

Facilitating Students' Interaction with Real Gas Properties Using a Discovery-Based Approach and Molecular Dynamics Simulations

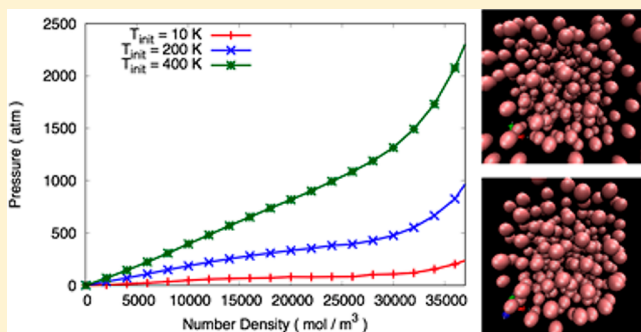
Chelsea Sweet,[†] Oyewumi Akinfenwa,[†] and Jonathan J. Foley, IV^{*†}

Department of Chemistry, William Paterson University, Wayne, New Jersey 07470, United States

S Supporting Information

ABSTRACT: We present an interactive discovery-based approach to studying the properties of real gases using simple, yet realistic, molecular dynamics software. Use of this approach opens up a variety of opportunities for students to interact with the behaviors and underlying theories of real gases. Students can visualize gas behavior under a variety of conditions; they can implement kinetic theory expressions as computer code to compute macroscopic quantities from microscopic/kinematic behaviors of the simulated gases, and they can explore strategies to quantitatively and qualitatively evaluate deviations from ideal gas behavior. A simple and fully functional molecular dynamics program written in the C programming language that has been tested on Windows, Mac, and Linux platforms accompanies this article; detailed suggestions are provided for implementing this exercise with or without programming activities.

KEYWORDS: Upper-Division Undergraduate, Physical Chemistry, Computer-Based Learning, Inquiry-Based/Discovery Learning, Gases, Kinetic-Molecular Theory



Gases are introduced in the elementary science curriculum as one of the fundamental states of matter, providing perhaps the first notion to students that matter is all around us all the time.¹ Chemists in particular encounter the ideal gas law in a variety of contexts throughout their training. The ideal gas law is an important part of the particle theory of matter, and is useful in illustrating and quantifying the relationships between different measurable quantities that characterize a gas system.^{1,2} The ideal gas law is also used extensively to motivate thermodynamic processes and to analyze the changes in energy and entropy associated with them.^{2,3} Gases also provide arguably the simplest regime for studying intermolecular interactions.^{3,4} The fact that the ideal gas law has such remarkable predictive power in a variety of circumstances despite its neglect of intermolecular interactions illustrates important concepts about the distance dependence of these forces. That is, the average intermolecular distance is large in low-density gases; therefore, the average intermolecular force is negligible. Despite repeated exposure, undergraduate chemistry students struggle to develop both a functional understanding of the ideal gas law and a solid conceptual understanding of the assumptions that define the ideal gas model (noninteracting point masses that undergo elastic collisions with the walls of their container).^{1–5} We have observed in our own physical chemistry curriculum that students also struggle to develop a functional understanding of “real gas” equations of state (e.g., van der Waals equation) and sound physical intuition about the microscopic intermolecular forces in gases. Large deviations from ideal behavior can occur in highly compressed gases, and

so a functional understanding of real gas properties and equations of state is of high practical value across a variety of industries that produce or rely on gaseous reagents, including the petroleum industry and industries that rely on nanofabrication via atomic layer deposition, to name a few.

In this article, we present a molecular dynamics (MD) simulation module that can aid discovery-driven study of monatomic gases across the continuum of regimes between ideal and non-ideal behavior. We have utilized this program for physical chemistry laboratories where students discover the conditions under which deviations from ideality are observed in terms of qualitative behaviors observed in 3-D animations derived from the simulations, and in terms of quantitative observables such as the macroscopic compressibility ($Z = PV/nRT$) that can be computed from the simulation.^{3,6,7} Students may also use the simulation to explore connections between macroscopic indicators of non-ideality and microscopic quantities like the potential energy and kinetic energy. We have found that the exercises described in this article are helpful to students by (i) enabling them to use computer visualization to refine their mental models of real gas behavior, (ii) enabling them to connect quantitative measures of deviations from ideality to qualitative behavior as observed through animation of MD simulations, and (iii) empowering them to design

Received: September 25, 2017

Revised: January 17, 2018

numerical experiments to test their reasoning and intuition in ways that would be challenging in an experimental setting.

Directions for obtaining, compiling, and running the code on Windows, Mac, and Linux platforms are distributed as [Supporting Information](#). The simulation is based upon classical MD simulation of particles that interact through the Lennard-Jones potential.^{6–9} In this technique, Newton's equations of motion are solved for all particles to render their motion in 3-dimensional space. Our code outputs the coordinates of each particle at each step in time in a format that can be readily rendered as an animation by the open-source program VMD;¹⁰ instructions on obtaining and using VMD are also provided as [Supporting Information](#). Kinetic theory is used to compute macroscopic quantities including the temperature and pressure from the time average of the MD trajectories.^{3,6} Microscopic quantities like the kinetic and potential energy of individual particles are also readily computed.

In the remainder of the paper, we provide a brief overview of how MD simulations have impacted both the study of chemical and material systems, and the teaching and learning of chemistry. We discuss how quantities relevant to the study of gas properties may be extracted from MD simulations, referencing a discussion of the salient theoretical and computational details of MD simulations that is provided in the [Supporting Information](#). We provide several suggestions for how the MD simulation module, written by the authors and freely accessible to readers,¹¹ may be used to facilitate discovery-driven exercises that can empower students to explore connections between microscopic and macroscopic properties of gases. We discuss our implementation of these exercises for several cohorts of physical chemistry students and provide a summary of student feedback.

