"Understanding the Phase behavior (VLE) of propane using Gibbs Ensemble Monte Carlo Simulations"

Sai Harshit Balantrapu
Department of Chemical and Biomedical Engineering
University of Wyoming
Laramie, 82072, WY, USA
Email ID: hbalantr@uwyo.edu

***Abstract*— The adsorption and absorption properties of fluids in porous media are key to designing and improving numerous energy-related applications. Molecular simulations of these systems require accurate force fields that capture the involved chemical reactions and can describe the vapor–liquid equilibrium (VLE). This paper aims to explore the phase behavior of propane gas using the Gibbs ensemble Monte Carlo method with the generalized Amber force field (GAFF and validate it with traditional force fields, such as TRAPPE-UA. For validation, we compared the VLE behavior of propane with the TRAPPE-UA force field. The computed VLE data agree with the National Institute of Standards and Technology reference data. This paper showed the importance of stochastic sampling in obtaining the phase diagram to explore different configurations in the potential energy space.**

***Index Terms*— Statistical thermodynamics, Gibbs ensemble Monte Carlo Simulations, GAFF, VLE**

I. INTRODUCTION

Separating olefin and paraffin mixtures is one of the petrochemical industry's most energy-intensive and challenging chemical separations due to their low relative volatility and similar molecular sizes. [1] Cryogenic distillation is the best method for separating propane from propylene. However, due to their similar physical properties and low volatility coefficients this process requires a high reflux ratio, high energy, and prolonged separation columns.[2] Therefore, the process is not affordable due to its high energy consumption and operating costs. There are various processes, such as extractive distillation, membrane, hydrogenation, adsorption, and physical adsorption, to separate propane from propylene. However, adsorption is the most efficient separation process for this purpose. To study the adsorption processes efficiently using the computational approach, we need to understand the phase behavior of the gases. In this paper, we studied the phase behavior of propane gas using the Gibbs ensemble Monte Carlo simulations. Additionally, we solved the example of a stochastic differential equation in the form of potential energy to show how stochasticity plays an important role in sampling the potential energy by mapping to the Boltzmann distribution.

II. THEORY ON MONTE CARLO SIMULATIONS IN STATISTICAL THERMODYNAMICS

A. Metropolis Acceptance Criteria

Monte Carlo moves implemented in Cassandra preserve detailed balance between each pair of microstates m and n. The metropolis criteria :

$$\Pi_{m,n}\alpha_{m,n}p_m = \Pi_{n,m}\alpha_{n,m}p_n$$

where:

- $\Pi_{m,n}$ is the probability of accepting the move from microstate m to microstate n,
- $\alpha_{m,n}$ is the probability of attempting the move that will form n from m,
- p_m is the probability of m in the ensemble of interest.

The acceptance probability is given by:

$$\Pi_{m,n} = \min\left(1, \frac{\alpha_{n,m}p_n}{\alpha_{m,n}p_m}\right) \tag{1}$$

The ratio in Eq. (2) will often involve an exponential, e.g., $e^{-\beta\Delta U}$. To preserve precision in the energy calculation, the acceptance probability is computed as:

$$\Pi_{m,n} = \exp\left\{-\max\left[0, \ln\left(\frac{\alpha_{m,n}p_m}{\alpha_{n,m}p_n}\right)\right]\right\} \tag{2}$$

The logarithm, defined in code as `ln_pacc`, is tested in the function `accept_or_reject()`. If `ln_pacc` is greater than 0 and less than a maximum numerical value, $\Pi_{m,n}$ is computed and compared to a random number.

Algorithm 1: Acceptance or Rejection Logic for Monte Carlo Move

```
accept_or_reject = .FALSE. if ln_pacc ≤ 0.0_DP then
    accept_or_reject = .TRUE.
ln_pacc < max_kBT
pacc = DEXP(-ln_pacc) if rranf() ≤ pacc then
    accept_or_reject = .TRUE.
```

III. Implementation of the Langevin Monte Carlo

As seen in the above section regarding the metropolis acceptance criteria, in this section, we will be discussing a Langevin stochastic differential equation, which uses the approach of Monte Carlo to sample the potential energy in the landscape of the Boltzmann energy distribution.

New samples are proposed by simulating the Langevin Stochastic Differential Equation (SDE), which is given by

$$dX_t = -\nabla U(X_t)dt + \sqrt{2d}B_t,$$

where U is the potential function and B_t is the standard Brownian motion.

Example

Let $U(z) = \frac{1}{2}\left(\frac{\|z\|_2^2}{0.4}\right)^2 - \log\left(e^{-0.5\left[\frac{z_1^2+z_2^2}{0.6}\right]} + e^{\left[\frac{z_1^2+z_2^2}{0.6}\right]}\right)$, and $p(z) \propto e^{-U(z)}$ be the distribution we want to sample from.

Let's visualize the (unnormalized) density.

```
def npdensity1(z):
    z = np.reshape(z, [z.shape[0], 2])
    z1, z2 = z[:, 0], z[:, 1]
    norm = np.sqrt(z1 ** 2 + z2 ** 2)
    exp1 = np.exp(-0.5 * ((z1 - 2) / 0.6) ** 2)
    exp2 = np.exp(-0.5 * ((z1 + 2) / 0.6) ** 2)
    u = 0.5 * ((norm - 2) / 0.4) ** 2 - np.log(exp1 + exp2)
    return np.exp(-u)

def npdensity2(z):
    z = z.reshape(-1, 2)
    x, y = z[:, 0], z[:, 1]
    u = 0.8 * x ** 2 + (y - ((x**2)**(1/3)))**2
    u = u / 2**2
    return np.exp(-u)
```

Fig. 2 Code for solving Stochastic Differential Equation - Langevin Dynamics

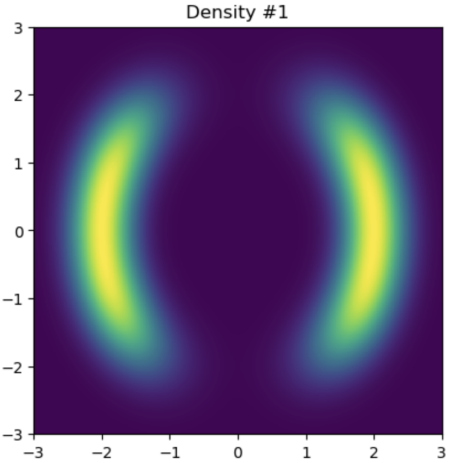


Fig. 3 Result of the density plot

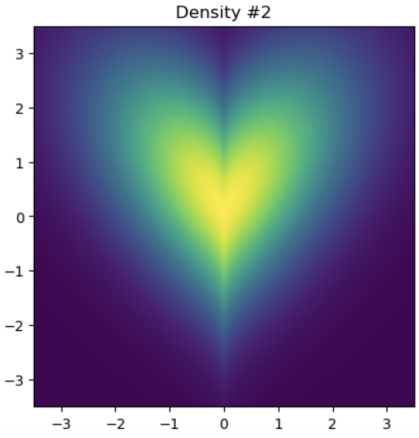


Fig. 4 Result of the density plot

We solved a stochastic differential equation (SDE) describing Langevin dynamics, where both deterministic forces and stochastic Brownian noise drove the evolution of a particle under a potential energy landscape. This stochastic framework mirrors the core principle of Gibbs Ensemble Monte Carlo simulations, where random trial moves, governed by Metropolis acceptance criteria, ensure thorough sampling of the system's configuration space. In both approaches, stochasticity is fundamental to achieving equilibrium ensemble sampling, allowing efficient exploration of phase space compared to purely deterministic methods. The random fluctuations in Langevin dynamics correspond to the trial moves in GEMC. At the same time, the energy-based bias introduced by the Metropolis criterion parallels the deterministic drift term in the stochastic differential equation (SDE), reinforcing the conceptual similarity between stochastic sampling and equilibrium thermodynamics. By solving a stochastic differential equation (Langevin dynamics) and comparing it with deterministic evolution, we observed that stochastic trajectories effectively explore regions of low potential energy, consistent with the Boltzmann distribution.

