AIP | The Journal of Chemical Physics

Crystal-structure prediction via the Floppy-Box Monte Carlo algorithm: Method and application to hard (non)convex particles

Joost de Graaf, Laura Filion, Matthieu Marechal, René van Roij, and Marjolein Dijkstra

Citation: J. Chem. Phys. **137**, 214101 (2012); doi: 10.1063/1.4767529
View online: http://dx.doi.org/10.1063/1.4767529
View Table of Contents: http://jcp.aip.org/resource/1/JCPSA6/v137/i21
Published by the American Institute of Physics.

Crystal-structure prediction via the Floppy-Box Monte Carlo algorithm: Method and application to hard (non)convex particles

Joost de Graaf,[1] Laura Filion,[1] Matthieu Marechal,[2] René van Roij,[3] and Marjolein Dijkstra[1]

[1]*Soft Condensed Matter, Debye Institute for Nanomaterials Science, Utrecht University, Princetonplein 5, 3584 CC Utrecht, The Netherlands*

[2]*Heinrich-Heine Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany*

[3]*Institute for Theoretical Physics, Utrecht University, Leuvenlaan 4, 3584 CE Utrecht, The Netherlands*

(Received 4 September 2012; accepted 31 October 2012; published online 3 December 2012)

In this paper, we describe the way to set up the floppy-box Monte Carlo (FBMC) method [L. Filion, M. Marechal, B. van Oorschot, D. Pelt, F. Smallenburg, and M. Dijkstra, Phys. Rev. Lett. **103**, 188302 (2009)] to predict crystal-structure candidates for colloidal particles. The algorithm is explained in detail to ensure that it can be straightforwardly implemented on the basis of this text. The handling of hard-particle interactions in the FBMC algorithm is given special attention, as (soft) short-range and semi-long-range interactions can be treated in an analogous way. We also discuss two types of algorithms for checking for overlaps between polyhedra, the method of separating axes and a triangular-tessellation based technique. These can be combined with the FBMC method to enable crystal-structure prediction for systems composed of highly shape-anisotropic particles. Moreover, we present the results for the dense crystal structures predicted using the FBMC method for 159 (non)convex faceted particles, on which the findings in [J. de Graaf, R. van Roij, and M. Dijkstra, Phys. Rev. Lett. **107**, 155501 (2011)] were based. Finally, we comment on the process of crystal-structure prediction itself and the choices that can be made in these simulations. © *2012 American Institute of Physics*. [http://dx.doi.org/10.1063/1.4767529]

I. INTRODUCTION

The prediction of crystal structures for atomic, colloidal, and nanoparticle systems, based solely on knowledge of the inter-particle interactions, is of primary importance in guiding the development of new materials. This problem has received a lot of attention over the past few decades,[1] but there still remain many unanswered questions concerning the efficient prediction of crystal structures. Commonly used techniques rely on simulated annealing,[2,3] genetic algorithms,[4,5] or Monte Carlo (MC) basin hopping.[6] However, these methods do not work well for systems that have a large entropic contribution to the free energy. Hard-particle systems pose a particular problem, since the entropy is the only factor that imposes the crystal structure at a fixed pressure, temperature, and particle shape. These entropy-driven systems can in principle be studied using an ergodicity search algorithm[7] or a metadynamics method,[8] but both approaches have their limitations. Other methods which start with a guess of the crystal structure (from, e.g., experiments) have also been used.[9]

Interest in the subject of crystal-structure prediction has intensified over the past few years due to the remarkable advancements made in colloid and nanoparticle synthesis.[10,11] Not only can spherical particles be synthesized with a high level of precision and reproducibility, but also a wide variety of convex (faceted) shapes, such as cubes,[12–14] octahedra,[15–17] tetrahedra,[18–20] and many more.[21–25] Perhaps most remarkable of all is the level of control that has been attained over the synthesis of nonconvex, irregular, and even punctured particles. Branched colloids and nanocrystals, such as octapods[26–30] and tetrapods,[31,32] have been created, as well as other nonconvex shapes, e.g., nanostars[15,33–35] and colloidal caps.[36–38] Moreover, there is a better understanding of the way to achieve phase behaviour that is dominated by entropic contributions in experimental systems.[39–42]

To address the problem of crystal-structure prediction at finite pressures, for systems that are mostly entropy driven, the method of floppy-box Monte Carlo (FBMC) was recently introduced.[43] It is based on Monte Carlo simulations and uses compression from the fluid to determine candidate structures. The FBMC technique was used successfully on a wide variety of systems consisting of, for instance: spheres that interact via a combination of hard and attractive pair potentials, binary mixtures of hard spheres, star polymers which have semi-long-range soft interactions,[43] as well as directionally anisotropic pair potentials (i.e., "patchy" particles).[44] Truly long-range dipole-dipole interactions, for which Ewald Sums were employed to determine the total energy of the system, were also considered.[43] Moreover, the FBMC algorithm was applied to determine a lower limit to the packing fraction of the densest configuration for highly shape-anisotropic solids.[45] Employing the FBMC technique in this way connects the field of materials science with fields as diverse as discrete geometry, number theory, and computer science.[46–50] We only briefly go into this here and refer the reader to Ref. 45 for a discussion of the recent developments in analysing densest packings. The FBMC technique led to the discovery of a wide variety of new crystal structures[43,45,51]

and it has proven itself to be a remarkably efficient and robust method. However, to date a fully comprehensive article describing in detail the implementation of the method has not been published.

In this paper, we describe the elements required to set up a FBMC crystal-structure-prediction algorithm, as implemented in Refs. 43 and 45. We assume only basic familiarity with thermodynamics and simulation techniques, such that this text can be used as a self-explanatory recipe. In Sec. II, we discuss the components that define a FBMC simulation. This is followed by an introduction of the MC trial moves and acceptance rules that we employ in Sec. III. Section IV presents a way to efficiently check for particle overlaps in the system, which is based on minimizing the number of periodic images that have to be taken into account. This, as we will show in Sec. IV, is a nontrivial problem due to the size of the box in comparison to the range of the interaction between the particles. Two types of hard-particle overlap routines are explained in Sec. V, by which simulations of (nonconvex) irregular faceted particles can be performed. The technique of lattice reduction, which is essential to prevent unphysical distortions of the simulation box, is introduced in Sec. VI. We briefly comment on soft interactions and external fields in Sec. VII, before we examine the way to use the FBMC algorithm to allow for efficient crystal-structure prediction in Sec. VIII. In Sec. IX, we discuss the properties of our algorithm and give a brief comparison to other hard-particle algorithms developed to predict dense structures. We also mention some results for the crystal structures we obtained by applying the FBMC algorithm (see the supplementary material,[52] which contains the structures corresponding to the results in Ref. 45) in relation to other work in the field. We conclude with an outlook in Sec. X.

II. CHARACTERISATION OF THE METHOD

The FBMC algorithm is based on an ordinary MC simulation in the isothermal-isobaric ensemble, i.e., with the number of particles N, pressure P, and temperature T fixed during the simulation. There are four properties that turn an ordinary *NPT* MC simulation into a tool for crystal structure prediction.