■ BRIEF OVERVIEW OF MOLECULAR DYNAMICS SIMULATIONS

Our simulation module draws inspiration from the pioneering work of Aneesur Rahman, who reported the first molecular dynamics simulation that utilized a physically realistic interparticle potential in 1964.⁸ Rahman's seminal paper described a simulation of 864 argon atoms interacting through a model potential energy function, known as the Lennard-Jones potential, and showed that many known experimental properties could be reproduced by such simulations with surprising fidelity.⁸ A tremendous amount of progress has been made in the development and application of MD methods in the decades since its invention so that now this method impacts a variety of societally relevant problems in the chemical, biological, and material sciences.^{12–14} Distributed computing paradigms such as Folding@Home have enabled citizen scientists to contribute their home computers to simulate protein behavior to further the understanding of diseases such as Alzheimer's.¹⁵ Theoretical and computational advances have also improved the overall efficiency of MD simulation of multiscale problems so that complex and rare events like ligand binding¹⁶ and the behaviors of trans-membrane proteins¹⁷ can now be simulated. Parallel efforts in the field have also dramatically improved the accuracy with which the forces between particles can be modeled, which has enabled MD to be used as a tool which can both explain and predict the behaviors of molecules in extreme environments. The development of so-called reactive force fields, which model bond-breaking and -making events,¹⁸ have facilitated the discovery of novel nanoscale phenomena like superlubricity in graphene.¹⁹ A

class of simulations known as *ab initio* molecular dynamics use rigorous quantum chemical methods to compute intermolecular forces during the runtime of the simulation, and have recently been leveraged to *discover* novel reaction pathways under extreme conditions.²⁰

MD Simulations in Chemical Education

As the use of MD simulations has become increasingly important in the chemical and biological sciences, it has also seen increased use in chemical and biological pedagogy, both to expose students to this increasingly important tool, and to leverage some of its unique capabilities to achieve specific learning goals.^{21–33} Molecular dynamics simulations, and molecular modeling techniques more generally, have been integrated into many aspects of the undergraduate science curricula with a variety of positive outcomes.^{7,21–37} Two common features of many of these activities are that they: (i) leverage visualization capabilities to help students refine mental models and develop an intuition for underlying physical and chemical phenomena and (ii) permit students to study observable quantities that would be difficult to measure experimentally. For example, Speer et al. described activities where physical chemistry students used the Amber³⁸ MD suite to simulate water, ethanol, and benzene to illustrate connections between intermolecular forces and temperature on liquid structure as quantified by the radial distribution function, a quantity whose experimental determination via X-ray scattering would not be feasible in many undergraduate laboratories.²¹ Students using these activities are also able to visualize the liquid structure by animations of the MD trajectories, reinforcing their intuition regarding how intermolecular forces impact order in the liquid phase. Eckler and Nee also implemented exercises within the Amber³⁷ suite enabling students to calculate the temperature dependence of viscosity in *n*-hexane and draw visual connections between this viscosity and the microscopic behavior of the liquid via animations of the MD trajectories.²² Schroeder implemented an interactive web-based MD simulator in 2D that illustrates a variety of rich phase behavior that can be realized by Lennard-Jones particles in two-dimensions and suggested their use for beginning and intermediate physics courses.⁷ Schroeder's web-based simulator has presets that allow students to explore the behavior of particles under extreme conditions that give rise to explosions, chaos, and other interesting physical phenomena. Cole and co-workers described the use of the Odyssey suite to facilitate exploration of intermolecular forces and their implications for micelle formation, forming an important bridge between physical chemistry concepts and topics typically discussed in biology and biochemistry courses.²³ There have also been several recent examples of utilizing MD simulation to help students understand dynamical processes in chemical and biological systems. Fuson has recently described complementary experimental/computational activities that enable students to visualize the dynamical processes involved in the spin relaxation of bromobenzene through MD simulations.²⁸ Messina and co-workers have described a discovery-driven module where students use bioinformatics tools to identify protein regions of interest in which to introduce site mutations, and subsequently use MD simulations to visualize the structural and dynamical differences that result from these mutations.³⁰ Hati and Bhattacharyya have described the use of MD simulations in discovery-driven laboratories that help students connect protein structure to protein function via their

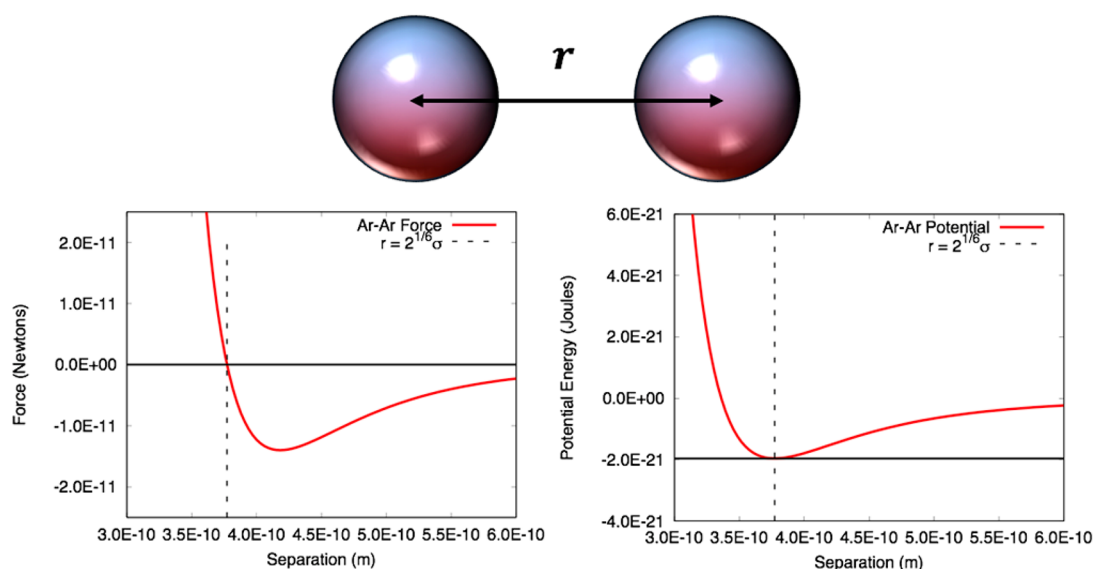


Figure 1. Illustration of Lennard-Jones force (left plot) and Lennard-Jones potential (right plot) as a function of separation between two argon atoms. All quantities are plotted in SI units where $\epsilon = 1.961 \times 10^{-21}$ J and $\sigma = 3.605 \times 10^{-10}$ m for argon. Note that the potential energy is a minimum, and the force is zero, at the separation $r = 2^{1/6}\sigma$. A complete table of L-J parameters in SI units for the noble gases is provided in [Table S1](#) in the Supporting Information.

