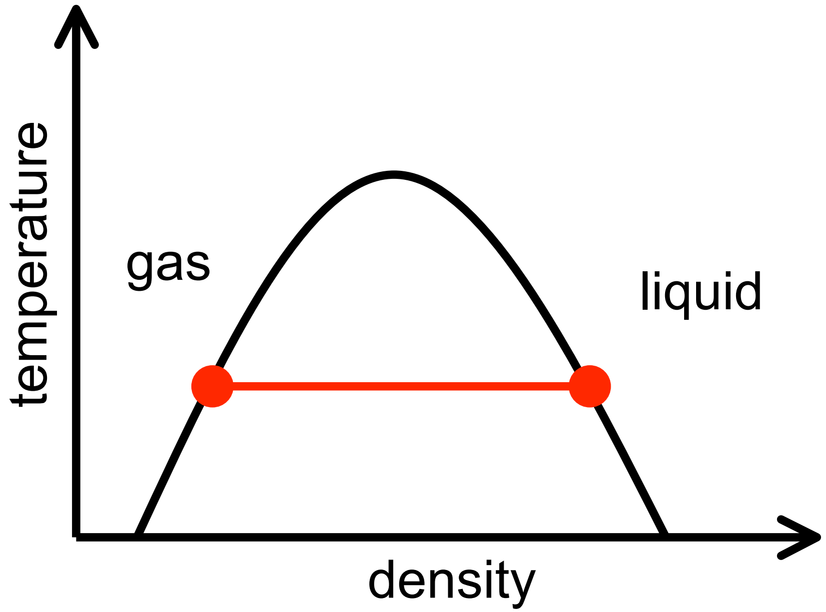
Ensembles II:

1. The Gibbs Ensemble (Chap. 8)

Bianca M. Mladek
University of Vienna, Austria
bianca.mladek@univie.ac.at

Our aim:

We want to determine the phase diagram of a given system.



For this, we need to know the coexistence densities at given temperature.

In experiments:

first order phase transition is easy to locate:

at right density and temperature ⇨

phase separation (two distinct phases devided by interface)

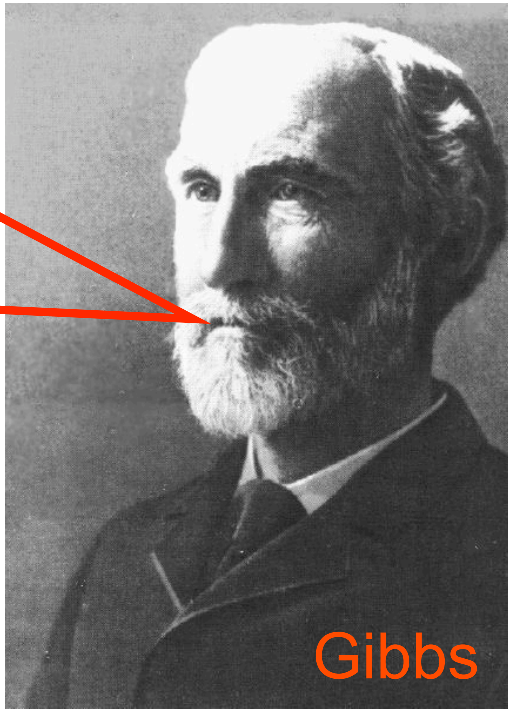
In simulations: ???

Find points where:

$$T_1 = T_2$$

$$P_1 = P_2$$

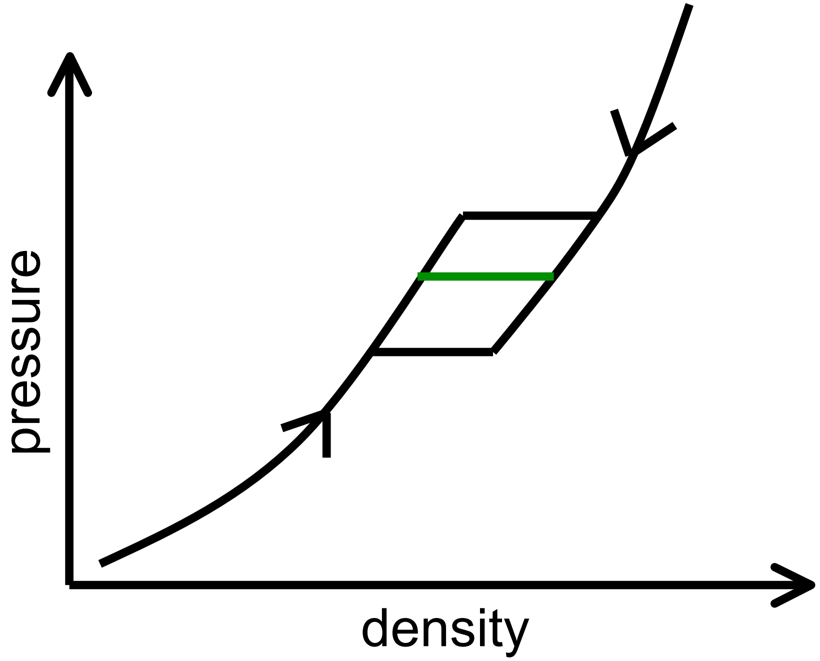
$$\mu_1 = \mu_2$$



…which is the condition for phase coexistence in a one-component system.

Problem in simulations:

Hysteresis:



NVT Ensemble

fluid

fluid

Let‘s lower the temperature…

NVT Ensemble

Problem:

The systems we study are usually small ⇨
large fraction of all particles resides in/near interface.

Possible solution #1

larger systems:

particles	% of part. in interface
1 000	49%
64 000	14%
1 million	6%

⇨ we need huge systems ⇨ computationally expensive

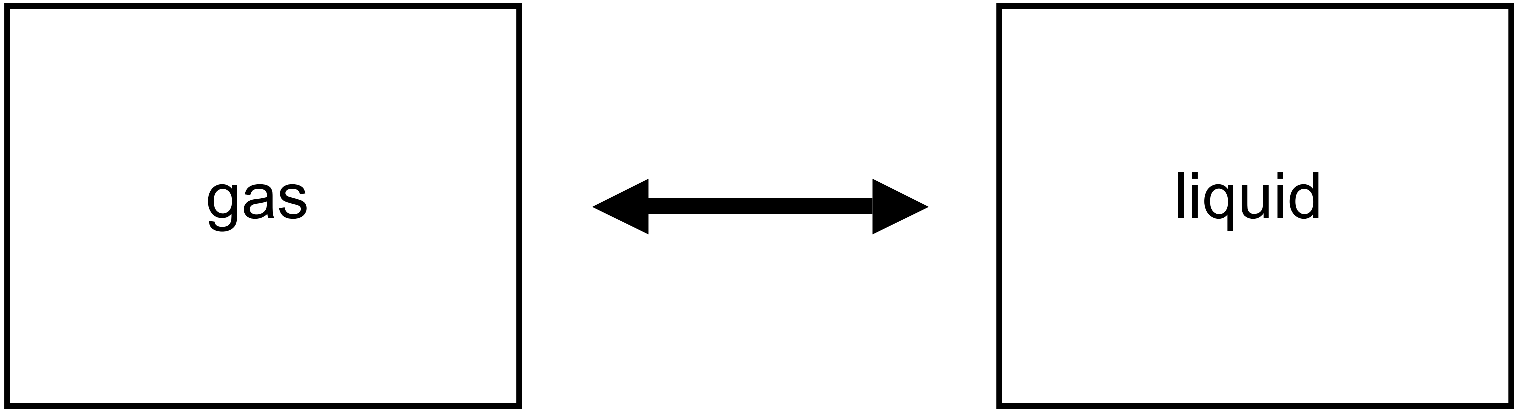
Possible solution #2: “ μPT ”-Ensemble

Problem: no such ensemble exists!

- μ, P and T are intensive parameters
- extensive ones unbounded

We have to fix at least one extensive variable (such as N or V)

Possible solution #3:
The Gibbs ensemble

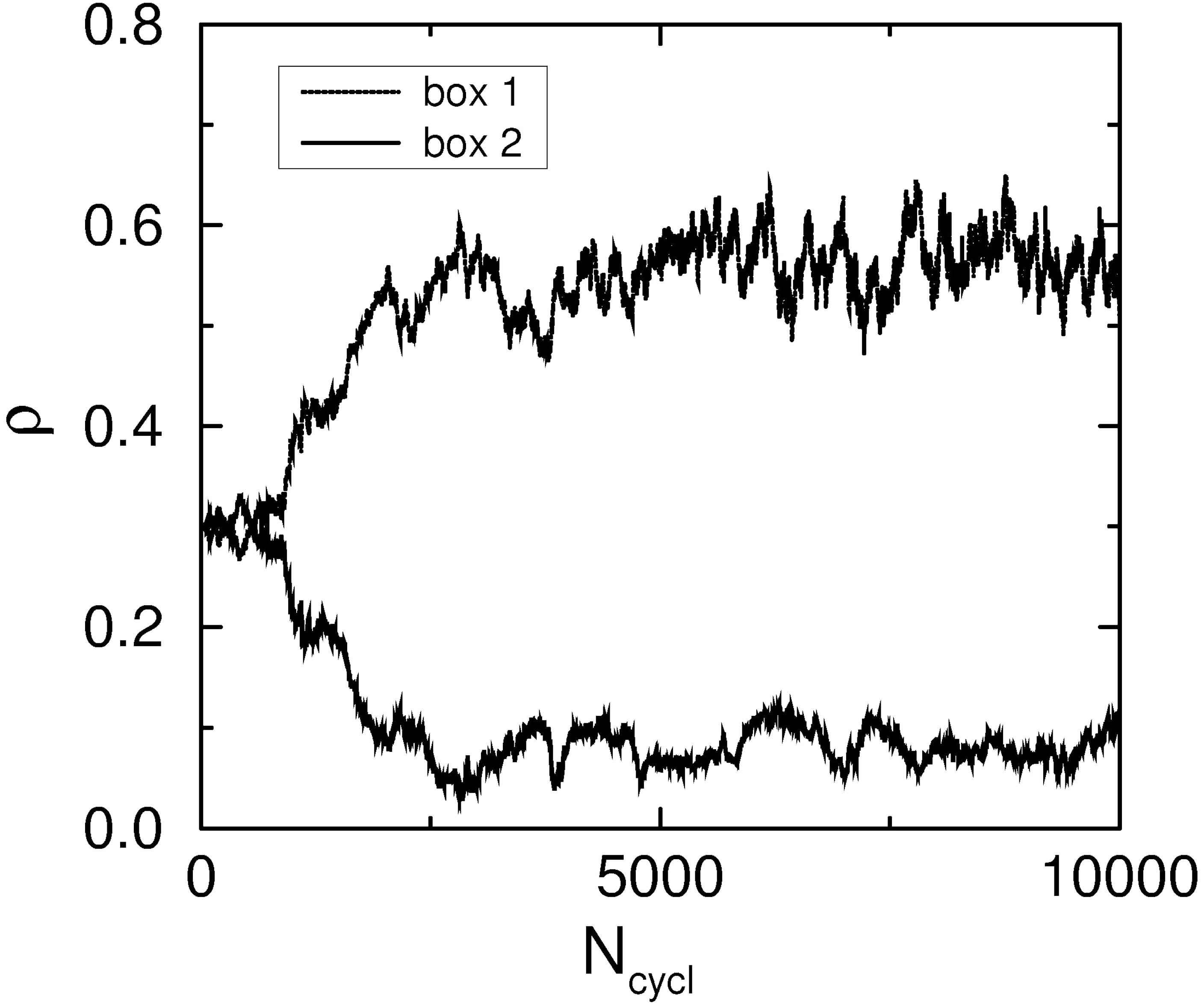


gas

liquid

achieve equilibrium
by coupling them

A. Z. Panagiotopoulos, 1987.

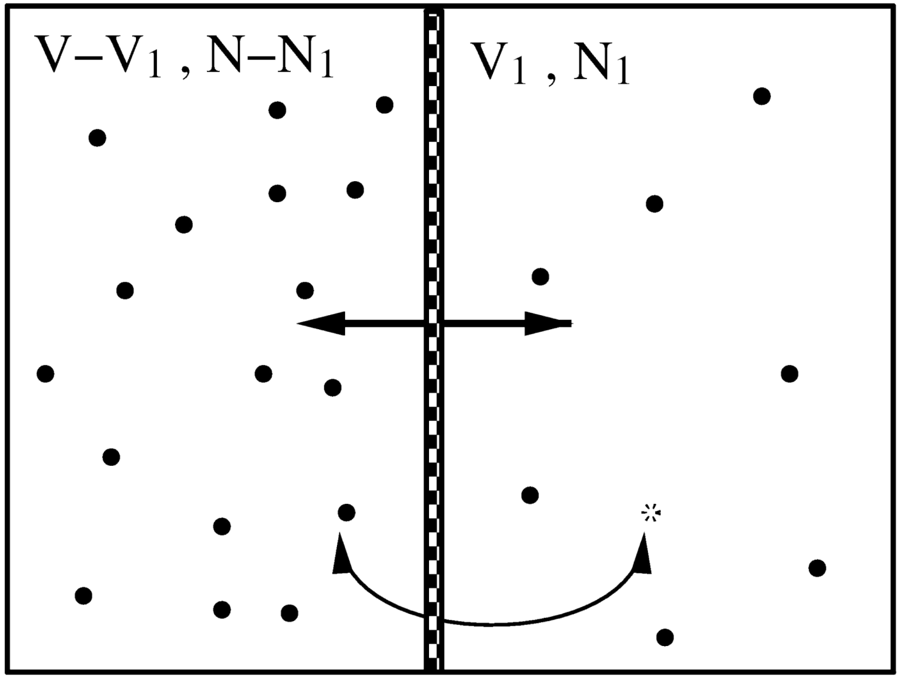

0.8
0.6
0.4
0.2
0.0
ρ
0
5000
10000
N_{cycl}
box 1
box 2

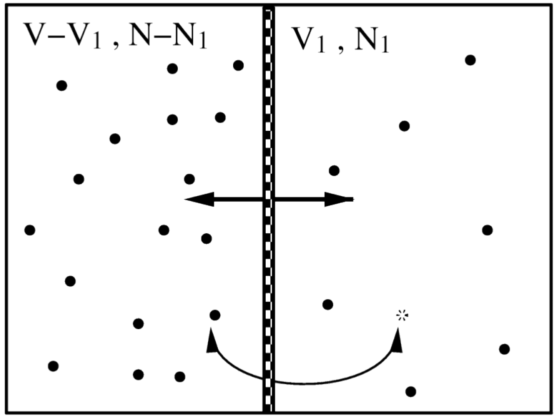
Overall system: NVT ensemble

$$N = N_1 + N_2$$

$$V = V_1 + V_2$$

$$T_1 = T_2$$



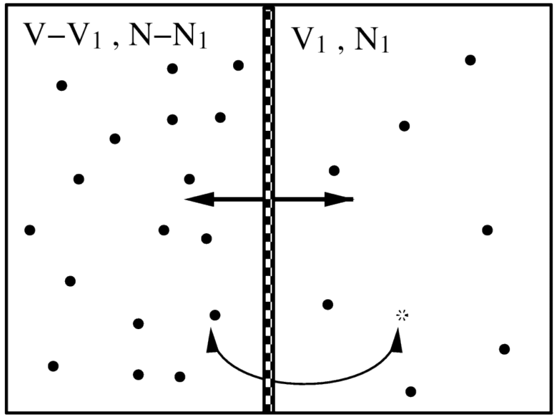


- distribute N_1 particles
- change the volume V_1
- displace the particles

partition function:

$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 V_1^{N_1} (V-V_1)^{N-N_1}$$

$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

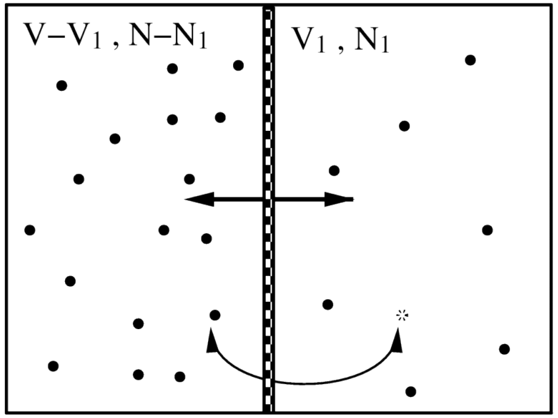


partition function:

$$Q_G(N,V,T) \;=\; \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 V_1^{N_1}(V-V_1)^{N-N_1}$$
$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

Distribute N_1 particles over two volumes:

$$\left(\begin{array}{c} N \\ N_1 \end{array}\right) = \frac{N!}{N_1!(N-N_1)!}$$

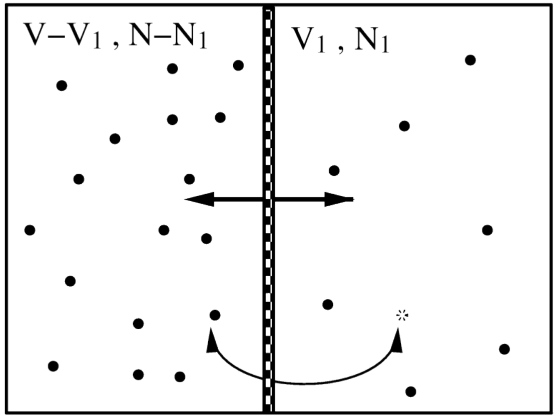


partition function:

$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 V_1^{N_1}(V-V_1)^{N-N_1}$$

$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

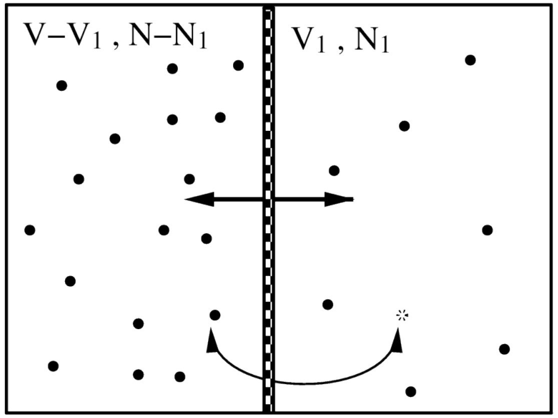
Integrate volume V_1



partition function:

$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 V_1^{N_1} (V-V_1)^{N-N_1}$$
$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

Displace the particles in box1 and box2



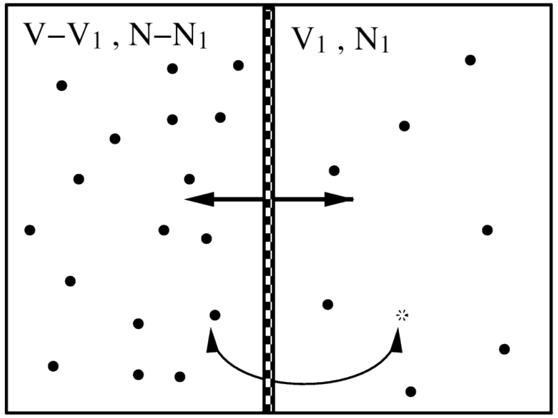
partition function:

$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 \boxed{V_1^{N_1}(V-V_1)^{N-N_1}}$$

$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

Displace the particles in box 1 and box2

scaled coordinates in [0,1)



partition function:

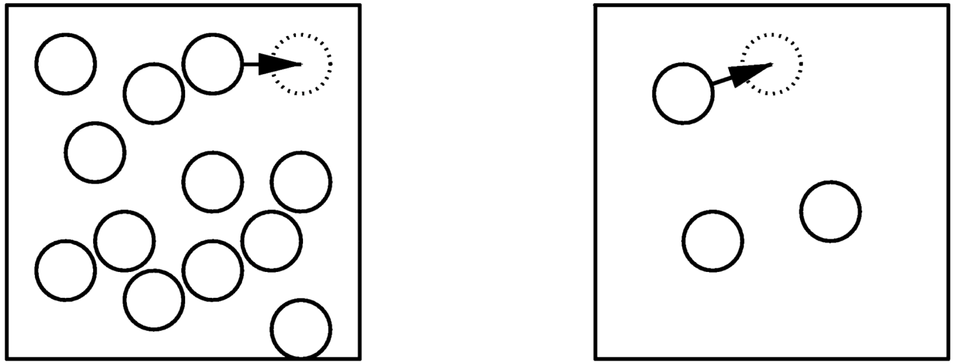
$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \frac{1}{V\Lambda^{3N} N_1!(N-N_1!)} \int_0^V \mathrm{d}V_1 V_1^{N_1}(V-V_1)^{N-N_1}$$

$$\int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]$$

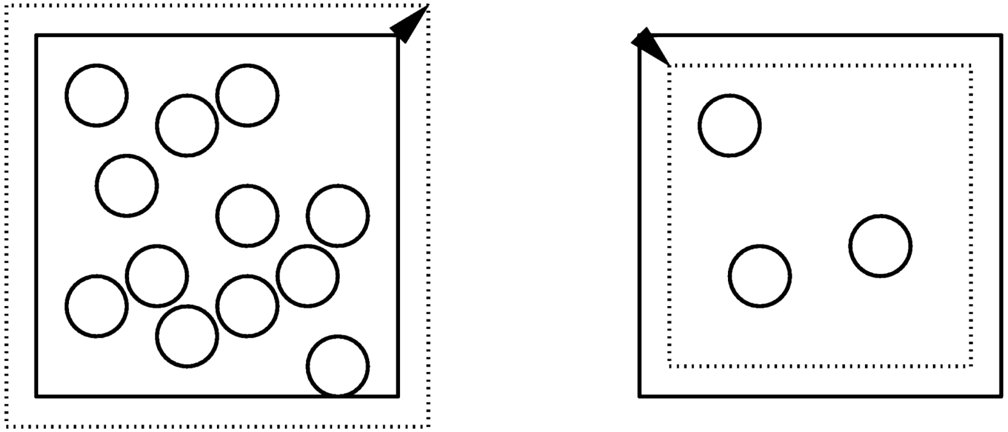
probability distribution:

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(s_1^{N_1}) + U(s_2^{N-N_1})\right]\right\}$$

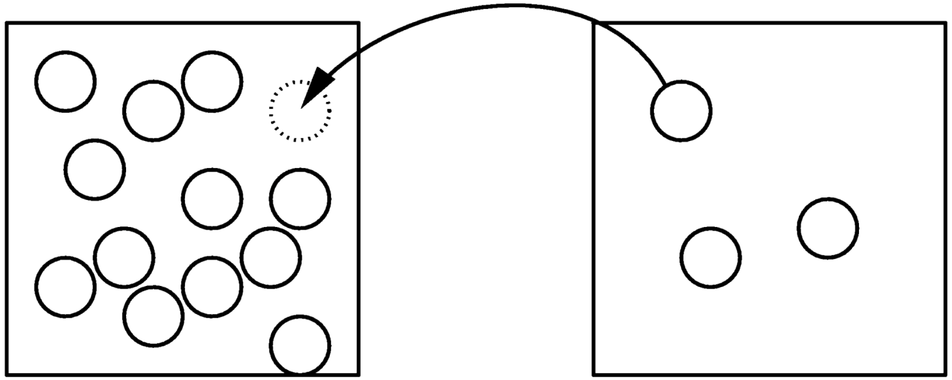
3 different kinds of trial moves:



Particle displacement



Volume change
⇨ equal P



Particle exchange
⇨ equal μ

Acceptance rules

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(s_1^{N_1}) + U(s_2^{N-N_1})\right]\right\}$$

Detailed Balance:

$$K(o \rightarrow n) = K(n \rightarrow o)$$

$$\mathcal{N}(o) \times \alpha(o \rightarrow n) \times \text{acc}(o \rightarrow n) = \mathcal{N}(n) \times \alpha(n \rightarrow o) \times \text{acc}(n \rightarrow o)$$

$$\frac{\text{acc}(o \rightarrow n)}{\text{acc}(n \rightarrow o)} = \frac{\mathcal{N}(n) \times \alpha(n \rightarrow o)}{\mathcal{N}(o) \times \alpha(o \rightarrow n)}$$

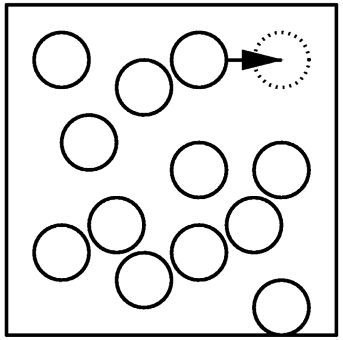
$$\frac{\text{acc}(o \rightarrow n)}{\text{acc}(n \rightarrow o)} = \frac{\mathcal{N}(n)}{\mathcal{N}(o)}$$

Displacement of a particle in box1

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U(s_1^{N_1})+U(s_2^{N-N_1})\right]\right\}$$

$$\mathcal{N}(n) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U(n)+U(s_2^{N-N_1})\right]\right\}$$

$$\mathcal{N}(o) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U(o)+U(s_2^{N-N_1})\right]\right\}$$



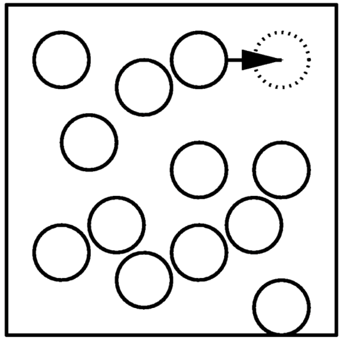
$$\frac{\text{acc}(o \rightarrow n)}{\text{acc}(n \rightarrow o)} = \frac{\cancel{\frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}}\exp\left\{-\beta\left[U(n)+\cancel{U(s_2^{N-N_1})}\right]\right\}}{\cancel{\frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}}\exp\left\{-\beta\left[U(o)+\cancel{U(s_2^{N-N_1})}\right]\right\}}$$

Displacement of a particle in box1

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(s_1^{N_1}) + U(s_2^{N-N_1})\right]\right\}$$

$$\mathcal{N}(n) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(n) + U(s_2^{N-N_1})\right]\right\}$$

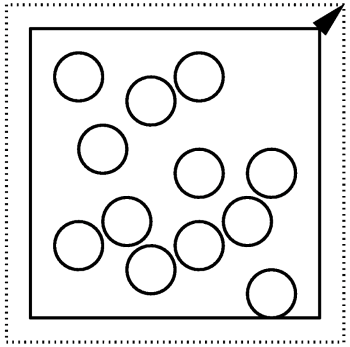
$$\mathcal{N}(o) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(o) + U(s_2^{N-N_1})\right]\right\}$$



$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\exp\{-\beta U_1(n)\}}{\exp\{-\beta U_1(o)\}}$$

Volume change

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U(s_1^{N_1})+U(s_2^{N-N_1})\right]\right\}$$



$$V_1^n = V_1^o + \Delta V$$

$$\frac{\mathcal{N}(n)}{\mathcal{N}(o)} = \frac{\frac{{V_{1,n}}^{N_1}(V-V_{1,n})^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U({s_{1,n}}^{N_1})+U({s_{2,n}}^{N-N_1})\right]\right\}}{\frac{{V_{1,o}}^{N_1}(V-V_{1,o})^{N-N_1}}{N_1!(N-N_1)!}\exp\left\{-\beta\left[U({s_{1,o}}^{N_1})+U({s_{2,o}}^{N-N_1})\right]\right\}}$$

$$\frac{\text{acc}(o \rightarrow n)}{\text{acc}(n \rightarrow o)} = \frac{V_{1,n}^{N_1}(V-V_{1,n})^{N-N_1}\exp\left\{-\beta U(s_n^N)\right\}}{V_{1,o}^{N_1}(V-V_{1,o})^{N-N_1}\exp\left\{-\beta U(s_o^N)\right\}}$$

Volume change

More efficient: random walk in ln [$V_1/(V-V_1)$]

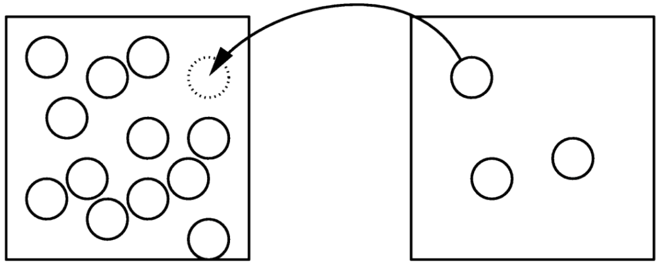
$$
\begin{aligned}
Q_G(N,V,T) &= \frac{1}{\Lambda^{3N} N!} \sum_{N_1=0}^{N} \binom{N}{N_1} \\
&\times \int_{-\infty}^{\infty} \mathrm{d}\ln\left(\frac{V_1}{V-V_1}\right) \frac{V_1(V-V_1)}{V} V_1^{N_1}(V-V_1)^{N-N_1} \\
&\times \int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right] \int \mathrm{d}s_2^{N-N_1} \exp\left[-\beta U(s_2^{N-N_1})\right]
\end{aligned}
$$

$$
\mathcal{N}(n) \propto \frac{V_{1,n}^{N_1+1}(V-V_{1,n})^{N-(N_1-1)}}{V N_1!(N-N_1)!} \exp\left[-\beta U(s_n^N)\right]
$$

$$
\frac{\mathrm{acc}(o \rightarrow n)}{\mathrm{acc}(n \rightarrow o)} = \frac{V_{1,n}^{N_1+1}(V-V_{1,n})^{N-N_1+1} \exp\left\{-\beta U(s_n^N)\right\}}{V_{1,o}^{N_1+1}(V-V_{1,o})^{N-N_1+1} \exp\left\{-\beta U(s_o^N)\right\}}
$$

