GRADE: A code to determine clathrate hydrate structures

Farbod Mahmoudinobar *, Cristiano L. Dias

The New Jersey Institute of Technology, Department of Physics, University Heights, Newark, NJ, 07102-1982, United States

A R T I C L E   I N F O

Article history:
Received 10 December 2018
Received in revised form 9 May 2019
Accepted 3 June 2019
Available online xxxx

Keywords:
Clathrate hydrate
All-atom simulation
F₄ order parameter

A B S T R A C T

GRADE is an open-source software package that identifies clathrate hydrate structures given the atomic coordinates of oxygen atoms of water. These structures are commonly found in natural gases and they are relevant to understand antifreezing proteins as well as the hydrophobic effect. In this manuscript, we describe basic algorithms used by GRADE to determine 5₁₂, 6²₅₁₂ and 6⁴₅₁₂ water cages. These algorithms are based on the connectivity of five and six-folded rings made of first-neighbor water molecules. Schemes to speed up the calculation of these cages are discussed and implemented in the code. GRADE also computes the four-body order parameter F₄ commonly used to identify different phases of water. An example of how GRADE can be used to compute the spontaneous nucleation of methane clathrates is also provided.

Program summary

Program Title: GRADE
Program Files doi: http://dx.doi.org/10.17632/h8d7zgmchf.1
Licensing provisions: GNU GPLv3
Programming language: C++
Nature of problem: Analyze gas clathrate formation from atomic positions of water molecules in computer simulations
Solution method: Algorithms to find 5₁₂, 6²₅₁₂ and 6⁴₅₁₂ cages and calculate F₄ order parameter

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

Gas clathrates are solid compounds formed by natural gas molecules trapped within solid cage-like water structure [1]. These structures are found on the seabed, and in ocean/lake sediments where conditions of high pressure and low temperature are favorable to their formation [2–6]. As the extraction of gas clathrates is becoming economically viable, these natural deposits are expected to account for an important fraction of the world’s energy supply [7,8]. However, the existence of these natural deposits is also a source of concern as the release of large quantities of methane during earthquakes and tsunamis could contribute significantly to global climate. Moreover, methane clathrate represents a dangerous problem for the gas industry due to potential plug formation in pipelines [19,10]. Accordingly, extensive efforts are being dedicated to understand the formation, stability, and inhibition of gas clathrates [11,12].

Nucleation of gas clathrates occurs at nanometer length scale and microsecond timescale, which are not easily accessible experimentally but can be probed using all-atom computer simulations [13]. The latter is providing insights into the sequence of molecular events leading to the nucleation of clathrate structures [14–19]. In particular, simulations have shown that an isolated water cage hosting a gas molecule in aqueous solution is short-lived [20–22] and clathrate nucleation requires local concentration of guest molecules at supersaturated conditions, known as “blobs” [21,23,24]. The formation of these blobs in water is the rate limiting step of clathrate nucleation which prompts water molecules to freeze into cages hosting gas molecules. Initially, cages are stacked together in an amorphous manner before relaxing into the crystalline clathrate phase [25]. This sequence of events, i.e., blobs → amorphous cages → clathrate phase, is known as the multistep hypothesis (MSH) of clathrate formation.

The formation of water clathrates also plays a role in biological systems. In particular, water molecules around non-polar side chains in proteins are more structured than in the bulk solution with a tendency to form incomplete cage-like structures [26–30]. In antifreeze proteins, e.g., the Antarctic bacterial antifreeze protein, the more structured water molecules around non-polar side chains mediate binding of ice surfaces to the protein inhibiting ice propagation [31]. Also, semi-clathrate water structures have

**Please cite this article as:** F. Mahmoudinobar and C.L. Dias, GRADE: A code to determine clathrate hydrate structures, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.06.004.
been shown to contribute to the stabilization of antifreeze protein folds [32]. Thus, understanding the formation of clathrates in biological systems is important to understand protein structure formation and to discover new antifreezing mechanisms. The aim of this paper is to present an open source computer code that analyzes atomic positions of oxygen atoms of water to compute the number of cages and account for their three-dimensional structures. The latter can be used for visual inspection using software such as VMD (Visual Molecular Dynamics) [33]. We named the code GRADE which stands for “cage” in Portuguese. The number of cages is commonly used as an order parameter to estimate the level of clathrate structures that have been formed in a simulation. Moreover, visual inspection of the time evolution of these structures is essential to provide insights into the sequence of molecular events leading to clathrate formation. Codes to analyze water structures have been developed by other labs including Molinerò’s group who developed CHILL+ to compute different order parameters that distinguishes between hydrates, cubic ice, hexagonal ice, and liquid water [34], as well as a cage analysis code [17], Bi et al. [19], Rodger et al. [35], Walsh et al. [14] and Guo et al. [36] have also written codes to identify water structures. Here we introduce an open-source C++ code, called GRADE, which employs a hierarchical algorithm to identify the evolution of rings, cups and cages in molecular dynamics simulations of water molecules and computes the four-body order parameter F4. This code is freely available on GitHub [37] and we anticipate that GRADE will serve as a C++ template code by the community which can be modified by research groups to account for their specific needs. Also, users who have significantly improved GRADE and would like to see their changes appended to new versions of the code are encouraged to contact the authors of this manuscript.