dynamical behaviors.³¹ Several reports in this *Journal* have also discussed the use of MD simulations to explore intermolecular and intramolecular forces from a microscopic point of view^{25–27} while other authors have discussed ways to involve undergraduates in the development of MD simulations tools.^{27,28}

It is the aim of this article to describe several exercises that we have implemented that draw on the strengths of MD simulation to reinforce critical concepts of intermolecular forces and real gas behavior. We have found that these exercises are helpful to students by (i) enabling them to use computer visualization to refine their mental models of real gas behavior, (ii) enabling them to connect quantitative measures of deviations from ideality to qualitative behavior as observed through animation of MD simulations, and (iii) empowering them to design numerical experiments to test their reasoning and intuition in ways that would be challenging in an experimental setting. The described exercises are typically conducted in the third or fourth week of Physical Chemistry I, after students have been introduced to kinetic theory of gases and reminded of intermolecular forces.

Real Gas Properties from Molecular Dynamics Simulations

The underlying concept behind MD simulations is surprisingly simple: atoms and molecules exert forces on one another that affect their motion, and the changes in motion due to these forces can be predicted to high accuracy by solving classical equations of motion, namely, Newton's second law.^{6–9} This concept is similar in spirit to the types of kinematics problems undergraduate students encounter in their general physics course: identify the forces acting on a particle of known mass (e.g., a baseball), and then find its position and velocity at some time in the future given its current position and velocity. The two challenges in the case of MD simulations are (i) the calculation of the appropriate forces on the particles and (ii) the solution of the equations of Newton's second law for the very large number N of particles that arise in typical simulations. In the simulations we describe here, monatomic noble gases interact with each other through forces that arise

from a model potential known as the Lennard-Jones potential,^{6–8} which has the following form:

$$U(r_{ij}) = 4\epsilon \left(\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right) \quad (1)$$

The force on particle i arising from interactions with all other particles is defined as^{6–8}

$$\vec{F}_i = - \sum_{j \neq i}^N \frac{\partial U(\vec{r}_{ij})}{\partial \vec{r}_{ij}} \quad (2)$$

The separation-dependence of the forces arising from the Lennard-Jones potential as well as the potential energy function itself are shown in [Figure 1](#) for argon. We provide the Lennard-Jones parameters, taken from refs 7 and 39, for the noble gases in [Table S1](#) in the Supporting Information. An activity employing quantum chemistry calculations to estimate the parameters in the Lennard-Jones potential has been previously described in this *Journal*,³⁶ and could augment the simulations described here for courses focusing on computational chemistry in more detail. Further details on the Lennard-Jones potential, interparticle forces, and numerical foundations of MD simulations within the context of this study are discussed in the [Supporting Information](#).

MD simulations solve for information about the instantaneous positions and velocities of all particles being simulated at every time step of the simulation; this information constitutes the molecular dynamics trajectory. Our simulation module computes trajectories for noble gas particles constrained by a cubic box with a volume that is specified by the user at runtime through the number density (the number of particles is hard-coded to be 216). The thermodynamic state functions temperature and pressure may be readily extracted from the MD trajectories of these systems.

Pressure is defined as force divided by area,

$$P = \frac{F}{A} \quad (3)$$

To determine the pressure of a gas, one needs to compute the average force exerted by the gas molecules on the walls of its containers divided by the total surface area of the container. According to kinetic theory, instantaneous forces arise from the changes in particle momentum associated with the gas particles' elastic collisions with their "container walls"; that is, the cubic box that constrains the particles. All gas particles in the simulation are given initial positions within the boundaries of this box, and if a gas particle's position reaches the boundary along a particular axis (e.g., the x -axis), the corresponding (x -) component of velocity is simply reversed, simulating an elastic collision (see Figure 2). An instantaneous force exerted by the

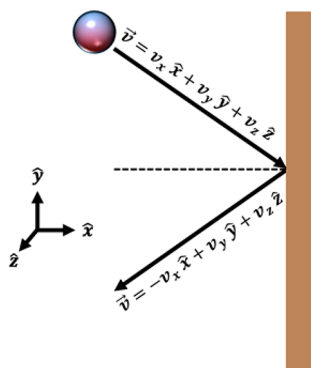


Figure 2. Schematic of an elastic collision of a particle with its container wall. The particle reaches the container boundary along the x -axis, and so it experiences a change in its x -component of velocity. This change in velocity is associated with a change in the x -component of momentum, as well. The sum over all such momentum changes in a given time interval can be interpreted as the instantaneous pressure within the kinetic theory of gases.

particles on the wall ($F_{\text{wall}}(t)$) can be defined by accounting for all such momentum changes that occur in a particular time interval, and an instantaneous pressure can be defined as this instantaneous force divided by the total surface area of the container walls, $P(t) = F_{\text{wall}}(t)/A$. An explicit expression for $F_{\text{wall}}(t)$ in terms of particle velocities is given in the Supporting Information (see eq S6). Figure 2 illustrates a particle colliding elastically with a wall and experiencing a change in the x -component of its velocity vector; this collision results in a change in the x -component of the particle's momentum vector with magnitude $\Delta p_x = 2mv_x$.

Similarly, within kinetic theory, the instantaneous temperature can be related to the mean squared velocity at a given instant in time by

$$T(t) = \frac{m\langle v^2(t) \rangle}{3k_B} \quad (4)$$

where $\langle v^2(t) \rangle$ denotes the mean squared velocity (see Supporting Information, eq S8), m is the particle mass, and k_B is Boltzmann's constant. The thermodynamic pressure and temperature can be approximated as the time average of the instantaneous $T(t)$ and $P(t)$ over the duration of the simulation; for temperature, this average is computed as

$$T = \frac{1}{N_{\text{steps}}} \sum_{i=1}^{N_{\text{steps}}} T(t_i) \quad (5)$$

and the pressure average is computed analogously.