(i) The number of particles N is small, typically $N < 12$.
(ii) The three vectors that span the simulation box are allowed to vary independently of each other in both their length and orientation. This is the origin of the term *floppy box*, which was adopted to emphasize that the box does not have a fixed shape. It is a variable-box-shape method, which is in common use in computational studies of colloids.[8,53] However, by the adjective "floppy-box" (MC) we refer to the whole of the technique to predict crystal structures, rather than just the fact that the box shape is variable. Note that an *NPT* variable-box-shape simulation is essentially an isothermal-isotension simulation,[53] for which a special form of the imposed stress tensor is used. This tensor only has diagonal elements, which are all the same and are directly proportional to the pressure, i.e., according to the hydrostatic-pressure assumption. Hence, we prefer the wording "isothermal-isobaric" or *NPT* ensemble.
(iii) To predict crystal structures the *NPT* part of the simulation is preceded by a compression from a dilute phase (gas or liquid), which is accomplished by increasing the pressure to drive the system towards higher densities.
(iv) To effectively predict crystal structures, it is necessary to perform FBMC simulations for the same system many times with different random seeds, initial conditions, compression paths, etc.

The first two properties – that the number of particles is small and that the box can deform – essentially allow the box to act as a *unit cell* for the crystal structures we are interested in. The periodicity of the boundary conditions, in combination with the small number of particles, ensures that the end result is a crystal. Fast and efficient exploration of crystal-structure candidates is made possible by choosing the method of compression (iii) and the way in which the algorithm is repeatedly applied (iv).

There is, however, a computational bottleneck that has to be overcome for any FBMC simulation to obtain such fast and efficient sampling: determining the number of periodic images that has to be checked. A particle in the unit cell of a crystal (simulation box) interacts with its neighbours, which are its own periodic images in the FBMC method. For instance, a single particle in a cubic unit cell is surrounded by a layer of 26 neighbours, followed by another layer that contains 98 (secondary) neighbours, etc. If a particle has an interaction range larger than the dimensions of the box, which is typical for FBMC simulations, it will notice the particles in one or more adjacent layers. When the particle is translated or rotated, the interaction between it and all these neighbours/images has to be checked to determine whether the move is accepted or rejected. Even worse, the number of relevant image layers may have increased when the box is deformed. Each additional layer that has to be considered increases the number of interaction computations quadratically, which quickly leads to unreasonable computational overhead even for simple interaction potentials.

However, it is often not necessary to check all images in a layer, e.g., a sphere centred in a cubic box with a diameter slightly larger than the box length can only interact with 6 of the 27 images in the next layer. As we will show, determining a small, yet sufficient number of images to be taken into account, is not straightforward in general, especially for very deformed boxes. In Sec. IV, we explain a procedure to obtain these images in a fast way. We shall first discuss the case where there are only pairwise hard-core interactions in a system of shape-anisotropic particles, before discussing the particles with soft-interactions.

III. THE ENSEMBLE AND MONTE CARLO MOVES

We assume that the system of interest consists of N particles at positions $\mathbf{r}_j$, of which the orientation is specified by orientation vectors $\mathbf{q}_j$, where j is an index that runs from 1 to N. The particles are contained in a box spanned by the three box vectors $\mathbf{v}_i$, with $i = 1, 2, 3$. In this description, the vertices

of the box are given with respect to a standard Cartesian coordinate frame and one of them is located at the origin. The set of $\mathbf{r}_j$ gives the location of the particles' centre of rotation, also with respect to this coordinate frame. The orientation of the jth particle is obtained by applying the rotation matrix generated by $\mathbf{q}_j$ to a predetermined initial orientation.[53] Let the volume of the jth particle be given by $V_j > 0$. The packing fraction is then $\phi = (1/V)\sum_j V_j$, with $V \equiv |\mathbf{v}_1 \cdot (\mathbf{v}_2 \times \mathbf{v}_3)|$ the volume of the system. Each particle has an outscribed sphere, of which the centre is located at the rotation point $\mathbf{r}_j$ and the radius is given by $R_{O,j}$. Note that this sphere does not necessarily satisfy the mathematical definition of the *minimum* outscribed sphere. The radius of the largest outscribed sphere of all N particles in the system of interest is denoted $R_O = \max_j R_{O,j}$. Finally, U gives the total energy of the system, which implicitly depends on the sets $\mathbf{v}^3 \equiv \{\mathbf{v}_i\}$, $\mathbf{r}^N \equiv \{\mathbf{r}_j\}$, and $\mathbf{q}^N \equiv \{\mathbf{q}_j\}$, i.e., $U \equiv U(\mathbf{v}^3, \mathbf{r}^N, \mathbf{q}^N)$. For a system with only hard-particle interactions U assumes two values: $\beta U = 0$, when there are no overlaps, and $\beta U = \infty$, when there are overlaps.

For a FBMC simulation we consider four types of trial moves to sample phase space: translation and rotation of the particles and scaling and deformation of the box, see Fig. 1. These moves must satisfy the acceptance criterion for detailed balance in the *NPT* ensemble, which is based on the Metropolis algorithm.[53,54] We write the probability of a move to be accepted as $\mathrm{acc}(o \to n)$, where "o" and "n" are the labels for the old and new state, respectively. Translation, rotation, and scaling moves are performed according to Ref. 53, where we apply Ref. 55 to construct the random perturbation of the orientations. The acceptance criterion for these moves is given by

$$\mathrm{acc}(o \to n) = \min\left(1, \exp\left[-\beta\left(U_n - U_o + P\,(V_n - V_o)\right) + (N+1)\log\left(\frac{V_n}{V_o}\right)\right]\right), \quad (1)$$

where the subscripts indicate the new and old values, P is the pressure, and V is the volume of the system, as before. Here, we assume logarithmic steps in V for the scaling moves, hence the natural logarithm in Eq. (1) is preceded by the factor $(N+1)$.[53] For the deformation moves, we select one element of one of the box vectors at random. This element is perturbed by a small number, which is chosen from a uniform probability distribution over an interval symmetric around zero. These moves also satisfy detailed balance for the *NPT* ensemble, when the acceptance rule

$$\mathrm{acc}(o \to n) = \min\left(1, \exp\left[-\beta\left(U_n - U_o + P\,(V_n - V_o)\right) + N\log\left(\frac{V_n}{V_o}\right)\right]\right) \quad (2)$$

is used.[53,56] It is easy to verify that Eq. (2) holds for this type of move using properties of the cross- and dot-product to show that the deformations described above allow for perturbations of the volume that scale linearly with the perturbation of the box vectors.

We introduced two types of moves that change the volume of the box – scaling and deformation – because we found that the combination of the two typically led to faster equilibration and exploration of phase space than using deformation moves only. The scaling moves essentially create space in the system for deformations to be more readily accepted at high volume fractions. The order in which we apply these moves is chosen at random and satisfies the following probability distribution. For $N = 1$ particles, we only require deformation and scaling moves, but sampling is sped up by also allowing rotation moves. We do *not* make any assumptions concerning the orientation of the box here. A commonly used choice for variable-box-shape simulations is to have one of the box vectors along the x axis, another vector in the positive part of the xy-plane, and the third in the $z > 0$ half space. However, this choice *necessitates* the use of rotation moves for $N = 1$ in order to fully explore phase space. We found that roughly 70% rotation, 15% scaling, and 15% deformation trial moves yields relatively efficient sampling. For $N \geq 2$ we typically used 35% translation, 35% rotation, 15% scaling, and 15% deformation trial moves. The ratio assigned to the volume moves may appear to be somewhat high, when compared to typical ($N \gg 10$) MC simulations, for which they are usually applied with probability $\sim 1/N \ll 1$. However, these values are not unreasonable, since there are only a few particles in the box and deformation plays an important role in sampling possible structures.

