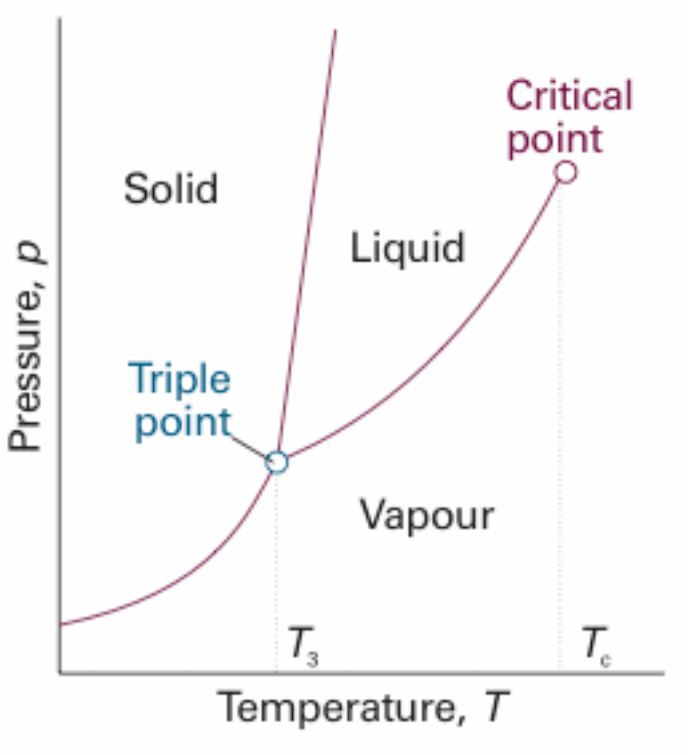
Ensembles II

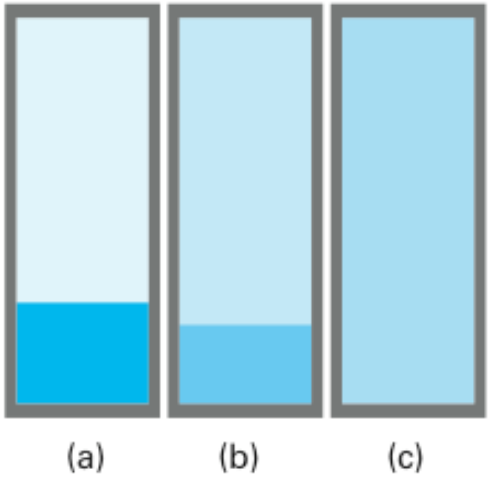
Free energies and phase equilibria 7
Chemical potentials 7.2
Gibbs ensemble 8

Phase behavior

- Simulations are good for predicting phase behavior
  - Prediction of thermodynamic stability of phases,
  - Coexistence lines
  - Critical points
  - Triple points
  - First order/second order phase transitions

Phase diagrams



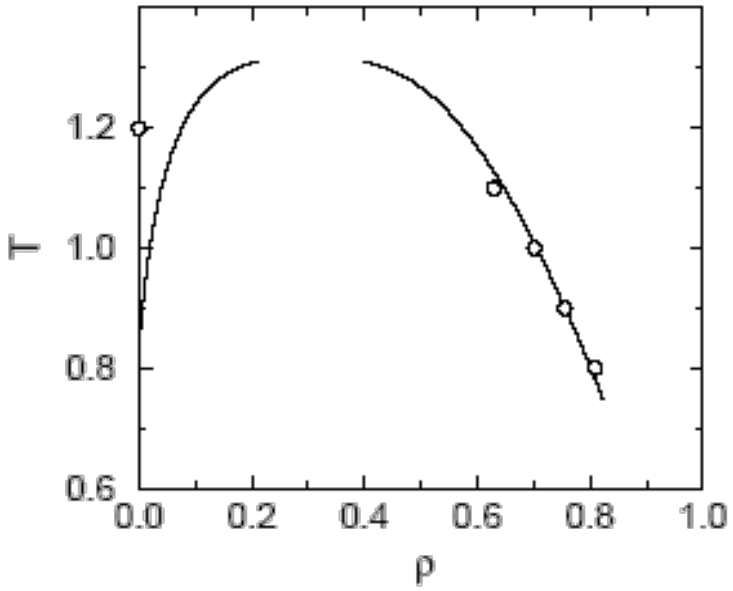


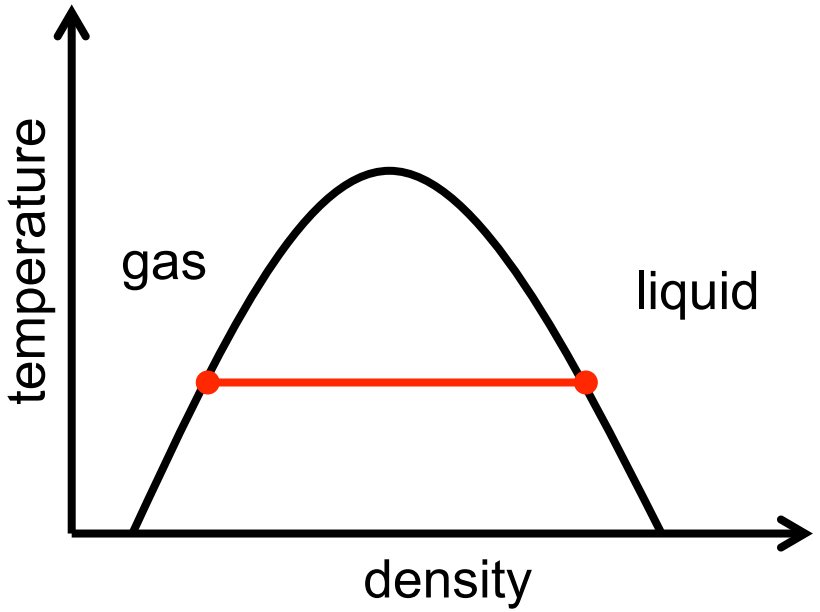
Critical point: no difference between liquid and vapor
Triple point: liquid, vapor and solid in equilibrium.

How do we compute these lines?

Phase diagram

Along the liquid gas coexistence line increasing the pressure and temperature at constant volume the liquid density becomes lower and the vapor density higher.





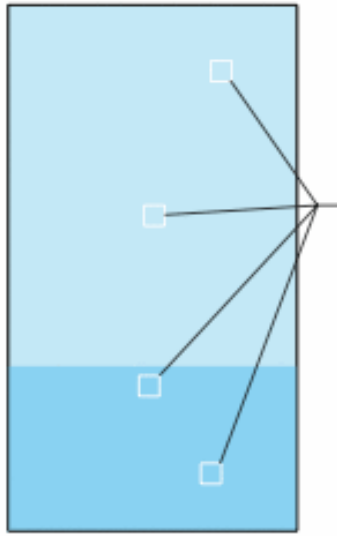
To determine the phase diagram of a given system we need to know the coexistence densities at given temperature (and pressure).

Easy in experiment, but what about simualtions?

Phase equilibrium

Chemical potential

$$\mu = \left(\frac{\partial F}{\partial N}\right)_{V,T} = \left(\frac{\partial G}{\partial N}\right)_{P,T} = G_m$$

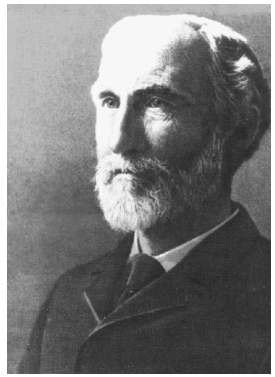


Identical chemical potential

Criterion for equilibrium (for single component)

$$T_\alpha = T_\beta \qquad P_\alpha = P_\beta \qquad \mu_\alpha = \mu_\beta$$

If $\mu_\alpha > \mu_\beta$ transport of particles from phase α to phase β.



Stable phase: lowest chemical potential

For single component stable phase has lowest Gibbs free energy G=μ

Relation between thermodynamic potentials

$$F = U - TS$$

$$G = F + PV$$

Suppose we have F(n,V,T)

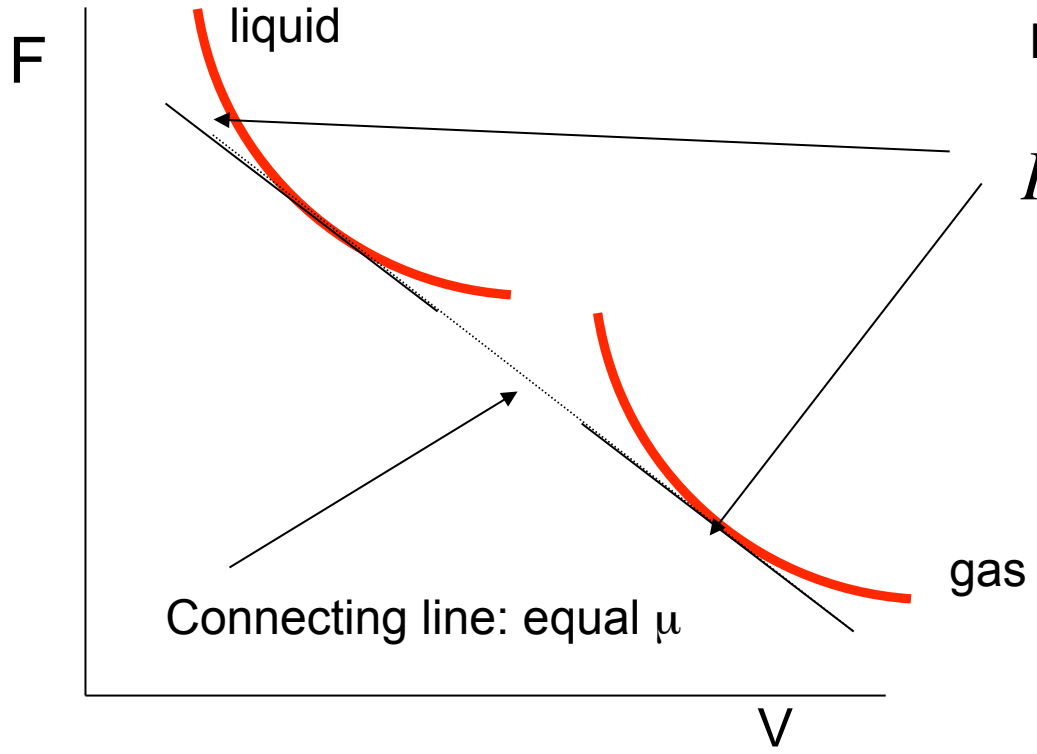
Then we can find G from F by realizing $P = -\left(\frac{\partial F}{\partial V}\right)_{n,T}$

$$G = F - V\left(\frac{\partial F}{\partial V}\right)_{n,T}$$

All thermodynamic quantities can be derived from F and its derivatives

Common tangent construction

- Phase equilibria follows from F(V,T)