It is important to note that although the user supplies an initial temperature, this temperature does not necessarily correspond to the thermodynamic average temperature of the simulation. The initial temperature is used to determine the initial distribution of particle velocities via the Maxwell–Boltzmann distribution.^{3,6} However, because the particles interact with each other through the Lennard-Jones potential (see Figure 1 and eq 1), the energy is partitioned among potential and kinetic energy, so the average kinetic energy, and therefore the temperature, may change as the system comes to equilibrium. For simulation parameters in which the particle experiences strong interactions with each other, the temperature defined in eq 5 may deviate significantly from the initial temperature specified by the user.

MOLECULAR DYNAMICS SIMULATION FOR DISCOVERY-DRIVEN STUDY OF GASES

Implementation in Brief

Here we provide a brief overview of the practical implementation of the activities in a classroom setting, including required software and hardware resources for running the simulation, and suggested allotment of time for instruction and the discovery-driven activities.

Required Resources

Software Requirements. The molecular dynamics source code (written in standard C code) has been tested on Windows (running the terminal emulator *cygwin*), Mac, and Linux platforms with the software requirements given in Table 1.

Table 1. Required Software for Implementing MD Activities

Software Programs ^a	Purpose
GNU Compiler Collection	Compiles molecular dynamics source code
Make	Used in conjunction with GCC for compilation of molecular dynamics source code
git	Used to obtain molecular dynamics source code ^b
VMD	Used to visualize MD trajectories
Cygwin (Windows only)	Emulates terminal environment on Windows systems, installs GCC, Make, and git

^aSee the Supporting Information document, Software Installation Instructions, for more details on obtaining and downloading these free resources. ^bSee ref 11.

Hardware Requirements. The hardware requirements are minimal and should be satisfied by any relatively modern Windows, Mac, or Linux desktop, laptop, or workstation. Simulation runtime will vary by processor speed, and memory and storage requirements should be uniform across machines. Resource requirements observed for a single simulation on a MacBook Pro with a 2 GHz Intel Core i5 are as follows:

- Simulation runtime: ~1 min
- Simulation memory requirements: <1 MB
- Hard drive requirements for storage of output data: ~125 MB

Suggested Allotment of Classroom/Laboratory Time

Table 2 provides a recommended implementation of the activities broken down by objective and meeting time.

Table 2. Suggested Implementation of Discovery-Driven Activities and Optional Programming Activities^a

Objective	Format	Approximate Duration
Introduction to MD Simulations	Prelab lecture given by instructor	1 h
Demo of software workflow (compilation, execution, data analysis)	Lab meeting 1: Instructor gives demo	10 min
Discovery-Driven Activities, Part 1: Qualitative/Quantitative assessment of real gas behavior	Lab meeting 1: Students working in small groups	Data collection, discussion, and analysis: 2 h
Discovery-Driven Activities, Part 2: Testing the limits of the ideal gas law and Computing quasi-isotherms	Lab meeting 2: Students working in small groups	Data collection, discussion, and analysis: 2 h
(Optional) Introduction to essentials of programming in C/C++	Lab meeting 3: Instructor leads hands-on introduction and gives assignments to work on at home to practice essential skills	Lab meeting: 2 h
(Optional) Programming kinetic theory equations	Lab meeting 4: Instructor leads formulation of kinetic theory equations in pseudo code, students work in small groups to implement code	Homework: 1–2 h Lab meeting: 2 h
		Homework (validation, data collection, and analysis): 1–2 h

^aDiscovery-driven activities, including several guiding questions, are described in the [Detailed Discussion of Implementation](#) section of this article. More details to guide the development of lecture material on the principles of MD simulations can be found in the [Supporting Information](#). Details on the software workflow can be found in the [Supporting Information](#); optional programming exercises can be found in the [Supporting Information](#).

Detailed Discussion of Implementation

Realistic computer simulation offers many opportunities to study gas properties and behavior that would not be feasible in the undergraduate experimental laboratory. In particular, it offers access to microscopic insight and yields high-precision information about macroscopic quantities provided the simulations are suitably performed. Here we describe several molecular dynamics simulations that can be implemented in physical chemistry courses that can be performed without modification to the software. Using default simulation parameters, 216 noble gas particles (He, Ne, Ar, Kr, or Xe can be chosen by the user at runtime) are simulated for 0.1 ns; each simulation takes approximately 1 min of computer time running on a 2 GHz Intel Core i5. The user may also specify an initial temperature and a number density at runtime, along with a title for the simulation to be used for naming all output files. Time averages from the simulations are used to compute the thermodynamic temperature and pressure, from which the compressibility and other related quantities are calculated.

We use the following guiding questions to motivate students to design simulations to explore real gas behavior:

- In certain cases, the ideal gas law ($P = nRT/V$) can fail to accurately predict the pressure of a gas given the temperature and number of moles of gas are known, and the volume of the container is known. Identify the conditions that are likely to lead to failure of the ideal gas law:
 - High number density (n/V), low temperature, strong (attractive) intermolecular forces
 - Low number density (n/V), moderate temperature, weak intermolecular forces
- For the choice you made above, explain why you think the ideal gas law will fail. Further explain in which way it will fail; that is, will the predicted pressure be too high or too low compared to actual pressure?
- One way of quantifying the error in the ideal gas law is to measure the pressure and temperature of a sample of gas with known number density and computing the quotient

$$Z = \frac{PV}{nRT}$$

where Z is known as the compressibility. When $Z = 1$, the ideal gas law can be expected to be accurate, whereas values larger or smaller than 1 indicate deviations from ideal behavior. For high number density, low temperature, and strong attractive intermolecular forces, do you expect the compressibility to be greater than 1, less than 1, or equal to 1?

- The noble gases can be ordered from weakest to strongest attractive intermolecular forces as follows: He < Ne < Ar < Kr < Xe. Design a set of MD simulations to test this ordering: Does the compressibility provide quantitative evidence that is consistent with this ordering in your simulations? Animate each simulation and comment on any differences you notice in the animations.