To further explore the potential energy by sampling from the Boltzmann distribution, we explored how a particle explores a complex 2D potential energy landscape under thermal fluctuations. To address this problem, we solved and modified the Rosenbrock potential and analyzed barrier crossings between regions (or wells) of the potential.

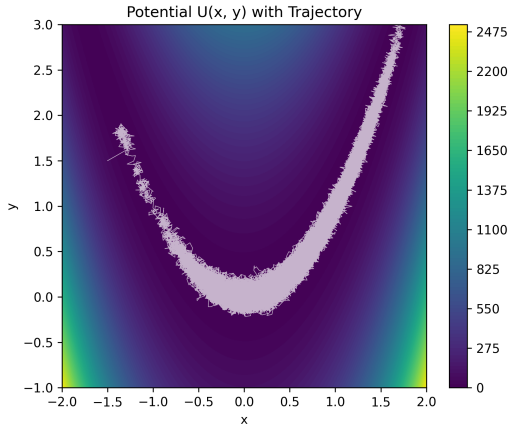


Fig. 5 Potential Energy Landscape

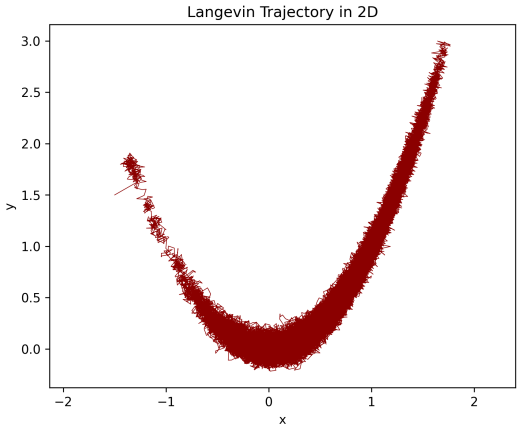


Fig. 6 Langevin Trajectory of the Potential Energy

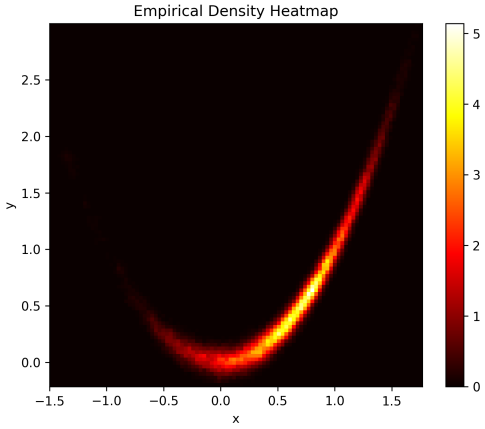


Fig. 7 Empirical Density Heatmap

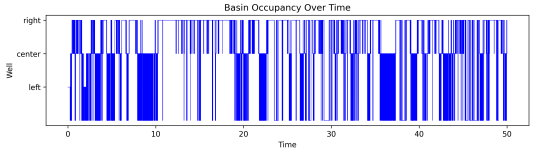


Fig. 8 Basin transitions over time

The results from the Langevin dynamics simulation reveal how a particle explores a complex 2D potential energy landscape under thermal fluctuations. The first plot, showing the potential energy surface with the trajectory overlaid, illustrates that the particle moves through the tilted Rosenbrock valley, transitioning between local basins (wells) separated by energy barriers. The second plot, a 2D trajectory, highlights the path traced in phase space and shows confinement within and transitions between regions of attraction. The third plot, a heatmap of the particle's empirical density, captures the probability distribution, showing higher occupation in low-energy areas, particularly near the central and right wells, indicating thermodynamic favorability. The fourth plot, depicting basin occupancy over time, explicitly tracks transitions between defined basins ("left," "center," and "right"), revealing rare but clear barrier crossings due to thermal noise. The plots demonstrate how stochasticity enables barrier hopping and equilibrium exploration, echoing principles foundational to methods like Gibbs ensemble Monte Carlo.

In deterministic evolution, the system quickly relaxes to a local minimum and becomes confined. This fundamental difference illustrates why stochastic sampling is critical: it ensures a broader exploration of phase space, convergence to thermodynamic equilibrium, and accurate ensemble averages, all of which are essential for the reliability of GEMC simulations. Therefore, stochastic move strategies are more suitable than deterministic ones to accurately model phase co-existence phenomena. This theory introduces the concept of ensembles in Monte Carlo simulations.

IV. Ensembles

A. Canonical Monte Carlo

In the canonical ensemble, the number of molecules N, the volume V, and the temperature T are all constant. The position, orientation, and conformation of a semi-flexible molecule with fixed bond lengths containing M atoms are given by a $2M+1$-dimensional vector $\mathbf{q}$. The positions, orientations, and conformations of all N molecules are denoted $\mathbf{q}^N$.

Microstate Probabilities

The probability of observing microstate m with configuration $\mathbf{q}_m^N$ is given by:

$$p_m = \frac{e^{-\beta U(\mathbf{q}_m^N)}}{Z(N,V,T)}\, d\mathbf{q}^N \tag{3}$$

where β is the inverse temperature $1/k_B T$, U is the potential energy, the differential volume $d\mathbf{q}^N$ is included to make p_m dimensionless, and Z is the configurational partition function:

$$Z(N,V,T) = \int e^{-\beta U(\mathbf{q}^N)}\, d\mathbf{q}^N. \tag{4}$$

The integral is over all $N(2M+1)$ degrees of freedom. The ratio of microstate probabilities follows from Eq. (4):

$$\frac{p_m}{p_n} = \frac{e^{-\beta U(\mathbf{q}_m^N)} d\mathbf{q}^N / Z(N,V,T)}{e^{-\beta U(\mathbf{q}_n^N)} d\mathbf{q}^N / Z(N,V,T)} = e^{\beta(U_n - U_m)} = e^{\beta \Delta U}. \tag{5}$$

The configurational partition function Z and differential volume $d\mathbf{q}^N$ both cancel, leaving only the ratio of Boltzmann factors.

Monte Carlo Moves

New configurations are generated by attempting moves that translate, rotate, and regrow a randomly selected molecule. For more information on the acceptance rules of these moves, please refer to the sections on Molecule Translation, Molecule Rotation, and Molecule Regrowth, respectively.

Partition Functions in the Canonical Ensemble

The microstate probability in Eq. (4) is normalized by the configurational partition function Z because the only relevant degrees of freedom are configurational. In other ensembles, the full canonical partition function Q appears, integrated over both configuration space $\mathbf{q}^N$ and momentum space $\mathbf{p}_q^N$:

$$Q(N,V,T) = \frac{1}{h^{N(2M+1)}N!}\int e^{-\beta H(\mathbf{p}_q^N,\mathbf{q}^N)}\,d\mathbf{p}_q^N\,d\mathbf{q}^N \quad (6)$$

where the $2M+1$ momenta $\mathbf{p}_q$ are conjugate to the generalized coordinates $\mathbf{q}$. The momentum and configuration integrals are separable, and the single-molecule momentum integrals are all identical. Thus:

$$\begin{aligned} Q(N,V,T) &= \frac{1}{N!}\left[\int e^{-\beta U(\mathbf{q}^N)}\,d\mathbf{q}^N\right]\left[\frac{1}{h^{2M+1}}\int e^{-\beta K(\mathbf{p}_q)}\,d\mathbf{p}_q\right]^N \\ &= \frac{1}{N!}Z(N,V,T)\left[\frac{Q(1,V,T)}{Z(1,V,T)}\right]^N \end{aligned}$$

where $Q(1,V,T)$ is the partition function of a single molecule in a box. The center-of-mass integrals for a single molecule are separable from the integrals over rotational and internal degrees of freedom:

$$Q(1,V,T) = Q_{\text{com}}Q_{\text{rot+int}} = V\Lambda^{-3}Q_{\text{rot+int}} \quad (7)$$

where Λ is the de Broglie wavelength of the molecule, and the rotational and internal momentum integrals in $Q_{\text{rot+int}}$ are not separable since the moments of inertia depend on the conformation adopted by the molecule.

