Simulating Chemical Reactions using a Swarm of Miniature Robots

Audrey Randall, John Klingner and Nikolaus Correll*

Department of Computer Science
University of Colorado at Boulder

Abstract. We wish to simulate basic rules of chemistry using a swarm of miniature robots, which mimic atoms and forming molecules. Atomic scale interactions are difficult to observe and computer simulations or ball-and-stick models capture either behavioral or embodied aspects, but not both. Miniature robots that are able to determine their orientation and position with respect to each other and provide audible, visual, and tactile feedback to a user could make such simulations both interactive and tangible. We describe a working prototype of our swarm-robotic chemistry simulation which demonstrates concepts including electronegativity, reaction spontaneity, the octet rule, and hybridization. Here, the key challenge is that once we go beyond the most simple set of atoms, the outcome of reactions cannot be calculated from first principles. We solve this problem by letting robots exchange local measurements, the nearby atoms, their geometry, and molecules that have formed and then using a compact look-up table implementation, which suggests avenues of further studies for both physical chemistry and swarm robotics. We also present preliminary data recorded from a high-school demonstration evaluating using a tangible simulation of chemistry reactions as a teaching tool.

1 Introduction

We are interested in simulating the basic rules of chemistry in a swarm of miniature robots for two reasons: First, we wish to encourage chemists to think about atomic interactions in terms of distributed systems and self-organization. Second, we hope to produce a teaching tool for chemistry students. While the latter use case might justify a simulation that is not perfectly faithful to the functioning of the physical system as long as it produces the correct phenomenological results, our long term goal is for those two applications of chemistry simulation to merge, thereby advancing our understanding of both chemistry and education.

Chemistry simulations fall broadly into two categories: teaching tools and research tools. From an educational perspective, interactive simulations of chemical processes are interesting as they may improve visual-spatial cognition [1–4]. In

* This research has been supported by NSF grant #1150223.

order to improve the pedagogical value of 3D simulations of molecular interactions, educators have begun embracing augmented reality [5] that combines tangible devices (some of which include olfactory and auditory sensations) [6], with simulations to improve learning [7]. Most teaching tools, such as those collected in MERLOT's database [8], are divided into lessons, each of which demonstrates one chemistry concept. A tool might create molecules that perform an acid-base reaction, but be incapable of a reduction-oxidation reaction. We have not seen any interactive teaching tools designed to imitate the behavior of real atoms and molecules in all situations, and we believe such a simulation could be extremely useful and interesting for students.

A similar gap exists in the area of chemistry simulations for research. Most chemistry research simulations do not simulate individual particles (either atoms or molecules), trading the extreme level of detail for computational expediency. These simulations are also usually optimized toward representing a single environment, such as a specific set of reactions in the troposphere [9] or the interaction of amino acids and mild steel [10], to name just two examples from a very large body of work in computational chemistry [11].

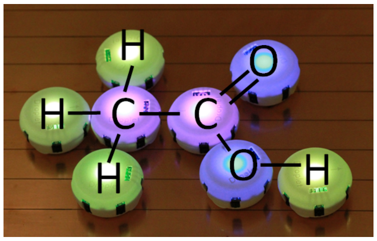

Fig. 1: This figure shows an example of Acetic Acid ($C_2H_4O_2$) in the formation our algorithm would dictate.

The closest example of research similar to our own uses physical concepts to simulate particles [12] as an interactive homogeneous swarm, which demonstrates the emergence of macroscopic patterns from microscopic particles. However, this simulation is composed of only one type of particle, and does not simulate different elements [12]. It is also implemented on a traditional centralized computing platform, not on a physical swarm of individual robots.