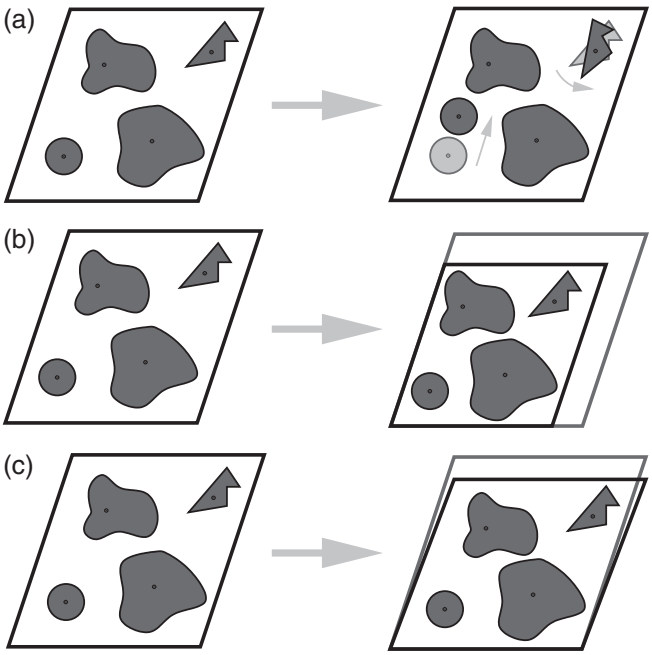


FIG. 1. A two-dimensional (2D) representation of the 4 types of Monte Carlo (MC) trial moves that we consider for the floppy-box Monte Carlo (FBMC) simulations. (a) Translation and rotation of the particles. (b) Uniform scaling of the box. (c) Box deformation.

IV. CONSTRUCTION OF THE IMAGE-LISTS

In the following, we do not make any additional assumptions on the composition of the system other than the ones

specified in Sec. III: mixtures of highly shape-anisotropic particles are allowed. When there are only hard-particle interactions in the system, the acceptance criteria of Eqs. (1) and (2) are substantially simplified. Translation and rotation moves are always accepted when they do not result in overlaps. However, as explained before, checking for overlaps in even the simplest of systems can be time consuming due to the strong influence of periodicity. For scaling and deformation moves it is even more expensive, since the minimum number of periodic images that has to be taken into account also needs to be redetermined. Hence, we first verify that such a move is not rejected on the basis of the pressure/volume part of $\mathrm{acc}(o \rightarrow n)$ [Eqs. (1) and (2)] before checking for overlaps. It is also convenient to use the packing fraction ϕ as an early out at high densities, since $\phi \leq 1$ must always hold.

In order to check for overlaps, as well as perform the various moves, it is convenient to introduce a set of scaled coordinates in analogy to the procedure outlined in Ref. 53. Let $\mathbf{s}_j \in [0, 1)^3$, with $j = 1, \ldots, N$ an index, denote the scaled position coordinates of the particles. There is a bijective function that relates these scaled coordinates to the real-space positions of the particles in the box, i.e., $M : [0, 1)^3 \rightarrow \mathbb{R}^3$, $\mathbf{r}_j = M\mathbf{s}_j$. Here, M is the matrix generated by the box vectors: $M = (\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3)$, where it is understood that the box vectors are the columns of the matrix. This function is bijective when the volume of the box $V > 0$, since this implies that $\det(M) \neq 0$. The scaled coordinate frame allows us to determine the number of periodic images of the box that we need to check for overlaps in order to confirm that there are no overlaps in the entire system. Here, we use the fact that checking for overlaps in the entire system is equivalent to checking for overlaps between a particle in the box and

(i) another particle in the box,
(ii) its own periodic images, and
(iii) other particles' periodic images,

for all particles in the box. It is efficient to first carry out step (i) for all particles, making sure not to double check, then step (ii) for all particles, and finally step (iii). For $N = 1$ the algorithm reduces to step (ii). It is computationally favourable to only check for overlaps between a particle and a minimum number of periodic images. However, obtaining this minimum number of images should not go at the expense of the overall speed of the algorithm, since volume changes occur frequently. Hence, we construct the lists of images as follows.

Recall that the largest outscribed-sphere radius is given by R_O. When any two particles/images are a distance of $2R_\mathrm{O}$ apart, they do not overlap. However, it is difficult to determine how many layers of images we need to take into account in each direction such that every image-point inside this sphere is considered. We first present the calculation of the number of "self-images" [for step (ii)]; the other case [step (iii)] for which a second image list is required, will be addressed afterwards. We may assume, without loss of generality, that the particle in the box is located in the origin. We pick a plane that goes through the origin and that is orthogonal to the vector $\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$, to bisect the sphere of radius $2R_\mathrm{O}$, see Fig. 2(a). By choosing the plane orthogonal to the sum of the box vectors we can (usually) avoid it intersecting image points. We only consider images that are on one side of the plane or that lie in it. This is justified because of point-symmetry as can be easily understood by considering Fig. 2(a). Take any one of the green points in Fig. 2(a) and mirror it in the origin. This image interacts with the particle at the origin in the same way as the particle at the origin interacts with the image located at the original (green) point, because the orientation of all images is the same. That is to say, the periodicity allows us to translate the entire crystal by the distance vector between the two points and this translated structure coincides with the original. Hence, we have to check only one of the two point-symmetric possibilities.

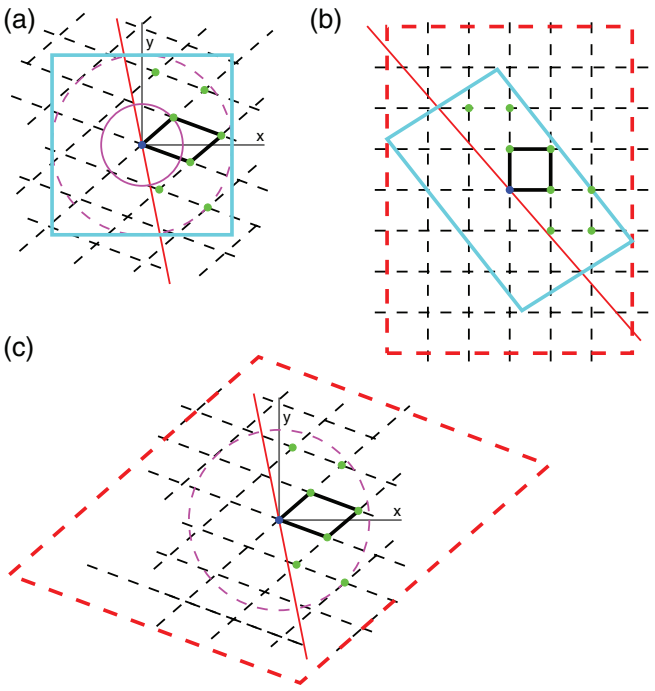


FIG. 2. The construction of the self-image list in two dimensions; the three-dimensional (3D) case is analogous. (a) The simulation box (black parallelogram), the particle's position (blue dot), the coordinate frame (grey lines), the periodic images (dashed black lines), the outscribed circle of radius R_O (solid magenta), and double-radius circle (dashed magenta). The solid red line indicates the point-symmetry axis: all images relevant for the overlap check are indicated with green dots. An axis-aligned cyan square envelops the $2R_\mathrm{O}$ circle. (b) In the scaled coordinate frame a bounding rectangle $[-N_1, N_1] \times [-N_2, N_2]$ (dashed red line) can be constructed. (c) The red bounding rectangle can be mapped back to the original frame and gives an upper bound to the number of images that have to be checked in each direction.