The configurational partition function is further separable into center-of-mass (translational), orientational, and internal degrees of freedom:

$$Z(1,V,T) = VZ_{\text{rot}}Z_{\text{int}} \quad (8)$$

where the volume V is the translational partition function, and Z_{rot} equals 4π for a linear molecule and $8\pi^2$ for a nonlinear molecule.

B. Isothermal-Isobaric Ensemble

In the isothermal-isobaric ensemble, the number of particles N, the pressure P, and temperature T are all constant while the volume V and energy E fluctuate. The partition function is

$$\Delta(N,P,T) = \int e^{-\beta PV}Q(N,V,T)\,dV \quad (10)$$

where Q is dimensionless and Δ has dimensions of volume. The kinetic contribution to Δ is independent of the pressure or volume and consequently separable from the configurational contribution, ΔZ

$$\Delta Z(N,P,T) = \int e^{-\beta PV}Z(N,V,T)\,dV \quad (11)$$

The probability of the system having volume V is

$$p(V) = Z(N,V,T)e^{-\beta PV}\Delta Z(N,P,T)\,dV \quad (12)$$

The probability of observing microstate m with configuration $\mathbf{q}_{Nm}$ and volume V_m is

$$p_m = \frac{e^{-\beta U(\mathbf{q}_{Nm})}d\mathbf{q}_{Nm}}{Z(N,V_m,T)}\cdot\frac{Q(N,V_m,T)e^{-\beta PV_m}dV}{\Delta(N,P,T)} \quad (13)$$

where the differential element $d\mathbf{q}_{Nm}$ has subscript m because it scales with the volume V_m. The ratio of microstate probabilities is

$$\frac{p_m}{p_n} = e^{\beta(U_n-U_m)+\beta P(V_n-V_m)}\left(\frac{d\mathbf{q}_m}{d\mathbf{q}_n}\right)_N = e^{\beta\Delta U+\beta P\Delta V}\left(\frac{d\mathbf{q}_m}{d\mathbf{q}_n}\right)_N \quad (14)$$

New configurations are generated via Molecule Translation, Molecule Rotation, Molecule Regrowth, and Volume Scaling.

C. Grand Canonical Monte Carlo

In the grand canonical ensemble, the chemical potential μ, the volume V, and temperature T are held constant while the number of molecules N and energy E fluctuate. The partition function is

$$\Xi(\mu,V,T) = \sum_{N=0}^{\infty}Q(N,V,T)\,e^{\beta\mu N} \quad (15)$$

The probability of the system having N molecules is

$$p(N) = \frac{Q(N,V,T)e^{\beta\mu N}}{\Xi(\mu,V,T)} \quad (16)$$

The probability of observing microstate m with N_m molecules and configuration $\mathbf{q}_{N_m m}$ is

$$p_m = \frac{e^{-\beta U(\mathbf{q}_{N_m m})}d\mathbf{q}_{N_m}}{Z(N_m,V,T)}\cdot\frac{Q(N_m,V,T)e^{\beta\mu N_m}}{\Xi(\mu,V,T)} \quad (17)$$

Note that Eq. (17) does not contain the factorial $N_m!$ that accounts for indistinguishable particles. In a simulation, particles are distinguishable: they are numbered and specific particles are picked for MC moves. The ratio of microstate probabilities is

$$\frac{p_m}{p_n} = e^{\beta\Delta U-\beta\mu\Delta N}\left[Q(1,V,T)Z(1,V,T)\,d\mathbf{q}\right]^{-\Delta N} \quad (18)$$

Alternatively, Eq. (18) can be recast to use the fugacity f instead of the chemical potential μ. The relationship between μ and f is

$$\mu = -k_BT\ln\left(\frac{Q(1,V,T)}{N}\right) = -k_BT\ln\left(Q(1,V,T)\beta fV\right) \quad (19)$$

Inserting Eq. (19) into Eq. (18) yields

$$\frac{p_m}{p_n} = e^{\beta\Delta U}\left[\beta fVZ(1,V,T)\,d\mathbf{q}\right]^{-\Delta N} \quad (20)$$

Fluctuations in the number of molecules are achieved by inserting and deleting molecules. A successful insertion increases the number of molecules from N to $N+1$, i.e.,

$\Delta N = 1$. A successful deletion decreases the number of molecules from N to $N-1$, i.e., $\Delta N = -1$.

Random insertions and deletions (see Inserting a Molecule Randomly and Deleting a Molecule Inserted Randomly) in the liquid phase typically have very large ΔU due to core overlap and dangling bonds, respectively, making the probability of acceptance very low. To overcome this challenge, insertions in Cassandra are attempted using Configurational Bias Monte Carlo. See Inserting a Molecule with Configurational Bias Monte Carlo and Deleting a Molecule that was Inserted via Configurational Bias Monte Carlo for details.

D. Gibbs Ensemble Monte Carlo

The Gibbs Ensemble Monte Carlo method is a standard technique for studying phase equilibria of pure fluids and mixtures. It is often used to study vapor-liquid equilibria due to its intuitive physical basis. In Cassandra, the NVT and NPT versions of the Gibbs Ensemble (GEMC-NVT and GEMC-NPT) are implemented. The GEMC-NVT method is suitable for simulating vapor-liquid equilibria of pure systems, since pure substances require the specification of only one intensive variable (temperature) to completely specify a state of two phases. By contrast, mixtures require the specification of an additional degree of freedom (pressure). Thus, in the GEMC-NPT method, the pressure is specified in addition to temperature.

In principle, the fundamental understanding of the Gibbs ensemble Monte Carlo (GEMC) algorithm is the direct simulation of gas–liquid phase coexistence, and was first introduced by Panagiotopoulos et al. (13,14). GEMC simultaneously models the gas and liquid phases in two different simulation boxes, as shown in Figure. Both boxes start with a given number of molecules, a given volume, and thus a given density. During the simulation, molecules and volume are exchanged between the two boxes. One box will equilibrate to the gas phase and the other to the liquid phase. As a result, the phase coexistence of a fluid is modeled, at a given temperature and pressure, without an interfering interface between the phases. The Gibbs ensemble provides accurate coexistence densities for relatively small systems, provided that one is not too close to the critical point.

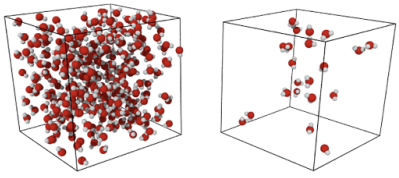


Fig. 5 Examples of liquid and gas boxes in GEMC simulations. Besides thermalization, the two boxes can exchange molecules and volume

The partition functions and microstate probabilities are derived for Gibbs Ensemble-NVT and Gibbs Ensemble-NPT, below. In both cases, thermal equilibrium is attained by performing translation, rotation, and regrowth moves. The acceptance rules for these moves are identical to those presented in Molecule Translation, Molecule Rotation, Molecule Regrowth, and Regrowing a Molecule with Configurational Bias Monte Carlo. Pressure equilibrium is achieved by exchanging volume in the case of GEMC-NVT, or independently changing the volume of each box in the case of GEMC-NPT. The acceptance rule for the exchanging volume in GEMC-NVT is derived and its Cassandra implementation is presented in Volume Exchange Moves. The acceptance rule for swapping a molecule in either GEMC-NVT or GEMC-NPT is derived in Molecule Exchange Moves.

