RASPA Workshop/School Exercises Day 3

4 September 2024

Exercise 1: Gibbs simulation of Vapor-Liquid Equilibrium (VLE) of CO_2

The Gibbs ensemble is a way of computing coexistence without interfaces. It is one the most used methods to study vapor-liquid and liquid-liquid equilibria. It is not suitable for very dense systems. The conditions for coexistence of two or more phases I, II, ... is that the pressure and temperature of all the phases must be equal, as well as the chemical potential of all the species. The Gibbs ensemble example for the single component CO_2 is listed below. Two boxes will be used, one will correspond to the liquid phase, the other one to the gas phase. The 'GibbsVolumeChange' move changes the individual volume leaving the total volume in tact, the 'GibbsSwap' move swaps particles from one box to the other. One of the practical problems is to make sure both boxes remain larger than twice the cutoff length. If not, the program will exit with an error message, and the simulation should be restarted with a bigger volume. Note that RASPA uses orientational biased insertions for small rigid molecules like CO_2. For this example about 10000-20000 cycles are needed to equilibrate properly and about 20000-50000 cycles to have decent statistics.
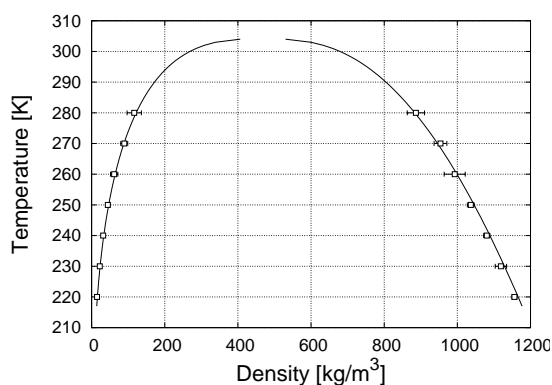

Figure 1: Gibbs ensemble simulation of CO2 at 250K. Two simulation boxes are used: one for the gas-branch and one for the liquid branch. The simulation can only be conducted below a certain temperature because otherwise the boxes can swap between gas and liquid. At 250K, the boxes are initialized with an equal amount of molecules, but soon split into gas and liquid. The average densities are straightforward to measure. As shown, the TraPPE model for CO2 does a good job when compare to experimental data (NIST database).

The input for a Gibbs-ensemble simulation looks like:

{

```
  "SimulationType" : "MonteCarlo",
  "NumberOfCycles" : 200,
  "NumberOfInitializationCycles" : 0,
  "PrintEvery" : 50,

  "Systems" :
  [
    {
      "Type" : "Box",
      "BoxLengths" : [30.0, 30.0, 30.0],
      "ExternalTemperature" : 240.0,
      "ChargeMethod" : "Ewald",

      "OutputPDBMovie" : true,
      "SampleMovieEvery" : 1,

      "GibbsVolumeMoveProbability" : 0.01
    },
    {
      "Type" : "Box",
      "BoxLengths" : [30.0, 30.0, 30.0],
      "ExternalTemperature" : 240.0,
      "ChargeMethod" : "Ewald",

      "OutputPDBMovie" : true,
      "SampleMovieEvery" : 1,

      "GibbsVolumeMoveProbability" : 0.01
    }
  ],

  "Components" :
  [
    {
      "Name" : "CO2",
      "TranslationProbability" : 0.5,
      "RotationProbability" : 0.5,
      "ReinsertionProbability" : 0.5,
      "GibbsSwapProbability" : 1.0,
      "CreateNumberOfMolecules" : [150, 150]
    }
  ]
}
```

Questions:

1. Examine the simulation input. Why do you need *two* boxes?

2. The simulation takes too long to run from start to finish. Instead we are going to make a movie of what happens at the start. Run the simulation, and view the movies of the system. Load both movie-files into `iRASPA`. Re-run and re-view several times. What do you notice? Is the sequence of events always the same?

3. Why is Gibbs-Ensemble sampling difficult at low T and at high T?

4. What is a good strategy to compute the whole VLE curve?

Exercise 2: Helium void-fraction

To compute the excess adsorption the void fraction of a structure needs to be specified using 'HeliumVoidFraction [real]'. RASPA automatically uses an equation of state (default: Peng-Robinson) to compute the fugacities from the pressure and mol-fraction. It also computes the amount of excess molecules from this equation of state.

The void fraction is the empty space of a structure divided by the total volume. In experiment it is measured using helium, because helium does (almost) not adsorb. It would be consistent to also measure this fraction using helium at room temperature. In practice it is easily computed from Widom particle insertion as the void fraction corresponds to the new Rosenbluth weight (here $\langle e^{-\beta U} \rangle$).