Moving a particle from box1 to box2

$$\mathcal{N}\left(N_1, V_1, s_1^{N_1}, s_2^{N-N_1}\right) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U(s_1^{N_1}) + U(s_2^{N-N_1})\right]\right\}$$



acceptance rule:

$$\mathcal{N}(n) \propto \frac{V_1^{N_1-1}(V-V_1)^{N-(N_1-1)}}{(N_1-1)!\,(N-(N_1-1))!} \exp\left\{-\beta\left[U_1(n) + U_2(n)\right]\right\}$$

$$\mathcal{N}(o) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U_1(o) + U_2(o)\right]\right\}$$

$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\frac{V_1^{N_1-1}(V-V_1)^{N-(N_1-1)}}{(N_1-1)!(N-(N_1-1))!} \exp\left\{-\beta\left[U_1(n) + U_2(n)\right]\right\}}{\frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left\{-\beta\left[U_1(o) + U_2(o)\right]\right\}}$$

Moving a particle from box1 to box2

$$\mathcal{N}(n) \propto \frac{V_1^{N_1-1}(V-V_1)^{N-(N_1-1)}}{(N_1-1)!\,(N-(N_1-1))!}\exp\{-\beta\,[U_1(n)+U_2(n)]\}$$

$$\mathcal{N}(o) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\{-\beta\,[U_1(o)+U_2(o)]\}$$

$$\frac{\mathrm{acc}(o\rightarrow n)}{\mathrm{acc}(n\rightarrow o)} = \frac{\dfrac{\cancel{V_1^{N_1-1}}(V-V_1)^{\cancel{N-(N_1-1)}}}{\cancel{(N_1-1)}!(N-(N_1-1))\cancel{!}}\exp\{-\beta\,[U_1(n)+U_2(n)]\}}{\dfrac{V_1^{\cancel{N_1}}\cancel{(V-V_1)^{N-N_1}}}{N_1\cancel{!}\cancel{(N-N_1)!}}\exp\{-\beta\,[U_1(o)+U_2(o)]\}}$$

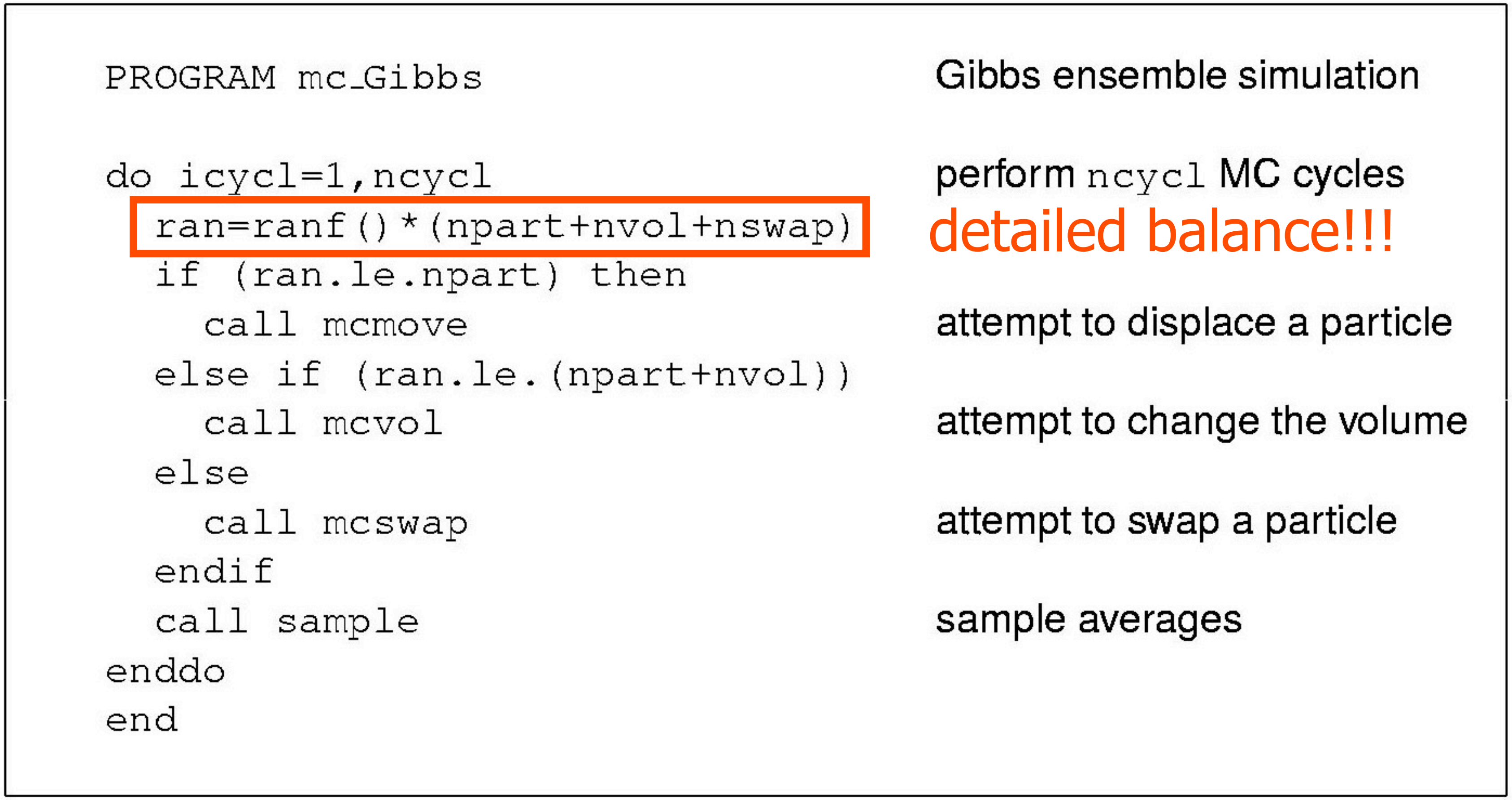
Moving a particle from box1 to box2

$$\mathcal{N}(n) \propto \frac{V_1^{N_1-1}(V-V_1)^{N-(N_1-1)}}{(N_1-1)!\,(N-(N_1-1))!}\exp\{-\beta\left[U_1(n)+U_2(n)\right]\}$$

$$\mathcal{N}(o) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}\exp\{-\beta\left[U_1(o)+U_2(o)\right]\}$$

$$\frac{\mathrm{acc}(o \rightarrow n)}{\mathrm{acc}(n \rightarrow o)} = \frac{N_1(V-V_1)}{(N-N_1+1)V_1}\,\frac{\exp\left\{-\beta U(s_n^N)\right\}}{\exp\left\{-\beta U(s_o^N)\right\}}$$

Algorithm 17 (Basic Gibbs Ensemble Simulation)



```
PROGRAM mc_Gibbs                          Gibbs ensemble simulation

do icycl=1,ncycl                          perform ncycl MC cycles
  ran=ranf()*(npart+nvol+nswap)
  if (ran.le.npart) then
    call mcmove                           attempt to displace a particle
  else if (ran.le.(npart+nvol))
    call mcvol                            attempt to change the volume
  else
    call mcswap                           attempt to swap a particle
  endif
  call sample                             sample averages
enddo
end
```

detailed balance!!!

Algorithm 18 (Attempt to Change the Volume in the Gibbs Ensemble)

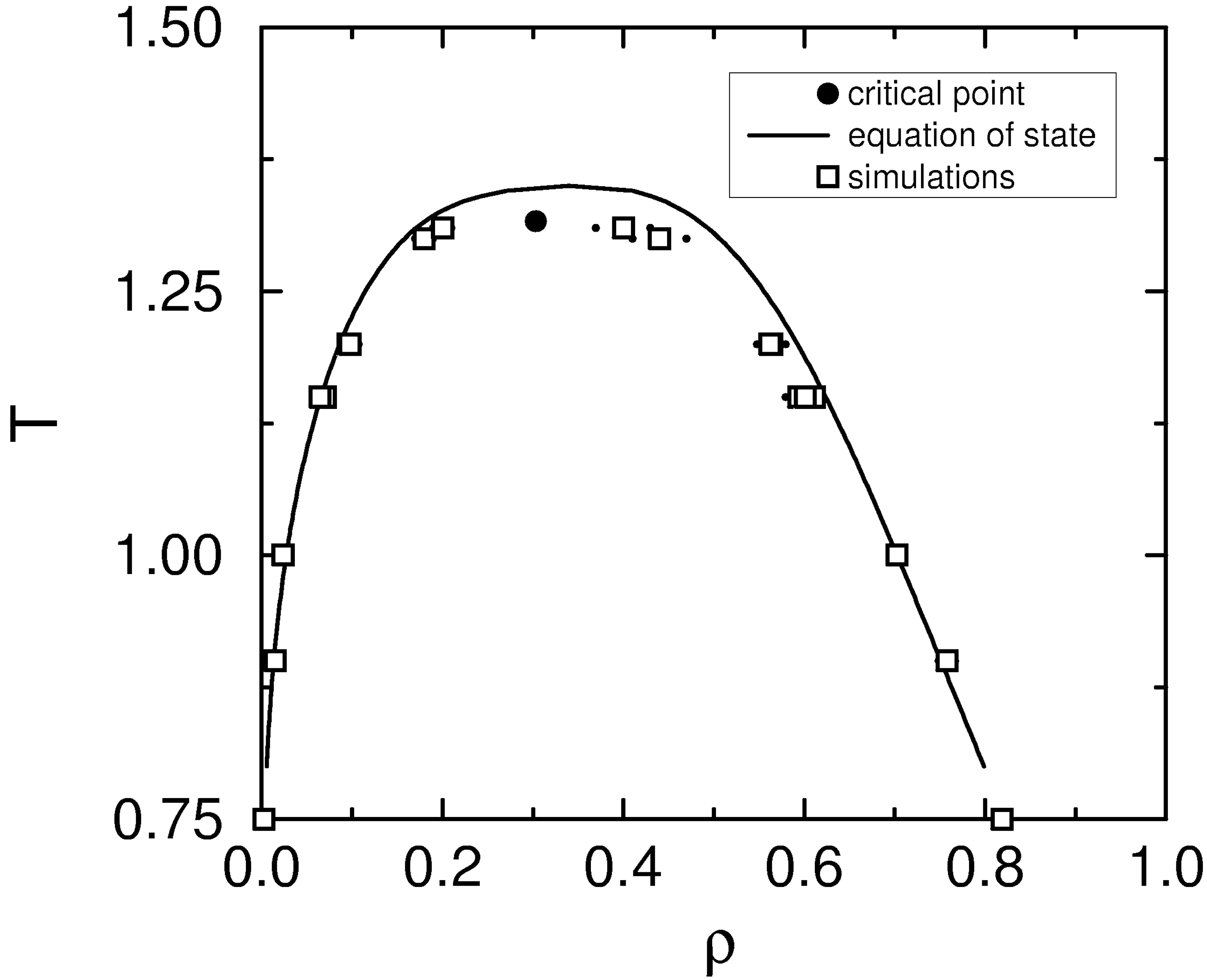
```
  SUBROUTINE mcvol                        attempt to change
                                          the volume
  call toterg(box1,en1o)                  energy old conf. box 1
  call toterg(box2,en2o)                  and 2 (box1: box length)
  vo1=box1**3                             old volume box 1 and 2
  vo2=v-vo1
  lnvn=log(vo1/vol2)+                     random walk in ln V1/V2
+     (ranf()-0.5)*vmax
  v1n=v*exp(lnvn)/(1+exp(lnvn))           new volume box 1 and 2
  v2n=v-v1n
  box1n=v1n**(1/3)                        new box length box 1
  box2n=v2n**(1/3)                        new box length box 2
  do i=1,npart
    if (ibox(i).eq.1) then                determine which box
      fact=box1n/box1o
    else
      fact=box2n/box2o
    endif
    x(i)=x(i)*fact                        rescale positions
  enddo
  call toterg(box1n,en1n)                 total energy box 1
  call toterg(box2n,en2n)                 total energy box 2
  arg1=-beta*((en1n-en1o)+
```

```
    box1n=v1n**(1/3)                              new box length box 1
    box2n=v2n**(1/3)                              new box length box 2
    do i=1,npart
      if (ibox(i).eq.1) then                      determine which box
        fact=box1n/box1o
      else
        fact=box2n/box2o
      endif
      x(i)=x(i)*fact                              rescale positions
    enddo
    call toterg(box1n,en1n)                       total energy box 1
    call toterg(box2n,en2n)                       total energy box 2
    arg1=-beta*((en1n-en1o)+
   + (npbox(1)+1)*log(v1n/v1o)/beta)              appropriate weight function
    arg2=-beta*((en2n-en2o)+                      acceptance rule (8.3.3)
   + (npbox(2)+1)*log(v2n/v2o)/beta)
    if (ranf().gt.exp(arg1+arg2)) then
      do i=1,npart                                REJECTED
        if (ibox(i).eq.)  then                    determine which box
          fact=box1o/box1n
        else
          fact=box2o/box2n
        endif
        x(i)=x(i)*fact                            restore old configuration
      enddo
    endif
    return
    end
```