V. Monte Carlo Moves

A. Moelcule Translation

A molecule is translated by moving its center of mass in each Cartesian direction by a random amount chosen from the uniform distribution on the interval $[-\delta r_{\max}, \delta r_{\max}]$. The maximum displacement $\delta r_{\max}$ must be given in the input file. The translation move is symmetric in forward and reverse directions. That is, either microstate n can be formed from microstate m and vice versa by moving one molecule within $\delta r_{\max}$ in each Cartesian direction, or microstate n cannot be formed at all. As a result, $\alpha_{mn} = \alpha_{nm}$.

The acceptance probability for a translation move follows from Eq. (6)

$$\ln\left(\frac{\alpha_{mn}}{\alpha_{nm}}\frac{p_m}{p_n}\right) = \ln\left(\frac{p_m}{p_n}\right) = \beta\Delta U \tag{28}$$

In Cassandra, the translation move is implemented in the subroutine `Translate` defined in `move_translate.f90`. The variable names in the `move_translate.f90` code are identified with the symbols.

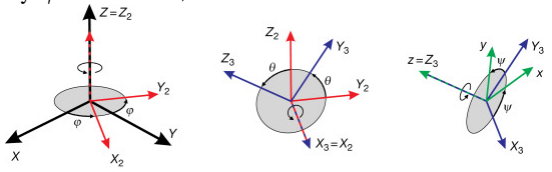
```
ln_pacc = beta(ibox) * delta_e
accept = accept_or_reject(ln_pacc)
```

B. Molecule Rotation

A linear molecule is rotated differently than a nonlinear molecule. A molecule is identified as linear if it is composed of 2 atoms or if all the angles are rigid with a bond angle of 180°.

If the molecule is linear:

- Pick three random angles: ϕ on $[-\pi,\pi]$, $\cos(\theta)$ on $[-1,1]$, and ψ on $[-\pi,\pi]$.
- With the origin at the molecule's center of mass, rotate by ϕ around z, rotate by θ around x', and rotate by ψ around z', as shown below.



- **Fig. 6** Procedure for rotating linear molecules.

Even though three angles are randomly chosen, the probability of the resulting orientation is $d\cos(\theta)d\phi/4\pi$.

If the molecule is nonlinear:

- Randomly select an axis: x, y, or z.
- Choose a random angular displacement $\delta\theta$ from $[-\delta\theta_{\max}, \delta\theta_{\max}]$. $\delta\theta_{\max}$ must be given in the input file.
- Rotate the molecule around a vector parallel to the selected axis and through its center of mass by $\delta\theta$.

In either case, the rotation move is symmetric, $\alpha_{mn} = \alpha_{nm}$, and the acceptance criteria is given by Eq. (28). The rotation move is implemented in subroutine `Rotate` defined in `move_rotate.f90`.

```
ln_pacc = beta(ibox) * delta_e
accept = accept_or_reject(ln_pacc)
```

C. Molecule Regrowth

Internal degrees of freedom in flexible molecules are sampled by deleting one or more fragments from the molecule and replacing the deleted fragments with conformations from a reservoir of fragment conformations. If the molecule consists of only a single fragment (e.g., water, all atom methane, united atom propane, all atom cyclohexane), the entire molecule is deleted and replaced as follows:

- Randomly select a molecule i with uniform probability $1/N$, record its center-of-mass position and delete it.
- Select a molecular conformation with Boltzmann probability $e^{-\beta U(\mathbf{q}^{(i)}_{\text{int},n})}/Z_{\text{int}}$, where $\mathbf{q}^{(i)}_{\text{int},n}$ are the internal bond or improper angles of molecule i in microstate n and Z_{int} is the configurational partition function over internal degrees of freedom (see Eq. (9)).
- Pick three random angles: ϕ on $[-\pi,\pi]$, $\cos(\theta)$ on $[-1,1]$, and ψ on $[-\pi,\pi]$. Rotate the molecule as shown in Fig. 2. The probability of the resulting orientation is $d\mathbf{q}_{\text{rot}}/Z_{\text{rot}}$, which for a nonlinear molecule is $d\cos(\theta)d\phi d\psi/8\pi^2$.
- Place the molecule with the selected conformation and orientation at the same center-of-mass position as the deleted molecule.

Regrowing a monoatomic particle has no effect. Regrowing a linear molecule is the same as rotating it. The overall probability α_{mn} of regrowing a molecule with the selected orientation and conformation is

$$\alpha_{mn} = \frac{1}{N}\frac{d\mathbf{q}_{\text{rot}}}{Z_{\text{rot}}}\frac{e^{-\beta U(\mathbf{q}^{(i)}_n)}d\mathbf{q}_{\text{int}}}{Z_{\text{int}}} \tag{29}$$

where $\mathbf{q}^{(i)}_n$ denotes the position, orientation, and conformation of molecule i in microstate n and $U(\mathbf{q}^{(i)}_n)$ is the potential energy of the isolated molecule i, i.e., the intramolecular potential energy. The reverse probability α_{nm} is identical except for the intramolecular potential energy $U(\mathbf{q}^{(i)}_m)$ of molecule i in microstate m. Using Eqs. (6) and (29), the acceptance criteria for the regrowth of a single fragment molecule are

$$\ln\left(\frac{\alpha_{mn}}{\alpha_{nm}}\frac{p_m}{p_n}\right) = \beta\left[(U(\mathbf{q}_{Nn}) - U(\mathbf{q}_{Nm})) - (U(\mathbf{q}^{(i)}_n) - U(\mathbf{q}^{(i)}_m))\right] \tag{9}$$

$$= \beta\Delta U - \beta\Delta U^{(i)}_{\text{int}} \tag{30}$$

Only the change in the intermolecular potential energy between molecule i and the other $N-1$ molecules contributes to the acceptance criteria. The code that implements Eq. (30) is shown in code in Regrowing a Molecule with Configurational Bias Monte Carlo.

If the molecule consists of more than one fragment (e.g., all-atom ethane, all-atom toluene, united-atom butane), a bond is cut, and the severed fragments are regrown using Configurational Bias Monte Carlo (CBMC). See Regrowing a Molecule with Configurational Bias Monte Carlo for more details.

D. Volume scaling

In Cassandra, new volumes are sampled as follows:

- Pick a random volume ΔV with uniform probability from the interval $[-\delta V_{\max}, \delta V_{\max}]$. The trial volume is $V + \Delta V$.
- Scale the box lengths uniformly.
- Scale the center of mass of each molecule uniformly.

The probability of selecting ΔV is the same as selecting $-\Delta V$, which makes scaling the volume symmetric, $\alpha_{mn} = \alpha_{nm}$. Scaling the configurations changes the differential element $d\mathbf{q}_{Nm}$ surrounding configuration $\mathbf{q}_{Nm}$. Only the molecular centers of mass change, so we can separate $d\mathbf{q}$ into 3 center of mass coordinates $d\mathbf{r}_{\text{com}}$ and $2M-2$ orientational and internal coordinates $d\mathbf{q}_{\text{rot+int}}$. The scaled center of mass positions are held constant, making $d\mathbf{r}_{\text{com}} = V d\mathbf{s}_{\text{com}}$. The acceptance probability for a volume scaling move is

$$\ln\left(\frac{\alpha_{mn}}{\alpha_{nm}}\frac{p_m}{p_n}\right) = \ln\left(\frac{p_m}{p_n}\right) = \beta\Delta U + \beta P\Delta V + N\ln\left(\frac{V_m}{V_n}\right) \tag{31}$$

The volume scaling move is implemented in subroutine `Volume_Change` defined in `move_volume.f90`.