A sufficiently large list of self-images is constructed by considering a cube with vertices $\mathbf{c}_n = 2R_\mathrm{O}(\pm\hat{\boldsymbol{x}} \pm \hat{\boldsymbol{y}} \pm \hat{\boldsymbol{z}})$ with $n = 1, \ldots, 8$ and $\hat{\boldsymbol{x}}$, $\hat{\boldsymbol{y}}$, and $\hat{\boldsymbol{z}}$ the Cartesian unit vectors. This cube envelops the sphere at the origin with radius $2R_\mathrm{O}$, see Fig. 2(a). By applying the inverse matrix M^{-1} to the eight $\mathbf{c}_n$ vertices of the cube, we obtain eight vertices that span a parallelepiped (Fig. 2(b)), say $\mathbf{p}_n = M^{-1}\mathbf{c}_n$. Using the vertices $\mathbf{p}_n$ the upper bounds to the number of images that need to be checked in each direction can be constructed

$$\begin{aligned} N_1 &= \lceil \max_n (\mathbf{p}_n \cdot \hat{\boldsymbol{x}}) \rceil; \\ N_2 &= \lceil \max_n (\mathbf{p}_n \cdot \hat{\boldsymbol{y}}) \rceil; \\ N_3 &= \lceil \max_n (\mathbf{p}_n \cdot \hat{\boldsymbol{z}}) \rceil, \end{aligned} \tag{3}$$

respectively, where $\lceil \cdot \rceil$ indicates the ceiling function, i.e., $\lceil x \rceil$ is the smallest integer larger than x. Since the parallelepiped is symmetric around the origin, the positions of the relevant images in the scaled coordinate frame are located in the rectangle $[-N_1, N_1] \times [-N_2, N_2] \times [-N_3, N_3]$ (Fig. 2(b)). For images that fall outside of this rectangle we can be certain that they do not overlap with the particle at the origin. Using the matrix M, we may now establish the equivalent set of images in the regular coordinate frame $P_{\text{im}} = \{i\mathbf{v}_1 + j\mathbf{v}_2 + k\mathbf{v}_3\}$, with $i = -N_1, \ldots, N_1$; $j = -N_2, \ldots, N_2$; $k = -N_3, \ldots, N_3$; and $i + j + k \neq 0$, see Fig. 2(c).

The list P_{im} can contain many elements which are irrelevant to the overlap check – i.e., points which fall outside of the halfsphere with radius $2R_{\text{O}}$ – when the box becomes strongly deformed or the particle has an "odd" shape. We employ the following steps to remove these images. Using the dot-product of a vector in P_{im} with the normal of the point-symmetry plane, we can efficiently eliminate all images that are not on the right side of this sphere-bisecting plane. Points that have a distance to the origin greater than $2R_{\text{O}}$ are also removed from the list by simply calculating the length of the position vector. Inscribed-sphere checks between the particle at the origin and a particle at a lattice-site in the reduced set of images $\tilde{P}_{\text{im}}$ are used to confirm that there are no situations where two particles clearly overlap, before switching to a more computationally expensive overlap algorithm. Here we employ a "concentric approach" to check for overlaps. That is, we first consider all images for which $|i| + |j| + |k| = 1$, followed by the set of images for which $|i| + |j| + |k| = 2$, etc. If there are overlaps it is far more likely that they are encountered close to the original particle, rather than further away. If there are no overlaps detected in the entire system – having checked steps (i), (ii), and (iii) – the reduced list is accepted and can be used to check for overlaps until an attempted volume move (scaling or deformation) requires a new self-image list to be constructed.

Checking for overlaps between a particle in the box and another particle's images [step (iii)] is a simple matter of extending the above reduced self-image list. We add 1 layer of images in the positive $\mathbf{v}_i$ directions to the set $\tilde{P}_{\text{im}}$. It is easily understood that it is sufficient to check up to this range in periodic images. The list $\tilde{P}_{\text{im}}$ already contained all relevant image points with a distance $2R_{\text{O}}$ and the distance added by considering two particles in the box is at most one box length in each $\mathbf{v}_i$ direction. The fact that the problem is no longer point symmetric is overcome by checking one particle with another particle's images and vice versa.

V. HARD-PARTICLE OVERLAP ALGORITHMS

In this section, we briefly discuss two hard-particle overlap algorithms, the method of separating axes and a triangular-tessellation-based technique, which can be used in combination with the FBMC technique to study crystal structures for faceted (nonconvex) particles. These algorithms were recently employed to great effect in Refs. 28 and 45. The two routines are predominantly used in computer-game and engineering applications.[57–59] However, such methods have gained popularity in the physics and mathematics communities and show great promise for future studies, especially with the emergence of the graphics card as a powerful platform for particle simulations and image processing.[60–62] Moreover, there has been a marked increase in the ability to synthesize a stunning array of faceted (nonconvex) particles,[15,26,27,31–35] as well as a large improvement in the level of control with which such particles can be prepared.[28] This has led to particular interest from the materials science community in these overlap algorithms to perform simulations on nanoparticle and colloid systems.[63–71]

A. The method of separating axes

The method of separating axes is an overlap algorithm that can be applied to convex particles. This technique is based on the (Hahn-Banach) separating-hyperplane theorem for convex sets in Euclidean space.[72] The theorem implies that for two disjoint convex sets there exists a plane between these sets that does not intersect either. That is, one of the particles is in one of the halfspaces defined by the separating plane and the other particle is in the opposite halfspace, respectively. Any axis orthogonal to the separating plane is referred to as a *separating axis*, because orthogonal projections of the convex sets onto this axis are disjoint, also see Fig. 3. Therefore, the problem of determining if two objects overlap can be reduced to finding a separating axis: if there is such an axis, the objects do not overlap; if there is no such axis, they do. The procedure to find a separating axis for two arbitrary convex particles cannot be performed in a finite time, since an infinite number of directions may result in a possible separating axis. However, it can be shown that for two convex three-dimensional (3D) polyhedra, only a finite number of directions has to be checked.[72] This makes it possible to turn the separating-hyperplane theorem into an efficient algorithm.