We describe four exercises (Qualitative Assessment of Ideal and Non-Ideal Behavior; Quantitative Assessment of Ideal and Non-Ideal Behavior; Testing the Limits of the Ideal Gas Law; and Computing Quasi-Isotherms) that can be run with the MD module as-is to support students' answers to the above questions. In the [Supporting Information](#), we describe an

additional exercise (Programming Kinetic Theory Equations) that can engage students in writing code, for example, in advanced courses that cover computational chemistry content. Details on executing the simulation steps in a terminal environment on PC, Mac, and Linux systems are provided in the [Supporting Information](#) with screenshots to aid users who may be unfamiliar with terminal environments. We provide instructions for installing the program Cygwin on a PC to facilitate the use of the MD simulation in a terminal environment within Windows as a supplement, as well. While this protocol is intended to be detailed enough to allow students to run the simulations with no directions from their instructors, we have found that a brief demonstration from the instructor enables virtually all students to quickly become self-sufficient with the software. A demonstration of steps 1–7 of the protocol can be given by the instructor in approximately 10 min.

One of the authors has implemented versions of these exercises with 5 sections of students enrolled in Physical Chemistry I (Thermodynamics and Kinetics) at William Paterson University since Fall of 2015 for a total of approximately 60 students. Our students have completed the MD simulations on Dell PCs running Windows 10, iMacs running MacOS Sierra, and multiuser workstations running Fedora 22 with similar runtimes in all cases (between 1 and 2 min per simulation). Students enrolled in Physical Chemistry I have completed a general physics course focusing on mechanics and kinematics, and have completed (or are coenrolled in) a multivariable calculus course. They have encountered the ideal gas equation of state and qualitative models of intermolecular forces in their general chemistry coursework, but may not be aware of the van der Waals equation of state or the Lennard-Jones potential. The implementation of these exercises has been refined on the basis of feedback received from student participants, as well as on the observations and judgements of the authors. In the final section of this paper, we provide a summary of feedback based on a survey given to two recent sections of Physical Chemistry students on the current implementation (one taught by one of the authors, J.J.F., one taught by a professor who is not an author of this paper).

Qualitative Assessment of Ideal and Non-Ideal Behavior

An extremely powerful aspect of molecular dynamics simulations is that they yield physically meaningful trajectories of chemical systems that can be rendered as movies, and these movies enable investigators to visualize the behavior of complex systems in ways that are extremely difficult, or even impossible, using experimental means. Our students have used the accompanying MD program in conjunction with the free visualization software VMD¹⁰ to animate their MD trajectories, which has helped them to refine their mental models of intermolecular forces and gaseous behavior (Figure 3). Each time a molecular dynamics simulation is run, the entire MD trajectory is written in a VMD-friendly format to a file called `title_traj.xyz` (`title` is specified by the user at runtime); the user is also prompted to open this file with VMD at the end of the simulation. For details on installing VMD, see the accompanying [Software Installation Instructions](#).

We suggest instructors encourage students to use animations of MD trajectories to help them answer the guiding questions 1–4 above. Students may be asked to identify the density and temperature conditions that give rise to the three distinct types of gas behavior:

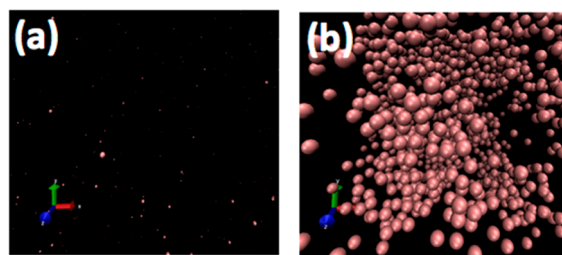


Figure 3. Animation images of molecular dynamics simulation of 1000 Ar atoms. Panel a shows Ar simulated under low-density conditions where nearly ideal behavior is observed ($Z \approx 1$). Panel b shows Ar simulated under higher density and low pressure where deviations from ideality are observed due to attractive forces ($Z < 1$). In Panel b, the presence of Ar clusters (dimers, trimers, etc.) can be seen. Students should be prompted to take note of the formation and persistence of these clusters as part of their qualitative assessment of ideal and non-ideal behavior. Note that the default number of particles is 216; the simulation conditions pictured here may be obtained by simple modification of the source code to increase the number of particles.

1. Ideal behavior
2. Non-ideal behavior where attractive forces dominate
3. Non-ideal behavior where repulsive forces dominate

Students may be guided to observe features of the gas behavior, including:

- The average separation of the gas particles
- The frequency of wall collisions relative to the frequency of interparticle collisions
- The apparent speed of the gas particles
- The persistence of particle clusters (dimers, trimers, etc.) resulting from attractive interactions (see [Figure 3](#))
- The frequency of collisions that lead to apparent acceleration due to repulsive interactions

Furthermore, students may be asked to hypothesize about the origin of these qualitative behaviors and devise further experiments with the simulation parameters to test their hypotheses. For example, a student may hypothesize that the frequency of “repulsive collisions” may increase with increasing average velocity of the particles, and could test this hypothesis by running a series of simulations with progressively larger initial temperatures.

Quantitative Assessment of Ideal and Non-Ideal Behavior

Guiding questions 3 and 4 above challenge students to quantify non-ideal behavior through the compressibility, which is computed and printed by our MD module. We find that reporting the macroscopic compressibility associated with each simulation is an important complement to making qualitative observations based on MD trajectories. The macroscopic compressibility, defined as $Z = PV/Nk_B T = PV/nRT$, can be seen as a quantitative measure of ideal and non-ideal behavior.³ For ideal gases, $Z = 1$; $Z > 1$ for non-ideal gases where repulsive forces dominate, and $Z < 1$ when attractive forces dominate. This quantity can serve as an anchor for students in their qualitative assessment of gas behavior, as well. For example, students may be asked to describe the differences between a non-ideal gas system where $Z \approx 0.9$ and a system where $Z \approx 0.7$ rather than attempt to identify attractive behavior, broadly speaking.

While the compressibility serves as a macroscopic indicator of deviations from ideality, the gas systems potential and kinetic energy is also computed by our MD module and written to a

data file called *title_output.txt* (*title* supplied by the user at runtime). This information provides a measure of ideality that is microscopic in origin. Tracking these quantities along with the compressibility will reinforce valuable concepts about the dominance of the kinetic energy in gas systems that are nearly ideal, which is why the ideal gas model works so well. The MD module prompts the user to analyze this data in Excel at the end of each MD simulation. For more information about the data provided in *title_output.txt*, see step 14 of the Protocol.