2 Background

The study of how and when atoms bond dates back to the 1800s, and between then and now, increasingly complex models have emerged to explain molecular and atomic behavior. The simplest atomic models, such as the Bohr model invented in 1913 by Niels Bohr, are sufficient to explain only the behavior of a small subset of atoms, with irregularities and exceptions. As our understanding of what atoms are and how they function has improved, better behaved models have emerged that can explain the behavior of a wider range of chemical species. But even today, chemistry has not advanced to the point where chemical reactions can be accurately simulated from physical principles. Our models are only as good as our understanding of the atom itself. Chemical physicists, in their attempts to create truly accurate models of the atom, are now delving into the realms of quantum physics and chemistry, which are developing fields that are not yet fully understood. In our attempt to implement the most accurate possible model for chemical reactions on a miniature robot, we found that all non-quantum models based on physical properties eventually break down. For example, the octet rule, which is often presented as the sole explanation for why atoms bond in the first weeks of a chemistry class, states that the valence shell of an atom can have up to eight valence electrons, and is not stable until it has eight. Students often ask why elements in the alkaline and alkaline-earth groups are "satisfied" with two valence electrons, and cannot have more. To solve this problem, the Lewis and Bohr models of the atom are introduced, but they cannot explain bonding between metals, which is why introductory chemistry classes often leave the subject of metallic bonding for the second class. The most complex non- quantum-based model taught to students to explain why atoms form bonds in the shapes they do is called hybridization, which we have chosen as a starting point for a distributed simulation presented here. But even this method is described in a textbook as existing only to predict molecular shapes we have no other explanation for [13]. Chemists doubt that it is an accurate model of what is truly happening.

For some purposes, look-up tables of experimental data are sufficient. Introductory chemistry classes across the world teach the use of tables of enthalpy, entropy, and Gibbs Free Energy values to determine reaction spontaneity. If the set of atoms in question is reduced to only a few specific, well- behaved species, the look-up tables have few enough exceptions that students can get a good idea of how the reactions work without confusion. We therefore believe that our initial attempt on a distributed, swarm robotic implementation described in this paper can convey information at the level of an introductory chemistry course very well, and might lead to similar insights as swarm robotics has afforded for the study of social insects [14,15], another natural system relying on self-organization [16].

3 Mapping basic chemistry concepts to distributed embodied swarms

The Octet Rule The Bohr atomic model describes electrons as nearly weightless particles arranged in a series of shells around the nucleus, and is considered the simplest possible abstraction of an atom. Within each shell, electrons are divided into orbitals, regions wherein electrons can probably be found. Each orbital contains at most two electrons. As distance from the nucleus increases, successive electron shells do not contain the same number of orbitals, and orbitals themselves have several different shapes. In this model, the outermost or "valence" shell determines how the atom will bond. It contains at most eight electrons. This is called the "octet rule," and though it doesn't always apply for larger atoms, it does apply for the nonmetals in the subset of elements we model.

Students are taught that atoms bond when each has an orbital with a single, unpaired electron in it. Those orbitals then overlap, allowing the atoms to share the electrons within them. This means that both orbitals effectively contain two electrons, which puts them both in a more stable state. If an atom has more than one orbital with an unpaired electron, it can bond with multiple other atoms. If two atoms have more than one orbital with an unpaired electron, they might form a double or triple bond. This model is the basis for our implementation of inter-atomic bonding, requiring individual robots to advertise their own number of unpaired electrons and listen to their neighbors. We assume that all individual atoms have a neutral formal charge. Ions are not currently implemented.

Electronegativity Students are also taught that atoms can form two types of bonds: *covalent* (where electrons are shared) and *ionic* (where electrons are taken by one atom from the other). Covalent bonds are divided into categories based on strength: the more polar the bond, the stronger it generally is. Polarity measures the difference in partial charge between bonded atoms. Students usually memorize the many types and strengths of covalent bonds, but there is a simpler model to determine bond type: electronegativity. Electronegativity is a property of an atom that determines its tendency to attract shared electrons. By this model, ionic bonds are simply covalent bonds that are so polar, one atom has full control of the electrons. This is reflected by our implementation, where the more electronegative atom has control of the electrons.

