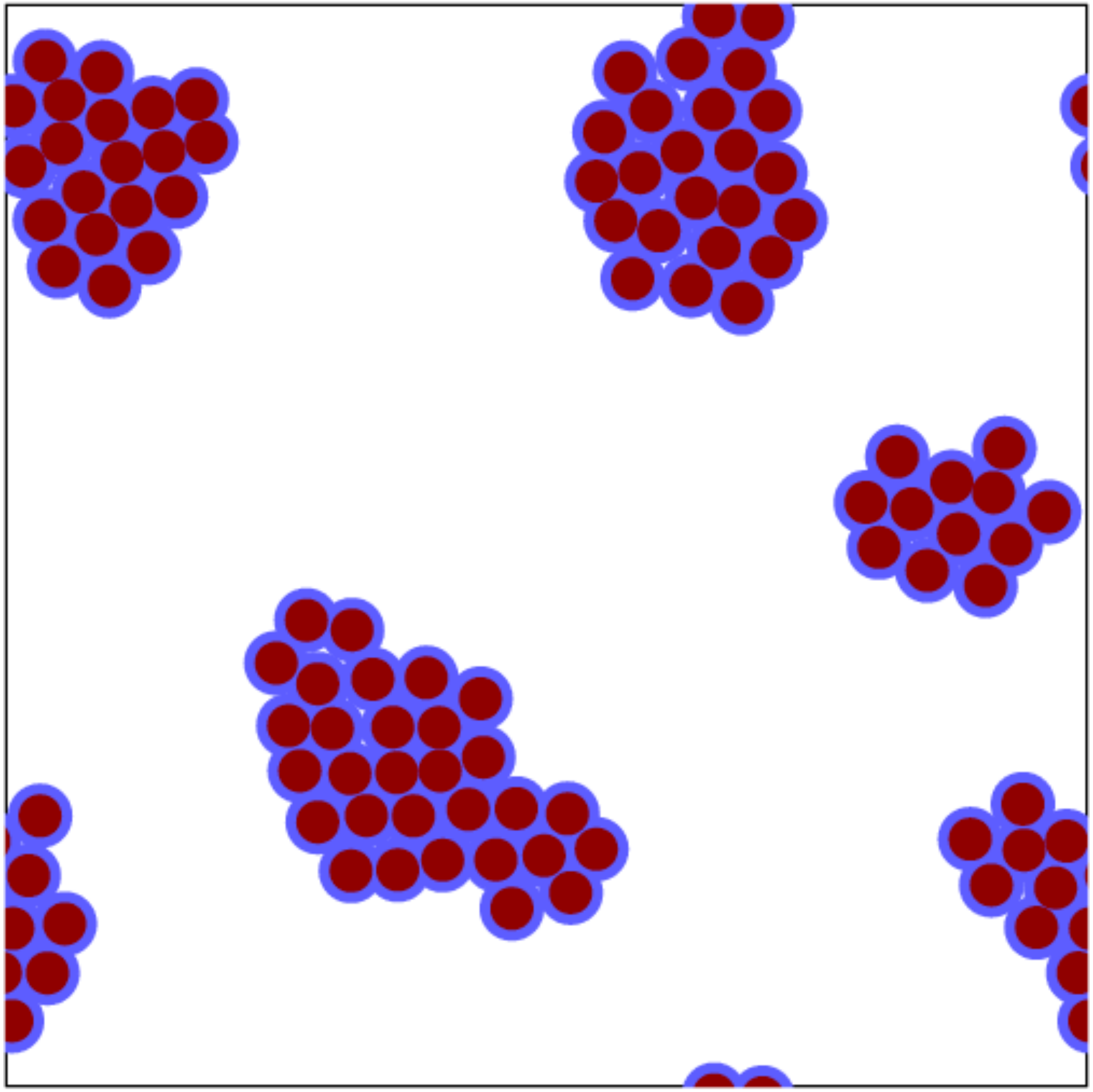
MC step

```
for j=1:n
   x0=xyz[1,j]; y0=xyz[2,j]; z0=xyz[3,j]
   x1,y1,z1=newcoordinates(ll,delta,x0,y0,z0)
   violate=false # will be true for “collison”
   n1=0   # number of overlapping pairs
   for i=1:n # count overlaps, terminate loop if violation
      if i==j continue
      dx=abs(x1-xyz[1,i]); dx=min(dx,ll-dx)
      dy=abs(y1-xyz[2,i]); dy=min(dy,ll-dy)
      dz=abs(z1-xyz[3,i]); dz=min(dz,ll-dz)
      r2=dx^2+dy^2+dz^2
      if r2 < r2_1
         violate=true   # violation found
         break          #  not allowed, terminate loop
      elseif r2 < r2_2
         n1=n1+1
      end
   end
   if violate; continue; end # terminate update if violation
   if rand() < exp(v0*(n1-n0[j])/temp);;;;
      xyz[1,j]=x1; xyz[2,j]=y1; xyz[3,j]=z1
      n0[j]=n1 # save new number of overlaps for particle j
      accepted=accepted+1
   end
end
```