Testing the Limits of the Ideal Gas Law

A remarkable feature of molecular dynamics simulations is that they can reproduce macroscopic properties with surprising accuracy even with a modest number of particles in the simulation. For example, the universal gas constant can emerge from the quotient $R = PV/nT = N_A PV/NT$ even for the default $N = 216$ provided the conditions of the simulation are such that ideal behavior is approximately observed.

The quotient $N_A PV/NT$ is computed by the MD module and printed for investigation by the user, who may be prompted to comment on its agreement with the universal gas constant when ideal behavior is expected. Note that the agreement of this quotient with the universal gas constant will be limited in precision by the number of significant figures to which the gas properties (mass and Lennard-Jones parameters) are specified; this could be used to prompt discussion of propagation of error. For simulation conditions that result in deviations from ideal behavior, students may be asked to evaluate the accuracy of equations of state that account for interparticle interactions, such as the van der Waals equation of state:

$$\frac{1}{nT} \left(P + a \left(\frac{n}{V} \right)^2 \right) (V - nb) = R \quad (6)$$

Computing Quasi-Isotherms

We encouraged our students to design systematic numerical experiments to address the guiding questions 1–4 above. One such systematic investigation of the impact of simulation conditions on real gas behavior involves computing plots of pressure versus number density for systems initialized at a given temperature, which we refer to as quasi-isotherms. True isotherms would require a thermostat for the simulation to approximate a constant NVT ensemble, whereas these simulations approximate constant NVE ensembles. Nevertheless, these quasi-isotherms provide students with another means of visualizing ideal and non-ideal regimes. For example, the point on the 400 K quasi-isotherm labeled A (see Figure 4) can be examined to reveal it has a compressibility larger than 1 and a pressure which is greater than would be predicted by the ideal gas law. Visualization of this molecular dynamics trajectory (Figure 4A) reveals vigorous collisions between the particles and with the container walls. On the other hand, the point on the 10 K quasi-isotherm labeled B (see Figure 4) has a compressibility less than 1 and a pressure which is less than would be predicted by the ideal gas law. Visualization of this molecular dynamics simulation reveals liquid-like behavior and wall collisions which are much less vigorous (see Figure 4B). In principle, students could compare these quasi-isotherms to true isotherms from ideal and real gas equations of state, but again we caution there will not be a perfect correspondence given the NVE nature of the MD simulations.

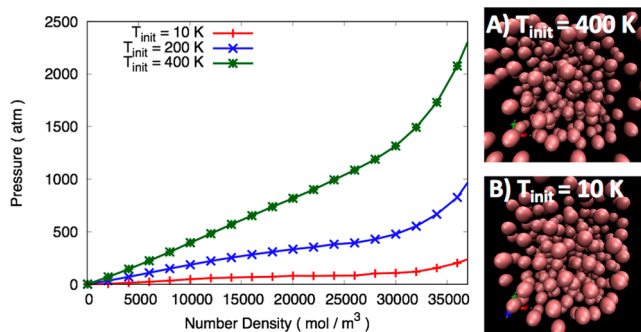


Figure 4. (Left panel) Several quasi-isotherms of argon gas showing the relationship between pressure (plotted in atmospheres) and number density (plotted in moles per meter cubed) for a given initial temperature of a MD simulation. (Right panel A) Snapshot from an MD simulation with number density of 35000 mol per meter cubed initialized at 400 K compared to (right panel B) snapshot with the same number density initialized at 10 K. In the instant of the snapshot, many more particles are colliding with the wall in A than in B, which represents well the origin of the lower pressure in B compared to A.

SUMMARY OF STUDENT FEEDBACK

One of the authors has implemented versions of these exercises with five sections of students enrolled in Physical Chemistry I (Thermodynamics and Kinetics) at William Paterson University since Fall of 2015 for a total of approximately 60 students impacted. The implementation of these exercises, particularly with respect to how students interact with the software and how questions are used to drive discovery-driven learning, has been refined on the basis of feedback received from student participants, as well as the observations and judgements of the authors. In earlier implementations (Fall 2015 and Fall 2016), more emphasis was placed on implementing kinetic theory equations as computer code, and less emphasis was placed on utilizing the MD tools to explore the phenomenology of the gas. While several students appreciated getting hands-on experience writing computer code, we found many students were frustrated by the numerous practical obstacles preventing them from successfully implementing their code. In many contexts, the programming exercises can provide impediments for students in terms of the objectives of (i) enabling them to use computer visualization to refine their mental models of real gas behavior, (ii) enabling them to connect quantitative measures of deviations from ideality to qualitative behavior as observed through animation of MD simulations, and (iii) empowering them to design numerical experiments to test their reasoning and intuition in ways that would be challenging in an experimental setting.

In Fall 2017, these exercises were implemented with two sections of Physical Chemistry with the guiding questions (1–4, above); the programming exercises were not implemented with these two sections. In these implementations, one section was led by one of the authors, J.J.F., and one section was led by a professor unaffiliated with this paper. We provided an end-of-lab survey to both sections (15 students total) to gauge the effectiveness of the MD module in (i) empowering them to engage in enquiry and design experiments, (ii) helping them to refine their mental models of macroscopic real gas behavior, and (iii) helping them to refine their mental models of microscopic intermolecular forces. Students were asked to rank the effectiveness of the tool on a scale of 1–3 on each of these dimensions, with 1 being not helpful at all, 2 being helpful but

slightly confusing, and 3 being helpful. The results of this survey are summarized in Figure 5.

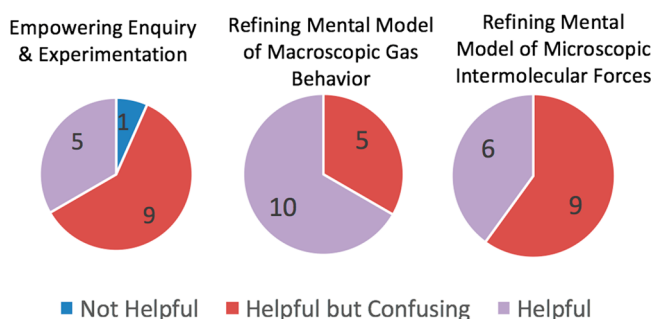


Figure 5. Summary of student survey responses from two Physical Chemistry sections that utilized MD simulations to answer questions about real gas behavior.