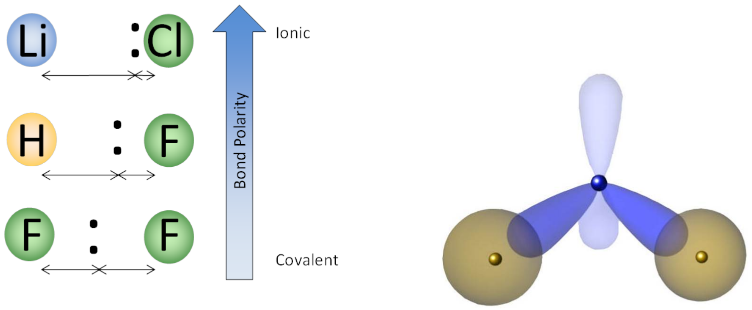

Fig. 2: Left: An increasing difference in electronegativity yields a bond where electrons are held more closely by one atom than the other, until the electrons transfer atoms entirely and the bond becomes ionic in character. Right: Water shown with four orbitals on the central oxygen. Light blue represents an orbital that does not contain bonded electrons (although it does contain a lone pair of unbonded electrons). Dark blue represents the two orbitals that form bonds with hydrogen. Since hydrogen only has one orbital, its shape is different from the orbitals of oxygen.

We use Mulliken-Jaffe electronegativity values on a Pauling scale [17] for our treatment of this concept. These values are given by the equation:

$$\chi = 3.48\left((IE_v + EA_v)/2 - 0.602\right)$$

EA_v is the electron affinity of the atom and IE_v is the first ionization energy. The electron affinity is the change in energy of an atom when an electron is added to it in its neutral state, and the first ionization energy is the energy required to take an electron away from the atom in its neutral state. Using χ values on a Pauling scale gives values that are between approximately 0 and 4. A larger χ means a stronger tendency to attract shared electrons. While attempting to form bonds, an atom will bond ionically if the difference in χ values between it and its potential partner is greater than 1.70 [13], and covalently otherwise. In a robotic implementation, it is sufficient to simply exchange the χ value between robots and choose bonding behavior accordingly.

Reaction Spontaneity and Thermodynamics The concept of reaction spontaneity is used to tell students whether, without outside influence, a reaction will occur. Initially, students learn about enthalpy (ΔH). Enthalpy is a quantity related to the heat that a reaction requires or releases. However, the exceptions to this rule lead to misconceptions. In the interest of avoiding these, we chose to implement the more complicated concept of Gibbs Free Energy (ΔG). ΔG values represent the difference in free energy between a reaction's products and reactants.

Reactions between atoms and molecules occur only when the products of the reaction have a lower energy level than the reactants. Thus, we calculate the difference between the sum of the ΔG values for the products and the sum of the ΔG values for the reactants, which gives us an overall ΔG for the reaction:

$$\Delta \mathrm{G}_{\mathrm{rxn}} = \sum \Delta \mathrm{G}_{\mathrm{products}} - \sum \Delta \mathrm{G}_{\mathrm{reactants}}$$

If $\Delta \mathrm{G}_{\mathrm{rxn}}$ is negative, then we say that the reaction occurs spontaneously. Reactions with $\Delta \mathrm{G}_{rxn} \geq 0$ do still occur, but only when energy is added to the system. In practice, enough energy can be added by kinetic interactions between molecules for such reactions to occur 'spontaneously'. However, due to the scale of our chemistry simulation, such effects are difficult to model, and thus our simulation only allows reactions with $\Delta \mathrm{G}_{rxn} < 0$ to occur. Both individual molecules