2. Methods

The aim of GRADE is to identify water structures, known as cages, that are the building blocks of gas clathrates. The current version of GRADE identifies 512, 6512 and 65512 cages that are made of 20, 24 and 28 water molecules, respectively. GRADE uses a hierarchical algorithm in which first-neighbors of all water molecules are identified followed by rings, cups, and cages. It computes all rings made of five and six first-neighbor water molecules [38]. Cups, i.e., half cages, are computed based on the connectivity of these rings to one another and cages are computed based on how cups are bound to each other. This approach was also used by other codes to identify water cages in solution [17] as well as to define order parameters to enable the development of new algorithm to simulate hydrate nucleation rates [19,39]. Hierarchical based algorithms have also been developed to identify polymorphic clathrate structures by Lauricella et al. [40]. Notice that the number of operations required to compute cages is very large and, thus, it can be very time consuming, specially for large systems and/or long trajectories. One advantage of GRADE is that it has built up schemes to speed up the calculation of cages. Below, we describe in detail how different quantities are computed within GRADE as well as the speed up schemes.

Two molecular dynamics setups are used to illustrate the different quantities computed by GRADE: one where water is in equilibrium in the liquid state and the other where water initially in the liquid state evolves towards forming 512, 6512 and 65512 cages around methane molecules in solution. Details of these setups are given in Table 1. Simulations were performed using the open source software GROMACS [41] version 5.1 with the leap-frog algorithm to integrate the equation of motion in the NPT ensemble. Temperature was controlled using the v-rescale thermostat (τT = 1 ps), and pressure was kept constant using the Parrinello–Rahman barostat (τP = 0.1 ps). Periodic boundary conditions were applied in all directions. A cutoff of 1.0 nm was used to account for short-range nonbonded interactions. Long-range electrostatics were calculated using the Particle Mesh Ewald (PME) algorithm with grid spacing of 0.16 nm and 1.0 nm real-space cutoff. The TIP4P/ice model [42] was used to mimic water, and methane molecules were modeled using a united atom representation with 6–12 Lennard-Jones interactions (σ = 0.373 nm and ε = 1.234 kJ/mol) [43]. Simulations A and B described in Table 1 lasted for 5 ns and 1 µs each, respectively.

2.1. First-neighbors

In Fig. 1, we show the oxygen–oxygen radial distribution function (RDF) of liquid water computed from simulation A in Table 1. First and second peaks of this RDF are located at distances R1 = 0.28 nm and R2 = 0.44 nm, respectively. The minimum between these peaks, i.e., Rcut, is commonly used as the cut-off radius to define first-neighbors. In other words, if the distance between oxygen atoms of water molecules i and j is smaller than Rcut, then these two water molecules are considered to be first-neighbors of each other. The default distance cut-off of GRADE is Rcut = 0.35 nm. This quantity can, however, be controlled using the flag ‘-c’ to define alternative Rcut values, e.g., to compute cages in simulations of coarse-grained models where bond-lengths are given in normalized units [44–47].

2.2. Rings

Rings are closed structures obtained by connecting first-neighbor water molecules to each other. For example, molecules i, j, k, l form a ring if j is first-neighbor of i and k, l is a first-neighbor of k and m, and m is first-neighbor of i. The size of a ring corresponds to the number of water molecules in the loop.
Thus, the size of the ring formed by molecules $i, j, k, l$, and $m$ is 5. Using a depth-first algorithm, GRADE searches for all five- and six-folded rings in the simulation box. The algorithm starts at one water molecule and explores as far as possible along each of the branches formed by first-neighbor water molecules before backtracking. An example is shown in Fig. 2a where the five-folded ring shown in red is found starting from molecule labeled 1. The branch formed by molecules 1 and 2 does not lead to the formation of a ring and, thus, the system is backtracked to explore the branch 1, 3, 4. Similarly, the latter branch does not lead to the formation of rings and the system goes on to explore branch 1, 3, 5, 6 and so on. All the iterations required to find the five-folded ring are shown in panel b of Fig. 2. When using these rings to identify cups, GRADE spends a significant amount of time on loops going over all five- and six-folded rings. To reduce this performance bottleneck, GRADE ignores deformed rings that cannot account for stable cups or cages. In particular, GRADE identifies two types of deformations related to the convexity and planarity of ring structures.