Overwhelmingly, the students found the MD module helpful in refining their mental model of macroscopic gas behavior. Students also found the simulations helpful in refining their mental model of microscopic intermolecular forces, though with a large number of students identifying at least some confusion. Students seemed to self-report the most confusion on the dimension that relates to empowering enquiry and experimentation; however, we note that all students were able to justify answers to the guiding questions (1–4 above) by making reference to simulations that they chose parameters for with little to no guidance from the instructor. In other words, students naturally adopted roles that are characteristic of constructivist learning cycles that discovery learning seeks to facilitate.^{40–42} We also note that the students participating in this lab have had little to no past exposure to discovery-driven pedagogy; the process of designing their own protocols to support enquiry is therefore new and challenging to the majority of these students. As noted by Ditzler and Ricci in their discussion of the “Discovery Chemistry” curriculum at College of the Holy Cross, there is some latitude inherent in such approaches with respect to the level of instructor guidance;⁴³ from this must necessarily follow some degree of variation in the level of intellectual discomfort experienced by the students during successful engagement with these exercises. The implementation described herein attempts to be cognizant of this latitude in the hopes that instructors represent the spectrum of perspectives on implementation of discovery-driven pedagogy, and their students will benefit from this MD simulation tool and associated discovery-driven activities.

■ ASSOCIATED CONTENT

§ Supporting Information

The Supporting Information is available on the ACS Publications website at DOI: 10.1021/acs.jchemed.7b00747.

Theoretical and computational details of molecular dynamics simulations in general, including a possible programming exercise (PDF, DOCX)

Detailed instructions for compiling and running the code, and rendering animations of the MD trajectories using the open-source software VMD¹⁰ (PDF, DOCX)

Instructions for installing supporting software, including VMD for visualizing MD trajectories and Cygwin for running a terminal environment natively in Windows (PDF, DOCX)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: foleyj10@wpunj.edu.

ORCID

Jonathan J. Foley IV: 0000-0001-8814-4444

Author Contributions

†C.S. and O.A. contributed equally to this work.

Notes

The authors declare no competing financial interest.

A well-commented and fully functional version of our molecular dynamics code written in the C programming language, along with a makefile for easy compiling, can be downloaded from github: <https://github.com/FoleyLab/MolecularDynamics> (accessed Jan 2018).

■ ACKNOWLEDGMENTS

J.J.F. gratefully acknowledges startup support from the College of Science and Health, and financial support from the Center for Research at William Paterson University.

■ REFERENCES

- (1) Novick, S.; Nussbaum, J. Pupils' Understanding of the Particulate Nature of Matter: A Cross-Age Study. *Sci. Educ.* **1981**, *65*, 187–196.
- (2) Kautz, C. H.; Heron, P. R. L.; Loverude, M. E.; McDermott, L. C. Students Understanding of the Ideal Gas Law, Part I: A Macroscopic Perspective. *Am. J. Phys.* **2005**, *73*, 1055.
- (3) Engel, T.; Reid, P. *Physical Chemistry*, 3rd ed.; Pearson: Boston, 2013.
- (4) Kautz, C. H.; Heron, P. R. L.; Shaffer, P. S.; McDermott, L. C. Students Understanding of the Ideal Gas Law, Part II: A Microscopic Perspective. *Am. J. Phys.* **2005**, *73*, 1064.
- (5) Privat, R.; Jaubert, J.-N.; Moine, E. Improving Students' Understanding of the Connections between the Concepts of Real-Gas Mixtures, Gas Ideal-Solutions, and Perfect-Gas Mixtures. *J. Chem. Educ.* **2016**, *93* (12), 2040–2045.
- (6) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*; Oxford Science Publications: Oxford, 2004.
- (7) Schroeder, D. V. Interactive Molecular Dynamics. *Am. J. Phys.* **2015**, *83*, 210–218.
- (8) Rahman, A. Correlations in the Motion of Atoms in Liquid Argon. *Phys. Rev.* **1964**, *136*, A405.
- (9) Verlet, L. Computer “Experiments” on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Phys. Rev.* **1967**, *159*, 98.
- (10) Humphrey, W.; Dalke, A.; Schulten, K. VMD: Visual Molecular Dynamics. *J. Mol. Graphics* **1996**, *14*, 33–38.
- (11) Simple Molecular Dynamics Code for Discovery-Driven Approach to Real Gas Properties. <https://github.com/FoleyLab/MolecularDynamics.git> (accessed Jan 2018).
- (12) Stilling, F. H.; Rahman, A. Improved Simulation of Liquid Water by Molecular Dynamics. *J. Chem. Phys.* **1974**, *60*, 1545.
- (13) Karplus, M.; McCammon, J. A. Molecular Dynamics Simulations of Biomolecules. *Nat. Struct. Biol.* **2002**, *9*, 646–652.
- (14) Brooks, B. R.; Brooks, C. L., III; Mackerell, A. D., Jr.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; Gafisch, A.; Caves, L.; Cui, Q.; Dinner, A. R.; Feig, M.; Fischer, S.; Gao, J.; Hodoseck, M.; Im, W.; Kuczera, K.; Lazaridis, T.; Ma, J.; Ovchinnikov, V.; Paci, E.; Pastor, R. W.; Post, C. B.; Pu, J. Z.; Schaefer, M.; Tidor, B.; Venable, R. M.; Woodcock, H. L.; Wu, X.; Yang, W.; York, D. M.; Karplus, M. CHARMM: The Biomolecular Simulation Program. *J. Comput. Chem.* **2009**, *30*, 1545–1614.
- (15) Shirts, M.; Pande, V. S. Screen Savers of the World Unite! *Science* **2000**, *290*, 1903–1904.