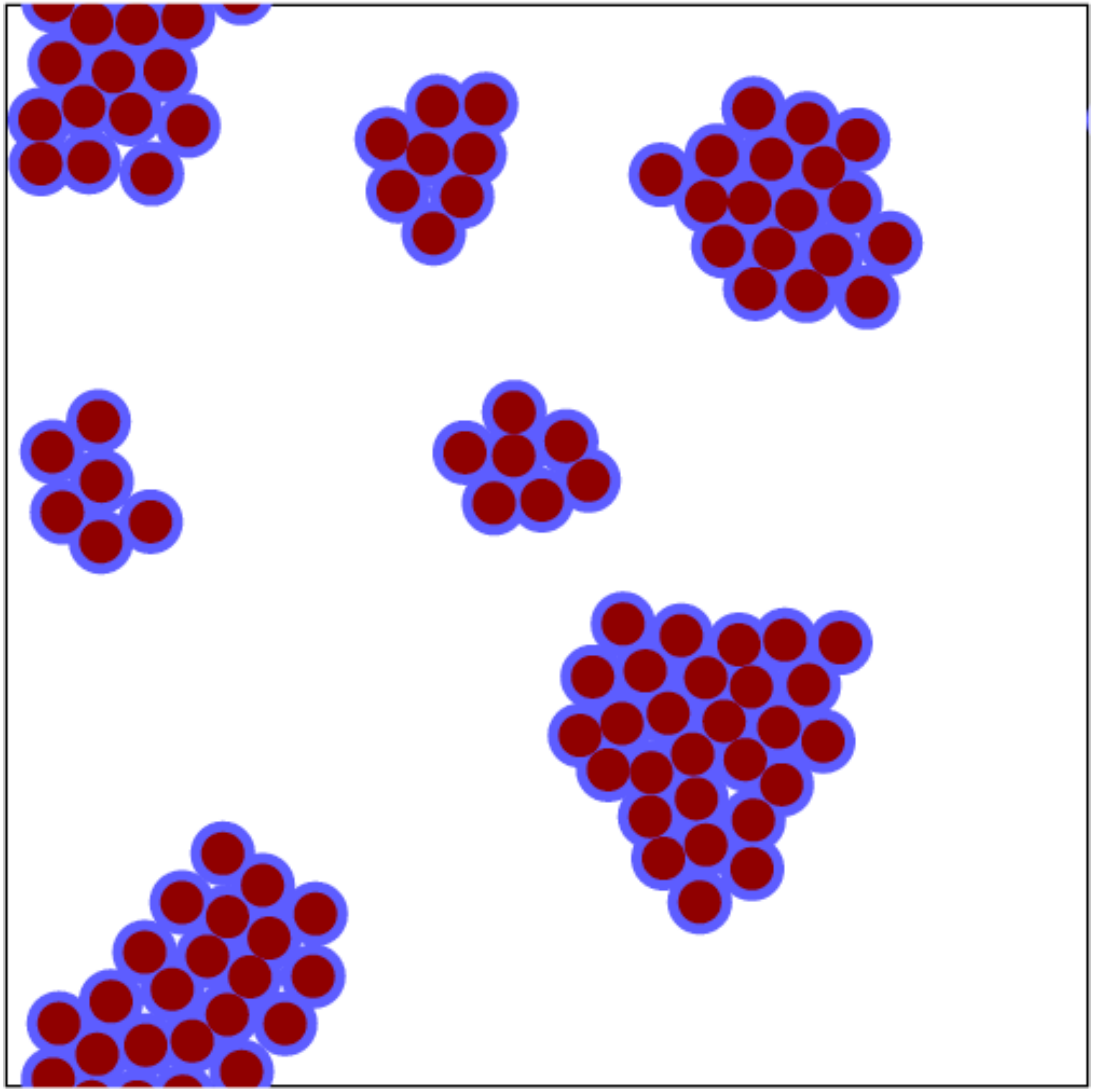
T = 1.00 1



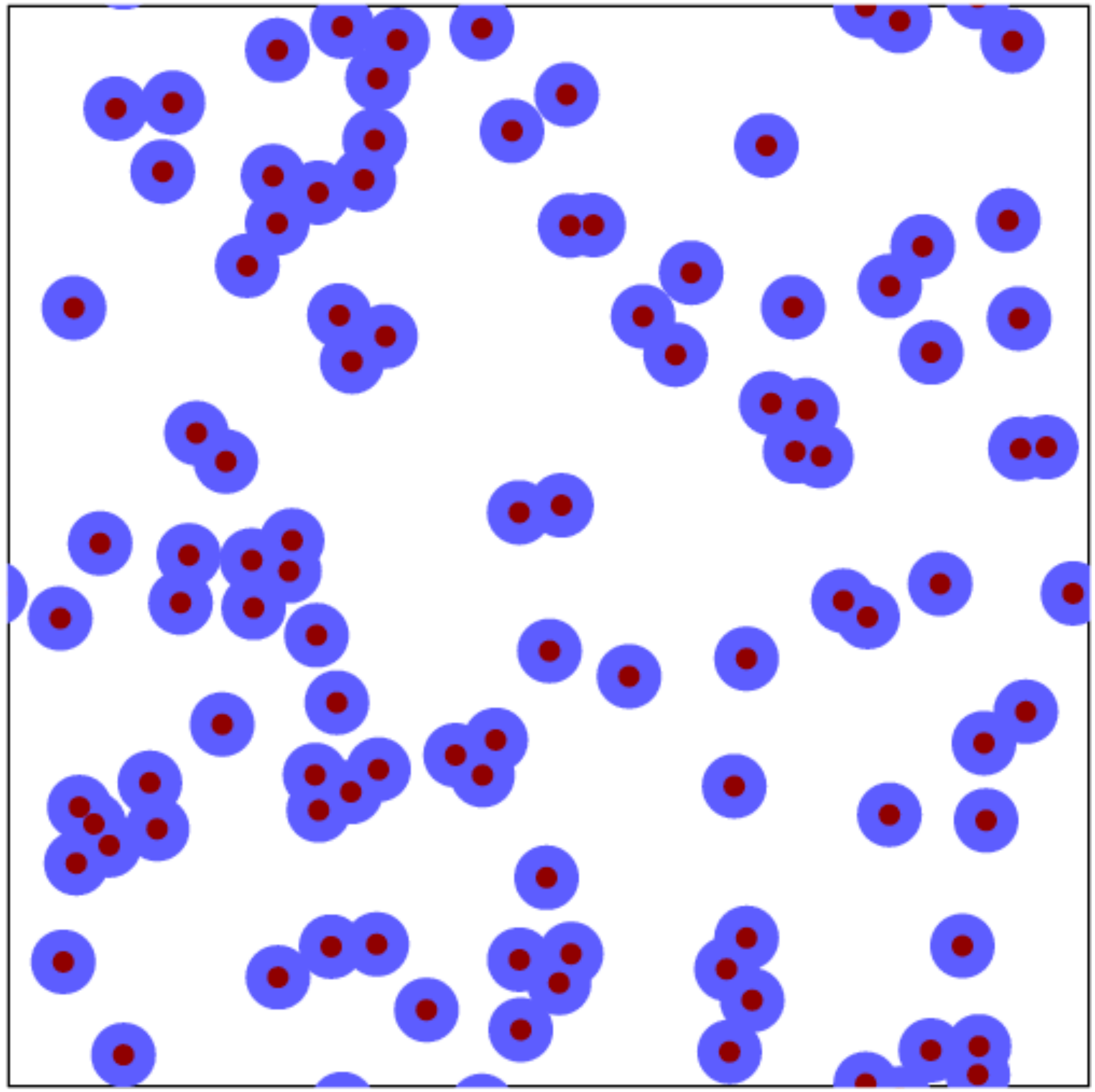
T = 0.20 10



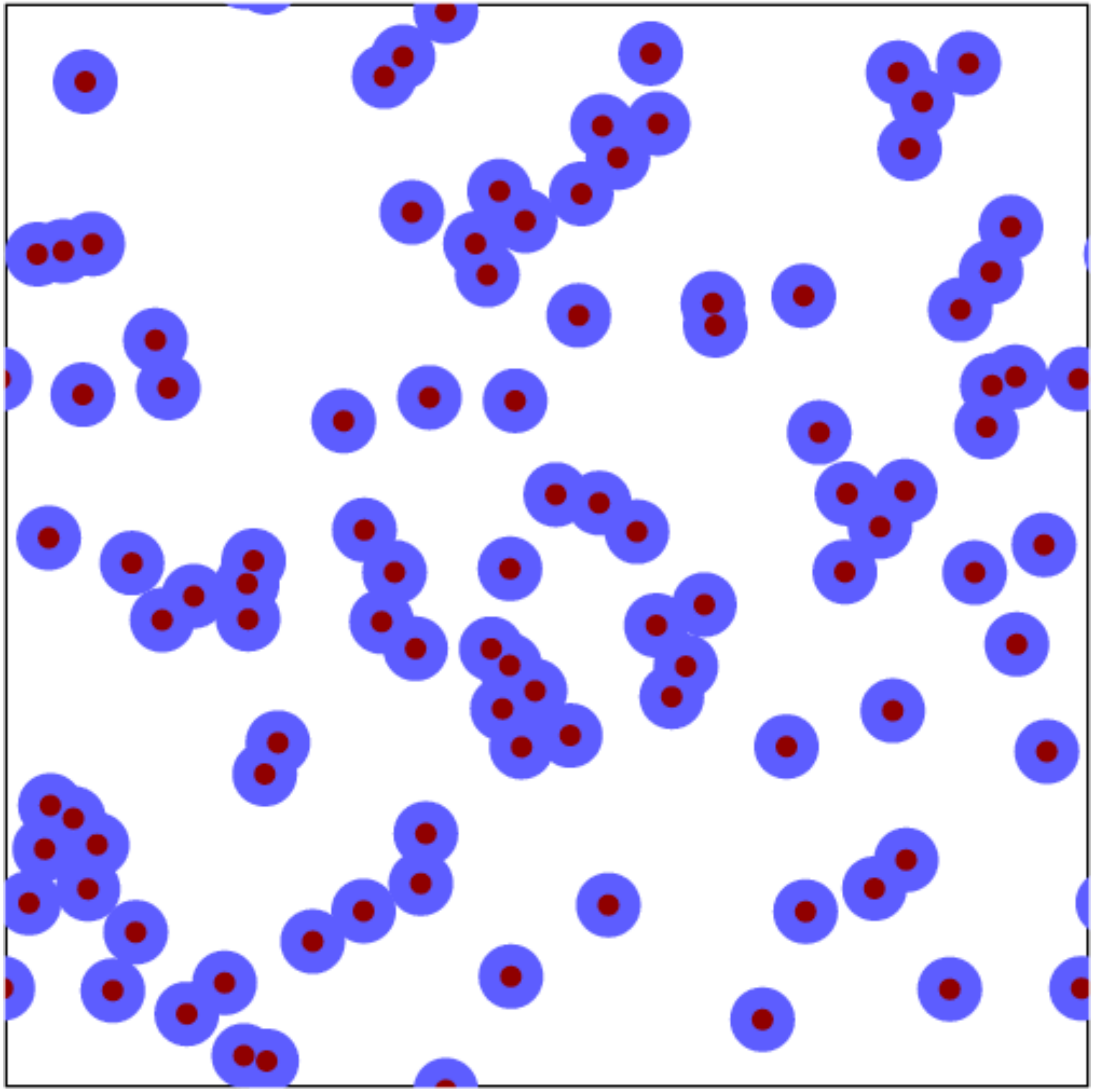