Only the set of vectors normal to the faces of the two polyhedra and the set of vectors generated by cross-products between two edges, one from each polyhedron, have to be

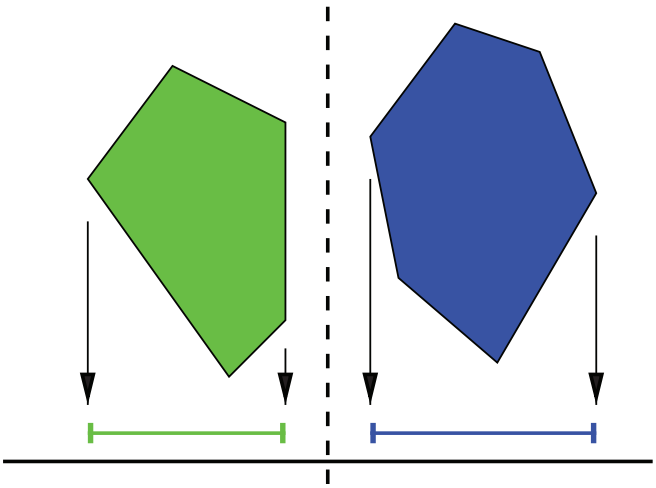


FIG. 3. A 2D representation of method of separating axes. We consider two disjoint convex polygons. The dashed vertical line indicates a possible separating plane for these particles, and the solid horizontal line an associated separating axis. Note that the projection of the particles on the separating axis results in disjoint domains, as indicated by the arrows and line segments.

checked for separation. The way in which to perform such checks efficiently is described in great detail for 3D polyhedral particles, as well as for two-dimensional (2D) polygons, in Ref. 73. One minor addition to the overlap routine of Ref. 73 should be made, when applying it in a FBMC simulation close to the maximum packing fraction. At these high densities it is possible that two particles come together in such a way that the cross-product of two of their respective edges becomes very small. This leads to numerical instabilities in the algorithm proposed in Ref. 73. Such instabilities can result in a separating axis being identified, despite the particles interpenetrating. We found that normalizing the cross-product vector eliminates the problem for all the systems we considered.

B. Triangular-tessellation-based overlap algorithms

The method of separating axes described above has the advantage that it is very easy to implement, can be applied without modification to a wide class of particles, and is also computationally efficient. However, due to the algorithm's dependence on the separating-hyperplane theorem it cannot be extended to handle nonconvex particles.

Checking for overlaps between nonconvex particles can be made possible by approximating the shape of a particle with a collection of spheres or rods.[70,74–78] The respective overlap algorithms are simple and efficient enough to justify the use of spheres and rods as building blocks for larger objects. Such approximations give exact results is the case of, e.g., dumbbells, which consist of two interpenetrating spheres. However, a prohibitive number of spheres or rods may be required to give a decent approximation, especially when an object contains both drastic changes in curvature and large flat parts. Only recently were the first attempts made to study systems that contain relatively complex curved particles. Nonconvex shapes with sharp edges and smooth surfaces, such as superdisks,[79] bowls,[80] and curved triangles,[71] have also come under investigation. For bowls[80] an overlap algorithm was devised unique to this shape. Moreover, Ref. 81 introduced a general method to handle collision detection for smooth (nonconvex) objects in molecular dynamics simulations. Unfortunately, none of the above methods are particularly suited to study nonconvex faceted particles.

Simple nonconvex polyhedral (faceted) particles may be broken up into convex constituents, for which the method of separating axes can be employed. However, an alternative to this type of partitioning exists, which is better suited to study more complex faceted shapes. In computer-game and engineering applications, the surface of the objects is usually defined by a polygonal mesh.[57] Two particles overlap when there is an intersection between a pair of polygons in the respective meshes. Such intersections are easily determined on a polygon-by-polygon basis, because of a polygon's simple shape, which also makes polygons ideally suited to describe objects. The algorithms employed to search for overlaps are often specifically designed to handle highly irregular objects and are typically very general in their setup. Moreover, they can usually be straightforwardly implemented, and a lot of effort has been put into their optimization. An example of such optimization is the use of an oriented-bounding-box-tree (OBB-tree)[82] for high-polygon models. An OBB-tree essentially breaks up the model into smaller pieces in a very efficient way. The tree allows the polygon-based overlap routine to be applied solely to parts of the model that are in close proximity.

Due to our familiarity with approximating the surface of an object with triangles, so-called triangular tessellation, also see Refs. 83 and 84, we decided to base our overlap algorithm for nonconvex hard particles on the *Robust and Accurate Polygon Interference Detection* (RAPID) library.[57] The RAPID library is designed specifically to quickly determine if there are triangle intersections; it employs an OBB-tree for larger objects and is easy to set up. See Refs. 58 and 59 for some alternative overlap-detection libraries. The procedure of performing an overlap check is as follows. Particles are tessellated with triangles, according to the method of Ref. 83, to generate simulation models. This triangular tessellation is exact for polyhedral particles, i.e., the shape is approximated perfectly. These models are passed to the RAPID library. To perform an overlap check we only need to input two position vectors and two rotation matrices in the relevant RAPID subroutine. This subroutine returns a Boolean value that specifies whether there are triangle intersections or not. We found RAPID to be extremely stable, even for FBMC simulations in the high-pressure dense-packed limit, where the numerical stability of any algorithm is put to the test. Unlike the method of separating axes inclusions may occur, when there is a substantial size difference between particles, since the triangular-tessellation-based algorithm only considers the surface area of a particle. We can use an inscribed-sphere check to reject trial moves that result in an inclusion. The RAPID algorithm also allows for interior and even disconnected triangles to be added to a model[57] that can be used to further prevent inclusions, when the gap between a particle and its inscribed sphere is too wide.

VI. UNPHYSICAL DISTORTIONS AND LATTICE REDUCTION

In the above discussion, we did not take into consideration that by allowing the box to deform, especially when compressing from a dilute phase, it may become very distorted. That is to say, the box can become very flat or elongated, when the angles between the lattice vectors become large or small. This slows down our algorithm, since an enormous number of periodic images needs to be taken into account in order to determine whether a move is accepted or not. Moreover, it can be difficult to determine whether two states actually have the same structure if one or both of the simulation boxes is/are heavily distorted. To improve the efficiency and to facilitate recognition of the structures that the FBMC algorithm produces, we require lattice reduction.

Lattice reduction is the process by which a set of basis vectors for a lattice is replaced by an equivalent set of basis vectors, which are shorter and more orthogonal. That is to say the surface-to-volume ratio of the box is minimized. In Ref. 4, an algorithm is proposed to accomplish this lattice reduction

in a 3D system. We modify this algorithm as follows. We measure the distortion of the simulation box using the function

$$\mathcal{C}(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3) = \frac{1}{9}\left(|\mathbf{v}_1| + |\mathbf{v}_2| + |\mathbf{v}_3|\right) \cdot \frac{\left(|\mathbf{v}_1 \times \mathbf{v}_2| + |\mathbf{v}_1 \times \mathbf{v}_3| + |\mathbf{v}_2 \times \mathbf{v}_3|\right)}{\mathbf{v}_1 \cdot (\mathbf{v}_2 \times \mathbf{v}_3)} \quad (4)$$

with $|\cdot|$ the vector norm. This function is obtained by multiplying the average box-vector length with the size of the box's surface, dividing by the total volume of the box, and normalizing this quantity such that $\mathcal{C} = 1$ for a cube. It can be shown that $\mathcal{C} > 1$, when the box is not cubic. We empirically established a criterion for lattice reduction: if $\mathcal{C} \lesssim 1.5$ the box is sufficiently orthorhombic and we do *not* perform lattice reduction; this number may be tweaked to better suit a particular system. When $\mathcal{C} > 1.5$, we follow Ref. 4 and generate a set of 12 lattice combinations

$$\begin{aligned} &\{\mathbf{v}_1 \pm \mathbf{v}_2, \mathbf{v}_2, \mathbf{v}_3\}, \{\mathbf{v}_1 \pm \mathbf{v}_3, \mathbf{v}_2, \mathbf{v}_3\}, \\ &\{\mathbf{v}_1, \mathbf{v}_2 \pm \mathbf{v}_1, \mathbf{v}_3\}, \{\mathbf{v}_1, \mathbf{v}_2 \pm \mathbf{v}_3, \mathbf{v}_3\}, \\ &\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \pm \mathbf{v}_1\}, \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \pm \mathbf{v}_2\}. \end{aligned} \quad (5)$$

We calculate the surface area for each of these potentially more orthorhombic boxes and select the one with the smallest surface area. For this new box, another set of 12 combinations is constructed according to Eq. (5) and the procedure is repeated. This process is terminated when the smallest surface area among these 12 candidates is greater than the surface area of the box in the previous iteration. See Fig. 4 for an illustration of the procedure.