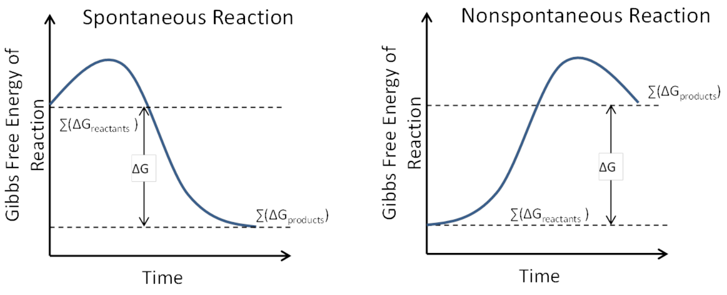

Fig. 3: Difference in Gibbs Free Energy for spontaneous and nonspontaneous reactions.

and reactions have ΔG values. In the case of individual molecules, the implied equation is the formation of that molecule from its most basic components. In a robotic implementation, the ΔG of a molecule can be determined in two ways: either by looking up a table of experimentally determined values or by using this equation [18]:

$$\Delta \mathrm{G} = \Delta \mathrm{H} - T\Delta \mathrm{S}$$

ΔH is the enthalpy of formation of that molecule, ΔS is the entropy of formation, and T is the temperature (assumed to be 298K). Most of our thermodynamic values were obtained from the DIPPR Chemical Database [19], which provides ΔG data for more than 2000 compounds. This database was selected due to its size and accessibility. We filled in gaps in this database, which were mostly radicals, with data from the NIST Chemistry WebBook [20].

In a robotics context, Gibbs Free Energy calculations require each robot to store the knowledge of the molecule it belongs to. When considering a reaction, the robot calculates the $\Delta \mathrm{G}_{\mathrm{rxn}}$. It uses its own molecule and the other robot's molecule as reactants, and using its own molecule with the other atom added and the other robot's molecule without that atom as products. Only single atom addition reactions are considered, meaning that the two robots considering reacting are the only ones that might switch molecules. In order to make the look-up of these $\Delta \mathrm{G}_{\mathrm{rxn}}$ values fast enough to run on our platform's microprocessor, we store a list of all possible molecules the robots could form as a prefix tree. More details on this implementation can be found in Section 4.

One value of using this system for determining if bonds form is that the robots can also demonstrate the basic principles of reaction direction: specifically, if a reaction occurs in the forward direction, it will never occur in the reverse direction unless energy is added to the system.

Hybridization and Molecular Geometry Hybridization theory determines the shape that molecules should form. Since electrons repel each other, the placement of the full or partially full orbitals around the atom can explain the shape that bonds make. For example, water has its v-shaped structure because of the placement of the four orbitals on the central oxygen in a tetrahedral shape (Figure 2, right). These orbitals have an angle to each other of approximately 109.5° [13]. Two out of the four have paired electrons and therefore do not form bonds, and the remaining two singly occupied orbitals bond with hydrogen, resulting in water's bent geometry.

As well as accurately predicting molecular geometry, hybridization gives a more accurate prediction of bond number than any other theory, since it was invented to explain bond structures that other theories couldn't account for. For a second example, according to Lewis bond theory, beryllium (Be) cannot bond because it has two electrons in its *s* orbital and thus has no orbitals with one unpaired electron that can overlap to form a bond. But *Be* can in fact form two bonds; hybridization theory explains this by stating that an electron gets "promoted" from the *s* orbital to an empty *p* orbital. The *s* orbital and *p* orbital, both now singly occupied, no longer have the character of an *s* and *p* orbital. They will each form a hybridized orbital denoted as sp^1 or simply sp. The hybridized orbitals now can form bonds in the same manner as ordinary orbitals. The repulsion of the hybridized orbitals gives the explanation for the molecule's shape, since the angle of the overlapping orbitals is also the angle between the bonded atoms.

In conjunction with the robotic platform's range, bearing, and heading measurement of the atoms to which it is bonded, hybridization is used in the code so that, for example, the two hydrogen atoms in H_2O are both constantly assessing and adjusting their position relative to the oxygen atom. This simulates traditional two-dimensional representations shown in textbooks such as Burdge and Overby's [21].