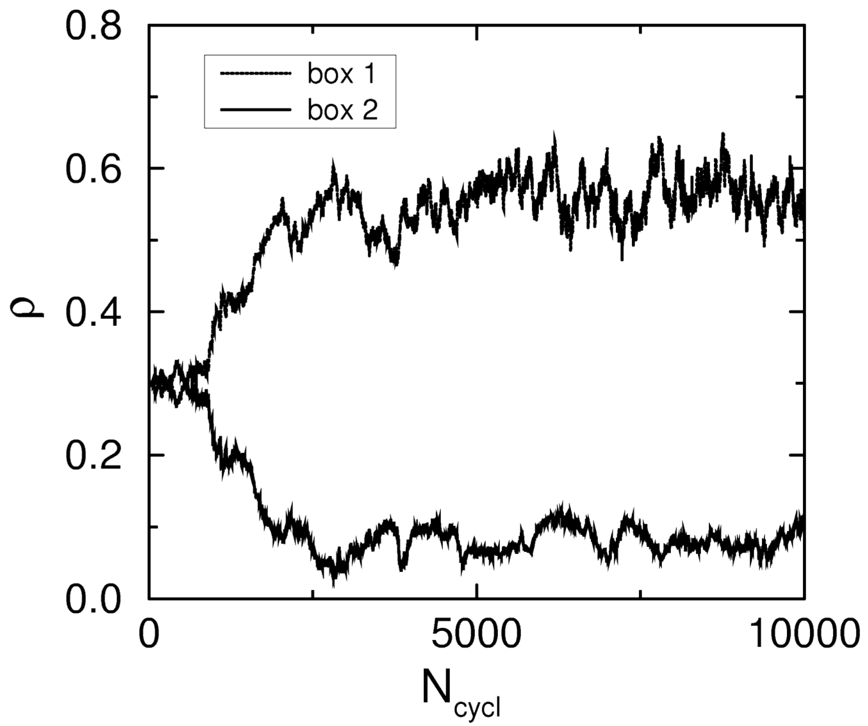
Algorithm 19 (Attempt to Swap a Particle between the Two Boxes)

```
   SUBROUTINE mcswap                       attempts to swap a particle
                                           between the two boxes
   if (ranf().lt.0.5) then                 which box to add or remove
     in=1
     out=2
   else
     in=2
     out=1
   endif
   xn=ranf()*box(in)                       new particle at a random position
   call ener(xn,enn,in)                    energy new particle in box in
   w(in)=w(in)+vol(in)*                    update chemical potential (8.3.5)
+  exp(-beta*enn)/(npbox(in)+1)
   if (npbox(out).eq.0) return             if box empty return
   ido=0                                   find a particle to be removed
   do while (ido.ne.out)
      o=int(npart*ranf())+1
      ido=ibox(o)
   enddo
   call ener(x(o),eno,out)                 energy particle o in box out
   arg=exp(-beta*(enn-eno +
+  log(vol(out)*(npbox(in)+1)/             acceptance rule (8.3.4)
+  (vol(in)*npbox(out)))/beta))
```

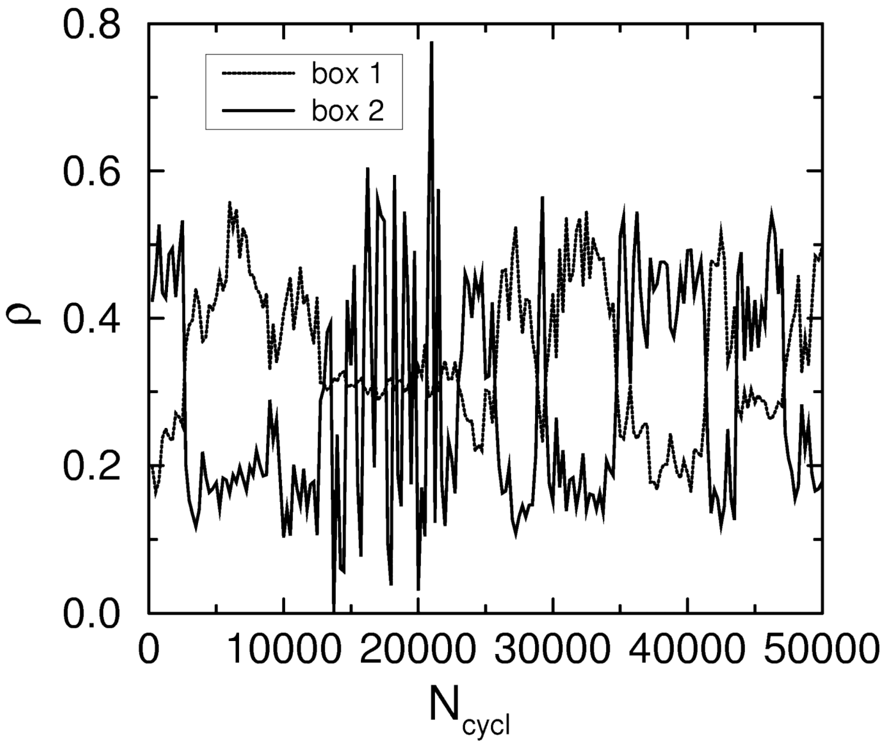
```
      else
        in=2
        out=1
      endif
      xn=ranf()*box(in)                 new particle at a random position
      call ener(xn,enn,in)              energy new particle in box in
      w(in)=w(in)+vol(in)*              update chemical potential (8.3.5)
     + exp(-beta*enn)/(npbox(in)+1)
      if (npbox(out).eq.0) return       if box empty return
      ido=0                             find a particle to be removed
      do while (ido.ne.out)
         o=int(npart*ranf())+1
         ido=ibox(o)
      enddo
      call ener(x(o),eno,out)           energy particle o in box out
      arg=exp(-beta*(enn-eno +
     + log(vol(out)*(npbox(in)+1)/      acceptance rule (8.3.4)
     + (vol(in)*npbox(out)))/beta))
      if (ranf().lt.arg) then
        x(o)=xn                         add new particle to box in
        ibox(o)=in
        nbox(out)=npbox(out)-1
        nbox(in)=npbox(in)+1
      endif
      return
      end
```


1.50
1.25
1.00
0.75
T
0.0
0.2
0.4
0.6
0.8
1.0
ρ
● critical point
equation of state
□ simulations

Analyzing the results (1)

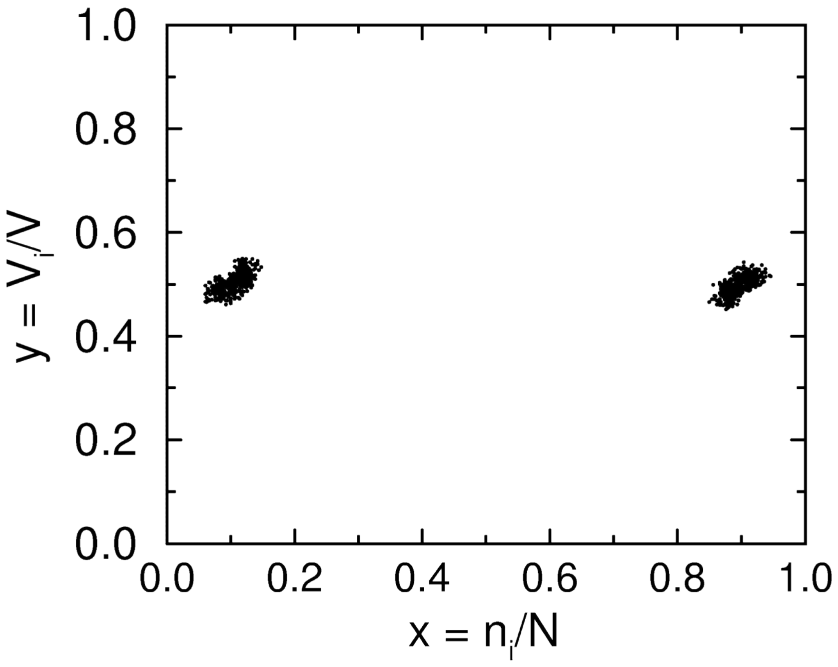


well below T_c

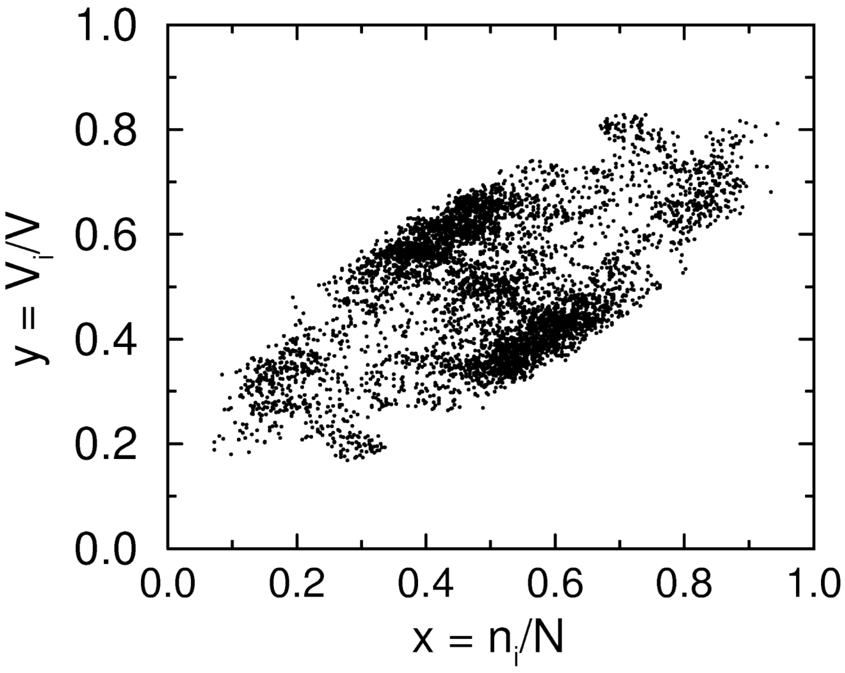


approaching T_c

Analyzing the results (2)

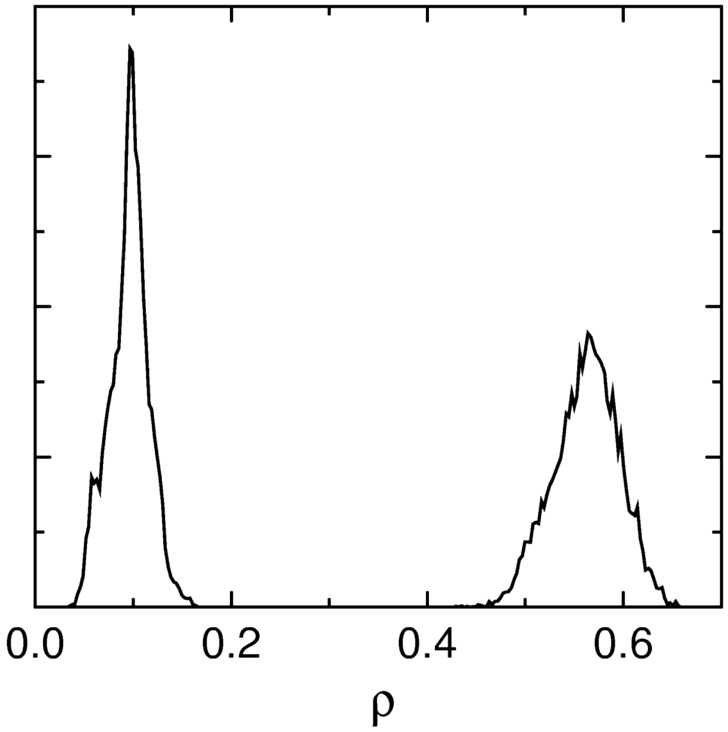


well below T_c

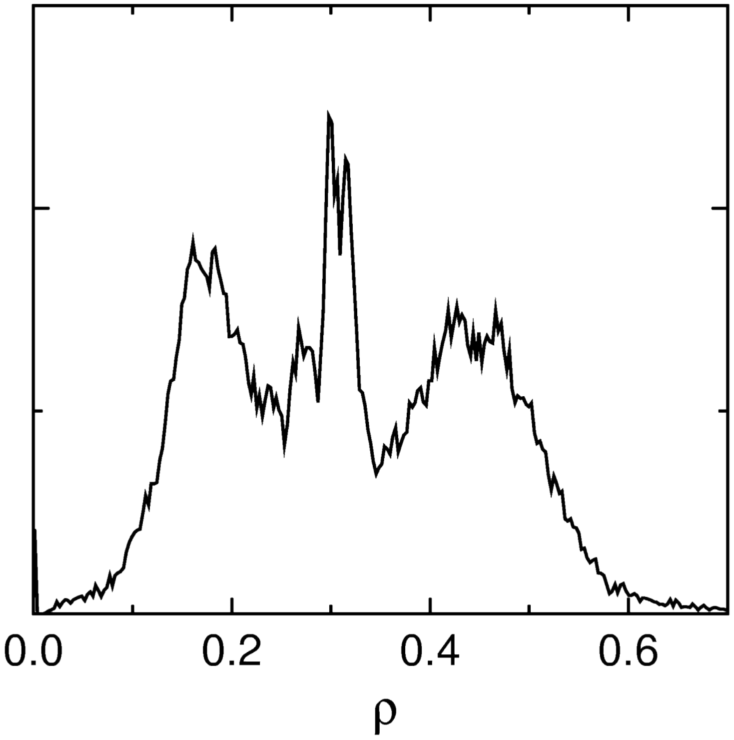


approaching T_c

Analyzing the results (3)



well below T_c



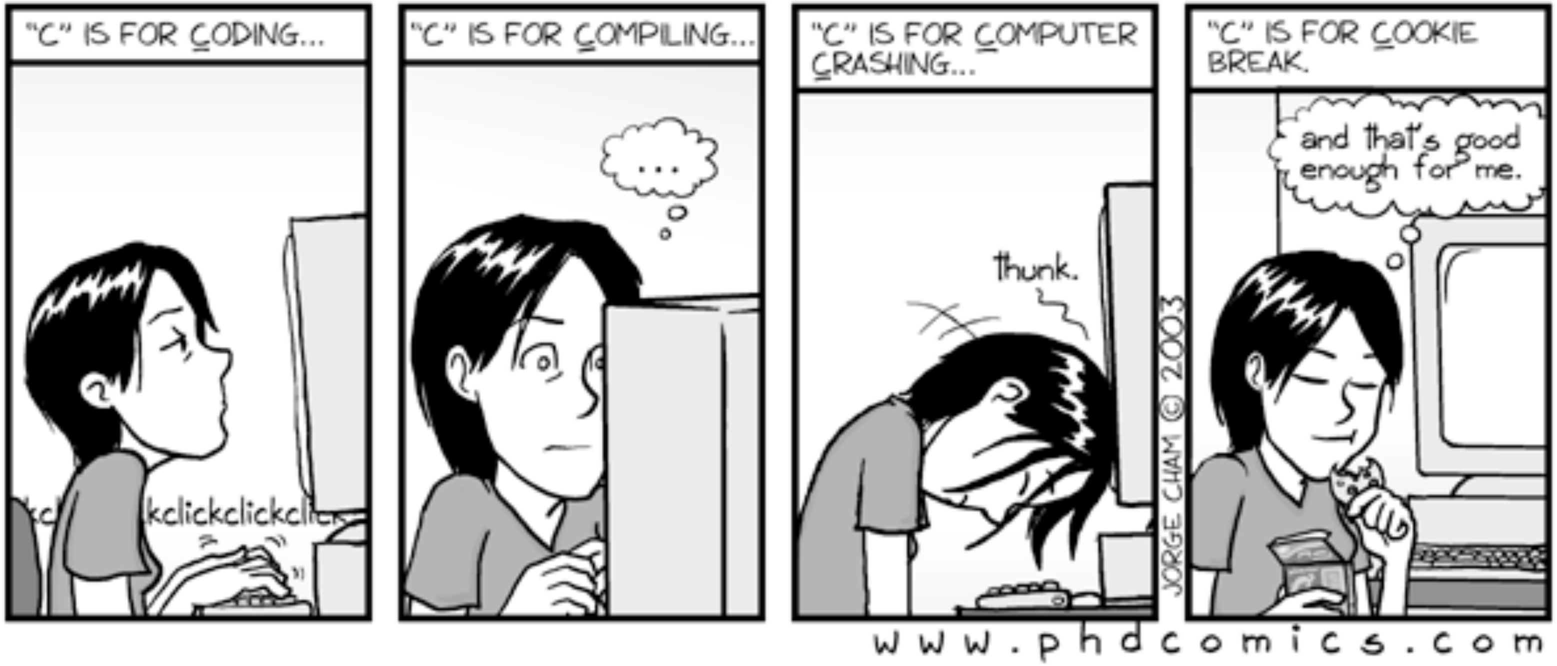
approaching T_c

Advantages:

- single simulation to study phase coexistence: system „finds“ the densities of coexisting phases
- free energies/chem. potentials need not be calculated
- significant reduction of computer time

Disadvantages:

- only for vapor-liquid and liquid-liquid coexistence
- not very successful for dense phases (particle insertion!)