T = 0.20 1000



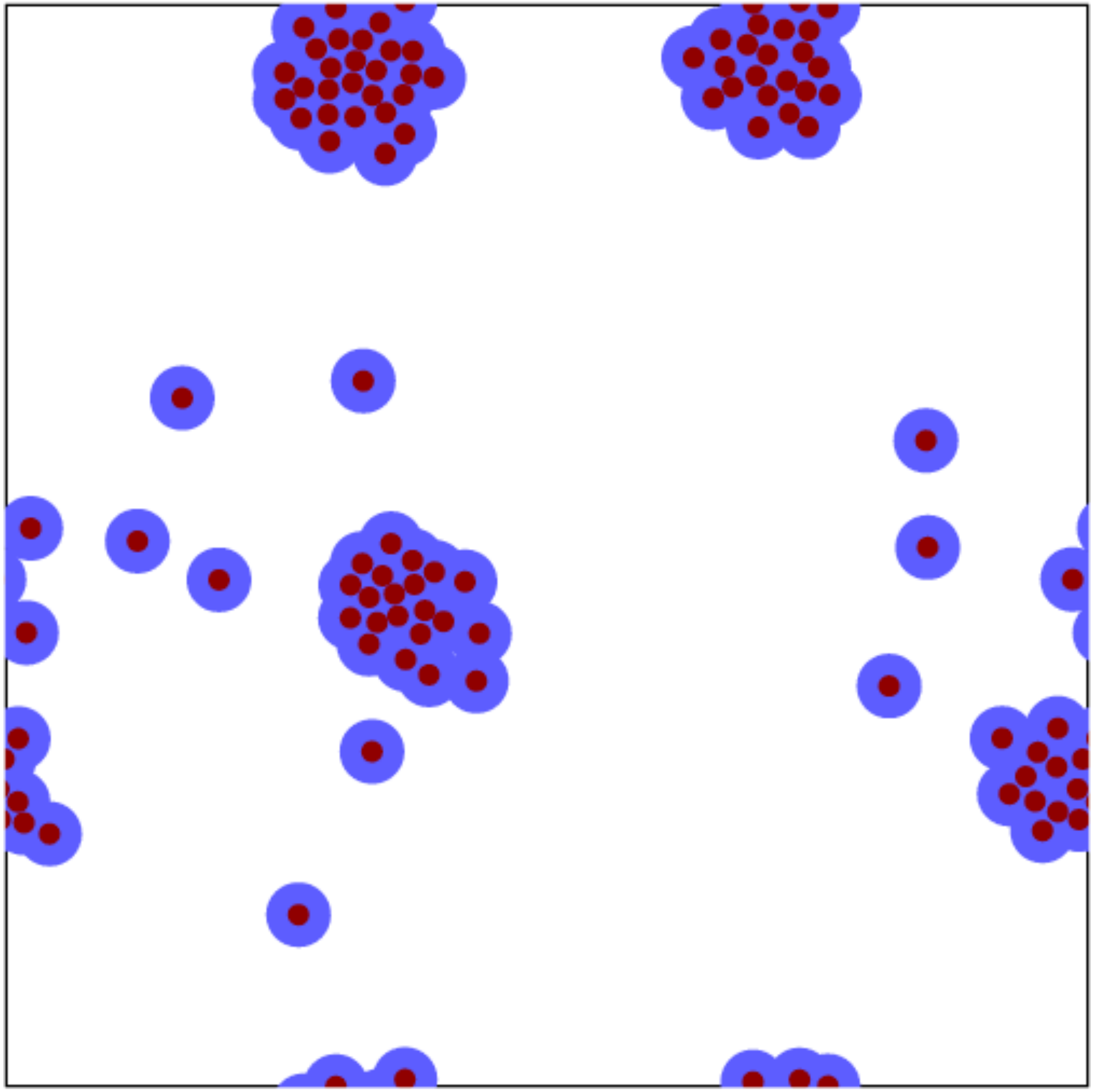
T = 5.00	5



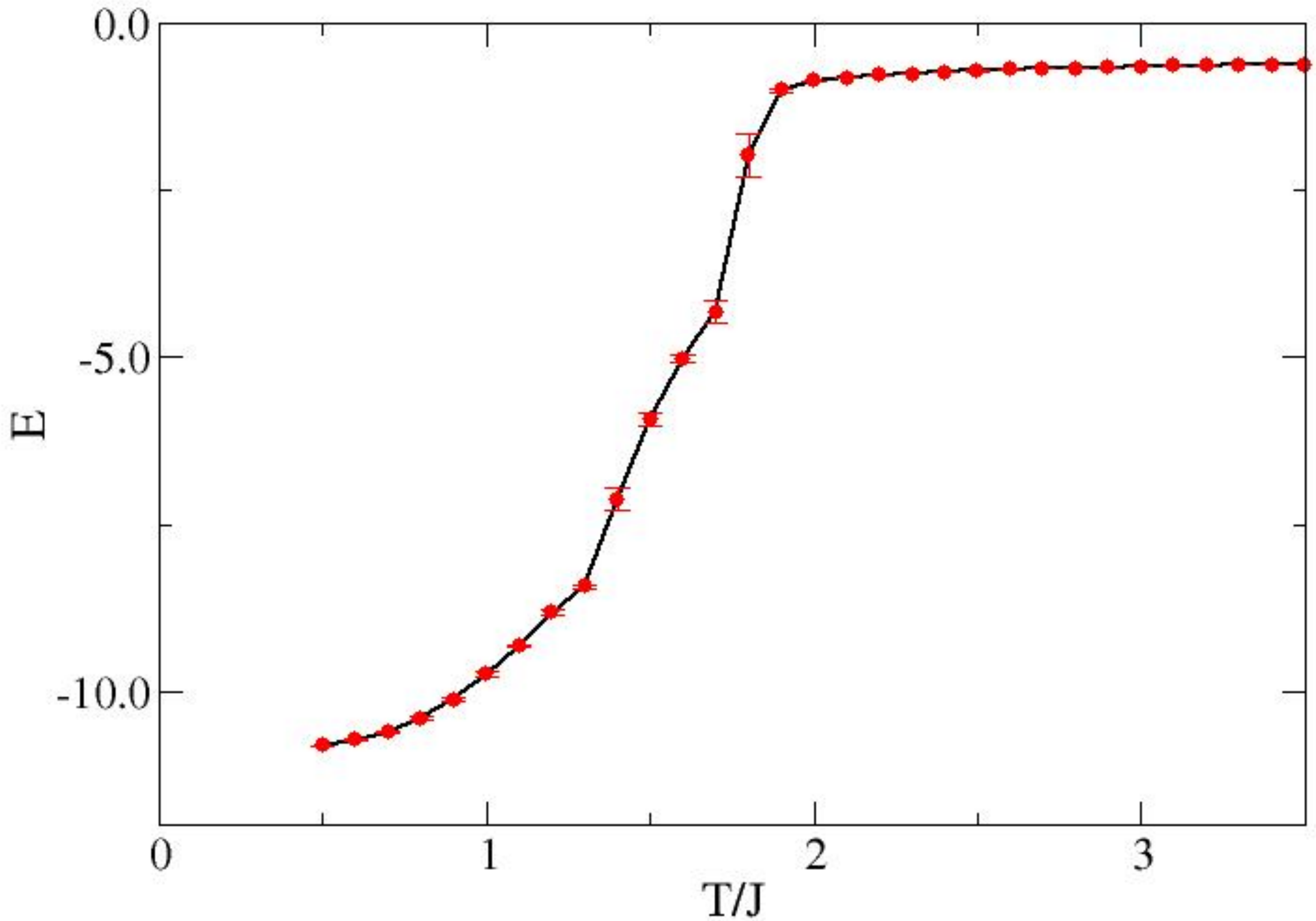
T = 1.00 5