4 Implementation

The Droplets are an open-source swarm robotic platform, with source code and manufacturing information available online[1]. Each Droplet is roughly cylindrical with a radius of 2.2 cm and a height of 2.4 cm. The Droplets use an Atmel xMega128A3U micro-controller, and receive power via their legs through a floor with alternating strips of $+5V$ and GND. Each Droplet has six infrared emitters, receivers, and sensors, which are used for communication, collision detection, and for the range and bearing system [22]. The top of each board has a light and color sensor, as well as an RGB LED. Each Droplet has three extended headers for legs located symmetrically around itself. The legs are spaced 120° apart and 1.5 cm from the center. Mounted symmetrically opposite from two of the legs are coin-type vibration motors of the type commonly used in cell phones and pagers. These motors are used as a low-cost locomotion method [23]. The third leg has a speaker mounted opposite it. Each Droplet has a unique ID number.

Our implementation of these chemistry concepts centers around a 'state message',' which each Droplet broadcasts periodically. This message includes:

- The IDs of each Droplet the sender is bonded to.
- A list of the atoms in the sender's molecule.
- The sender's atomic number.
- The state of the sender's valence shells.

A Droplet also receives updated measurements for the sender's range, bearing, and heading with this message [22]. If the Droplet receiving this message is bonded to the sender already, then the Droplet checks where it should move to maintain the appropriate molecular geometry as determined by hybridization. It also updates its list of atoms in its molecule based on the sender's other bonds.

The Droplet's color indicates what atom it is. Bonds are represented visually by the bonded Droplets all blinking red together. The simultaneous blinking occurs because all of the Droplets' clocks are synchronized with a firefly synchronization algorithm [24]. To avoid multiple Droplets trying to communicate

[1] `https://github.com/correlllab/cu-Droplet`

simultaneously, their state messages are sent out with a fixed period, but with phase determined by their ID number. Since every Droplet knows the ID number of every other Droplet in its molecule, we use the lowest ID number amongst the Droplets in a given molecule to determine the phase of the blink.

If the receiving Droplet (call it '**C**') is not bonded to the sender (call it '**O**'), then it considers whether or not it should be. A change in bond status represents a reaction, so the first consideration **C** makes is whether the reaction would be spontaneous; whether $\varDelta G_{rxn} < 0$. To do so, it must search the list of more than 350 molecules to find each product and reactant. For a modern desktop, such a search could be performed linearly, but on the Droplet's small and low-cost hardware we wanted the performance benefits of storing the list as a prefix tree. The downside of such a data structure, however, is its large memory footprint, as we have a node for every prefix of a molecule and each node has to store a number of pointers in addition to the values. Using the method described in [25], we packed the prefix tree (which contains 659 nodes with pointer size of two bytes, value size of two bytes, and key size of one byte) as a single byte array with memory footprint of 2719 bytes

Assuming **C** has determined a spontaneous reaction should occur, it then checks its valence shell and that of **O**, to ensure that bonding wouldn't violate the octet rule and, if it doesn't, to determine which orbitals to bond to as determined by hybridization. If these and the other pre-bond checks are passed, **C** updates its state: valence shell, molecule, and bonded atoms.

The next time **C** sends out its own state message, **O** will see that **C** is trying to bond to it, and perform all of the above checks again. This is important because there may be some reason (specifically, due to interactions between **O** and a third Droplet) that **O** can't bond with **C**. If **O** confirms with its checks that a bond should occur, then it updates its own state, and the bond is formed. Otherwise, when **C** gets a new state message from **O** and sees that **O** isn't bonded to it, **C** gives up on the bond and fixes its own state accordingly.