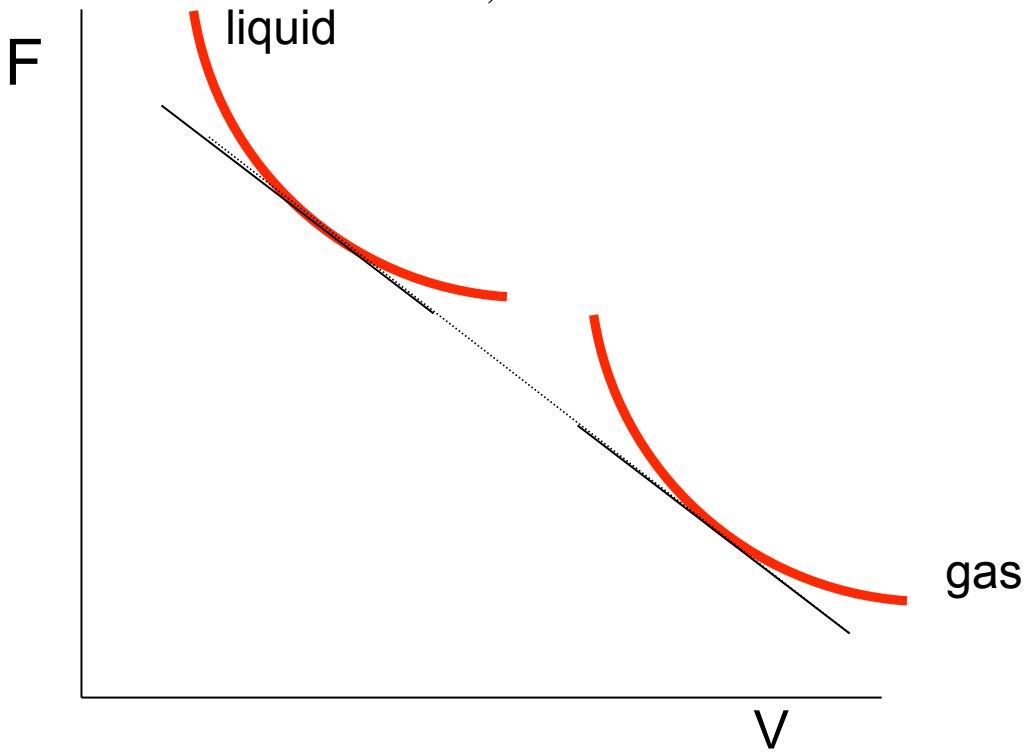


Equal tangents

$$P = -\left(\frac{\partial F}{\partial V}\right)_{n,T}$$

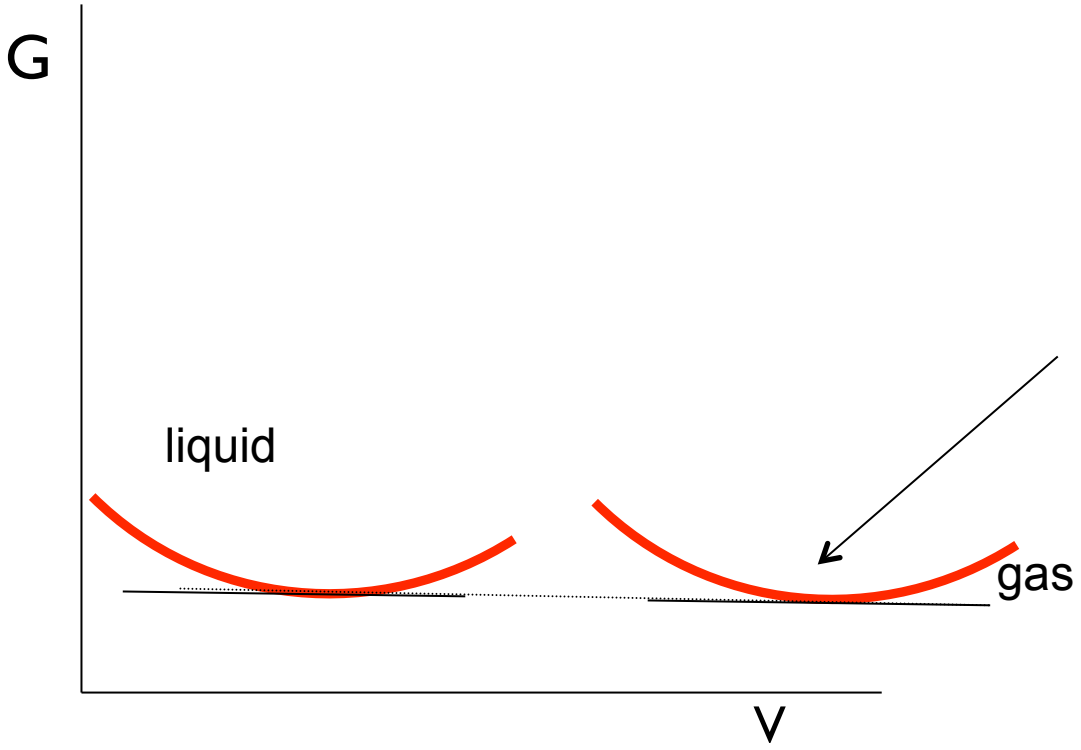
Common tangent construction

$$G = F - V\left(\frac{\partial F}{\partial V}\right)_{n,T}$$



Common tangent construction

$$G = F - V\left(\frac{\partial F}{\partial V}\right)_{n,T}$$



Both liquid and vapor G equal and minimal

Only equilibrium when P,T is on coexistence line.

We need F or μ

- So equilibrium from F(V) alone or from P and μ

$$F(V) = F(V_0) + \int_{V_0}^{V} \left(\frac{\partial \mathrm{F}}{\partial \mathrm{V}} \right)_{\mathrm{N,T}} \mathrm{dV} = F(V_0) - \int P \mathrm{dV}$$

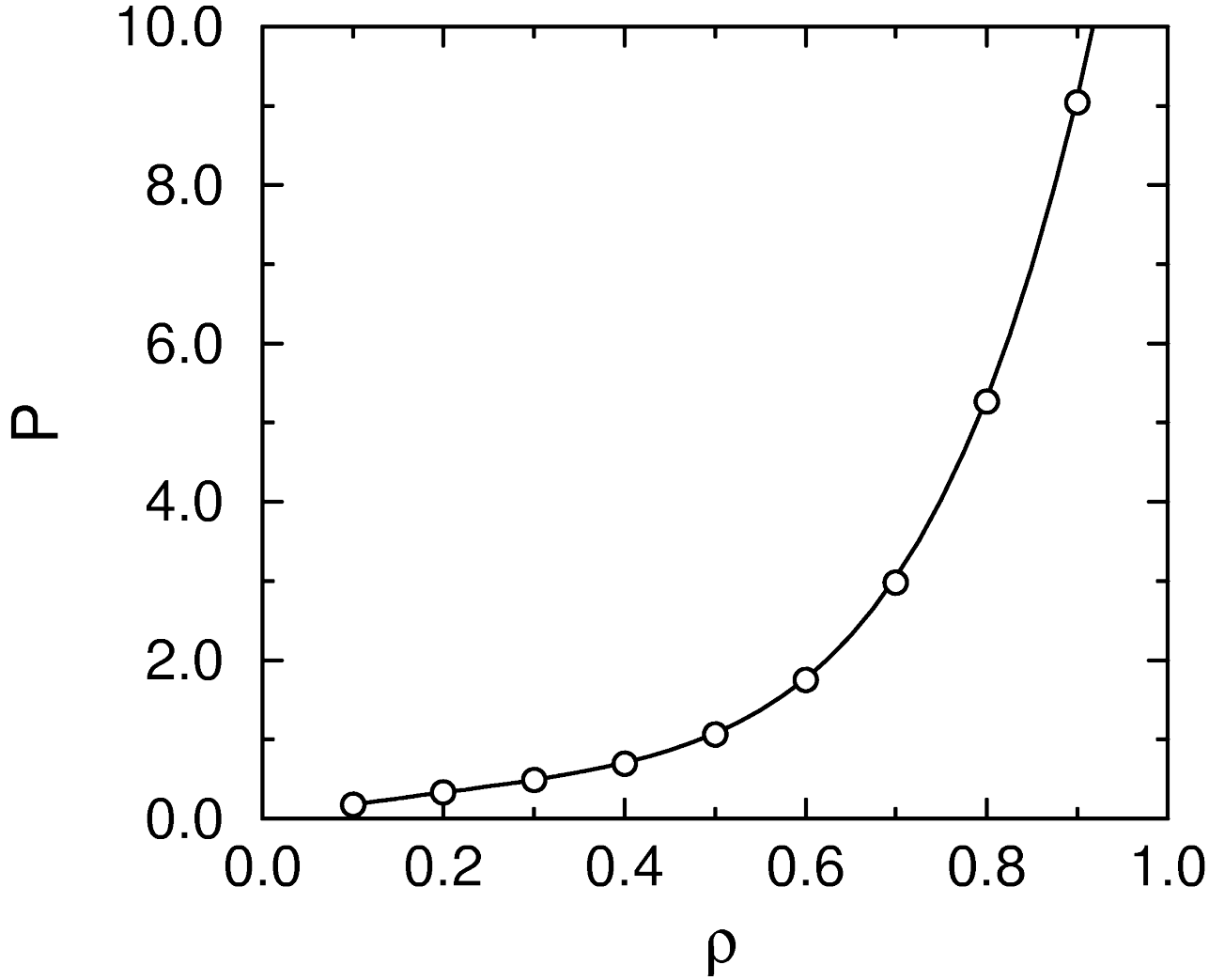
$$F(\rho) = F(\rho_0) + N \int_{\rho_0}^{\rho} \frac{P(\rho')}{\rho^2} \mathrm{d}\rho'$$