Full lattice reduction is always accomplished within a finite number of steps, but it is prudent to impose a cutoff at roughly 10 iterations. We found that the algorithm converges quickly, typically taking no more than 3–5 iterations, for boxes with $1.5 \lesssim \mathcal{C} \lesssim 10$. When the fully lattice-reduced box is found the particles are placed back in the box. Note that the number of particles in the box is preserved under lattice reduction. In addition to lattice reduction, one can also implement a simple constraint on the angles that the box vectors can make with each other, see Ref. 43. However, we did not implement this constraint to predict the crystal structures for the hard faceted particles as the lattice-reduction procedure efficiently removes any unwanted box deformations. Moreover, an angular constraint might bias the simulation for particles that have a highly anisotropic shape.

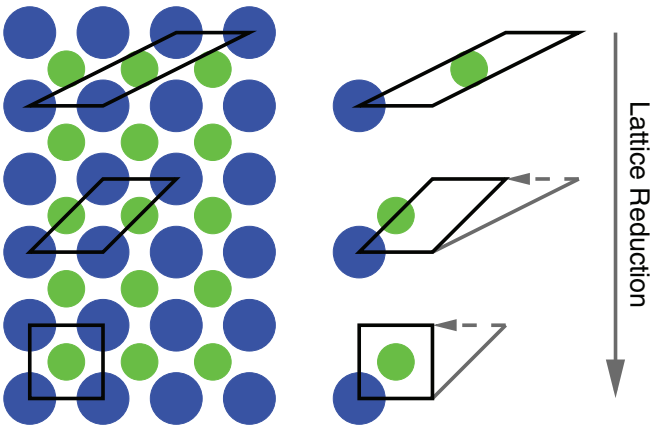


FIG. 4. Illustration of the lattice-reduction procedure. A 2D binary crystal of disks is shown, as well as three possible unit cells for this crystal. From top to bottom the lattice vectors of the unit cells become more orthogonal and shorter. The lattice vector that is modified is shown in grey. The one by which it is reduced, is indicated using a dashed grey arrow. The lattice-reduced unit cell has a square geometry.

VII. SOFT INTERACTIONS AND EXTERNAL FIELDS

Hard-particle FBMC simulations, for which a suitable overlap routine is chosen, can be performed using the above combination of acceptance criteria, image-list generation, and lattice reduction. Introducing soft interactions into the system requires extending the above approach as follows. The method of truncation and tail correction described in Ref. 53 is employed to determine the contribution to the system's total energy for short-range and semi-long-range interaction potentials. That is, soft interactions which decay faster than r^{-3}, with r the inter-particle distance. Here we assume a radially symmetric cut-off distance of R_C, which allows us to construct similar image lists as for the hard-particle interaction. A version of the FBMC method was also successfully applied to systems with long-range interaction potentials,[43] for which Ewald Sums[53,85] are required to compute the energy contribution. Interaction with an external field can also be easily introduced, since such a term only couples to the particles in the box in the expression for the total energy. It depends on the specifics of the system whether it is advantageous to first consider acceptance of the scaling and deformation moves on the hard-particle part of acc($o \rightarrow n$) or on the soft-interaction part. Of course, additional constraints on the box vectors will have to be introduced, because the external field is either periodic in the direction(s) the field varies, or the system has a fixed, finite extent in one or more directions.

VIII. COMPRESSION AND CRYSTAL STRUCTURE PREDICTION

In order to predict crystal structures, we apply the following procedure to a wide variety of systems including systems with various types of interactions, i.e., hard particles, anisotropic particles, and particles which interact via long-range potentials. See Sec. IX and Refs. 45 and 51 for some examples. Systems are prepared in a dilute phase by inserting the particles randomly while avoiding overlaps until the desired particle number is achieved. Next, the system is equilibrated at a low pressure. What constitutes a "low pressure" is strongly dependent on the system, but can easily be ascertained by a few trial runs. In order to obtain the pressure P of interest, the initial pressure is increased over a number of Monte Carlo cycles, where one cycle is understood to be one (translation, rotation, volume) trial move per particle. A slow increase of the pressure, for instance, according to a geometric series of pressure steps, can be applied to allow the system to sample more ways in which to "crystallize," in the hope that it chooses the optimal structure. If a slow increase of the pressure is used, we typically find that the FBMC simulation only has to be performed a small number of times in order to

obtain a good understanding of the possible crystal structures that can be found at the pressure of interest, say 25–100 compression runs. It is also possible to compress the system by rapidly increasing the pressure. In this case, effective sampling of crystal structures usually necessitates a larger set of compression runs than for the slower pressure increase, since there is less time for the system to explore different configurations before it becomes jammed in a (local) minimum of the crystal-structure free energy. Both slow and fast compression runs should be followed by a period of equilibration at the desired pressure to allow the system to settle in its (locally) optimal structure. When the fluctuations in the system are relatively large, it can be useful to follow this equilibration with a production run during which, for instance, the lattice vectors, particle positions, and particle orientations are averaged to obtain a representative crystal-structure candidate.

Increasing and decreasing the pressure several times around the P value of interest can prove useful in helping the system to cross free-energy barriers, thereby improving the chance of finding the global minimum in the free energy. Allowing the MC step size to adjust to predetermined acceptance ratios during the compression and equilibration part of the simulation also improves the results of the FBMC algorithm. We typically employ an acceptance ratio of 25% for translations and rotations, and a ratio of 10% for scaling and deformation moves. The set of crystal-structure candidates, that is obtained by slow and/or fast compression, can be analysed using a combination of software packages[86] and manually going through crystal-structure databases.[87,88] For a binary mixture of hard spheres this approach allowed the crystal structures of the candidates to be determined.[43] However, establishing the structure of a numerically obtained result is highly nontrivial in general. Moreover, a description of a system by the atomic equivalent of the lattice on which the particles' centres of rotation reside,[43,68,69,88] is not always adequate for nonspherical particles, because of the strong relevance of the orientation of the particles.[70] Finally, we note that the frequency with which structures are observed in the FBMC runs can give some insight into which of the structures we find is stable,[43] but only free-energy calculations can give a definitive answer.