5 Experiments & Results

Our molecule-forming algorithm reaches one or more of several stable states for every combination of atoms in the environment. We tested the algorithm by timing how long it took for one of these states to be reached for a certain number of individual atoms. In order to reduce the chance of error due to missing information, only hydrogen, oxygen, and carbon were used to test timing. These elements are the basic components of simple organic molecules, which means that there is a great deal of information available about the various molecules they form. Since the robots' IDs define the order in which they send messages, we switched the atomic identity of each robot to determine if a bias toward a particular state might be induced by having a predictable message ordering. We didn't find any obvious bias.

The total number of bonds present in all molecules usually increases with, but is not a direct function of, the number of atoms present. For example, if four hydrogen atoms and an oxygen atom are placed on the board, the total number of bonds in the stable state will be three, because these atoms will form a water molecule (H_2O) and dihydrogen (H_2). The maximum number of bonds we observed being formed from our choice of individual atoms was six, when nine atoms were present. Figure 4 shows this relationship.

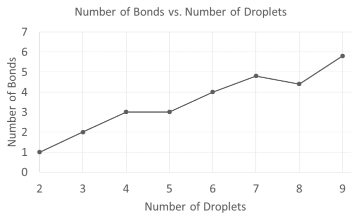

Fig. 4: The average number of bonds formed as a function of the number of Droplets.

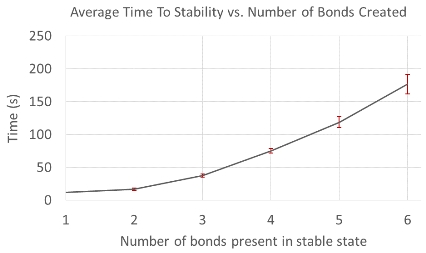

Fig. 5: Time to reach a stable state versus the number of bonds present in the stable state. Error bars represent standard error of the mean.

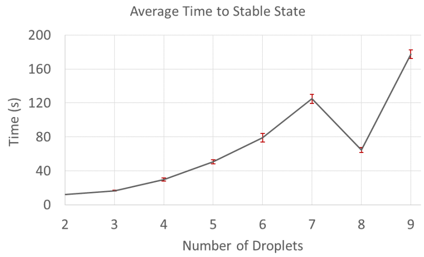

Fig. 6: Time to reach a stable state from unbonded component atoms. Error bars represent standard error of the mean, and sample size is 5.

The notable exception to this rule is when eight atoms are present. If we look at the average number of bonds present in stable configurations as a function of the number of atoms, we see a marked decrease for eight robots (Figure 4). This likely explains why the time to reach a stable state seems to scale predictably with the number of atoms present, with one exception in the case of eight atoms: the important factor in time-to-stability is actually the total number of bonds in the stable configuration, as shown in Figure 6.

While we could theoretically form any of the 659 encoded molecules, various factors make this difficult in practice. Specifically, as the size of the molecule increases, the likelihood that its components will form multiple, independently-stable smaller molecules increases as well. This is a consequence of the greedy nature of our molecule-forming algorithm.

We also presented the Droplet Chemistry Simulation to a group of ten students in grades 10–12. All students are in the St. Vrain School District in Longmont, Colorado. We split the students into three groups and gave each group a board with several chemistry Droplets, as well as several other Droplets with nothing but synchronization code (these extra Droplets help maintain the synchronization, which loses stability for small numbers of robots). The session began with a brief introduction to how the Droplets worked: both hardware and software. We explained that the Droplets communicate using IR sensors and emitters, that they can determine range and bearing based on their communication, and that they walk using vibration motors. We also presented a modified chart of the periodic table that only included our subset of atoms. This chart showed which color corresponded to which atom, as well as the Lewis dot structure of each atom. The information given allowed students to predict which bonds would form by looking at the chart, and then test their prediction by observing the Droplets. The students observed the rate at which bonds formed, the rate at which they broke, and which atoms bonded when, by observing the pattern of synchronized blinks as well as the Droplets' motion. After a period of interacting with the hardware, we asked them a series of qualitative questions, that are omitted here for brevity, to probe the Droplet's qualities as a teaching tool.