- So in fact for only 1 point of the equation of state the F is needed

- For liquid e.o.s even from ideal gas

$$\beta F(\rho)/N = \beta F^{id}(\rho)/N + \int_{0}^{\rho} \frac{\beta P(\rho') - \rho'}{\rho^2} \mathrm{d}\rho'$$

Equation of state



$$P = P(\rho, T)$$

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

$$F(\rho) = F(\rho_0) + N\int_{\rho_0}^{\rho} \frac{P(\rho')}{\rho^2}\,\mathrm{d}\rho'$$

Statistical Thermodynamics

Partition function

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

Ensemble average

$$\langle A \rangle_{NVT} = \frac{1}{Q_{NVT}} \frac{1}{\Lambda^{3N} N!} \int \mathrm{d}\mathbf{r}^N A\left(\mathbf{r}^N\right) \exp\left[-\beta U\left(\mathbf{r}^N\right)\right]$$

Probability to find a particular configuration

$$P\left(\mathbf{r}^N\right) = \frac{1}{Q_{NVT}} \frac{1}{\Lambda^{3N} N!} \int \mathrm{d}\mathbf{r}'^N \delta\left(\mathbf{r}'^N - \mathbf{r}^N\right) \exp\left[-\beta U\left(\mathbf{r}'^N\right)\right] \propto \exp\left[-\beta U\left(\mathbf{r}^N\right)\right]$$

Free energy

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

Problem: Q is in general inaccessible

Chemical potential

$$Q_{NVT} = \frac{V^N}{\Lambda^{3N} N!} \int ds^N \exp\left[-\beta U\left(s^N; L\right)\right]$$

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

$$= -\ln\left(\frac{V^N}{\Lambda^{3N} N!}\right) - \ln\left(\int ds^N \exp\left[-\beta U\right.\right.$$

$$= -N\ln\left(\frac{1}{\Lambda^3 \rho}\right) + N - \ln\left(\int ds^N \exp\left[-\beta U\left(s^N; L\right)\right]\right)$$

$$\beta F^{IG} = N\left[\ln\left(\Lambda^3 \rho\right) - 1\right]$$

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

$$= \ln\left(\Lambda^{3N} \rho\right)$$

$$\beta\mu^{IG} = \beta\mu^0 + \ln\left(\rho\right)$$

$$\left.\begin{aligned} \beta F &= \beta F^{IG} + \beta F^{ex} \\ \mu &\equiv \left(\frac{\partial F}{\partial N}\right)_{V,T} \end{aligned}\right\} \quad \begin{aligned} \beta\mu &= \beta\mu^{IG} + \beta\mu^{ex} \\ \beta\mu^{IG} &\equiv \left(\frac{\partial \beta F^{IG}}{\partial N}\right)_{V,T} \quad \beta\mu^{ex} \equiv \left(\frac{\partial \beta F^{ex}}{\partial N}\right)_{V,T} \end{aligned}$$

Widom test particle insertion

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

$$\beta\mu = \frac{\beta F(N+1)) - \beta F(N))}{N+1-N}$$

$$= -\ln\frac{Q(N+1)}{Q(N)}$$

$$= -\ln\left(\frac{\frac{V^{N+1}}{\Lambda^{3N+3}(N+1)!}}{\frac{V^N}{\Lambda^{3N}N!}}\right) - \ln\left(\frac{\int ds^{N+1}\exp\left[-\beta U\left(s^{N+1};L\right)\right]}{\int ds^{N}\exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

$$= -\ln\left(\frac{V}{\Lambda^3(N+1)}\right) - \ln\left(\frac{\int ds^{N+1}\exp\left[-\beta U\left(s^{N+1};L\right)\right]}{\int ds^{N}\exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

$$\beta\mu = \beta\mu^{IG} + \beta\mu^{ex}$$

$$\beta\mu^{ex} = -\ln\left(\frac{\int ds^{N+1}\exp\left[-\beta U\left(s^{N+1};L\right)\right]}{\int ds^{N}\exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

Widom test particle insertion

$$\beta\mu^{ex} = -\ln\left(\frac{\int ds^{N+1} \exp\left[-\beta U\left(s^{N+1};L\right)\right]}{\int ds^{N} \exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

$$U\left(s^{N+1};L\right) = \Delta U^{+} + U\left(s^{N};L\right)$$

$$\beta\mu^{ex} = -\ln\left(\frac{\int ds^{N} \int ds_{N+1} \exp\left[-\beta\left(\Delta U^{+} + U\left(s^{N};L\right)\right)\right]}{\int ds^{N} \exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

$$= -\ln\left(\frac{\int ds_{N+1} \int ds^{N} \left\{\exp\left[-\beta\Delta U^{+}\right]\right\} \exp\left[-\beta U\left(s^{N};L\right)\right]}{\int ds^{N} \exp\left[-\beta U\left(s^{N};L\right)\right]}\right)$$

$$= -\ln\left(\int ds_{N+1} \left\langle \exp\left[-\beta\Delta U^{+}\right]\right\rangle_{NVT}\right)$$

Ghost particle!

Algorithm 16 (Widom Test Particle Insertion)

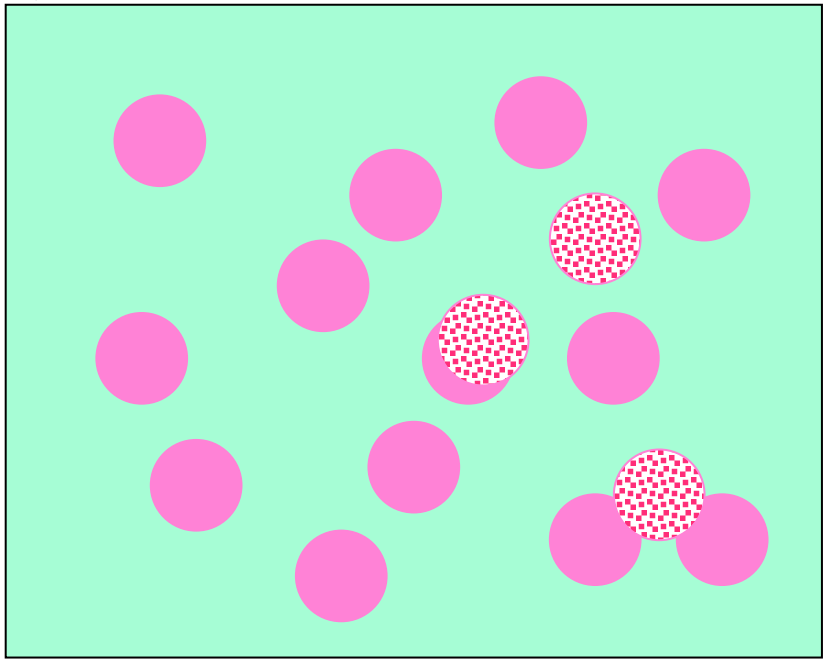
```
 subroutine Widom                    excess chemical potential
                                     via the addition of test particles
 xtest=box*ranf()                    generate a random position
 call ener(xtest,entest)             determine energy
 wtest=wtest                         update Boltzmann factor in (7.2.5)
+       +exp(-beta*entest)
 return
 end
```

Hard spheres

$$\beta\mu^{ex} = -\ln\left(\int \mathrm{ds}_{N+1} \left\langle \exp\left[-\beta\Delta U^{+}\right]\right\rangle_{NVT}\right)$$