The input for a void-fraction simulation looks like:

```
{
  "SimulationType" : "MonteCarlo",
  "NumberOfCycles" : 100000,
  "NumberOfInitializationCycles" : 1000,
  "PrintEvery" : 5000,

  "Systems" : [
    {
      "Type" : "Framework",
      "Name" : "ITQ-29",
      "NumberOfUnitCells" : [2, 2, 2],
      "ChargeMethod" : "None",
      "CutOff": 11.8,
      "ExternalTemperature" : 300.0
    }
  ],

  "Components" : [
    {
      "Name" : "helium",
      "WidomProbability" : 1.0,
      "blockingPockets" : [
          [0.0,        0.0,        0.0,        4.0],
          [0.5,        0.0,        0.0,        0.5],
          [0.0,        0.5,        0.0,        0.5],
          [0.0,        0.0,        0.5,        0.5]
        ],
      "CreateNuMberofmolecules" : 0
    }
  ]
}
```

Questions:

1. Why is, *in this case*, no initialization needed?

2. Re-examine the ITQ-29-type structure. Make a guess of how much of the structure is void. Compute the value by running the code. Find the Helium void fraction in the output-file by searching for the `Rosenbluth weight` and average `Rosenbluth weight`. What percentage of the framework is void?

3. Switch off the blocking of the small β-cages. What (wrong) answer do you get?

Exercise 3: Adsorption isotherm of CO_2 in ITQ-29

Adsorption isotherms can be easily obtained by specifying a list of (increasing) pressures which will be subsequently run. If no `FugacityCoefficient` keyword is specified these pressure are converted to fugacity using the Peng-Robinson equation of state. Important: it is essential to specify the 'ideal gas Rosenbluth weight' for a component. This value needs to be computed separately and depends only on temperature (see auxiliary examples in the RASPA manual). This value is the reference state of the ideal gas. It is convenient to specify it in advance and it only depends on temperature, otherwise the correct pressure needs to deduced afterwards and is different from the specified input. For mixtures this becomes cumbersome when the ideal

gas Rosenbluth weight of the components is different. It is also convenient to specify the 'void fraction' of the materials (probed with helium) in advance (see previous exercise). If you do, the excess adsorption is automatically computed properly. At high pressures and temperatures the excess adsorption can be substantially lower than absolute adsorption.

The input for a grand-canonical ensemble simulation looks like:

```
{
  "SimulationType" : "MonteCarlo",
  "NumberOfCycles" : 100000,
  "NumberOfInitializationCycles" : 5000,
  "PrintEvery" : 1000,

  "Systems" : [
    {
      "Type" : "Framework",
      "Name" : "ITQ-29",
      "NumberOfUnitCells" : [2, 2, 2],
      "CutOff" : 11.8,
      "HeliumVoidFraction" : 0.401506,
      "ChargeMethod" : "Ewald",
      "ExternalTemperature" : 300.0,
      "ExternalPressure" : 1e5,

      "ComputeNumberOfMoleculesHistogram": true,
      "MinimumRangeNumberOfMoleculesHistogram" : 0,
      "MaximumRangeNumberOfMoleculesHistogram" : 80,
      "SampleNumberOfMoleculesHistogramEvery" : 1,
      "WriteNumberOfMoleculesHistogramEvery" : 1000,

      "OutputPDBMovie" : false,
      "SampleMovieEvery" : 1
    }
  ],

  "Components" : [
    {
      "Name" : "CO2",
      "FugacityCoefficient" : 1.0,
      "IdealGasRosenbluthWeight" : 1.0,

      "TranslationProbability" : 0.5,
      "RotationProbability" : 0.5,
      "ReinsertionProbability" : 0.5,
      "SwapProbability" : 1.0,

      "blockingPockets" : [
          [0.0,        0.0,        0.0,        4.0],
          [0.5,        0.0,        0.0,        0.5],
          [0.0,        0.5,        0.0,        0.5],
          [0.0,        0.0,        0.5,        0.5]
        ],

      "CreateNuMberofmolecules" : 0
    }
  ]
}
```

Questions:

1. Experiments use *pressure*, simulation *chemical potentials* (the μVT-grand-canonical ensemble). How are pressure and the chemical potential related?

2. Why do you, in general, need to specify in the input a pre-computed `IdealGasRosenbluthWeight`?

3. Each simulation is at a different temperature and pressure. How would you go about computing an isotherm as a function of pressure?

4. Describe the (at least) two ways in which you can do this (Hint: how would you compute adsorption and how would you compute desorption?)

Questions:

5. Compute several (e.g. 2 or 3) points on the isotherm. Investigate the loading and the error in the loading. How can you reduce the error?