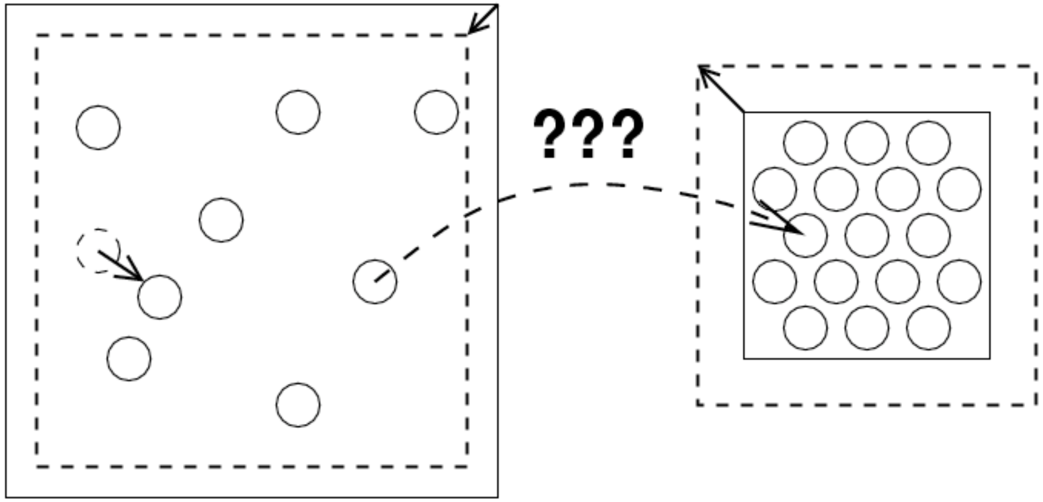


"C" IS FOR COFFEE BREAK!

Ensembles II:

2. Free energy of solids (Chap 7 & 10)
3. Case study: cluster solids

Gibbs ensemble for **solid** phases?



problem: swap moves are unlikely to be accepted.

$\mu_1 \neq \mu_2$ ⇨ Gibbs ensemble does not work

What other method?

Problem:

With normal Monte Carlo simulations, we cannot compute "thermal" quantities, such as S, F and G, because they depend on the **total volume of accessible phase space**.

For example:

$$F = -kT \ln Q$$

with

$$Q = \frac{1}{\Lambda^{3N} N!} \int d\mathbf{r}^N \exp[-\mathcal{U}(\mathbf{r}^N)/k_B T]$$

Solution: thermodynamic integration

$$U(\lambda) = (1-\lambda)\, U_I + \lambda U_{II}$$

Coupling parameter

$U(0) = U_I$ system of interest

$U(1) = U_{II}$ reference system

$$\boxed{F = -\frac{1}{\beta}\ln\left(Q_{NVT}\right)}$$

$$Q_{NVT}(\lambda) = \frac{1}{\Lambda^{3N} N!}\int \mathrm{dr}^N \exp\left[-\beta U(\lambda)\right]$$

$$\left(\frac{\partial F(\lambda)}{\partial \lambda}\right)_{N,T} = -\frac{1}{\beta}\frac{\partial}{\partial \lambda}\ln(Q) = -\frac{1}{\beta}\frac{1}{Q}\frac{\partial Q}{\partial \lambda}$$

$$= \frac{\int \mathrm{dr}^N \left(\partial U(\lambda)/\partial \lambda\right)\exp\left[-\beta U(\lambda)\right]}{\int \mathrm{dr}^N \exp\left[-\beta U(\lambda)\right]}$$

$$= \left\langle \frac{\partial U(\lambda)}{\partial \lambda}\right\rangle_\lambda$$

$$F(\lambda = 0) = F(\lambda = 1) - \int_0^1 \mathrm{d}\lambda \left\langle \frac{\partial U(\lambda)}{\partial \lambda}\right\rangle_\lambda$$

Free energy as ensemble average!

$$\boxed{\left\langle \frac{\partial U(\lambda)}{\partial \lambda}\right\rangle_\lambda = \left\langle U_{II} - U_I\right\rangle_\lambda}$$

Why $U(\lambda) = (1-\lambda)\,U_I + \lambda U_{II}$?

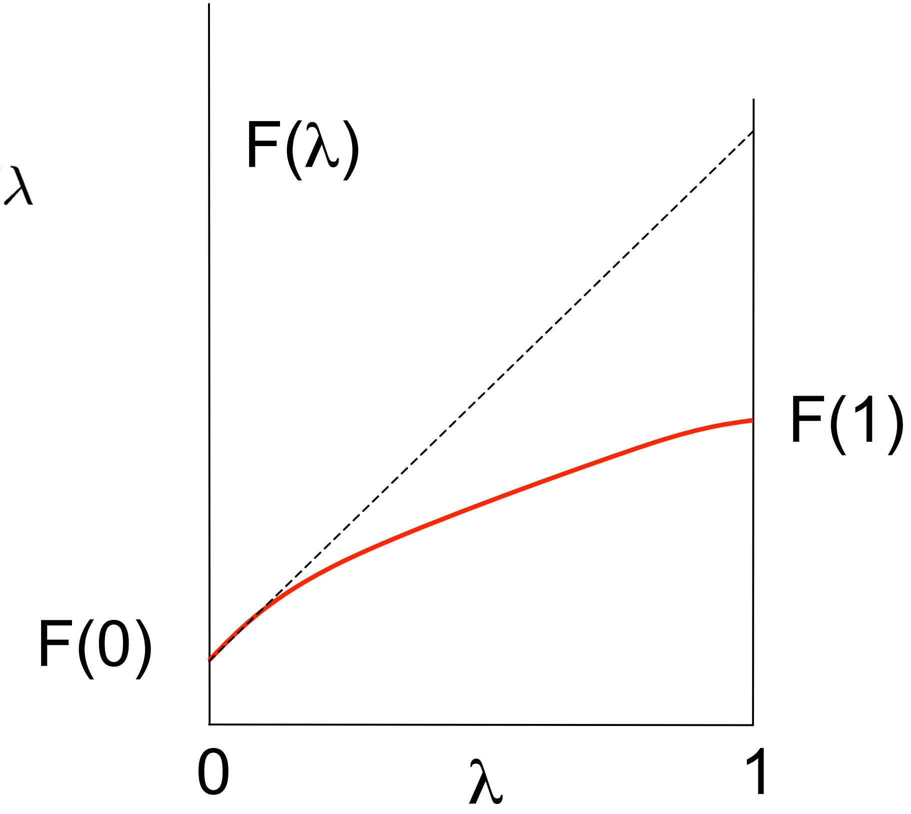
The second derivative is ALWAYS negative:

$$\left(\frac{\partial^2 F}{\partial \lambda^2}\right)_{N,V,T,\lambda} = -\beta\left[\left\langle \left(U_{II} - U_I\right)^2 \right\rangle_\lambda - \left\langle U_{II} - U_I \right\rangle_\lambda^2\right] \leq 0$$

Therefore:

$$\left(\frac{\partial F}{\partial \lambda}\right)_{NVT\lambda=0} \geq \left(\frac{\partial F}{\partial \lambda}\right)_{NVT\lambda}$$

Good test of simulation results…!

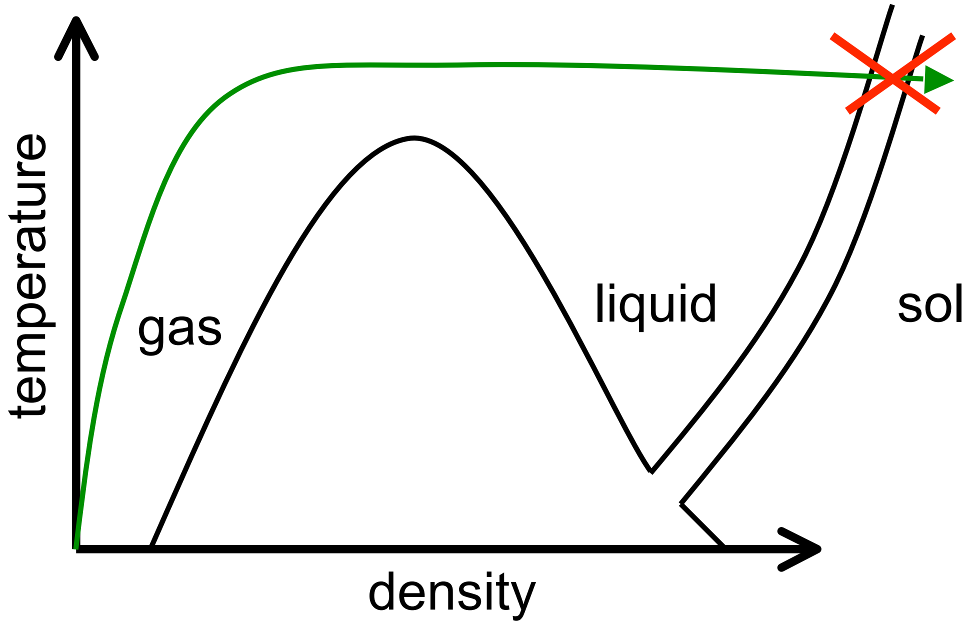


Words of caution

Integration has to be along a *reversible* path.

interested in solid ⇨ reference system has to be solid

Why? Because there is no reversible path coming from the ideal gas



Standard reference system: Einstein solid

Einstein crystal: non-interacting particles coupled to their lattice sites by harmonic springs

$$U(r^N;\lambda) = U(r_0^N) + (1-\lambda)\left[U(r^N) - U(r_0^N)\right] + \lambda\sum_{i=1}^{N}\alpha_i(r_i - r_{0,i})^2$$

$$F = F^{Einstein} - \int_{\lambda=0}^{\lambda=1} \mathrm{d}\lambda \left\langle \frac{\partial U(\lambda)}{\partial\lambda} \right\rangle$$

$$F^{Einstein} = U(r_0^N) - \frac{3}{2\beta}\sum_{i=1}^{N}\log\left(\frac{\pi}{\alpha_i\beta}\right)$$

$$\left\langle \frac{\partial U(\lambda)}{\partial\lambda} \right\rangle_\lambda = \left\langle \sum_{i=1}^{N}\alpha_i\left(r_i - r_{0,i}\right)^2 - \left[U(r^N) - U(r_0^N)\right] \right\rangle_\lambda$$

Einstein solid: recipe

For fixed crystal structure:

At fixed T and ρ:

- make simulations at different values of λ
- measure $\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_{\lambda}$

- Numerically integrate using e.g. Gauss-Legendre quadrature
- determine

$$F = F^{Einstein} - \int_{\lambda=0}^{\lambda=1} \mathrm{d}\lambda \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle$$

Standard reference system: Einstein solid

- special care for discontinuous potentials:
  not possible to linearly switch off interaction

- diverging short-range repulsion:
  for $\lambda=1$, particles can overlap ⇨ weak divergence in $\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_{\lambda}$

- choose α's such that $\left\langle \sum_{i=1}^{N} (r_i - r_{0,i})^2 \right\rangle_{\lambda=0} = \left\langle \sum_{i=1}^{N} (r_i - r_{0,i})^2 \right\rangle_{\lambda=1}$

- *Frenkel - Smit: „Thermodyn. integration for solids"*
  ~everyone else: „Frenkel-Ladd" (JCP, 1984)

Common tangent construction

Now that we have *F(V;T)* at hand:

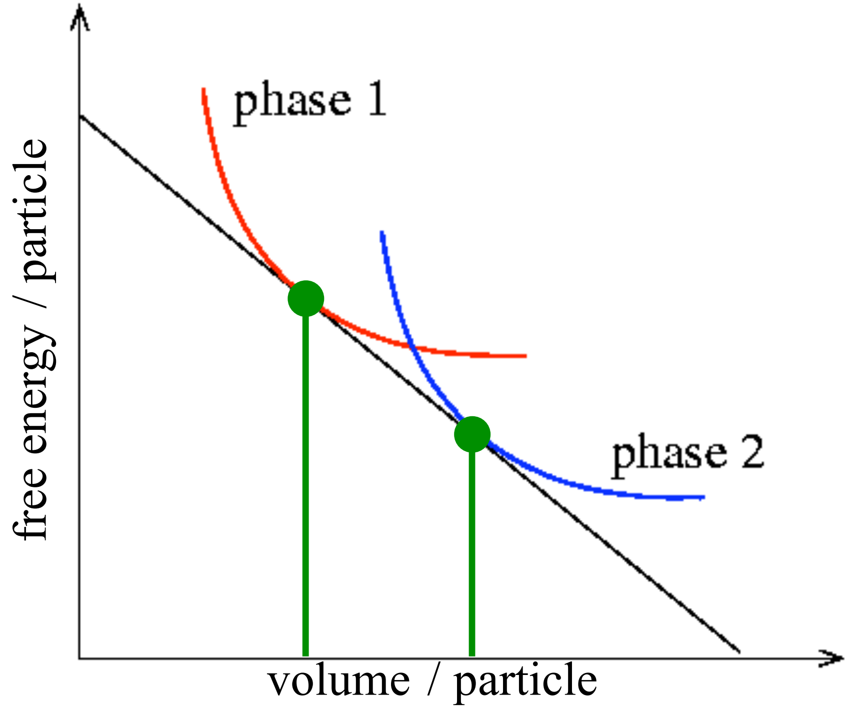
common tangent construction:

$P = -(\partial F/\partial V)_{N,T}$

equal pressure ⇨
equal slope

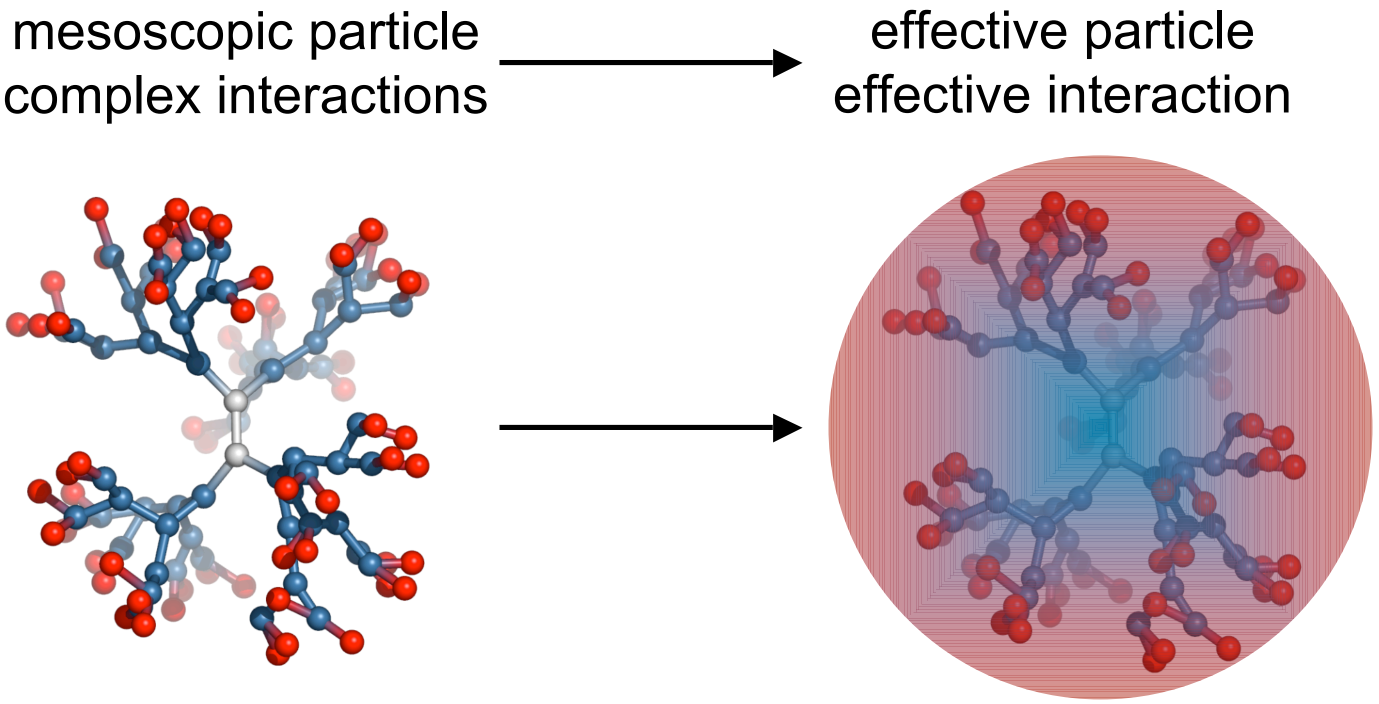
$F = \mu - PV$

equal chemical potentials ⇨
equal intercept



But now consider the following systems:

Coarse graining:



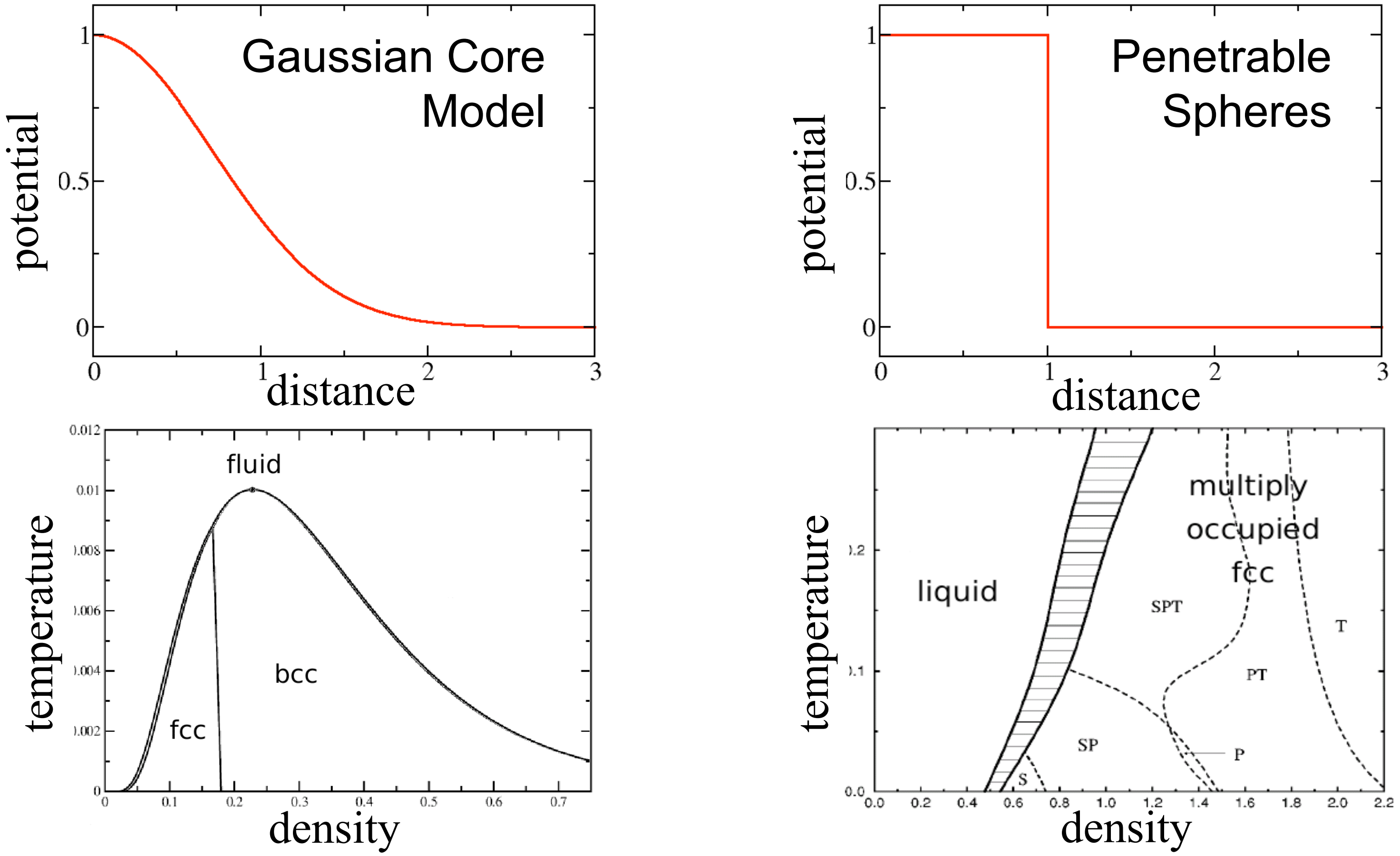
mesoscopic particle
complex interactions

effective particle
effective interaction

Effective interactions can be tuned

⇨ bounded, purely repulsive potentials

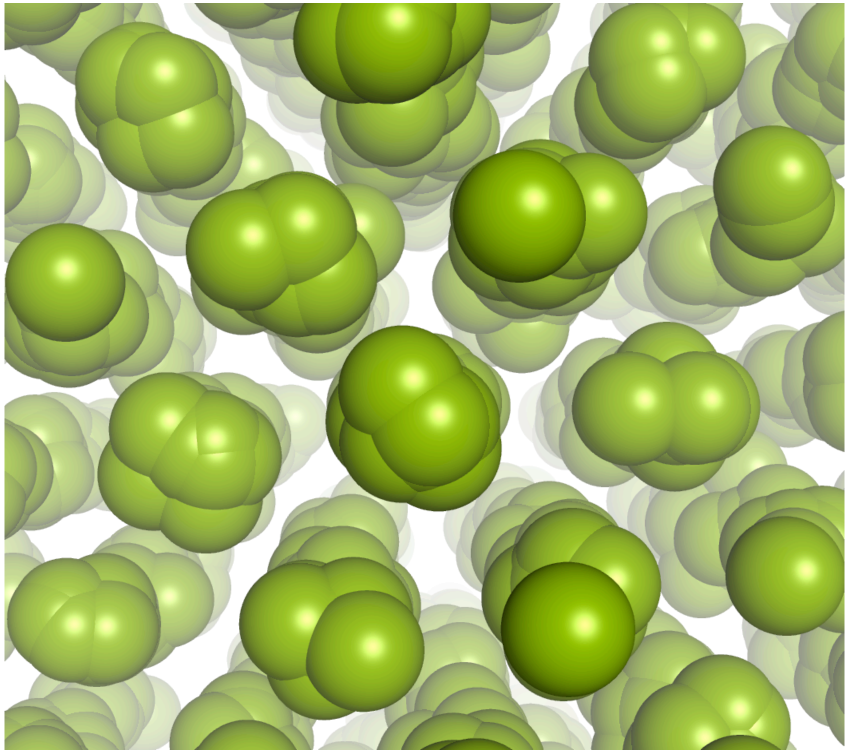
Effective interactions: soft and bounded



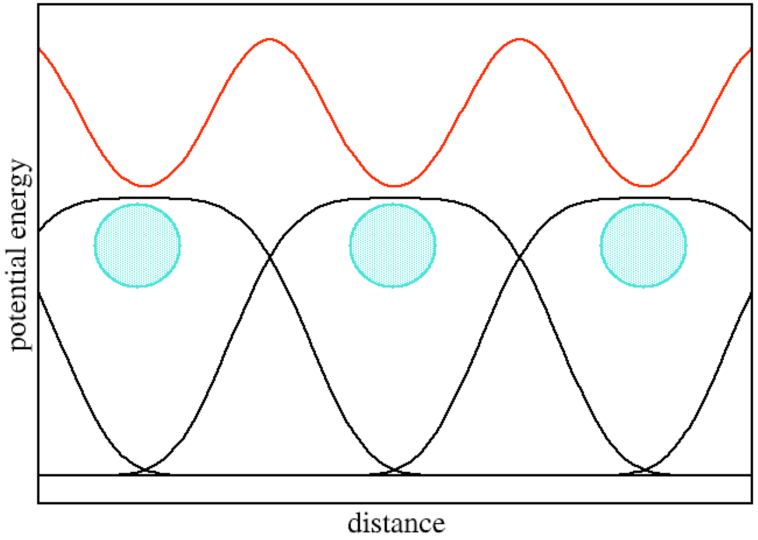
… are models for effective interactions of polymers etc.

Clustering

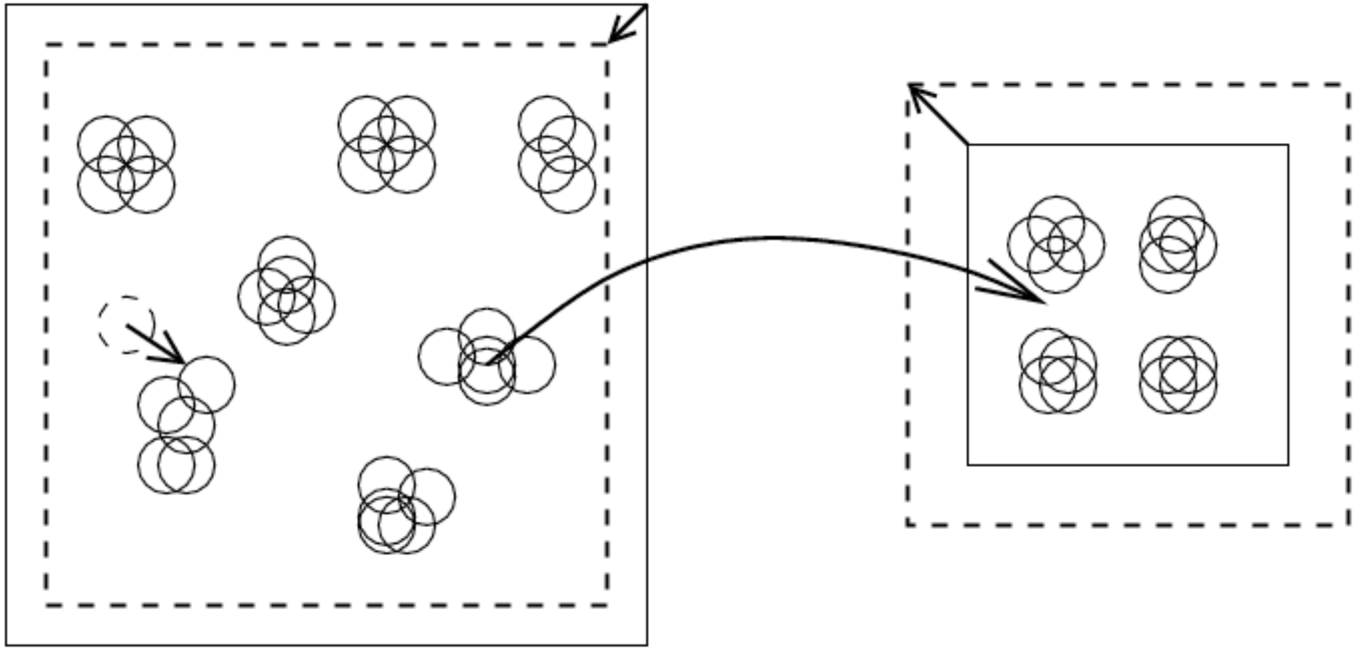
Potential energy surface in 1D:



somewhat steeper:



Gibbs ensemble for soft solids?



swap moves get accepted (soft potentials!!!) ⇨

“Great, in this case Gibbs ensemble works even for solids!“

…or does it not?

Gibbs ensemble for soft particles

It does NOT.

Because we find:
Different starting densities ⇨ different coexistence densities

But WHY???

Problem:

- We swap particles, that‘s good.
- We do NOT change the amount of lattice sites, that‘s bad.