- (16) Dickson, A.; Tiwary, P.; Vashisth, H. Kinetics of Ligand Binding through Advanced Computational Approaches: A Review. *Curr. Top. Med. Chem.* **2017**, *17*, 2626–2641.
- (17) Madsen, J. J.; Sinitskiy, A. V.; Li, J.; Voth, G. A. Highly Coarse-Grained Representations of Transmembrane Proteins. *J. Chem. Theory Comput.* **2017**, *13*, 935–944.
- (18) Senftle, T. P.; Hong, S.; Islam, M. M.; Kylasa, S. B.; Zheng, Y.; Shin, Y. K.; Junkermeier, C.; Engel-Herbert, R.; Janik, M. J.; Aktulga, H. M.; Verstraelen, T.; Grama, A.; van Duin, A. C. T. The ReaxFF Reactive Force-Field: Development, Applications, and Future Directions. *npj Comp. Mater.* **2016**, *2*, 15011.
- (19) Berman, D.; Deshmukh, S. A.; Sankaranarayanan, S. K. R. S.; Erdemir, A.; Sumant, A. V. Macroscale Superlubricity Enabled by Graphene Nanoscroll Formation. *Science* **2015**, *348*, 1118–1122.
- (20) Wang, L.-P.; Titov, A.; McGibbon, R.; Liu, F.; Pande, V. S.; Martinez, T. J. Discovering Chemistry with an *ab initio* Nanoreactor. *Nat. Chem.* **2014**, *6*, 1044–1048.
- (21) Speer, O. F.; Wengert, B. C.; Taylor, R. S. Molecular Dynamics Simulations of Simple Liquids. *J. Chem. Educ.* **2004**, *81*, 1330–1332.
- (22) Eckler, L. H.; Nee, M. M. A Simple Molecular Dynamics Lab To Calculate Viscosity as a Function of Temperature. *J. Chem. Educ.* **2016**, *93*, 927–931.
- (23) Burkholder, P. R.; Purser, G. H.; Cole, R. S. Using Molecular Dynamics Simulation To Reinforce Student Understanding of Intermolecular Forces. *J. Chem. Educ.* **2008**, *85*, 1071–1077.
- (24) Mathews, G. P. Molecular Dynamics Simulator. *J. Chem. Educ.* **1993**, *70*, 387.
- (25) Lykos, P. Modeling the Hydrogen Bond within Molecular Dynamics. *J. Chem. Educ.* **2004**, *81*, 147–149.
- (26) Lamberti, V. E.; Fosdick, L. D.; Jessup, E. R.; Schauble, C. J. C. A Hands-On Introduction to Molecular Dynamics. *J. Chem. Educ.* **2002**, *79*, 601–606.
- (27) Saiz, E.; Tarazona, M. P. Molecular Dynamics and the Water Molecule: An Introduction to Molecular Dynamics for Physical Chemistry Students. *J. Chem. Educ.* **1997**, *74*, 1350.
- (28) Fuson, M. M. Anisotropic Rotational Diffusion Studied by Nuclear Spin Relaxation and Molecular Dynamics Simulation: An Undergraduate Physical Chemistry Laboratory. *J. Chem. Educ.* **2017**, *94*, 521–525.
- (29) Carlotto, S.; Zerbetto, M. Computational Study of Environmental Effects on Torsional Free Energy Surface of N-Acetyl-L-methyl-alanylaminide Dipeptide. *J. Chem. Educ.* **2014**, *91*, 96–102.
- (30) Chiang, H.; Robinson, L. C.; Brame, C. J.; Messina, T. C. Molecular Mechanics and Dynamics Characterization of an *in silico* Mutated Protein: A Stand-Alone Lab Module or Support Activity for *In Vivo* and *In Vitro* Analyses of Targeted Proteins. *Biochem. Mol. Biol. Educ.* **2013**, *41*, 402–408.
- (31) Hati, S.; Bhattacharyya, S. Incorporating Modeling and Simulations in Undergraduate Biophysical Chemistry Course To Promote Understanding of Structure-Dynamics-Function Relationships in Proteins. *Biochem. Mol. Biol. Educ.* **2016**, *44*, 140–159.
- (32) Elmore, D. E. Why Should Biochemistry Students Be Introduced to Molecular Dynamics Simulations – And How Can We Introduce Them? *Biochem. Mol. Biol. Educ.* **2016**, *44*, 118–123.
- (33) Rodrigues, J. P. G. L. M.; Melquiond, A. S. J.; Bonvin, A. M. J. J. Molecular Dynamics Characterization of the Conformational Landscape of Small Peptides: A Series of Hands-on Collaborative Practical Sessions for Undergraduate Students. *Biochem. Mol. Biol. Educ.* **2016**, *44*, 160–167.
- (34) Rosenfeld, S. Molecular Modeling as an Integral Part of an Advanced Lab Course. *J. Chem. Educ.* **1991**, *68*, 488–489.
- (35) Halpern, A. M.; Glendening, E. D. Exploring the Nature of the H₂. 2. Using *Ab Initio* Molecular Orbital Calculations To Obtain the Molecular Constants. *J. Chem. Educ.* **2013**, *90*, 1459–1462.
- (36) Halpern, A. M. Structural and Thermodynamic Properties of the Argon Dimer. *J. Chem. Educ.* **2010**, *87*, 174–179.
- (37) Dugas, H. Teaching Molecular Modeling: An Introductory Course for Chemists, Implemented at the University de Montreal. *J. Chem. Educ.* **1992**, *69*, 533–535.
- (38) Case, D. A.; Cheatham, T. E., III; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M., Jr.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber Biomolecular Simulation Programs. *J. Comput. Chem.* **2005**, *26* (16), 1668–1688.
- (39) Barker, J. A.; Fisher, R. A.; Watts, R. O. Liquid Argon: Monte Carlo and Molecular Dynamics Calculations. *Mol. Phys.* **1971**, *21*, 657–673.
- (40) Spencer, J. N. New Directions in Teaching Chemistry: A Philosophical and Pedagogical Basis. *J. Chem. Educ.* **1999**, *76*, 566–569.
- (41) Farrell, J. J.; Moog, R. S.; Spencer, J. N. A Guided Inquiry General Chemistry Course. *J. Chem. Educ.* **1999**, *76*, 570–574.
- (42) Wink, D. J. Constructivist Frameworks in Chemistry Education and the Problem of the “Thumb in the Eye. *J. Chem. Educ.* **2014**, *91*, 617–622.
- (43) Ditzler, M. A.; Ricci, R. W. Discovery Chemistry: Balancing Creativity and Structure. *J. Chem. Educ.* **1994**, *71*, 685–688.