6. Compare the histogram of the number of adsorbates as a function of fugacity/pressure. Where is the distribution the broadest?

7. How do the block-averages behave and what do they tell you? (i.e. when is the simulation converged?)

8. Why is the excess loading always lower than the absolute adsorption? What happens with excess adsorption at very high loading? Can you explain this weird behavior?

9. Check the enthalpy of adsorption at low loading (computed from the fluctuation formula), does it match the result computed on day 1 using a single particle to compute the infinite dilution limit?

Go to the directory `exercise_3_adsorption_co2_in_itq-29/data`. Try to get the best fit by changing the initial values and refitting from the current values until convergence. In `gnuplot`, you can play around with the fitting:

```
f(x)=qA*bA*x**alphaA/(1.0+bA*x**alphaA) + qB*bB*x**alphaB/(1.0+bB*x**alphaB)
fit f(x) "data_CO2_300.data" us 1:($2/8) via qA,bA,alphaA,qB,bB,alphaB
```

Model	expression
Langmuir	f(x)=qA*bA*x/(1.0+bA*x)
dual Langmuir	f(x)=qA*bA*x/(1.0+bA*x) + qB*bB*x/(1.0+bB*x)
Langmuir-Freundlich	f(x)=qA*bA*x**alphaA/(1.0+bA*x**alphaA)
dual Langmuir-Freundlich	f(x)=qA*bA*x**alphaA/(1.0+bA*x**alphaA) + qB*bB*x**alphaB/(1.0+bB*x**alphaB)
n-sites models	. . .

Table 1: Various isotherm models.

Questions:

9. Use the precomputed adsorption data file, and try to fit several theoretical isotherm shapes (from Table 1): Langmuir, dual-Langmuir, Langmuir-Freundlich, dual-site Langmuir-Freundlich. Which one fits bets?

10. What does that tell you about the physics of adsorption?

Exercise 4: Adsorption isotherm of CO_2 in ITQ-29 using advanced methods

The acceptance probability in conventional Monte Carlo of the insertion/deletion move goes to zero at high density and/or low temperature. A new scheme to remedy this problem is the "Continuous Fractional Component Monte Carlo" (CFCMC) method of Shi and Maginn. The system is expanded with a "fractional" molecule that has a scaled interaction with the other particles and with the framework. The scaling parameter λ ranges from 0 to 1, with 0 meaning the particles is not felt, while 1 means the particle is fully present. In addition to the usual set of MC moves now also moves are performed on λ, attempting to increase and decrease it. Effectively, increasing λ corresponds to "inflating" the molecule, and decreasing λ corresponds to "deflating" the molecule. A change of λ greater than 1 leads to insertion. The fractional molecule is made integer, and a new fractional molecule is randomly inserted in the system with the remainder of λ. Similarly, a decrease of λ below zero leads to a deletion of a molecule. Further details will be provided in the methodology section. The point to note here, however, is that the λ moves can be biased, ideally making the λ-histogram flat. The method therefore is able to *force* molecules in and out of the system.

The input for CBMC/CFCMC adsorption of CO_2 in ITQ-29 is:

```
{
  "SimulationType" : "MonteCarlo",
  "NumberOfCycles" : 500000,
  "NumberOfEquilibrationCycles" : 250000,
  "PrintEvery" : 5000,

  "Systems" : [
    {
      "Type" : "Framework",
      "Name" : "ITQ-29",
      "NumberOfUnitCells" : [2, 2, 2],
      "CutOff" : 11.8,
      "HeliumVoidFraction" : 0.401506,
      "ChargeMethod" : "Ewald",
      "ExternalTemperature" : 300.0,
      "ExternalPressure" : 1e5,

      "ComputeNumberOfMoleculesHistogram": true,
      "MinimumRangeNumberOfMoleculesHistogram" : 0,
      "MaximumRangeNumberOfMoleculesHistogram" : 80,
      "SampleNumberOfMoleculesHistogramEvery" : 1,
      "WriteNumberOfMoleculesHistogramEvery" : 1000
    }
  ],

  "Components" : [
    {
      "Name" : "CO2",
      "FugacityCoefficient" : 1.0,

      "ThermodynamicIntegration" : true,

      "TranslationProbability" : 0.5,
      "RotationProbability" : 0.5,
      "ReinsertionProbability" : 0.5,
      "WdomProbability" : 1.0,
      "CFCMC_CBMC_SwapProbability" : 1.0,

      "blockingPockets" : [
          [0.0,        0.0,        0.0,        4.0],
          [0.5,        0.0,        0.0,        0.5],
          [0.0,        0.5,        0.0,        0.5],
          [0.0,        0.0,        0.5,        0.5]
        ],
```

```
        "CreateNuMberofmolecules" : 0
    }
  ]
}
```

So, `CFCMC_SwapProbability 1.0` and `CFCMC_CBMC_SwapProbability 1.0` for CBMC/CFCMC. An equilibration phase is additional used to measure the biasing factors that are needed in this methodology.