Swope and Anderson, 1992:

$$dF = -SdT - PdV + \mu dN + \boxed{\mu_c dN_c}$$

important when:

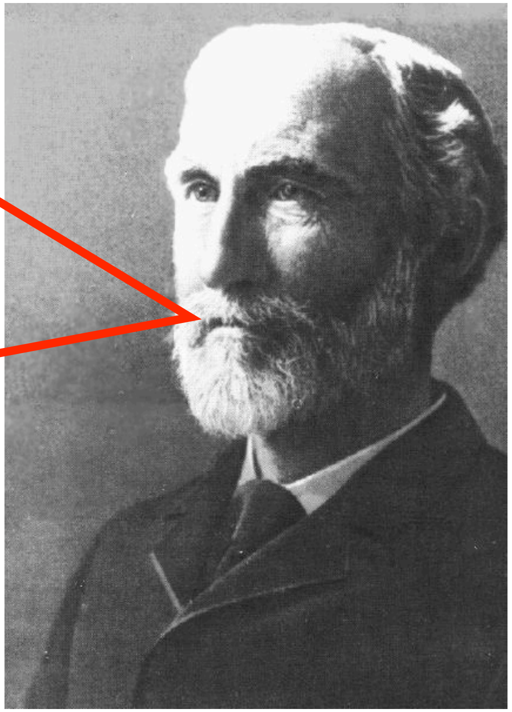
- vacancies and interstitials
- clustering

$$T_1 = T_2$$

$$P_1 = P_2$$

$$\mu_1 = \mu_2$$

$$\mu_c = 0$$



In bulk:

modifications at surfaces, interfaces and boundaries

In simulations:

Usually: geometry & periodic boundary conditions ⇨

$$N_c = const.$$

Therefore: ratio N/N_c is set at start of simulation and cannot equilibrate to real value

⇨ system gets stuck in states where $\mu_c \neq 0$.

This is the reason the Gibbs ensemble does not work for soft crystals. There are no moves ensuring that $\mu_c = 0$.

The solution: a recipe

For solid structure of interest (fcc, bcc,…) and N_c fixed:

- fix T and ρ:
  - choose N and carry out NVT simulation
  - measure:
    - P ……. virial equation
    - μ ……. Widom‘s insertion (P. Bolhuis, last Friday)
    - F ……. thermodynamic integration
  - determine μ_c using:

$$\mu_c = \frac{F(\mu_c) + P(\mu_c)V - \mu(\mu_c)N}{N_c}$$

  - Repeat for different values of N until $\mu_c = 0$ is found.

Widom's particle insertion, revisited

$$\mu \equiv \left(\frac{\partial F}{\partial N}\right)_{V,T}$$

But *N* is not a continuous variable. Therefore

$$\mu \approx \left(\frac{\boxed{F(N+1,V,T)-F(N,V,T)}}{(N+1)-N}\right)$$

$$\boxed{F(N+1,V,T)-F(N,V,T)} = -kT \ln \boxed{\frac{Q(N+1,V,T)}{Q(N,V,T)}}$$

$$\boxed{\frac{Q(N+1,V,T)}{Q(N,V,T)}} =$$

$$\frac{V}{(N+1)\Lambda^3}\,\frac{\int ds^{N+1} \exp(-\beta U(s^{N+1})}{\int ds^{N} \exp(-\beta U(s^{N})}$$

Now write

$$U\ \ (s^{N+1}) \equiv U(s^N) + \Delta U(s_{N+1}, s^N)$$

then

$$\frac{Q(N+1,V,T)}{Q(N,V,T)} =$$

$$\frac{V}{(N+1)\Lambda^3}\int ds_{N+1}\langle \exp(-\beta\Delta U(s_{N+1}, s^N))\rangle$$

And therefore

$$\mu = -kT\ln\left(\frac{V}{(N+1)\Lambda^3}\int ds_{N+1}\langle \exp(-\beta\Delta U(s_{N+1}, s^N))\rangle\right)$$

Finally:

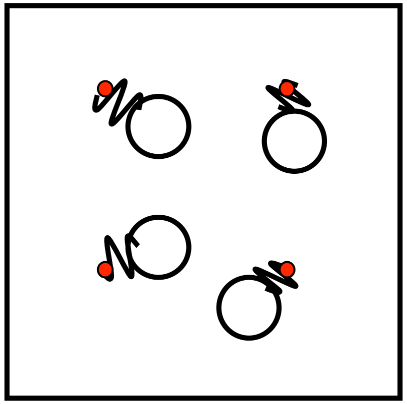
$$\mu = \mu_{id.gas} - kT\ \mathrm{ln}\left(\int ds_{N+1} \langle \exp(-\beta \Delta U(s_{N+1}, s^N)) \rangle \right)$$

Recipe

1. Evaluate ΔU for a random insertion of a particle in a system containing N particles.
2. Compute $\exp(-\beta \Delta U)$
3. Repeat M times and compute the average "Boltzmann factor" $< \exp(-\beta \Delta U) >$
4. Then $\mu_{excess} = -kT\ \mathrm{ln} < \exp(-\beta \Delta U) >$

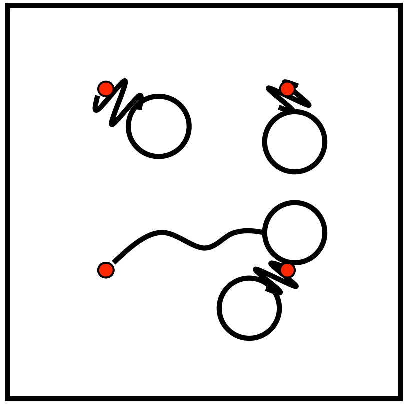
Thermodynamic intergration for soft particles

Einstein crystal not appropriate:



particles can hop
to other lattice sites

springs get stretched



Thermodynamic intergration for soft particles

Reference system (U_0): ideal gas confined by barriers

$$U(\lambda) = (1-\lambda)U_0 + \lambda U_{\text{system}}$$

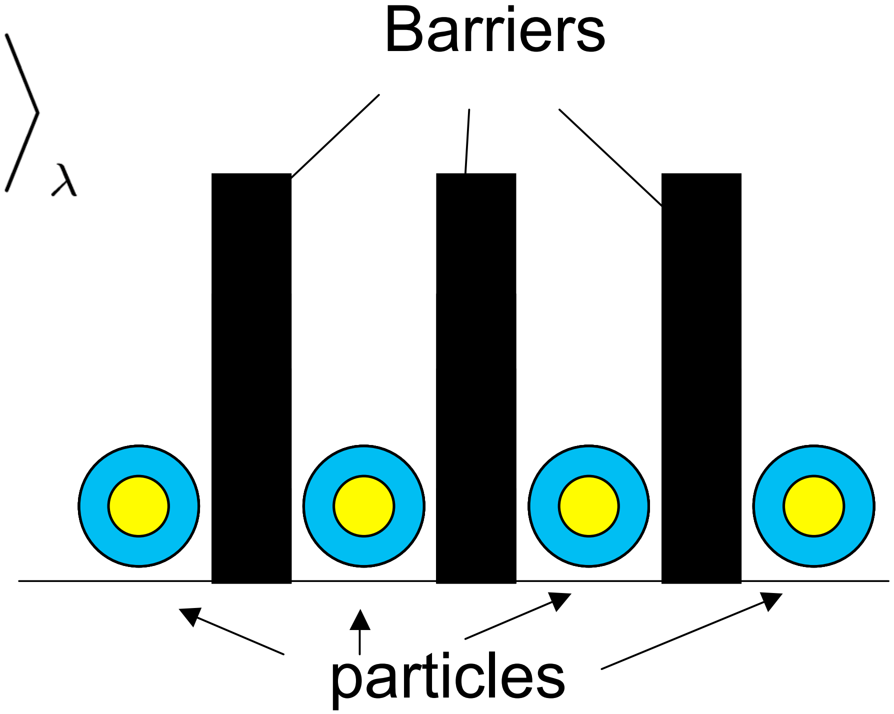
$$F_{\text{system}} = F_0 + \int_0^1 d\lambda \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda$$

$$\left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_\lambda = \left\langle U_0 - U_{system} \right\rangle_\lambda$$

$$\lambda = 0.75$$

$$Q_0 = \frac{1}{\Lambda^{3N} N!}\left(V_W + V_B \exp(-\beta U_{\max})\right)^N$$

$$\beta F_0 = \log Q_0$$



Mladek, Charbonneau, Frenkel, PRL (2007)

The solution: a recipe

For solid structure of interest (fcc, bcc,…) and N_c fixed:

- fix T and ρ:
  - choose N and carry out NVT simulation
  - measure:
    - P ……. virial equation
    - μ ……. Widom‘s insertion (P. Bolhuis, last Friday)
    - F ……. thermodynamic integration
  - determine μ_c using:

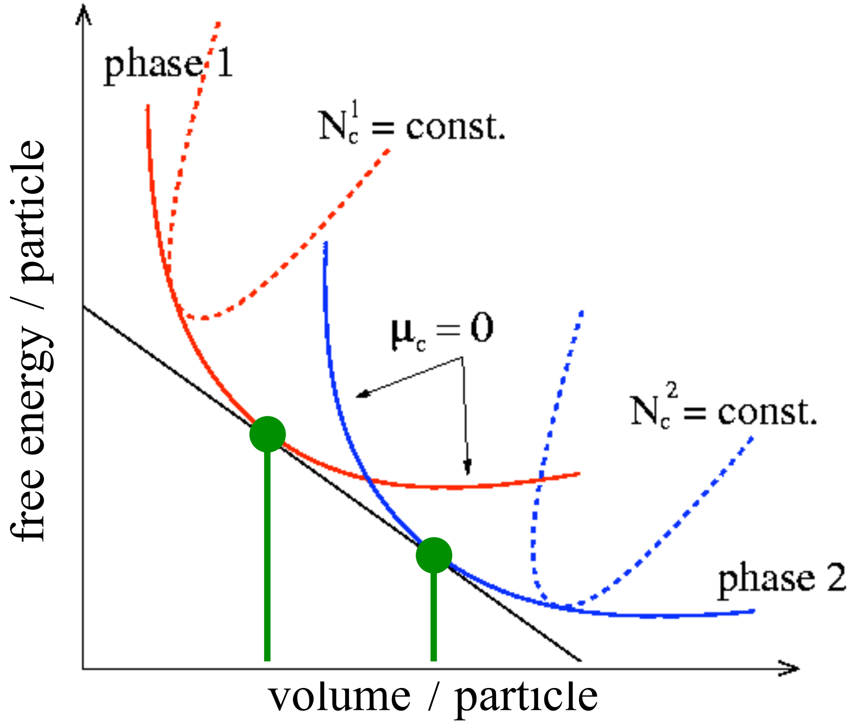
$$\mu_c = \frac{F(\mu_c) + P(\mu_c)V - \mu(\mu_c)N}{N_c}$$

  - Repeat for different values of N until $\mu_c = 0$ is found.

… and then???

Common tangent construction

For cluster crystals: only valid for $F/N(\mu_c = 0)$ curves!



Side note:
Once equilibrium is found

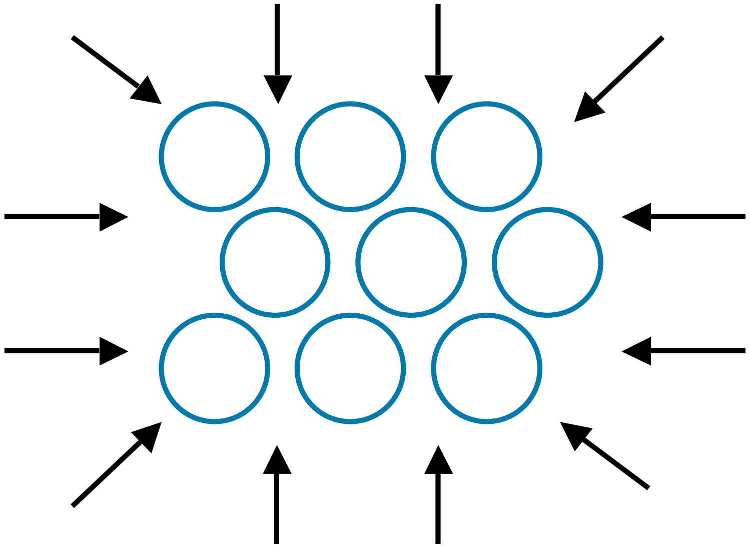
μ_c irrelevant?

No, because:
Modified thermodynamic formalism has impact on the second derivatives of the free energy,
e.g. the bulk modulus

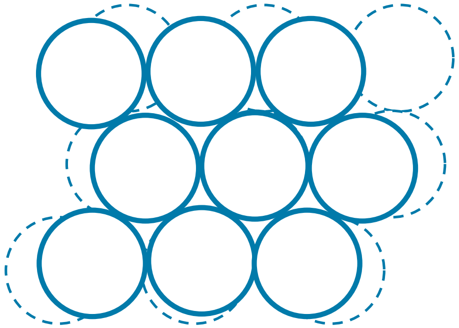
Example: bulk modulus

$$B = V\,(\partial^2 F/\,\partial V^2)_{N,T}$$
$$= -V\,(\partial P/\,\partial V)_{N,T}$$

$$P = P[N,V,T,N_c(N,V,T)]$$

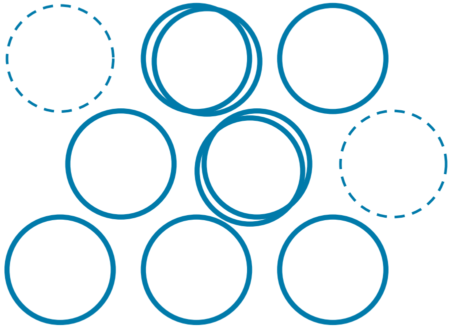


affine shrinking:



$$-V\,(\partial P/\,\partial V)_{N,T,N_c}$$

lattice site deletion:



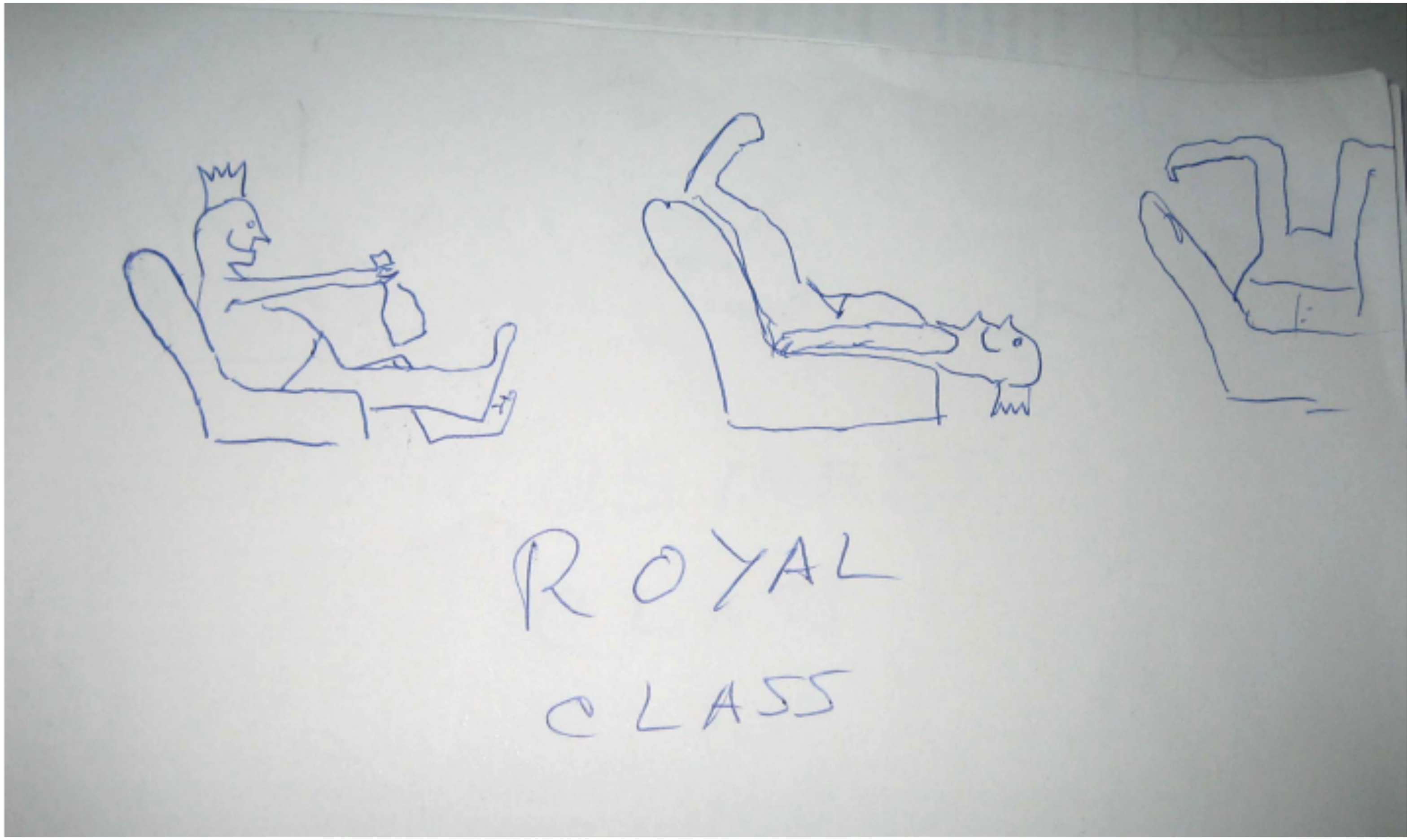
$$-V\,(\partial P/\,\partial N_c)_{N,T,V}\,(\partial N_c/\,\partial V)_{N,T,\mu_c=0}$$

How to visualise what we just learned?

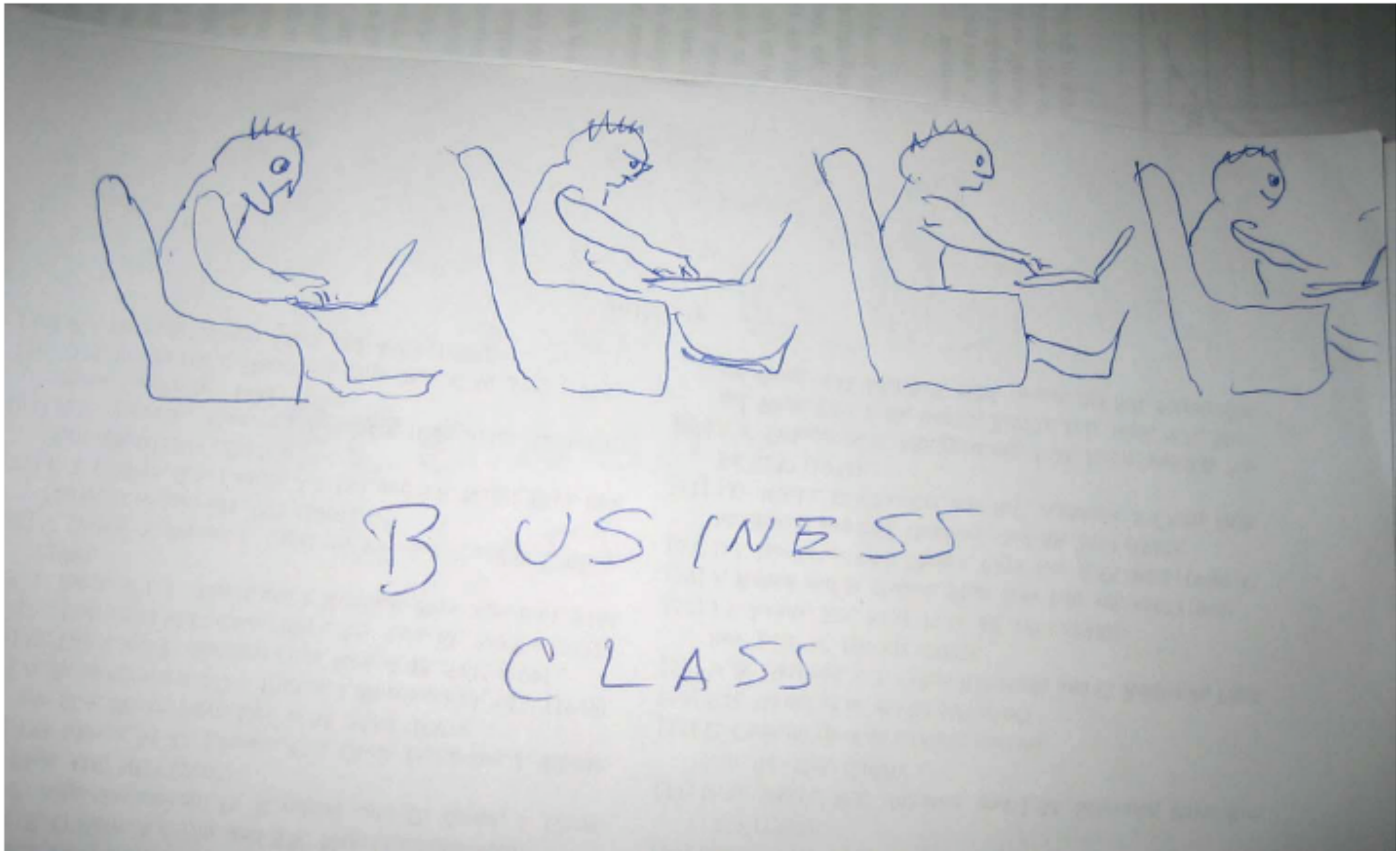
“The Future of Air Travel”

Short Essay On the Dangers of Applying Science

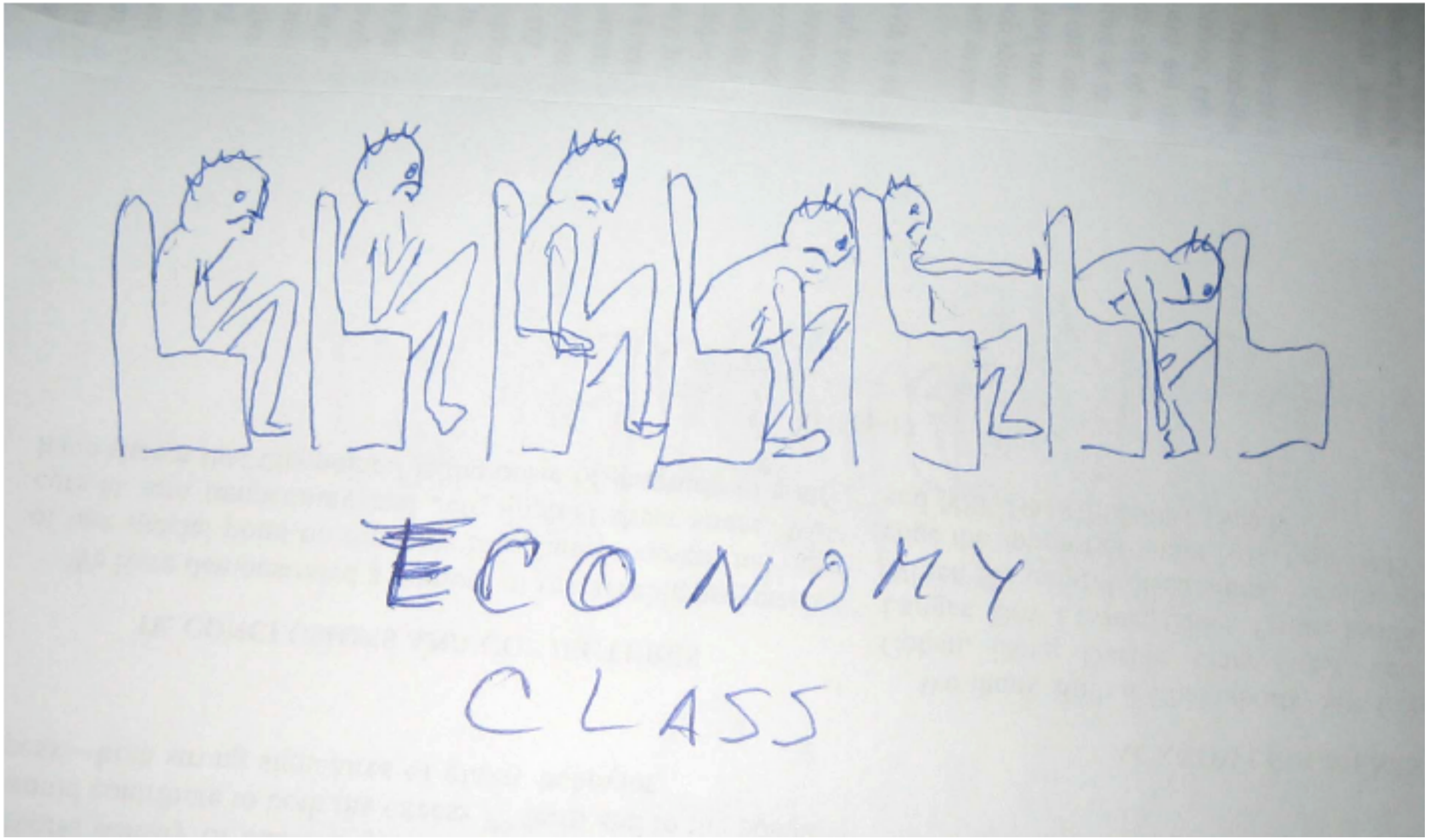
by Daan Frenkel



To save costs…



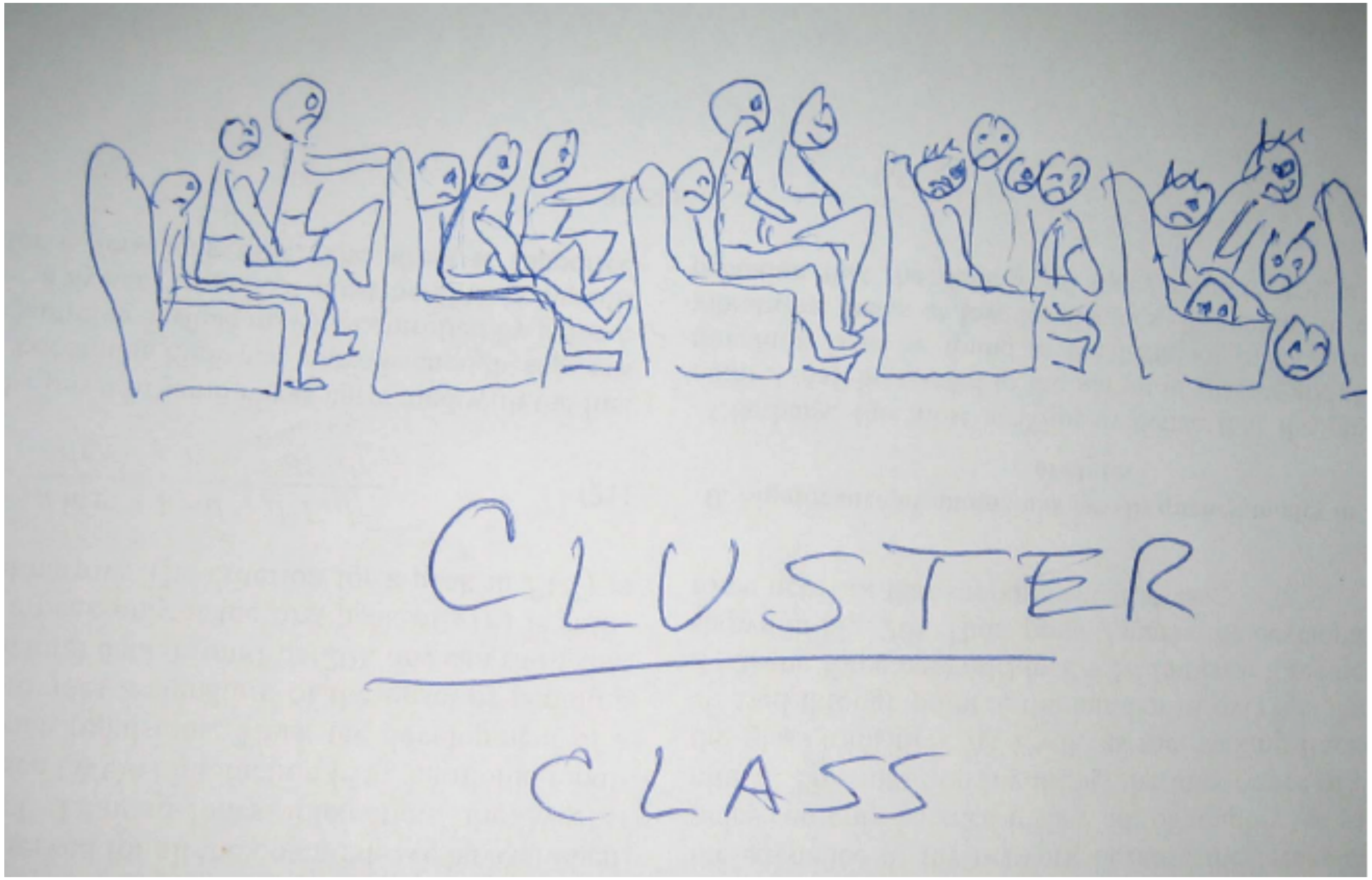
..airlines must increase the packing density of passengers.



They do this by decreasing the “lattice spacing”.
This is how most of us travel…

But if airlines ever find out about the this work…

…the future could be far worse:


CLUSTER
CLASS

Summed up:
determining phase diagram

- Which coexistence am I interested in?
  - gas-liquid, liquid-liquid ⇨ Gibbs
  - solids ⇨ thermodynamic integration, Widom

- Am I treating hard or soft systems?
  - thermodyn. int.: choice of reference system
  - Gibbs: can give wrong results!

  - Special care has to be taken whenever dealing with solids where particle number is not equal to amount of lattice sites