**Non-convexity.** Five- and six-folded loop structures that are non-convex are excluded from the list of rings identified by GRADE. For a five-folded ring to be non-convex, the distance between vertices $i$ and $i+2$ has to be smaller than $d_5 = 1.6 \times a$, where $a$ is the distance between neighboring vertices and $d_5$ is the distance between vertices $i$ and $i+2$ in a regular pentagon—see Fig. 3a. Thus, to flag non-convex five-folded loops, we impose that distances between oxygen atoms of water molecules $i$ and $i+2$ have to be greater than $1.6 \times R_{cut}$, where $R_{cut}$ is the adjustable parameter determining the degree of convexity to be tolerated. Similarly, we impose that distances between vertices $i$, $j$, and $i+3$ in six-folded rings have to be greater than $(2 \times R_{cut} - \delta_2)$, with $\delta_2 > 0$—see Fig. 3b.

In Fig. 3c, we show the dependence of the number of five- and six-folded rings on $\delta_1$ and $\delta_2$. As rings are allowed to exhibit more deformations, i.e., with increasing $\delta_1$ and/or $\delta_2$, the number of rings increases. To find reasonable values for $\delta_1$ and $\delta_2$ to be used by GRADE, the dependence of the number of cages (defined later in the text) on these parameters is shown in Fig. 3d. In this figure, the number of $6^5 12$ and $6^5 12$ cages is computed by fixing $\delta_1$ to its maximum (0.7) and by varying $\delta_2$. The number of cages increases sharply with increasing $\delta_1$ and $\delta_2$ before reaching a plateau at $\delta_1 = 0.18$ nm and $\delta_2 = 0.26$ nm. Notice that the plateau is reached while the number of rings in Fig. 3c is still increasing. This implies that many rings found by GRADE when using $\delta_1 > 0.18$ nm and $\delta_2 > 0.26$ nm do not contribute to the formation of cages as they are too deformed. These rings can, therefore, be ignored and GRADE uses $\delta_1 = 0.18$ nm and $\delta_2 = 0.26$ nm as a default. Values of $\delta_1$ and $\delta_2$ can be modified using input flag option ‘-d1’ and ‘-d2’.

**Non-planarity.** Some convex five- and six-folded rings in the simulation still exhibit high levels of deformity with some water molecules abnormally projected out of the plane formed by the other water molecules. Fig. 4 shows examples of deformed (panels a and b) and non-deformed (panels c and d) five- and six-folded rings. Since stable cages cannot be constructed with highly deformed rings, they do not need to be included in the list of identified rings. GRADE identifies this type of deformity by dividing five- and six-folded rings into three and four planes, respectively, as shown in Fig. 4 e, f. Angles between these planes (i.e., $\theta_1$, $\theta_2$, $\theta_3$, and $\theta_4$) are computed and are required to be smaller than a given cutoff value, i.e., $\theta_{cut}$. In Fig. 5a, we show that the number of five- and six-folded rings increases with increasing $\theta_{cut}$ as rings found by GRADE are allowed to be more deformed. In Fig. 5b, we show the dependence of the number of cages (defined using $\delta_1 = 0.18$ nm and $\delta_2 = 0.26$ nm) on $\theta_{cut}$. The number of cages increases with $\theta_{cut}$ up to $45^\circ$ where it reaches a plateau. This implies that there are many rings defined by a cutoff angle greater than $45^\circ$ but these deformed rings do not contribute to the formation of cages. Based on this analysis, GRADE uses $45^\circ$ as its default cut-off angle as this value of $\theta_{cut}$ reduces significantly the number of rings without affecting the number of cages found. However, the default cutoff angle can be modified using the input flag option ‘-theta’.

In Fig. 5c, we show that a large fraction of the rings found by GRADE are either non-convex (15% and 27% for five- and six-folded rings, respectively) or exhibit deformities related to planarity (7% and 16% for five- and six-folded rings, respectively). By ignoring these rings the run time of GRADE can be reduced by more than 80%. However, we recommend for users to test the effect of $\delta_1$, $\delta_2$, and $\theta_{cut}$ on their particular systems.
2.3. Cups

To define cups and cages, it is convenient to introduce a few new concepts. Two rings are considered neighbors to each other if they have two vertices (one edge) in common. The number of neighboring rings of a reference ring \( i \) is known as the coordination number of \( i \). Thus, a ring is fully-coordinated if it has the same number of neighboring rings as its size. Rings that are neighbors of a fully-coordinated ring are called lateral rings.