E. Inserting a molecule in Configurational bias Monte Carlo

In Configurational Bias Monte Carlo (CBMC), the molecular conformation of the inserted molecule is molded to the insertion cavity. First, the molecule is parsed into fragments such that each fragment is composed of (a) a central atom and the atoms directly bonded to it (see Fig. **??**), or (b) a ring of atoms and all the atoms directly bonded to them. Then, a position, orientation, and molecular conformation of the molecule to be inserted are selected via the following steps:

1) Select the order in which each fragment of the $(N+1)$-th molecule will be placed. The probability of the resulting sequence is p_{seq}. (See example in Table II)

Algorithm 2: Volume Scaling Move Acceptance in Cassandra

Input: β(this_box), delta_e, pressure(this_box), delta_volume, total_molecules, box_list(this_box)%volume, box_list_old%volume
Output: accept (Boolean indicating move acceptance)
Compute the log acceptance probability:
  ln_pacc ← β(this_box) · delta_e
  ln_pacc ← ln_pacc + β(this_box) · pressure(this_box) · delta_volume
  ln_pacc ← ln_pacc − total_molecules · ln$\left(\frac{\text{box_list(this_box)\%volume}}{\text{box_list_old\%volume}}\right)$
Determine acceptance:
  accept ← accept_or_reject(ln_pacc)
Return: accept

TABLE I

VARIABLE SYMBOLS AND CODE NAMES FOR VOLUME SCALING MOVE.

Symbol	Code name
β	`beta(this_box)`
ΔU	`delta_e`
P	`pressure(this_box)`
ΔV	`delta_volume`
N	`total_molecules`
V_n	`box_list(this_box)%volume`
V_m	`box_list_old%volume`

- The first fragment i is chosen with uniform probability $1/N_{\text{frag}}$.
- Subsequent fragments must be connected to a previously chosen fragment and are chosen with the uniform probability $1/N_{\text{cnxn}}$, where the number of connections $N_{\text{cnxn}} = \sum_{ij} \delta_{ij} h_i (1-h_j)$ is summed over all fragments i and j. h_i is 1 if fragment i has been previously chosen and 0 otherwise. δ_{ij} is 1 if fragments i and j are connected and 0 otherwise.

2) Select a conformation for fragment i with Boltzmann probability $e^{-\beta U(\mathbf{q}_{\text{frag}_i})} d\mathbf{q}_{\text{frag}_i} / Z_{\text{frag}_i}$, where $\mathbf{q}_{\text{frag}_i}$ are the internal degrees of freedom (angles and/or impropers) associated with fragment i.
3) Select an orientation with uniform probability $d\mathbf{q}_{\text{rot}} / Z_{\text{rot}}$.
4) Select a coordinate for the center of mass (COM) of fragment i:
   - Select κ_{ins} trial coordinates $\mathbf{r}_k$, each with uniform probability $d\mathbf{r}/V$. Since one of the trial coordinates will be selected later, the individual probabilities are additive. The probability of the collection of trial coordinates is $\kappa_{\text{ins}} d\mathbf{r}/V$.
   - Compute the change in potential energy ΔU_{ins_k} of inserting fragment i at each position $\mathbf{r}_k$ into configuration $\mathbf{q}_{N_m}$.
   - Select one of the trial coordinates with probability $e^{-\beta \Delta U_{\text{ins}_k}} / \sum_k e^{-\beta \Delta U_{\text{ins}_k}}$.
5) For each additional fragment j:

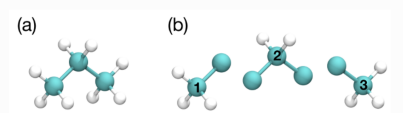


Fig. 1. **Fig. 7** (a) An all-atom model of propane. (b) The same model as in (a), but parsed into three fragments.

- Select a fragment conformation with Boltzmann probability $e^{-\beta U(\mathbf{q}_{\text{frag}_j})} d\mathbf{q}_{\text{frag}_j} / Z_{\text{frag}_j}$.
- Select the first of κ_{dih} trial dihedrals ϕ_0 with uniform probability from the interval $[0, \frac{2\pi}{\kappa_{\text{dih}}})$. Additional trial dihedrals are equally spaced around the unit circle, $\phi_k = \phi_{k-1} + 2\pi/\kappa_{\text{dih}}$. The probability of selecting ϕ_0 is $\kappa_{\text{dih}} d\phi / 2\pi$.
- Compute the change in potential energy ΔU_{dih_k} of attaching fragment j to the growing molecule with each dihedral ϕ_k.
- Select one of the trial dihedrals with probability $e^{-\beta \Delta U_{\text{dih}_k}} / \sum_k e^{-\beta \Delta U_{\text{dih}_k}}$.

TABLE II

POSSIBLE SEQUENCES AND PROBABILITIES FOR INSERTING THE FRAGMENTS OF THE ALL-ATOM MODEL OF PROPANE SHOWN IN FIG. **??**.

Sequence	p_{seq}
1 2 3	1/3
2 1 3	1/6
2 3 1	1/6
3 2 1	1/3

The overall probability α_{mn} of attempting the insertion with the selected position, orientation, and conformation is

$$= p_{\text{seq}}\, p_{\text{bias}} \frac{e^{-\beta U(\mathbf{q}_{\text{frag}})} d\mathbf{q}}{V Z_{\text{rot}} Z_{\text{frag}} \Omega_{\text{dih}}}, \tag{10}$$

where $Z_{\text{frag}} = \prod_i Z_{\text{frag}_i}$ is the configurational partition function over degrees of freedom internal to each fragment, $U(\mathbf{q}_{\text{frag}}) = \sum_i U(\mathbf{q}_{\text{frag}_i})$ is the summed potential energy of each of the (disconnected) fragments, $\Omega_{\text{dih}} = (2\pi)^{N_{\text{frag}}-1}$, and p_{bias} is

$$p_{\text{bias}} = \kappa_{\text{ins}} \frac{e^{-\beta \Delta U_{\text{ins}_k}}}{\sum_k e^{-\beta \Delta U_{\text{ins}_k}}} \left[\prod_{j=1}^{N_{\text{frag}}-1} \kappa_{\text{dih}} \frac{e^{-\beta \Delta U_{\text{dih}_k}}}{\sum_k e^{-\beta \Delta U_{\text{dih}_k}}} \right]. \tag{11}$$

Note that the term $V Z_{\text{rot}} Z_{\text{frag}} \Omega_{\text{dih}}$ in the denominator of Eq. (32) differs from $Z(1,V,T) = V Z_{\text{rot}} Z_{\text{int}}$.