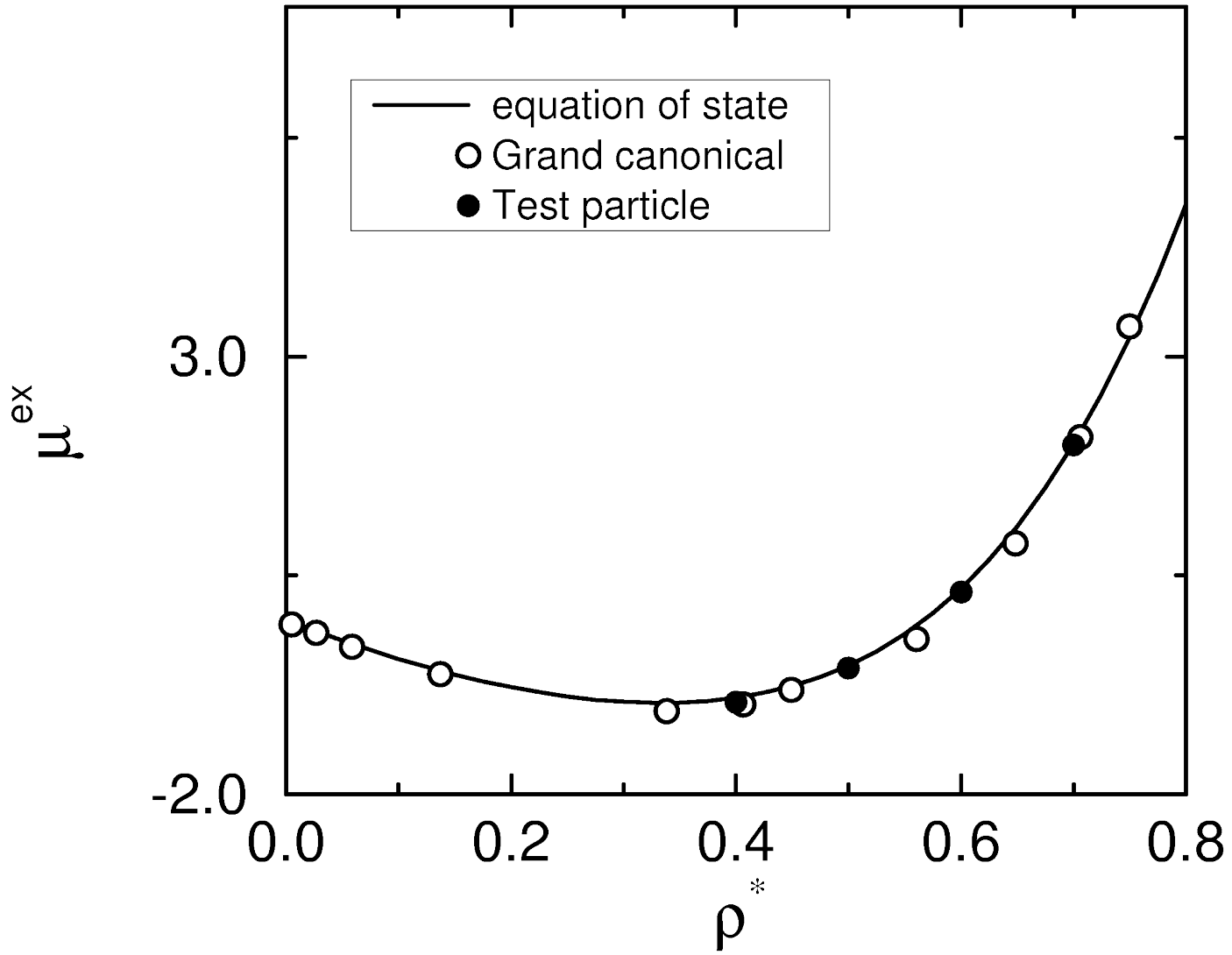
$$U(r) = \begin{cases} \infty & r \le \sigma \\ 0 & r > \sigma \end{cases}$$

$$\left\langle \exp\left[-\beta\Delta U^{+}\right]\right\rangle = \begin{cases} 0 & \text{if overlap} \\ 1 & \text{no overlap} \end{cases}$$



Probability to insert a test particle!

Lennard-Jones fluid



Other ensembles: NPT

NVT: Helmholtz free energy

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

NPT: Gibbs free energy

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

$$\beta G = -\ln\left(Q_{NPT}\right) \qquad Q_{NPT} = \frac{1}{\Lambda^{3N} N!}\int \mathrm{d}V V^N \exp\left(-\beta V P\right)\int \mathrm{d}s^N \exp\left[-\beta U\left(s^N; L\right)\right]$$

$$\beta\mu = \frac{\beta G\left(N+1\right) - \beta G\left(N\right)}{N+1-N} \qquad \beta\mu = -\ln\frac{Q\left(N+1\right)}{Q\left(N\right)}$$

$$\beta\mu = -\ln\frac{\frac{1}{\Lambda^{3N+3}\left(N+1\right)!}\left(\int \mathrm{d}V V^{N+1}\exp\left(-\beta V P\right)\int \mathrm{d}s^{N+1}\exp\left[-\beta U\left(s^{N+1};L\right)\right]\right)}{\frac{1}{\Lambda^{3N} N!}\left(\int \mathrm{d}V V^{N}\exp\left(-\beta V P\right)\int \mathrm{d}s^{N}\exp\left[-\beta U\left(s^{N};L\right)\right]\right)}$$

$$\beta\mu = -\ln$$

The volume fluctuates!

$$\left\langle \frac{\beta P V}{N+1}\int \mathrm{d}s_{N+1}\exp\left(-\beta\Delta U^{+}\right)\right\rangle \neq \left\langle \frac{\beta P V}{N+1}\right\rangle\left\langle \int \mathrm{d}s_{N+1}\exp\left(-\beta\Delta U^{+}\right)\right\rangle$$

$$\beta\mu = \ln$$

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

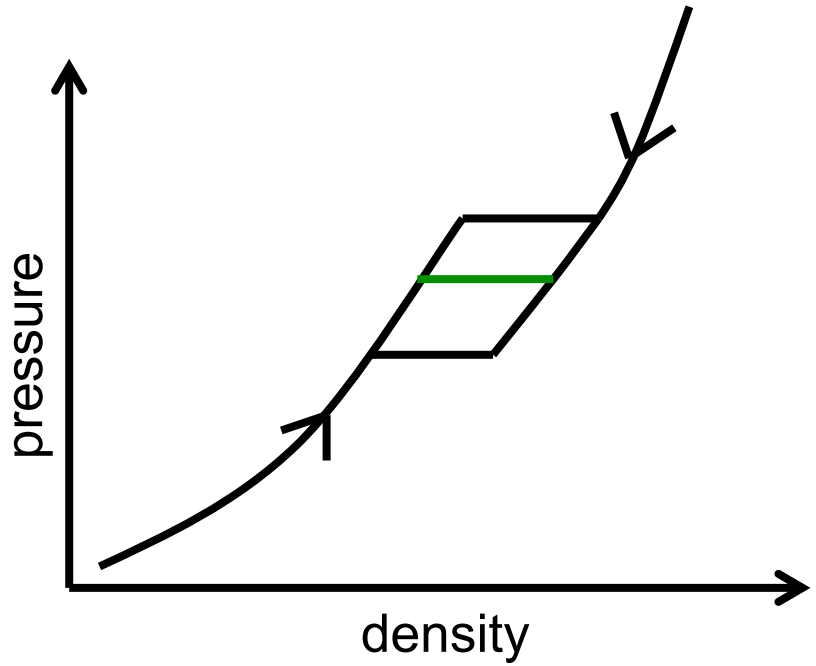
NVT:

$$\beta\mu = \beta \ln(\rho) - \ln\left\langle \int \mathrm{ds}_{N+1} \exp\left[-\beta\Delta U^{+}\right]\right\rangle_{NVT}$$

NPT:

$$\beta\mu = \ln\left(\Lambda^{3}\beta P\right) - \ln\left\langle \frac{\beta PV}{N+1} \int \mathrm{ds}_{N+1} \exp\left(-\beta\Delta U^{+}\right)\right\rangle$$

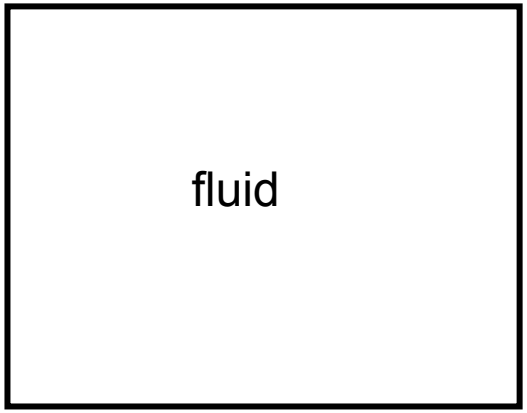
Phase equilibrium and hysteresis



solution : determine equilibrium from chemical potentials

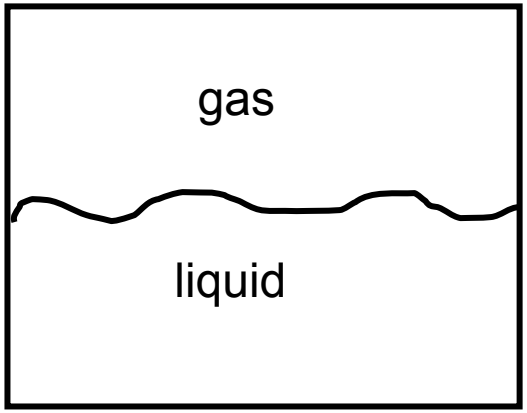
Idea: can’t we compute the equilibrium directly ?