IX. CLOSE-PACKED CRYSTAL STRUCTURES FOR ANISOTROPIC PARTICLES

To demonstrate the strength of the FBMC method in combination with a triangular-tesselation-based overlap routine, we used our algorithm to determine the densest packing crystal structures for 142 convex faceted particles and 17 nonconvex faceted shapes. In particular, we studied the close packing of the 5 Platonic solids, the 13 Archimedean solids, the 13 Catalan solids, the 92 Johnson solids, 7 regular prisms, 7 regular antiprisms, 11 nonconvex polyhedra, and 11 (non)convex miscellaneous shapes. The miscellaneous shapes included 4 nonconvex colloidal models: a cap, nanostar, octapod, and tetrapod. In Ref. 45, we presented and analysed the highest packing fraction we found for each of these particles. Here, we present the complete unit cells associated with the closed-packed crystal structures that we obtained along with the packing data. This includes the particle models we studied and configurations files required to – and detailed instructions on how to – construct the densest packing structures that we obtained. Figure 5 shows an example of what is to be found in the supplementary material.[52]

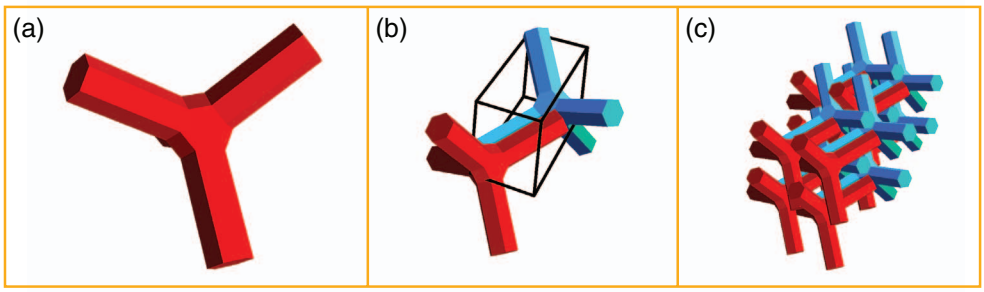
The best packings we found for the 5 Platonic solids and 13 Archimedean solids agreed excellently with literature,[48,63,65,89,90] yielding results within 0.002 of the literature value. For the truncated tetrahedra, we discovered a new crystal structure which improved upon the literature value of the densest packing. Subsequent numerical and analytic studies[68,91] showed that this value could be further improved upon.

For many of the other structures we studied, no previous best packings were known and so the crystal structures presented in the supplementary material[52] represent the first candidate densest packings for these shapes to date. In a recent publication, Damasceno *et al.*[69] examined the assembly of many of the same (convex) particle shapes from a dense fluid using Monte Carlo simulations of approximately 2000 particles. A large portion of the particle shapes they studied did not crystallize, particularly the Johnson solids. Others formed liquid crystal and plastic crystal phases. A more complete description of the equilibrium phase behaviour of these particles could be constructed by combining the phases they predict (crystalline, liquid crystal, or plastic crystalline) together with our best packed results. Our results can be used to determine the crystal branch(es) for the equation of state to a particular polyhedron, which in turn is required to calculate the free energy of these phases, and drawing the resulting phase diagram.

We note that the FBMC algorithm can only be used to establish a lower bound to the packing fraction of the densest configuration of shape-anisotropic (nonconvex) objects. Currently, several other techniques exist to estimate this lower bound numerically.[45,63,65,68,71,89,92,93] The FBMC method is similar to the adaptive-shrinking-cell (ASC) method of Refs. 63 and 89, since both allow for a sequential search of configurational space and lattice space using a Metropolis based MC procedure. However, FBMC uses a lattice-reduction technique[4] to avoid unwanted distortions of the unit cell, unlike the ASC algorithm. In addition, FBMC drives the systems towards its densest configuration by employing a (gradual) pressure increase according to a *NPT*-MC simulation, whereas ASC drives compression using the negative packing fraction as the basis of its Metropolis acceptance rule. The method used in Refs. 65 and 68, likely amounts to a different implementation of the principles that underlie the FBMC algorithm, whereas the techniques of Refs. 92–94 constitute a completely different means of determining densest packings. The method of Refs. 92 and 93 is based on a divide-and-conquer approach to achieve simultaneous satisfaction of multiple constraint equations. The approach of Ref. 94 is based on a penalty technique. It goes beyond the scope of this paper to determine which of these techniques is most suited to achieve densest-packed structures.

Of the aforementioned crystal-structure prediction algorithms FBMC has the advantage that it can be used to explore suboptimal packings in accordance with the statistical

PA04: Tetrapod



Model Parameters (d)

Model file:	Tetrapod.ply
Sphericity:	0.20303
Centrosymmetric (C/NC):	NC
Upper bound to the maximum packing fraction (MPF):	1.00000
Outscribed-sphere lower bound to the MPF:	0.04864
Oriented-bounding-box lower bound to the MPF:	0.10628

Unit Cell Parameters (e)

Number of particles:	2
Maximum packing fraction (MPF):	0.59207
Particles form a centrosymmetric compound (y/n/-):	y
Lattice Vectors:	$\mathbf{v1} = \{1.200440, -0.000000, 0.000000\}$
	$\mathbf{v2} = \{-0.336253, 2.503690, -0.000000\}$
	$\mathbf{v3} = \{-0.138403, 0.486549, 1.123920\}$

Particles in unit cell: (f)

Index	x	y	z	Rotation Matrix
1	0.564118	1.961490	0.292156	$\begin{pmatrix} 0.287425 & -0.949803 & 0.123536 \\ 0.912837 & 0.232588 & -0.335607 \\ 0.290027 & 0.209230 & 0.933867 \end{pmatrix}$
2	0.123058	0.297707	0.025395	$\begin{pmatrix} -0.290561 & 0.948728 & -0.124458 \\ -0.289587 & -0.211160 & -0.933569 \\ -0.911983 & -0.235217 & 0.336094 \end{pmatrix}$

FIG. 5. Close-packed result for a tetrapod: (a) model of a tetrapod, (b) unit cell, (c) eight replicas of the unit cell, which demonstrate what the crystal looks like, (d) model parameters including the name of the model .ply file, the sphericity of the model, the centrosymmetry of the model, as well as the associated upper and lower bounds to the packing fraction, (e) unit cell parameters including the number of particles in the unit cell, the maximum packing fraction that this lattice achieves, the formation of a centrosymmetric compound, and the vectors which describe the unit cell, (f) the basis particles, their location with respect to the origin and rotation with respect to the orientation in the .ply file. Further information on the various parameters and a page like this for each particle we studied can be found in the supplemental material.[52]

NPT-ensemble at finite pressures. This is particularly relevant, since densest-packed candidate crystal structure need not be thermodynamically stable at all pressures for which the system crystallizes.[28,51,66,67,95] It is important to realize that there are strong finite-size effects for the prediction of candidates at finite pressure. The pressure P at which we perform the FBMC simulations only sets a range from which we sample crystal structures. Large fluctuations in PV/k_BT, where P is the pressure, V is the volume, k_B is Boltzmann's constant, and T is the temperature, are to be expected in general due to the small number of particles considered. The advantage of these density fluctuations is that they allow the system to sample many different states at a single pressure. However, the presence of large fluctuations likely prevents phases with a small pressure range of stability to be discovered using our method. Some of these phases, in particular liquid crystals, may be found by melting, or by crystallizing a fluid.[69] Due to the level of fluctuations that is expected, the FBMC algorithm allows some flexibility in the precision to which soft-interaction terms are taken into account. For example, it is possible to ignore small cut-off corrections without incurring a large error, since this type of correction can be absorbed as a (small) pressure change in the acceptance rules. In this sense the FBMC technique is quite robust, but by the same token it is not *a priori* capable of attaining a high degree of accuracy at finite pressures.