In the reverse move, 1 of the $N+1$ particles is randomly selected for deletion. The probability α_{nm} of picking the molecule we just inserted is

$$\alpha_{nm} = \frac{1}{N+1}. \tag{12}$$

Combining Eq's gives the acceptance probability for a CBMC insertion move

$$\ln\left(\frac{\alpha_{mn}\alpha_{nm}p_m}{p_n}\right) = \beta\left[\Delta U - U(\mathbf{q}_{\text{frag},n}^{(N+1)})\right] + \ln\left(\frac{N+1}{\beta f' V}\right) + \ln(p_{\text{seq}}\, p_{\text{bias}}), \tag{13}$$

where μ' and f' are, respectively, a shifted chemical potential and a skewed fugacity,

$$\mu' = \mu + k_B T \ln\left(\frac{Q_{\text{rot+int}} Z_{\text{frag}} \Omega_{\text{dih}}}{Z_{\text{int}}}\right), \tag{14}$$

$$f' = f \frac{Z_{\text{frag}} \Omega_{\text{dih}}}{Z_{\text{int}}}. \tag{15}$$

All of the terms in Eqs are intensive. GCMC simulations using Eqs and (36) will converge to the same average density regardless of the simulation volume V. However, the values of μ' or f' that correspond to the converged density will not match tabulated values of μ or f computed from experimental data.

Note that the term Z_{IG}/Ω from Macedonia *et al.* would be equivalent to $Z_{\text{int}}/\Omega_{\text{frag}}\Omega_{\text{dih}}$ in the nomenclature used here. The configurational partition function of the disconnected fragments integrates over a Boltzmann factor, $Z_{\text{frag}} = \int e^{-\beta U(\mathbf{q}_{\text{frag}})} d\mathbf{q}_{\text{frag}}$, whereas the term $\Omega_{\text{frag}} = \int d\mathbf{q}_{\text{frag}}$ does not.

VI. Force Field

The Generalized Amber Force Field (GAFF)[6] was used to describe the inter - and intramolecular interactions. The partial atomic charges used with GAFF [6] were derived using the restrained electrostatic potential (RESP) method based on the optimized structure of each ion in the vacuum at the B3LYP/6-311++g(d,p) level using the package Gaussian 16[4]. Charges on the ions were uniformly scaled by 0.8 to account for polarization effects. A cutoff of 12 Å was used along with long-range tail corrections to the energy and pressure for the Lennard-Jones portion of the pair interactions. The equation of the total energy is as follows:

$$\begin{aligned} E^{\text{MM}} = & \sum_{\text{bonds}} K_r (r - r_{\text{eq}})^2 + \sum_{\text{angles}} K_\theta (\theta - \theta_{\text{eq}})^2 \\ & + \sum_{\text{torsions}} \sum_{n=0}^{5} C_n (\cos(\psi))^n + \sum_{\text{impropers}} k_d (1 + \cos(n_d \omega - \omega_d)) \\ & + \sum_{i>j}^{N} \left[4\epsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right) + \frac{1}{4\pi\epsilon_0} \frac{q_i q_j}{r_{ij}} \right], \end{aligned}$$

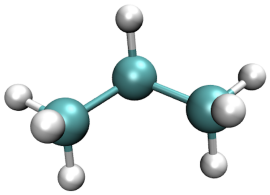
The energetic contributions due to bonds, angles, dihedrals, and improper torsions are described by the terms involving K_r, K_θ, C_n, and K_d, respectively. The Lennard-Jones (LJ) 12–6 potential is used to express van der Waals interactions, for which ϵ_{ij} and σ_{ij} denote the size and energy parameters between atoms i and j. q_i and q_j are the partial charges placed on the atomic sites i and j, respectively, and describe the electrostatic interactions in the system via Coulomb's law. As proposed for the model, the Lorentz–Berthelot combining rule was used to compute interactions between two different atom types. Intramolecular interactions between the terminal atoms in a dihedral, the so-called 1–4 interactions, were scaled by a factor of 0.5 for both the LJ and electrostatic interactions,

Algorithm 3: Configurational Bias Monte Carlo (CBMC) Insertion Move

Input: N (current number of molecules), V (volume), β (inverse temperature), μ' (shifted chemical potential), κ_{ins} (number of trial coordinates), κ_{dih} (number of trial dihedrals), molecule fragment data
Output: accept (Boolean indicating move acceptance)
Parse the $(N+1)$-th molecule into N_{frag} fragments
Select a sequence for placing fragments with probability p_{seq} ▷ See Table 10
Initialize ln_pbias $\leftarrow 0$, $\Delta U \leftarrow 0$, $U_{\text{frag}} \leftarrow 0$
for *each fragment* $i = 1$ *to* N_{frag} **do**
  Select conformation with probability $\frac{e^{-\beta U(\mathbf{q}_{\text{frag}_i})} d\mathbf{q}_{\text{frag}_i}}{Z_{\text{frag}_i}}$
  $U_{\text{frag}} \leftarrow U_{\text{frag}} + U(\mathbf{q}_{\text{frag}_i})$ **if** $i = 1$ **then**
    Select orientation with probability $\frac{d\mathbf{q}_{\text{rot}}}{Z_{\text{rot}}}$
    Generate κ_{ins} trial coordinates $\mathbf{r}_k$ with probability $\frac{d\mathbf{r}}{V}$
    for *each* $k = 1$ *to* κ_{ins} **do**
      Compute ΔU_{ins_k} for inserting fragment i at $\mathbf{r}_k$
    Select one $\mathbf{r}_k$ with probability $\frac{e^{-\beta \Delta U_{\text{ins}_k}}}{\sum_k e^{-\beta \Delta U_{\text{ins}_k}}}$
    ln_pbias $\leftarrow$ ln_pbias $+ \ln\left(\kappa_{\text{ins}} \cdot \frac{e^{-\beta \Delta U_{\text{ins}_k}}}{\sum_k e^{-\beta \Delta U_{\text{ins}_k}}}\right)$
    $\Delta U \leftarrow \Delta U + \Delta U_{\text{ins}_k}$ **else**
    Generate κ_{dih} trial dihedrals ϕ_k starting at ϕ_0 with probability $\frac{\kappa_{\text{dih}} d\phi}{2\pi}$ **for** *each* $k = 1$ *to* κ_{dih} **do**
      Compute ΔU_{dih_k} for attaching fragment i with dihedral ϕ_k
    Select one ϕ_k with probability $\frac{e^{-\beta \Delta U_{\text{dih}_k}}}{\sum_k e^{-\beta \Delta U_{\text{dih}_k}}}$
    ln_pbias $\leftarrow$ ln_pbias $+ \ln\left(\kappa_{\text{dih}} \cdot \frac{e^{-\beta \Delta U_{\text{dih}_k}}}{\sum_k e^{-\beta \Delta U_{\text{dih}_k}}}\right)$
    $\Delta U \leftarrow \Delta U + \Delta U_{\text{dih}_k}$
Compute the log acceptance probability:
  ln_pacc $\leftarrow \beta \cdot (\Delta U - U_{\text{frag}})$
  ln_pacc $\leftarrow$ ln_pacc $- \beta \cdot \mu'$
  ln_pacc $\leftarrow$ ln_pacc $+ \ln\left(\frac{(N+1)\Lambda^3}{V}\right)$
  ln_pacc $\leftarrow$ ln_pacc $+ \ln(p_{\text{seq}})$ + ln_pbias
Determine acceptance:
  accept $\leftarrow$ accept_or_reject(ln_pacc)
Return: accept

while the nonbonded interactions between the atoms connected by bonds and angles were excluded.

The optimized structure of the propane molecule :



•

Once the structure is optimized, the force field parameters are obtained using the antechamber module in the Amber force field.

VII. Simulation Details

Once we have derived the propane force field parameters, we will use Gibbs ensemble Monte Carlo simulations to obtain the phase diagram of the propane gas. During the GEMC simulation, the total number of molecules N and the total volume V remain constant. For thermalization trial moves, one can perform Monte Carlo trial moves and translate/rotate each molecule separately. On the basis of the energy, the thermalization is accepted or rejected. For convenience, we chose to perform a thermalization of all molecules in a single trial move by using a molecular dynamics (MD) algorithm. We computed the phase diagram of propane at different temperatures, 217 K, 249 K, 281 K, 312 K, and 344 K by performing Gibbs Ensemble Monte Carlo simulations for 2×10^7 steps using CASSANDRA[3]. All simulations were performed on GPUs using the Medicine-Bow supercomputing facility at the University of Wyoming. Once the Monte Carlo simulations were performed, we computed the densities of the propane in both the liquid and vapor boxes to obtain the vapor-liquid co-existence curve that leads to an understanding of thermodynamic properties.