NVT Ensemble



Let‘s lower the temperature…

NVT Ensemble



Problem:

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

leads to bad estimate of phase coexistence

Possible solution #1

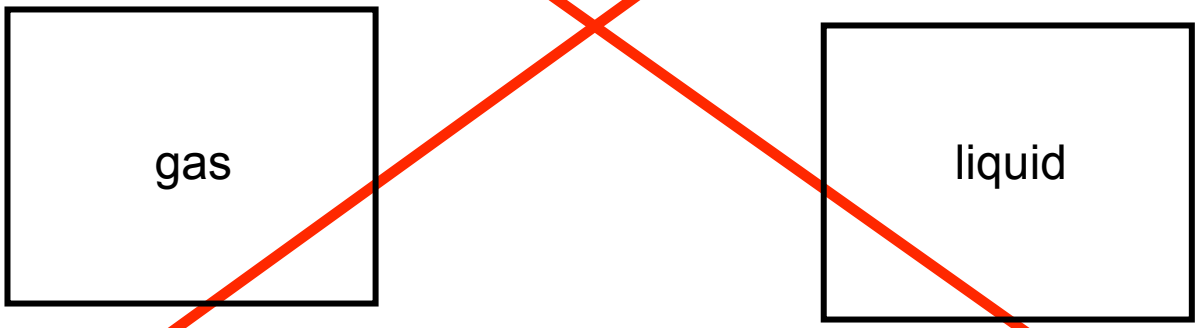
larger systems:

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

If we could compute liquid and gas in two separate simulations at constant μ, P and T

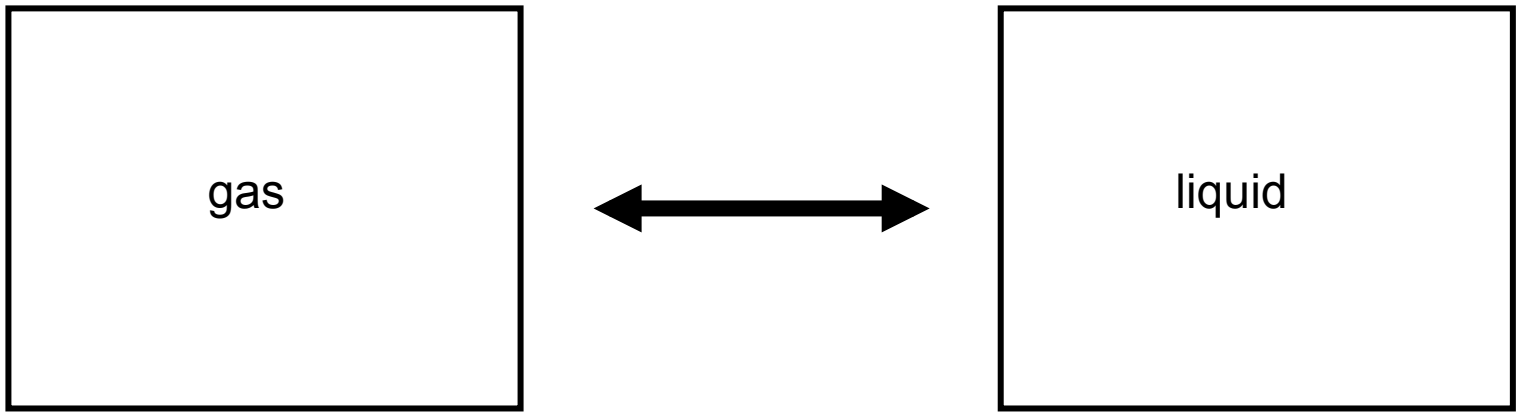


Problem: no such ensemble exists!

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

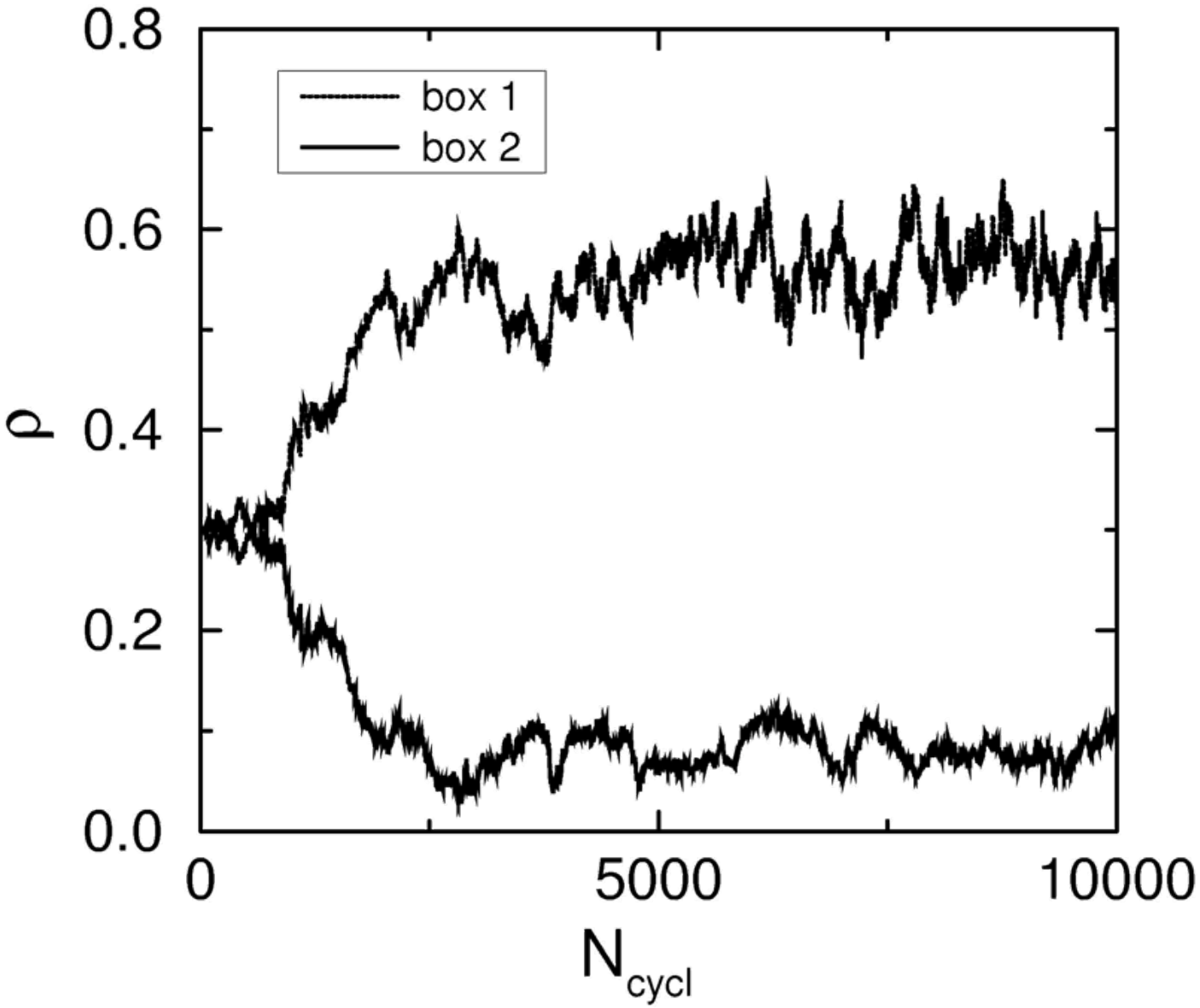
We have to fix at least one extensive variable (such as N or V)