The set of water molecules comprising fully-coordinated and lateral rings form a cup when each of the lateral rings are neighbors to two other lateral rings. To illustrate this definition, Fig. 6 depicts cups with their fully-coordinated five and six-folded rings in red and their lateral rings in black. Using five and six folded rings, the two most common types of cups that can be formed are:

- 5\(^6\) cup in which sizes of fully coordinated and lateral rings are five. The upper index 6 stands for the number of five-folded rings.
- 6\(^5\)\(^6\) cup in which the fully coordinated ring is a six-folded ring and the lateral rings are five-folded. Upper indices stand for the number of six- and five-folded rings.

GRADE finds shared edges between pentagon–pentagon and pentagon–hexagon rings to determine 5\(^6\) and 6\(^5\)\(^6\) cups. All cups are found and assigned a unique identifying number to avoid double counting.

2.4. Cages

The set of molecules comprised by two cups forms a 5\(^{12}\) or 6\(^5\)\(^{12}\) cage when each lateral ring of one cup is neighbor to two lateral rings of the other cup as depicted in Fig. 6a–b. A 6\(^5\)\(^{12}\) cage is formed by the set of molecules in four 6\(^1\)5\(^{6}\) cups in which each pair of these cups shares two five-folded rings with each other. The three types of cages that GRADE detects are:

- 5\(^{12}\) cages, formed by two 5\(^6\) cups and comprising 20 water molecules.
- 6\(^2\)5\(^{12}\) cages, formed by two 6\(^5\)\(^6\) cups and containing 24 water molecules.
- 6\(^4\)5\(^{12}\) cages, formed by four 6\(^1\)5\(^6\) cups and containing 28 water molecules.

To avoid double counting, GRADE assigns a unique identifier number to each cage. In addition to water molecules, GRADE can also process two types of solutes with arbitrary names in the input gro file. Distances between center of mass of each cage and atoms of solutes are calculated and if these distances are smaller than 0.2 nm, the solutes are considered trapped inside the cage. A summary file containing the number of filled and empty cages is provided as output file. Identifying less common types of cages, e.g., 6\(^3\)5\(^{12}\) cages [48], is planned for the next version of GRADE.

2.5. Order parameter \( F_4 \)

In addition to cages, another quantity that is often used to characterize the state of water molecules is the four-body order parameter \( F_4 = \langle \cos 3\phi \rangle \), where \( \phi \) is the torsion angle of the configuration H-O-·-·O-H formed with the outermost hydrogen atoms of two neighboring water molecules. This quantity can be used to distinguish between different tetrahedral networks adopted by water. The flag ‘-f4’ can be used to compute the \( F_4 \) order parameter averaged over all pairs of water molecules in the simulation box.

2.6. Program structure

2.6.1. Source files

GRADE is written in C++ and is made up of a main program file and two supporting resource files:

- GRADE.cpp — main program file which reads input files, writes output files, and calls several functions;
- Functions.hpp — header file, includes function prototypes;
- Functions.cpp — resource file, includes functions needed to find rings, cups and cages.

Files contain original source code written by the authors and they can be compiled using common C++ compilers. A complete list of input and output options of GRADE can be printed using ‘-h’ flag.
2.6.2. Input and output file
GRADE reads the three-dimensional position of atoms in a periodic box provided in gro format. Several software packages, including pdb2gmx which is part of the GROMACS [41] suite and InterMol [49] can be used to convert pdb or trajectory files to gro format. The name of the input file including the extension (.gro) should follow the ‘-i’ flag. The frequency with which frames in the input file are read can be controlled with the flag ‘-fr’.

Four output files are produced by default. One of these files has extension “.xvg” and it contains eight columns which are the frame number, simulation time (if time is provided in the input file), the total number and the number of filled 5\(^\text{12}\) , 6\(^\text{512}\) and 6\(^\text{4512}\) cages, respectively. Filled cages are cages that host a solute molecule, i.e., residue name other than SOL in the input file. Three files with “.gro” extension are also produced by GRADE. One file for each type of cages which contain positions of water molecules forming the cages, all solute molecules, followed by solute molecules trapped inside cages. The molecule name of the latter is “CBX”. This allows trapped solutes to be visualized independently from other solutes using VMD or other atomic visualization software. By default, names of these output files are formed by concatenating the input name with “_cage512_i.gro”, “_cage62512_i.gro” and “_cage64512_i.gro”, where \(i\) stands for the frame number. Alternative output names can be provided using the ‘-o’ flag. The frequency with which output files are written can be controlled with the ‘-dt’ flag. By default ‘-dt’ and ‘-fr’ are set to one.