VIII. Results and Discussion

A. VLE - TRAPPE(UA) validation

To validate our model, we compared the results of GEMC simulations with the TRAPPE classical forcefields. The critical point of the VLE can be calculated by fitting with the law of rectilinear diameters:

$\frac{\rho_l+\rho_g}{2} = \rho_c + A\left(1 - \frac{T}{T_c}\right)$

where ρ_l, ρ_g, and ρ_c are the liquid, gas, and critical densities, respectively, T and T_c are the temperature and critical temperature, and A is a fitting parameter.

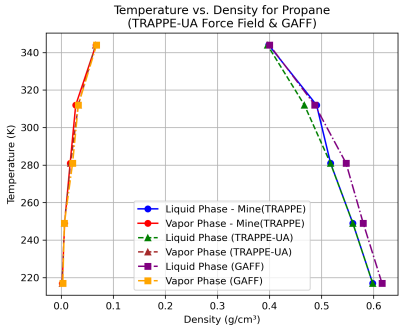
The density difference of the phases can be fitted to a scaling law:

$$\rho_l - \rho_g = B\left|1 - \frac{T}{T_c}\right|^{\gamma} \tag{16}$$

where γ is the critical exponent, with $\gamma = 0.32$ for 3D systems, and B is a fitting parameter obtained from the fit.

To configure a Gibbs Ensemble Monte Carlo (GEMC) simulation, we named propane.out to study propane's phase equilibrium at T K in two cubic boxes: one 40 Å in size with 240 molecules and the other 100 Å with 160 molecules, totaling 400 propane molecules defined by propane.mcf. Using a Lennard-Jones potential with a 14 Å cutoff and tail corrections, the simulation employs Lorentz-Berthelot mixing rules and calculates pairwise energies, starting with random seeds 1216131145 and 1216131146. It runs for 20 million steps in an equilibration phase, adjusting move parameters every 100 steps, with output properties—total energy, density, molecule count, and volume—recorded every 1,000 steps and coordinates every 2,200 steps for both boxes. Move probabilities include 0.40 for translation (0.15 in Box 1 with a 14 Å max displacement), 0.40 for rotation (max angles of 30° and 180°), 0.005 for volume (200 $Å^3$ max change), and 0.195 for CBMC swaps, supported by three fragment files and parameters like 12 insertion/dihedral trials and a 6.5 Å cutoff.

To compare our results with the TRAPPE-UA forcefield, we need to validate the results. We validated this by running the Gibbs Ensemble Monte Carlo Simulations for the propane molecule with TRAPPE - UA forcefield parameters under the same simulation settings.



•

The above plot shows the comparison of the vapor-liquid equilibrium data with the GAFF and TRAPPE force fields.

The above plot and the equation used for fitting the data will give us properties such as the critical temperature and density.

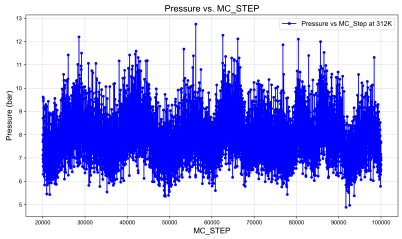
Force Field	T_c **(K)**	ρ_c **(g/cm³)**
TRAPPE (Validation Run)	366	0.21
TRAPPE Original	368	0.221
GAFF	368.5	0.22
Experimental	369	0.22

TABLE III

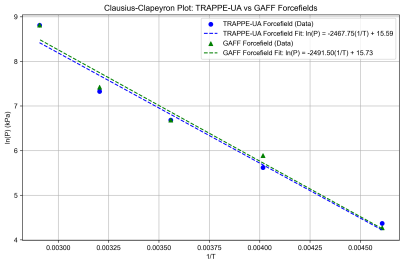
Critical Temperatures and Densities from the Simulations

Using the data from the Monte Carlo simulations, we can plot the Clausius-Clapeyron plot, which gives us an understanding of the vapor pressure and the enthalpy of vaporization.

First, we analyzed the variation of pressure at a specific temperature, 312K, to compute the average pressure, as the calculations in the ensemble techniques involve statistical uncertainty.



From the plot, we can estimate that the average pressure over the Monte Carlo steps is 7.87 bar at 312 K. Similarly, we computed the average pressure for all temperatures and plotted the curve between ln(P(kPa)) vs 1/TK-1).



The Clausius-Clapeyron equation was used to analyze the vapor pressure data for two force fields: TRAPPE-UA and GAFF. The natural logarithm of pressure, $\ln(P)$, was plotted against the inverse temperature, $1/T$, and a linear fit was applied to determine the slope and intercept. The enthalpy of vaporization (ΔH_{vap}) was calculated using the relation $\Delta H_{\text{vap}} = -\text{slope} \times R$, where $R = 8.314\,\text{J/mol·K}$ is the gas constant. The results are summarized below.

TABLE IV

CLAUSIUS-CLAPEYRON FIT PARAMETERS FOR TRAPPE-UA AND GAFF FORCEFIELDS

Forcefield	**Slope**	**Intercept**	ΔH_{vap} **(kJ/mol)**
TRAPPE-UA	-2467.75	15.59	20.52
GAFF	-2491.50	15.73	20.71

The data in the table show that the enthalpy of vaporization computed from both force fields is consistent with NIST data.

B. Initial and Final Snapshots of liquid and vapor phases of the Monte Carlo Simulations

Below is a series of figures showing the transition of a system from the liquid phase to the vapor phase, as observed in a molecular dynamics simulation. The system is confined within a cubic box, and the snapshots illustrate the molecules' initial, intermediate, and final states.

The figures demonstrated successful Gibbs ensemble Monte Carlo simulations of a liquid-to-vapor phase transition at 217 K. The initial liquid phase is characterized by a high density and strong intermolecular attractions, resulting in a closely packed molecular arrangement. The final vapor phase shows a low density and minimal intermolecular interactions, with molecules spread out sparsely across the simulation box. This transition likely occurred due to a change in thermodynamic conditions, such as a decrease in density or pressure, driving the system from a condensed phase to a gaseous phase. The simulation effectively captures the fundamental physics of phase transitions, including the role of intermolecular forces, density changes, and the influence of temperature, providing a visual confirmation of the expected behavior of the system under the given conditions.