Possible solution #3: The Gibbs ensemble



achieve equilibrium
by coupling them

A. Z. Panagiotopoulos, 1987.

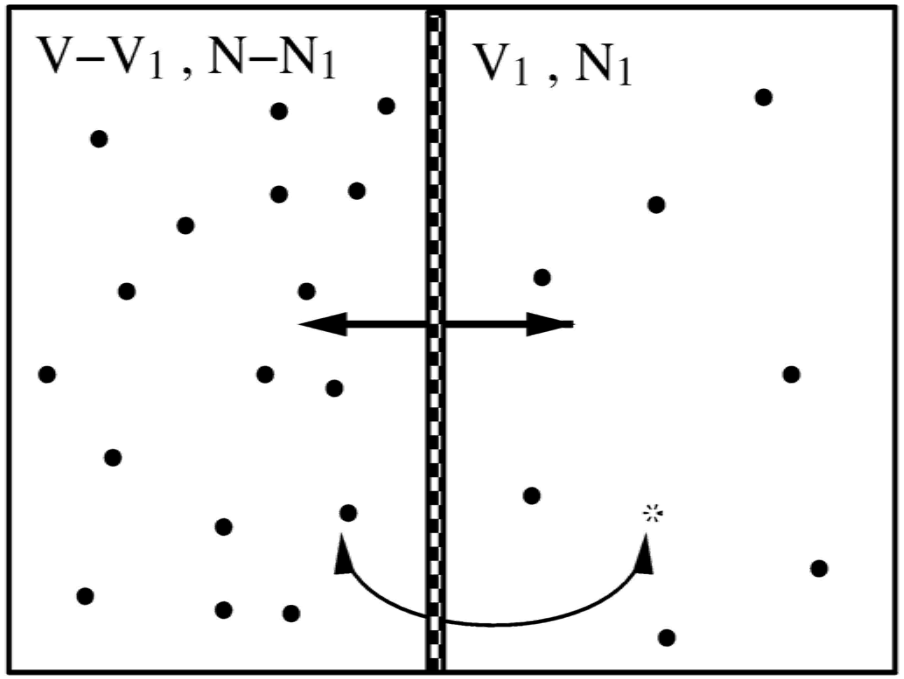

0.8
0.6
0.4
0.2
0.0
ρ
box 1
box 2
0
5000
10000
N_cycl

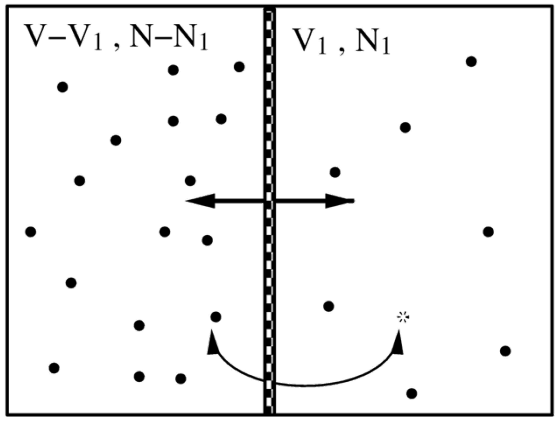
Overall system: NVT ensemble

$$N = N_1 + N_2$$

$$V = V_1 + V_2$$

$$T_1 = T_2$$

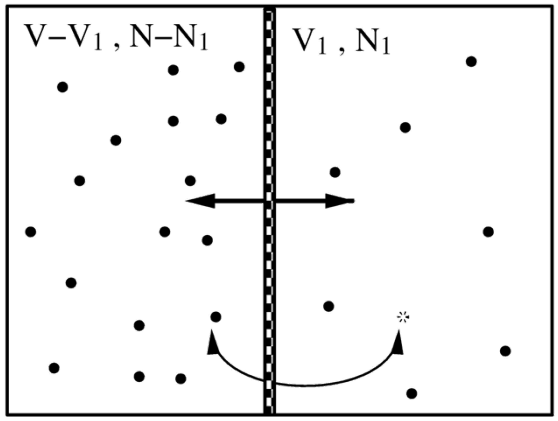




- distribute N_1 particles
- change the volume V_1
- displace the particles

partition function:

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

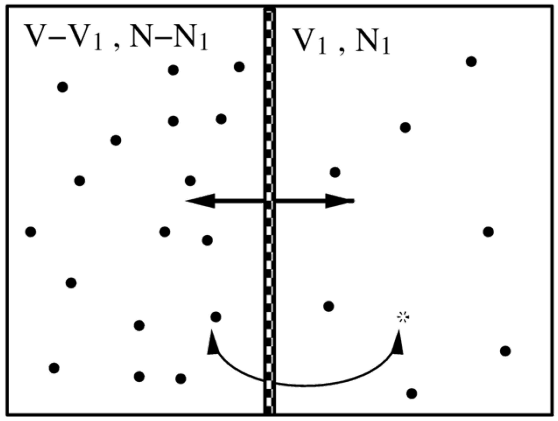


partition function:

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

Distribute N_1 particles over two volumes:

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

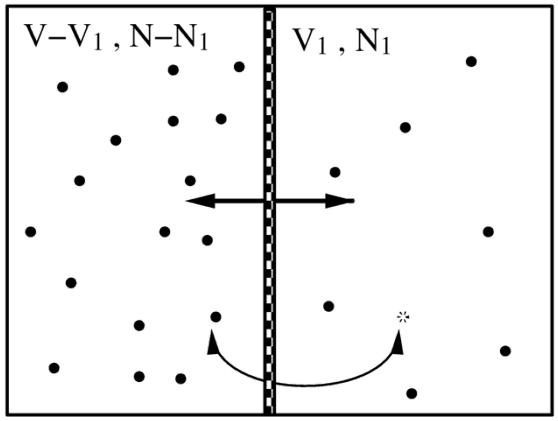


partition function:

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

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

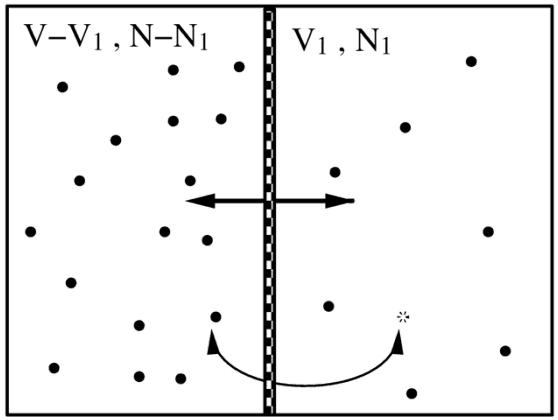
Integrate volume V_1



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



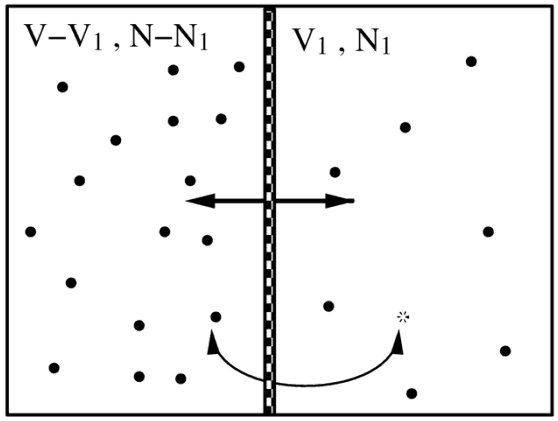
partition function:

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

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

Displace the particles in box 1 and box2

scaled coordinates in [0,1)



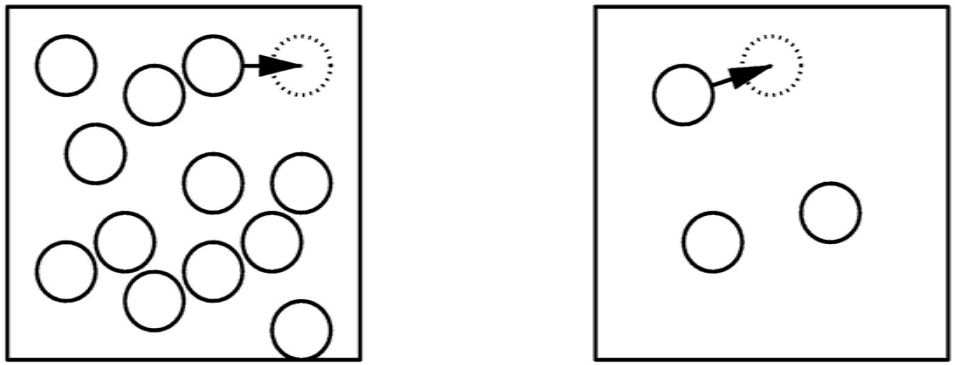
partition function:

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

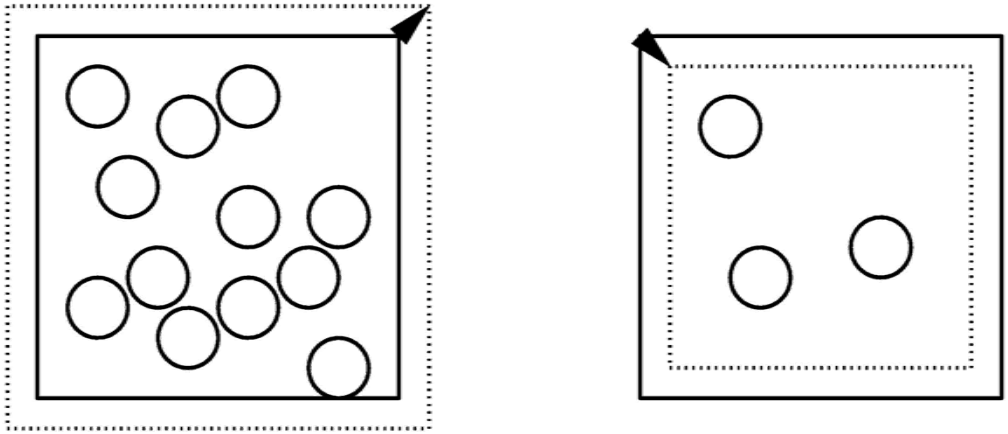
probability distribution:

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

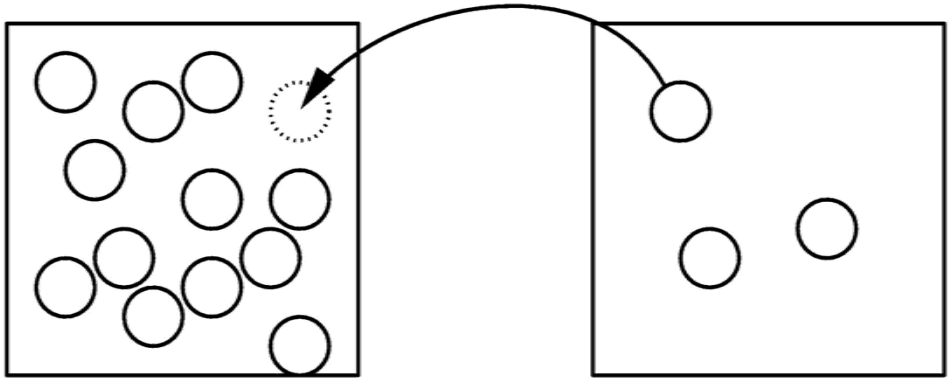
3 different kinds of trial moves:



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:

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

$$\mathcal{N}(o) \times \alpha(o \to n) \times \mathrm{acc}(o \to n) = \mathcal{N}(n) \times \alpha(n \to o) \times \mathrm{acc}(n \to o)$$

$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\mathcal{N}(n) \times \alpha(n \to o)}{\mathcal{N}(o) \times \alpha(o \to n)}$$