Start with making a movie of the start of the adsorption using a few hunderd cycles. Pay particular attention to the fractional molecule, which is in `iRASPA` colored white and scaled in size with the λ-parameter.
Questions:

1. Compute a single pressure point using CFCMC and CBMC/CFCMC. Do you get the same result?

2. What happens with the acceptance of the insertion/deletion move?

3. Are the histograms of the fluctuating number of molecules for conventional MC, CFCMC, and CBMC/CFCMC the same?

Exercise 5: Chemical potential of CH$_4$ in MFI using advanced methods

We are going to compare three different methods to compute chemical potentials:

1. The difference between $\lambda = 0$ and $\lambda = 1$ is the difference between a system with N and a system with $N + 1$ molecules. Hence, using

$$F(\lambda) = -k_B T \ln P(\lambda) \tag{1}$$

we can obtain the chemical potential in (CB-)CFCMC from

$$\mu^{\text{excess}} = -k_B T \ln \frac{\langle P(\lambda \uparrow 1) \rangle}{\langle P(\lambda \downarrow 0) \rangle} \tag{2}$$

2. Thermodynamic integration is a method used to calculate the difference in free energy between two given states. For example, to calculate the free energy of inserting a molecule into a system. The integration runs from state A, with $\lambda = 0$, to state B, with $\lambda = 1$, where the interactions of the inserted molecule are gradually scaled from 0 to that of a full molecule. The change in free energy between states A with $\lambda = 0$ and state B with $\lambda = 1$, i.e. the insertion of a particle, can be computed from the integral of the ensemble averaged derivatives of the potential energy over the coupling parameter λ

$$\mu^{\text{excess}} = \Delta F \left(A_{\lambda=0} \to B_{\lambda=1} \right) = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial U(\lambda)}{\partial \lambda} \right\rangle_{\lambda} d\lambda \tag{3}$$

This is performed by defining a potential energy function $U(\lambda)$, sampling the ensemble of equilibrium configurations at a series of λ values, calculating the ensemble-averaged derivative of $U(\lambda)$ with respect to λ at a series of λ values from 0 to 1, and finally computing the integral over the ensemble-averaged derivatives.

3. The Widom test-particle method is a general simulation technique to obtain the excess chemical potential μ of a fluid. Here the variable is the number of particles N to calculate

$$\mu^{\text{excess}} = \mu - \mu^{\text{ideal}} = -k_B T \log \left\langle e^{-U^+/KB_T i} \right\rangle \tag{4}$$

at constant T and V. This method is now extensively used in simulations of gases and fluids.

The input for the chemical potential measurement of CH_4 in MFI is:

```
{
  "SimulationType" : "MonteCarlo",
  "NumberOfCycles" : 50000,
  "NumberOfEquilibrationCycles" : 20000,
  "NumberOfInitializationCycles" : 5000,
  "PrintEvery" : 25000,

  "Systems" : [
    {
      "Type" : "Framework",
      "Name" : "MFI_SI",
      "NumberOfUnitCells" : [2, 2, 2],
      "CutOff" : 12.0,
      "HeliumVoidFraction" : 0.265280,
      "ChargeMethod" : "None",
      "ExternalTemperature" : 300.0,
      "ExternalPressure" : 1e5
    }
  ],

  "Components" : [
    {
      "Name" : "methane",
      "FugacityCoefficient" : 1.0,

      "ThermodynamicIntegration" : true,

      "TranslationProbability" : 0.5,
      "RotationProbability" : 0.5,
      "ReinsertionProbability" : 0.5,
      "WidomProbability" : 1.0,

      "CFCMC_CBMC_SwapProbability" : 1.0,

      "CreateNuMberofmolecules" : 0
    }
  ]
}
```

Since we, in this exercise, impose the chemical potential (via the fugacity), we know exactly what chemical potential (fugacity) we should measure.

Questions:
1. Run some short simulations at various pressure/temperatures. Compare the results of the three methods (see results written at the end of the output file). Which of these is the best method? which one the worst? Why?

2. Is it because of fugacity/pressure or something more fundamental?

3. Examine the pre-run results. Plot the λ-histograms and biasing as a function of λ.

4. What are the strengths of the expanded ensembles? and weaknesses?

5. For what systems are the expanded ensembles really essential to use?