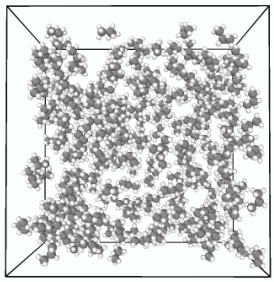


Fig. 2. Initial snapshot: Liquid phase at 217K

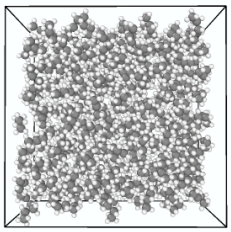


Fig. 3. Final snapshot: Liquid phase at 217K with dense arrangement of propane molecules

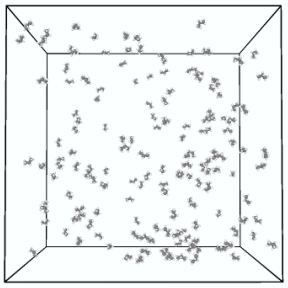


Fig. 4. Initial snapshot: Vapor phase at 217K

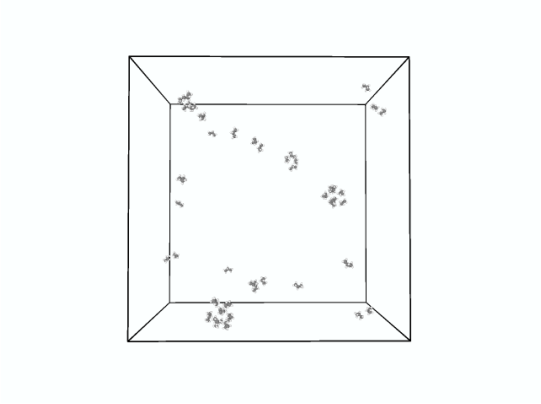


Fig. 5. Final snapshot: Vapor phase at 217K

IX. CONCLUSIONS

The results and comparisons with experimental and NIST data show that the GAFF force field accurately captures propane vapor-liquid equilibrium (VLE) behavior. The Monte Carlo simulations successfully modeled the fundamental physics of phase transitions, as demonstrated through the VLE curves and the Clausius-Clapeyron analysis. Furthermore, stochastic modeling — particularly through the Gibbs Ensemble Monte Carlo (GEMC) method — proves essential for VLE studies by efficiently sampling complex phase spaces, enforcing thermodynamic equilibrium, and enabling accurate ensemble averaging. In contrast, deterministic approaches are limited by computational cost, slow equilibration, and poor sampling of diverse configurations. Thus, GEMC's stochastic framework offers a robust and computationally efficient route for the reliable study of vapor-liquid equilibria. Additionally, we explored the basin transitions of the particles to sample the potential energy over the Boltzmann distribution, where sufficient and random sampling plays an essential role in capturing the phase space of the propane gas. This paper finally concludes that the stochasticity plays a crucial role in exploring the different configurations of the potential energy to sample it over the Boltzmann distribution and ensures ergodicity, where the entire phase space is eventually explored, and finally obtains the vapor-liquid equilibrium curve using Gibbs Ensemble Monte Carlo Simulations.

ACKNOWLEDGMENT

I am grateful to the High-Performance Computing Center at the University of Wyoming for providing me with the computational resources to run the extensive Monte Carlo simulations.

REFERENCES

[1] Murali, R. Surya, K. Yamuna Rani, T. Sankarshana, A. F. Ismail, and S. Sridhar. Separation of binary mixtures of propylene and propane by facilitated transport through silver incorporated poly (ether-block-amide) membranes. *Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles*, 70(2):381–390, 2015.

[2] Kiss, A. A., Lange, J. P., Schuur, B., Brilman, D. W. F., van der Ham, A. G. J., & Kersten, S. R. A. (2016). Separation technology—Making a difference in biorefineries. *Biomass and Bioenergy*, *95*, 296–309. Elsevier.

[3] Shah, J. K., Marin-Rimoldi, E., Mullen, R. G., Keene, B. P., Khan, S., Paluch, A. S., Rai, N., Romanielo, L. L., Rosch, T. W., Yoo, B., & Maginn, E. J. (2017). Cassandra: An open source Monte Carlo package for molecular simulation. *Journal of Computational Chemistry*, *38*(19), 1727–1739. Wiley Online Library.

[4] Frisch, M. J., Trucks, G. W., Schlegel, H. B., Scuseria, G. E., Robb, M. A., Cheeseman, J. R., Scalmani, G., Barone, V., Petersson, G. A., Nakatsuji, H., Li, X., Caricato, M., Marenich, A. V., Bloino, J., Janesko, B. G., Gomperts, R., Mennucci, B., Hratchian, H. P., Ortiz, J. V., Izmaylov, A. F., Sonnenberg, J. L., Williams-Young, D., Ding, F., Lipparini, F., Egidi, F., Goings, J., Peng, B., Petrone, A., Henderson, T., Ranasinghe, D., Zakrzewski, V. G., Gao, J., Rega, N., Zheng, G., Liang, W., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Vreven, T., Throssell, K., Montgomery Jr., J. A., Peralta, J. E., Ogliaro, F., Bearpark, M. J., Heyd, J. J., Brothers, E. N., Kudin, K. N., Staroverov, V. N., Keith, T. A., Kobayashi, R., Normand, J., Raghavachari, K., Rendell, A. P., Burant, J. C., Iyengar, S. S., Tomasi, J., Cossi, M., Millam, J. M., Klene, M., Adamo, C., Cammi, R., Ochterski, J. W., Martin, R. L., Morokuma, K., Farkas, O., Foresman, J. B., & Fox, D. J. (2016). *Gaussian 16 Revision C.01*. Gaussian Inc., Wallingford CT.

[5] Wang, J., Wang, W., Kollman, P. A., & Case, D. A. (2006). Automatic atom type and bond type perception in molecular mechanical calculations. *Journal of Molecular Graphics and Modelling*, *25*, 247–260.

[6] Wang, J., Wolf, R. M., Caldwell, J. W., Kollman, P. A., & Case, D. A. (2004). Development and testing of a general AMBER force field. *Journal of Computational Chemistry*, *25*, 1157–1174.

[7] Paquet, E., & Viktor, H. L. (2015). Molecular dynamics, Monte Carlo simulations, and Langevin dynamics: A computational review. *BioMed Research International*, *2015*(1), 183918. Wiley Online Library.

[8] Lopes, J. N. C., & Tildesley, D. J. (1997). Multiphase equilibria using the Gibbs ensemble Monte Carlo method. *Molecular Physics*, *92*(2), 187–196. Taylor & Francis.

[9] Siepmann, J. I., & Frenkel, D. (1992). Configurational bias Monte Carlo: A new sampling scheme for flexible chains. *Molecular Physics*, *75*(1), 59–70. Taylor & Francis.

[10] Binder, K., & Heermann, D. W. (1992). *Monte Carlo Simulation in Statistical Physics* (Vol. 8). Springer.

APPENDIX

APPENDIX: ALGORITHMS

Algorithm 4: Metropolis Acceptance Criterion

Compute energy difference ΔU
Compute log acceptance probability: $\ln p_{\text{acc}} = -\beta \Delta U$
if $\ln p_{acc} > 0$ **then**
    Accept move
rand$(0,1) < e^{\ln p_{\text{acc}}}$
Accept move **else**
    Reject move

Algorithm 5: Langevin Monte Carlo Sampling

Initialize X_0 **for** $t = 1$ *to* T **do**
    Sample Brownian noise: $\xi_t \sim \mathcal{N}(0, I)$
Update: $X_{t+1} = X_t - \nabla U(X_t)\Delta t + \sqrt{2\Delta t} \cdot \xi_t$

Algorithm 6: Gibbs Ensemble Monte Carlo Move (GEMC)

Randomly select a move type: translate, rotate, volume exchange, or molecule swap
Propose trial configuration
Compute ΔU and other relevant terms (e.g., ΔV)
Compute log acceptance probability **if** *Accepted* **then**
    Apply move **else**
    Reject move

Algorithm 7: Volume Exchange Move Acceptance

Compute: $\ln p_{\text{acc}} = \beta \Delta U + \beta P \Delta V + N \ln\left(\frac{V_m}{V_n}\right)$
Accept move if $\ln p_{\text{acc}} > \ln(\text{rand}(0,1))$

Algorithm 8: Configurational Bias Monte Carlo (CBMC) Insertion

Parse molecule into fragments
For each fragment:
- Sample internal angles with Boltzmann weight
- Generate trial conformations
- Accept one based on Metropolis probability

Combine fragment contributions to compute total log acceptance probability
Accept or reject based on $\ln p_{\text{acc}}$