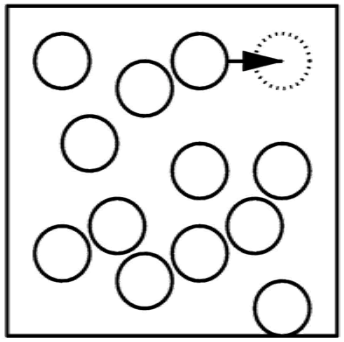
$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\mathcal{N}(n)}{\mathcal{N}(o)}$$

Displacement of a particle in box1

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

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

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



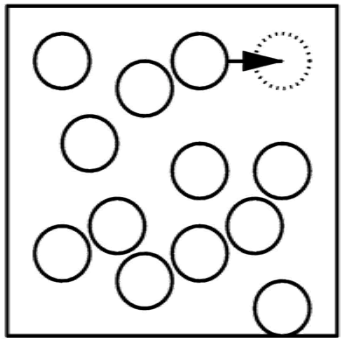
$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\cancel{\frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}} \exp\left\{-\beta\left[U(n) + \cancel{U(s_2^{N-N_1})}\right]\right\}}{\cancel{\frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!}} \exp\left\{-\beta\left[U(o) + \cancel{U(s_2^{N-N_1})}\right]\right\}}$$

Displacement of a particle in box1

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

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

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

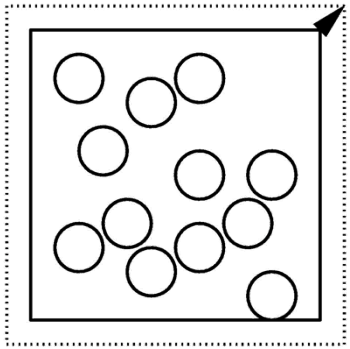


$$\frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{\exp\{-\beta U_1(n)\}}{\exp\{-\beta U_1(o)\}}$$

Volume change

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

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



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

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

Volume change

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

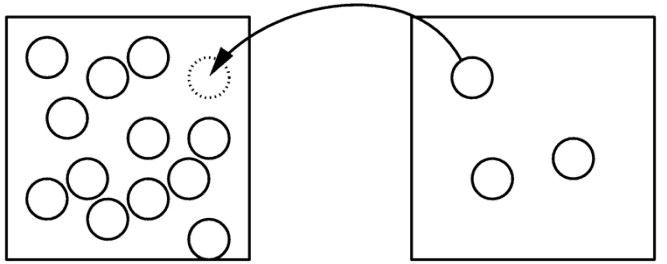
$$Q_G(N,V,T) = \frac{1}{\Lambda^{3N} N!} \sum_{N_1=0}^{N} \left(\begin{array}{c} N \\ N_1 \end{array} \right) \times \int_{-\infty}^{\infty} \mathrm{d} \ln \left(\frac{V_1}{V-V_1} \right) \frac{V_1(V-V_1)}{V} V_1^{N_1} (V-V_1)^{N-N_1} \times \int \mathrm{d}s_1^{N_1} \exp\left[-\beta U(s_1^{N_1})\right]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

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

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

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

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

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

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

Moving a particle from box1 to box2

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

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

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

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)           detailed balance!!!
  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
```

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

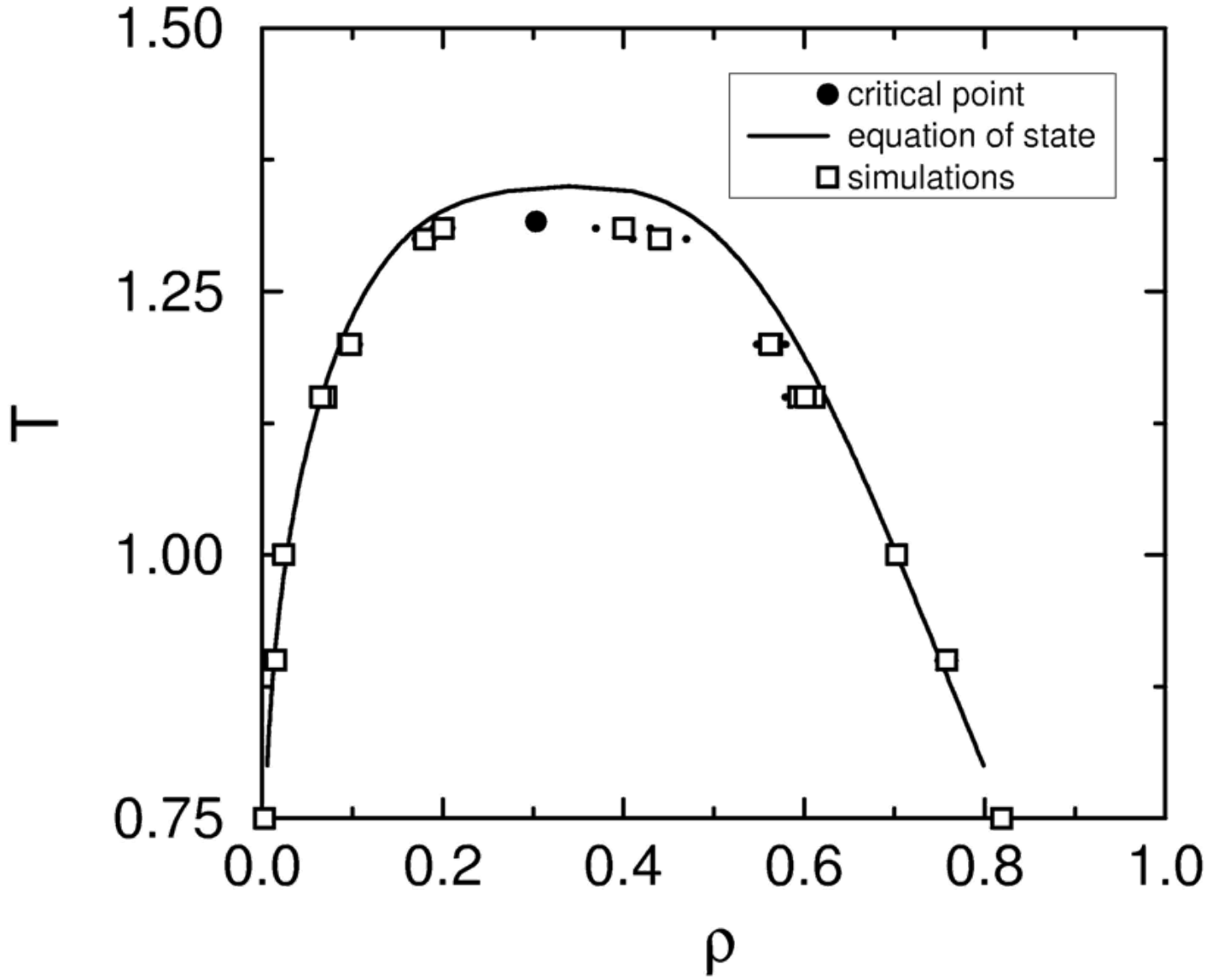
```
 SUBROUTINE mcvol                          attempt to change
                                           the volume
 call toterg(box1,en1o)                    energy old conf. box 1
 call toterg(box2,en2o)                    and 2 (box1: box length)
 vo1=box1**3                               old volume 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)+
+ (npbox(1)+1)*log(v1n/v1o)/beta)          appropriate weight function
```

```
 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)+
+ (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
```

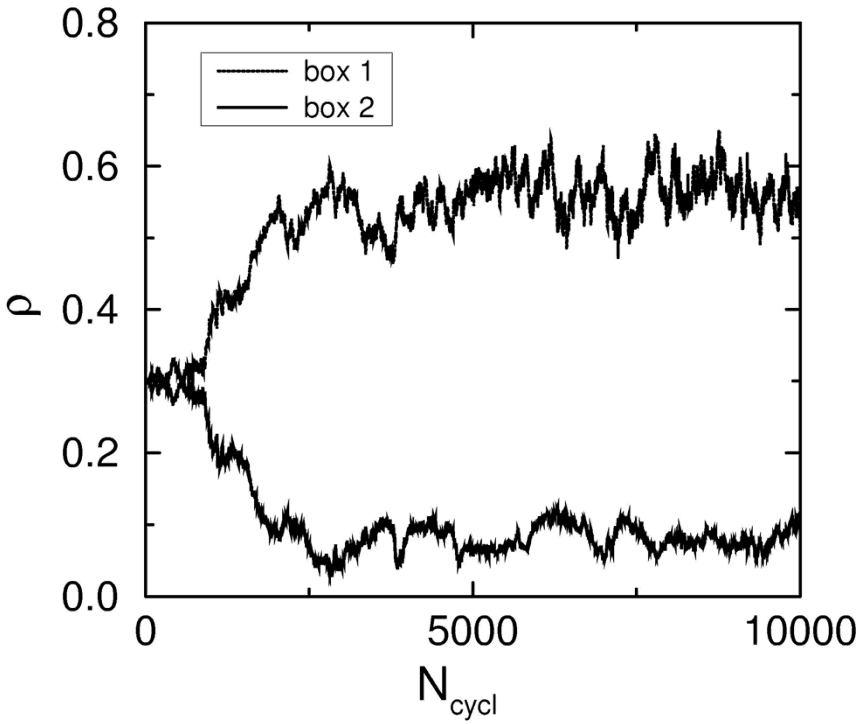
Algorithm 19 (Attempt to Swap a Particle between the Two Boxes)