The overall reaction we observed was very positive. Many students expressed the opinion that the Droplets would help or would have helped them to better understand chemistry at the time of their first class in the subject. All agreed that the Droplets would have increased their interest in introductory chemistry. Five students had taken a chemistry class before. When we asked them to rate their previous chemistry class, two students rated their class as a four and three students gave it a three. However, students did not express enthusiasm about the idea of chemistry when we asked them about their experiences with their first class. The Droplets, on the other hand, produced a significant amount of interest. Students were engaged and asked questions, volunteering ideas about other simulations the Droplets could perform. The students seemed in particular to appreciate the tangibility of the Droplets, as compared to what they had experienced in class.

6 Conclusion

Modeling the attractions that cause bonds between atoms intuitively seems like a trivial problem: perhaps one that is computationally expensive, but trivial nonetheless. However, it is important to remember that our ability to model physical forces is only as good as our knowledge of the forces themselves. Chemistry does not have an entirely accurate understanding of the atom yet, which makes modeling its behavior based on physical forces extremely difficult. The Droplets present a solution to this problem by using electronegativity to predict bond type, hybridization theory to predict molecular shape, and Gibbs Free Energy to predict reaction spontaneity. From a computational perspective, this is much more complicated than using the same concepts on a single computer. Instead of one “puppet-master” program controlling a set of virtual constructs in a perfect environment, the Droplets each have their own constant calculations running to determine their position in space relative to the robots around them.

Future work will include expanding the ability of the Droplets to be used as a teaching tool. Based on the positive reaction of the ten students who saw the chemistry simulation prototype, we think that it is worth performing a more formal educational study to judge the Droplets' efficacy in promoting interest in and understanding of chemistry. Our most immediate plans for future work are to conduct a larger scale controlled study of this in classrooms. The data from such a study would help guide which aspects of the simulation should be improved. One such aspect would be to use electronegativity for intermolecular interactions: causing the Droplets to move as if experiencing a force from surrounding molecules. This would allow the Droplets to demonstrate formation of lattice structures.

We also plan to expand the list of elements the Droplets can simulate. In particular, we would include metals, which have much more complicated interactions due to a more advanced orbital configuration. This would require restructuring the valence shell model and removing the assumption that the octet rule applies. We would also like to implement more complicated reactions using the atoms that we already simulate.

Although accurately modeling atomic and molecular interactions is an extremely difficult task, we are confident that the Droplets succeed in demonstrating atomic behavior with reasonable accuracy. Implementing a chemical simulation on a swarm platform also offers other advantages, such as interactivity and scalability.

References

1. M. Stieff and U. Wilensky, "Connected chemistry—incorporating interactive simulations into the chemistry classroom," *Journal of Science Education and Technology*, vol. 12, no. 3, pp. 285–302, 2003.
2. H.-K. Wu and P. Shah, "Exploring visuospatial thinking in chemistry learning," *Science education*, vol. 88, no. 3, pp. 465–492, 2004.
3. M. Limniou, D. Roberts, and N. Papadopoulos, "Full immersive virtual environment cave tm in chemistry education," *Computers & Education*, vol. 51, no. 2, pp. 584–593, 2008.
4. K. L. Vavra, V. Janjic-Watrich, K. Loerke, L. M. Phillips, S. P. Norris, and J. Macnab, "Visualization in science education," *Alberta Science Education Journal*, vol. 41, no. 1, pp. 22–30, 2011.
5. M. Fjeld and B. M. Voegtli, "Augmented chemistry: An interactive educational workbench," in *Mixed and Augmented Reality, 2002. ISMAR 2002. Proceedings. International Symposium on*, pp. 259–321, IEEE, 2002.
6. E. Richard, A. Tijou, P. Richard, and J.-L. Ferrier, "Multi-modal virtual environments for education with haptic and olfactory feedback," *Virtual Reality*, vol. 10, no. 3-4, pp. 207–225, 2006.
7. M. Fjeld, J. Fredriksson, M. Ejdestig, F. Duca, K. Bötschi, B. Voegtli, and P. Juchli, "Tangible user interface for chemistry education: comparative evaluation and redesign," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 805–808, ACM, 2007.