3. Results
As an example of how GRADE can be used to study clathrate formation, we now analyze the four simulations B defined in Table 1. First, GRADE needs to be compiled using for example the GNU compiler collection (or any C++ compiler):

```
g++ GRADE.cpp Functions.cpp -o GRADE
```

A Makefile provided with the source files compiles the executable and ensures that the minimum GNU version requirement is met. If trajectory of water and solute molecules are provided in a file named “trajectory.gro”, GRADE can be executed using the command line:

```
./GRADE -i trajectory.gro -f4 yes
```

This generates five output files: “trajectory.xvg”, “trajectory_cage512_frame.gro”, “trajectory_cage62512_frame.gro”, “trajectory_cage64512_frame.gro” and “F4.gro”. A separate gro-file is generated for each analyzed frame or as specified in the command line using the ‘-dt’ flag. Fig. 7a–d shows the time-evolution of the number of 5\(^\text{12}\) (in red), 6\(^\text{512}\) (blue) and 6\(^\text{4512}\) (green) cages of the four simulations in setup B, defined in Table 1. This time evolution is provided in the output file “trajectory.xvg”. Initially, there are no cages in the simulation box as the system is in the liquid state. After an induction time that varies from \(\sim 100\) to 250 ns for the four systems studied, the number of cages increases abruptly to approximately 10–15. This is consistent with theories of blob formation [50] in which the spontaneous nucleation of cages from the liquid state requires the formation of a critical nucleus. After the formation of this nucleus, the number of cages increases in a non-continuous manner whereby short periods of annihilation are followed by regrowth.

Fig. 7e shows snapshots of cages and methane molecules at four points of one of the trajectories (see panel b). Snapshot I

---

Please cite this article as: F. Mahmoudinobar and C.L. Dias, GRADE: A code to determine clathrate hydrate structures, Computer Physics Communications (2019), https://doi.org/10.1016/j.cpc.2019.06.004.
corresponds to the first $S^2$ cage that is filled with a methane molecule in the simulation (time = 110 ns). This cage is, however, unstable and it disappears in snapshot II (time = 180 ns) which shows the first stable cluster made of eight $S^2$ (in red) and six $6S^2$ cages (in blue). This cluster grows through the addition of newly formed cages as depicted in snapshots III (time 250 ns) and IV (time = 580 ns).

The four-body order parameter, $F_4 = \langle \cos 3\phi \rangle$, is plotted for the four simulations in Fig. 8. This parameter is provided in the output file “F4.xvg”. $F_4$ takes values of 0.7, 0 and −0.4 for SI hydrate, liquid water and ice, respectively to distinguish different phases of water [51, 52]. In our simulations, $F_4$ initially starts from values close to zero, which represents the liquid state, and it increases abruptly as water molecules evolve into clathrate structures.

4. Conclusion

In this paper, we present a computer code to analyze water structures related to the formation of gas clathrate. The code identifies $S^2$, $6S^2$ and $6S^2$ cages as well as the four-body order parameter ($F_4$) given the atomic coordinates of water molecules. These are main quantities used to quantify clathrate formation in molecular dynamics simulations. We anticipate that this open-source code will serve as a template by the community and it should be of interest to researchers studying not only natural gases but also antifreezing proteins and hydrophobic interactions. First, the method used to analyze rings in order to compute cages is presented followed by a brief discussion of the computer code. Second, an example of how the code can be used to study the spontaneous nucleation of methane clathrates is presented. We anticipate that the freely available source code will enable research groups to easily analyze their simulation results. Also, the code can be modified to allow the investigation of other order parameters to quantify water structures. It should be mentioned that we have implemented restrictions on the conditions required by water molecules to be considered part of five- and six-folded rings, however, these restrictions have no effect on the number of cages found by the code and they account for a significant speed up of the computer code. Further speed up of GRADE can be provided through parallelization which is planned for a future version of the code.

Acknowledgments

This work was made possible by a grant from the ACS-PRF, United States #58024-ND6. Computational resources for this work were provided by Compute Canada and NJIT’s High Performance Computer center. Authors thank Dr. Zhaoyian Su for his insightful comments on output file processing.

References