```
 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))
 if (ranf().lt.arg) then
   x(o)=xn                                  add new particle to box in
```

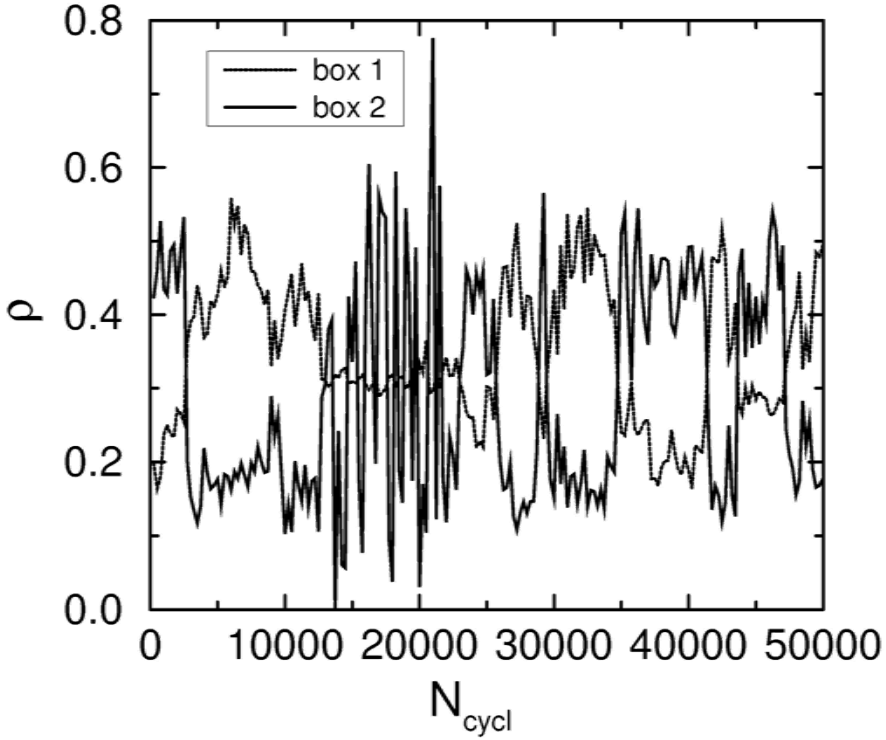
```
    out=2
  else
    in=2
    out=1
  endif
  xn=ranf()*box(in)                    new particle at a random position
  call ener(xn,enn,in)                 energy new particle in box in
  w(in)=w(in)+vol(in)*                 update chemical potential (8.3.5)
 + exp(-beta*enn)/(npbox(in)+1)
  if (npbox(out).eq.0) 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))
  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)

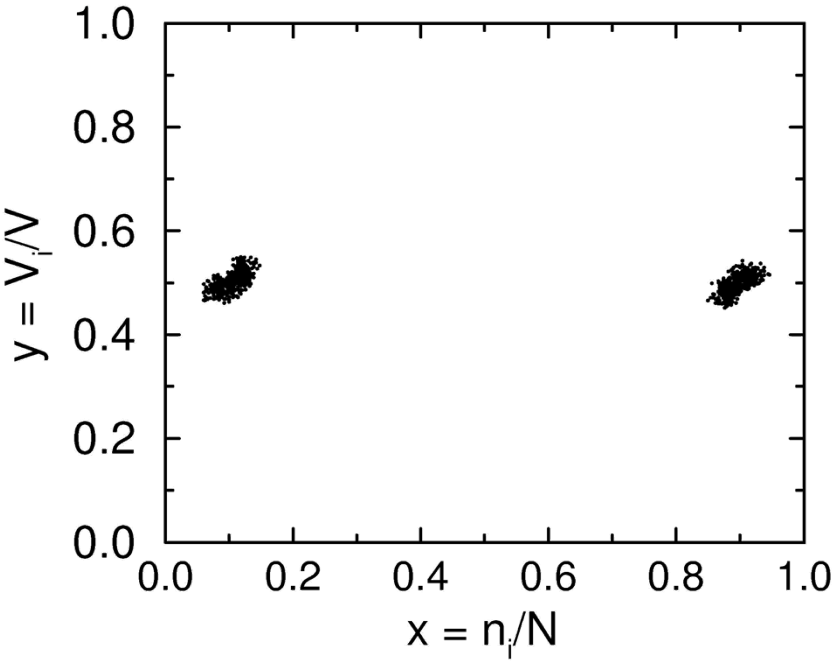


well below T_c

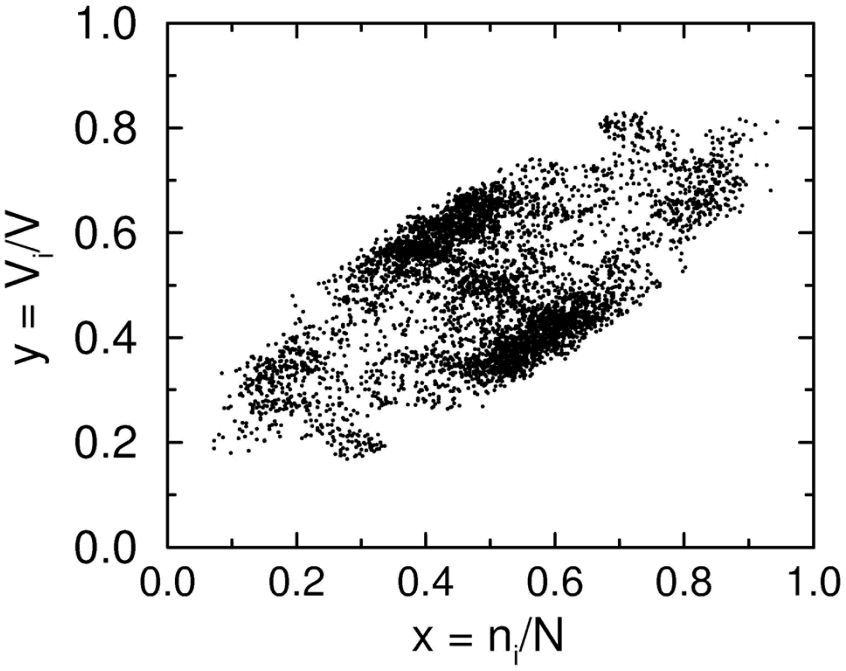


approaching T_c

Analyzing the results (2)

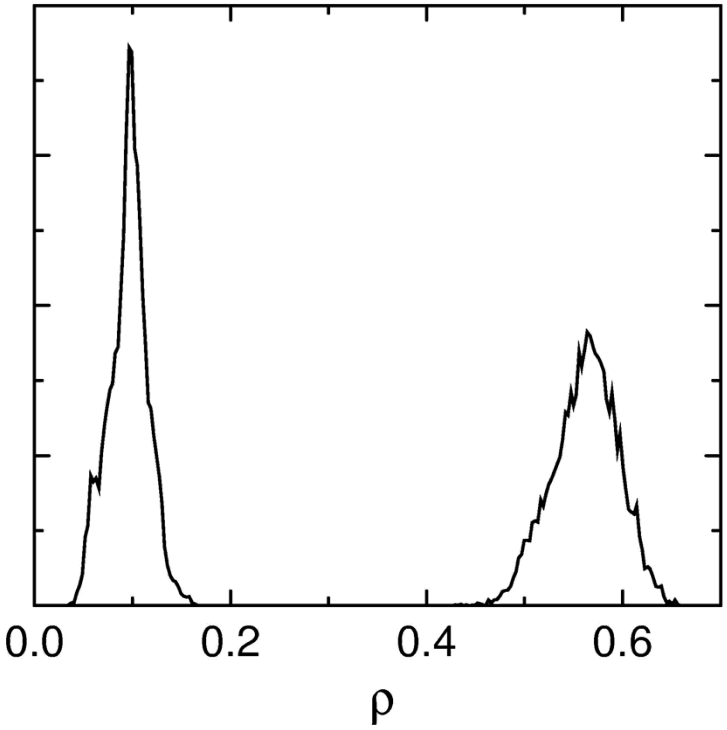


well below T_c

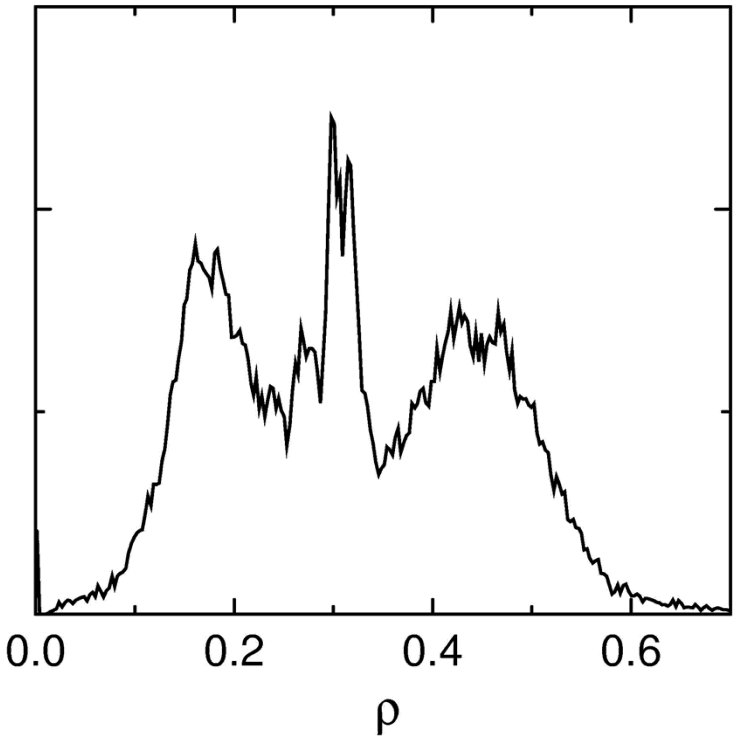


approaching T_c

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!)