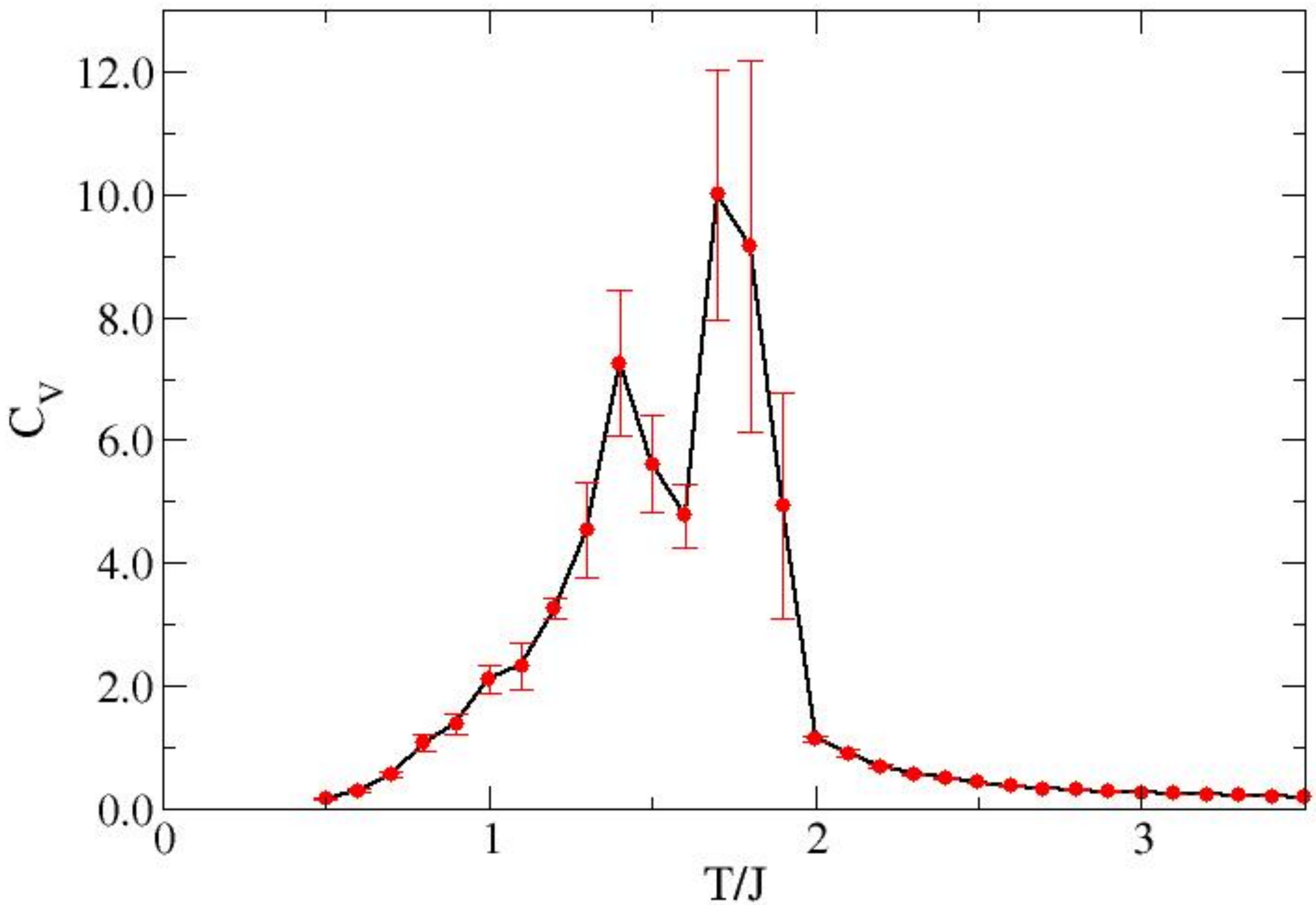
T = 1.00 1000



Internal energy



Specific heat



Simulated annealing

General, very useful optimization method

In optimization, some function of a number of variables is to be minimized or maximized.

In many systems one can think of this function (or its negative value) as an energy of a many-body system.

Introduce a fictitious temperature

Minimize E in a Monte Carlo simulation with slowly decreasing temperature.

May not give the absolutely best solution within finite time, but typically a very good solution is obtained.

T = 0.99 1000