8. "Merlot ii: Multimedia educational resource for learning and online teaching." `http://tinyurl.com/j6686uz`. Accessed: 4/6/2016.
9. Y. Wang, D. J. Jacob, and J. A. Logan, "Global simulation of tropospheric o3-no x-hydrocarbon chemistry: 1. model formulation," *Journal of Geophysical Research: Atmospheres*, vol. 103, no. D9, pp. 10713–10725, 1998.
10. N. O. Eddy, F. E. Awe, C. E. Gimba, N. O. Ibisi, and E. E. Ebenso, "Qsar, experimental and computational chemistry simulation studies on the inhibition potentials of some amino acids for the corrosion of mild steel in 0.1 m hcl," *Int. J. Electrochem. Sci*, vol. 6, pp. 931–957, 2011.
11. F. Jensen, *Introduction to computational chemistry.* John Wiley & Sons, 2013.
12. H. Sayama, "Seeking open-ended evolution in swarm chemistry," in *Artificial Life (ALIFE), 2011 IEEE Symposium on*, pp. 186–193, IEEE, 2011.
13. J. Burdge and J. Overby, *Chemistry Atoms First*, vol. 1, ch. 7.1. McGraw-Hill Learning Solutions, 2011.
14. J. Halloy, G. Sempo, G. Caprari, C. Rivault, M. Asadpour, F. Tâche, I. Said, V. Durier, S. Canonge, J. M. Amé, *et al.*, "Social integration of robots into groups of cockroaches to control self-organized choices," *Science*, vol. 318, no. 5853, pp. 1155–1158, 2007.
15. N. Correll and A. Martinoli, "Modeling and designing self-organized aggregation in a swarm of miniature robots," *The International Journal of Robotics Research*, vol. 30, no. 5, pp. 615–626, 2011.
16. E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems.* No. 1, Oxford university press, 1999.
17. J. Hinze and H. H. Jaffe, "Electronegativity. i. orbital electronegativity of neutral atoms," *Journal of the American Chemical Society*, vol. 84, no. 4, pp. 540–546, 1962.
18. Silberberg and Amateis, *Chemistry: The Molecular Nature of Matter and Change*, ch. 20.3. McGraw-Hill Education, 7 ed., 2015.
19. S. b. A. Design Institute for Physical Properties, *DIPPR Project 801 - Full Version.* Design Institute for Physical Property Research/AIChE, 2005; 2008; 2009; 2010; 2011; 2012.
20. D. Burgess, *NIST Chemistry WebBook.* Gaithersburg, MD: National Institute of Standards and Technology, 2005; 2008; 2009; 2010; 2011; 2012.
21. J. Burdge and J. Overby, *Chemistry: Atoms First.* McGraw-Hill New York, 2012.
22. N. Farrow, J. Klingner, D. Reishus, and N. Correll, "Miniature six-channel range and bearing system: Algorithm, analysis and experimental validation," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 6180–6185, IEEE, 2014.
23. J. Klingner, A. Kanakia, N. Farrow, D. Reishus, and N. Correll, "A stick-slip omnidirectional powertrain for low-cost swarm robotics: Mechanism, calibration, and control," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pp. 846–851, IEEE, 2014.
24. R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.
25. U. Germann, E. Joanis, and S. Larkin, "Tightly packed tries: How to fit large models into memory, and make them load fast, too," in *Proceedings of the Workshop on Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pp. 31–39, Association for Computational Linguistics, 2009.