Finally, we would like to comment on the possible use of the overlap algorithm described in Secs. II–VIII in analytic studies. Our methods have thus far been applied in a numerical context. However, using a potentially densest crystal structure that was obtained by the FBMC algorithm, a set of constrained (analytic) equations can be set up that

specify the overlaps between particles and possible ways in which they are allowed to move. The image-list and triangular-tessellation approach can be employed to minimize the number of equations that have to be simultaneously solved. This set of equations can be subsequently subjected to constrained optimization around the numerically predicted structure to determine the analytic densest packing for that particular dense crystal. As previously mentioned, similar approaches have been applied for tetrahedra[65,90] and truncated tetrahedra.[68,91] However, it is also possible that the numerical results are of sufficient quality to "guess" the analytic solutions which maximizes the density and verify that this structure does not give overlaps using an analytic form of our overlap techniques. We used this approach for rhombicuboctahedra and rhombic enneacontrahedra in Ref. 45 to *prove* that we obtained the *densest* structure for these particles.

X. SUMMARY AND OUTLOOK

Summarizing, we described in detail the way in which a FBMC simulation,[43] which allows for the prediction of crystal-structure candidates at finite pressure, can be set up. We also discussed two types of overlap algorithm, the method of separating axes and a triangular-tessellation based technique, by which hard-particle simulations for (nonconvex) faceted objects can be performed. When these overlap routines are combined with the FBMC algorithm, a powerful simulation technique is obtained with many applications to colloid research, as well as mathematical problems of a more fundamental nature.

The ability of the technique to predict the structure of low density phases, such as plastic-crystal (rotator) phases, has to be investigated further in the near future. Moreover, the analytic applications of the overlap algorithm described in this paper require further study. It is our hope that our method may eventually be used to prove some of the conjectures that have been proposed for the properties of the dense packings of polyhedra, e.g., Ulam's conjecture[96] and the conjectures proposed by Torquato and Jiao in their recent paper.[97]

ACKNOWLEDGMENTS

We would like to thank Tristan Hartskeerl, who introduced J.d.G. to the polygon-based overlap detection and with whom J.d.G. collaborated to perform an initial study of the behaviour of highly shape-anisotropic particles. We would also like to extend our gratitude towards Dr. F. Smallenburg for his help in setting up the supplementary material.[52] M.D. acknowledges financial support by a "Nederlandse Organisatie voor Wetenschappelijk Onderzoek" (NWO) Vici Grant, and R.v.R. by the Utrecht University High Potential Programme.

[1] S. M. Woodley and R. Catlow, Nature Mater. **7**, 937 (2008).
[2] J. Pannetier, J. Bassas-Alsina, J. Rodriguez-Carvajal, and V. Caignaert, Nature (London) **346**, 343 (1990).
[3] H. R. Karfunkel and R. J. Gdanitz, J. Comput. Chem. **13**, 1171 (1992).
[4] D. Gottwald, G. Kahl, and C. N. Likos, J. Chem. Phys. **122**, 204503 (2005).
[5] A. R. Oganov and C. W. Glass, J. Chem. Phys. **124**, 244704 (2006).
[6] D. J. Wales and H. A. Scheraga, Science **285**, 1368 (1999).
[7] J. C. Schon, Z. P. CanCarevic, A. Hannemann, and M. Jansen, J. Chem. Phys. **128**, 194712 (2008).
[8] R. Martoňák, A. Laio, and M. Parrinello, Phys. Rev. Lett. **90**, 075503 (2003).
[9] A. P. Malanoski and P. A. Monson, J. Chem. Phys. **110**, 664 (1999).
[10] S. C. Glotzer and M. J. Solomon, Nature Mater. **6**, 557 (2007).
[11] S. Sacanna and D. J. Pine, Curr. Opin. Colloid Interface Sci. **16**, 96 (2011).
[12] Y. Sun and Y. Xia, Science **298**, 2176 (2002).
[13] X. Zhang, C. Dong, J. Zapien, S. Ismathullakhan, Z. Kang, J. Jie, X. Zhang, J. Chang, C.-S. Lee, and S.-T. Lee, Angew. Chem., Int. Ed. **48**, 9121 (2009).
[14] H.-L. Wu, C.-H. Kuo, and M. H. Huang, Langmuir **26**, 12307 (2010).
[15] E. Matijevic, Acc. Chem. Res. **14**, 22 (1981).
[16] A. Tao, P. Sinsermsuksakul, and P. Yang, Angew. Chem. Int. Ed. **45**, 4597 (2006).
[17] A. R. Tao, S. Habas, and P. Yang, Small **4**, 310 (2008).
[18] A. Mews, A. V. Kadavanich, U. Banin, and A. P. Alivisatos, Phys. Rev. B **53**, R13242 (1996).
[19] F. Kim, S. Connor, H. Song, T. Kuykendall, and P. Yang, Angew. Chem. **116**, 3759 (2004).
[20] E. C. Greyson, J. E. Barton, and T. W. Odom, Small **2**, 368 (2006).
[21] B. Wiley, T. Herricks, Y. Sun, and Y. Xia, Nano Lett. **4**, 1733 (2004).
[22] A. S. Barnard, X. M. Lin, and L. A. Curtiss, J. Phys. Chem. B **109**, 24465 (2005).
[23] J. Tang, X. Zhou, D. Zhao, G. Q. Lu, J. Zou, and C. Yu, J. Am. Chem. Soc. **129**, 9044 (2007).
[24] K. Kwon, K. Y. Lee, Y. W. Lee, M. Kim, J. Heo, S. W. Ahn, and S. J. Han, J. Phys. Chem. C **111**, 1161 (2007).
[25] H. R. Vutukuri, "Complex colloidal structures by self-assembly in electric fields," Ph.D. dissertation (Utrecht University, 2012).
[26] S. Deka, D. Miszta, K. Dorfs, A. Genovese, G. Bertoni, and L. Manna, Nano Lett. **10**, 3770 (2010).
[27] R. Brescia, K. Miszta, D. Dorfs, L. Manna, and G. Bertoni, J. Phys. Chem. C **115**, 20128 (2011).
[28] K. Miszta, J. de Graaf, G. Bertoni, D. Dorfs, R. Brescia, S. Marras, L. Ceseracciu, R. Cingolani, R. van Roij, M. Dijkstra, and L. Manna, Nature Mater. **10**, 872 (2011).
[29] C. J. DeSantis, A. A. Peverly, D. G. Peters, and S. E. Skrabalak, Nano Lett. **11**, 2164 (2011).
[30] C. J. DeSantis and S. E. Skrabalak, Langmuir **28**, 9055 (2012).
[31] L. Manna, D. J. Milliron, A. Meisel, E. C. Scher, and A. P. Alivisatos, Nature Mater. **2**, 382 (2003).
[32] M. C. Newton and P. A. Warburton, Mater. Today **10**, 1369 (2007).
[33] G. Zhou, M. Lü, Z. Xiu, S. Wang, H. Zhang, Y. Zhou, and S. Wang, J. Phys. Chem. B **110**, 6543 (2006).
[34] N. Zhao and L. M. Qi, Adv. Mater. **18**, 359 (2006).
[35] T. Huang, Q. A. Zhao, J. Y. Xiao, and L. M. Qi, ACS Nano **4**, 4707 (2010).
[36] C. I. Zoldesi and A. Imhof, Adv. Mater. **17**, 924 (2005).
[37] D. Jagadeesan, U. Mansoori, P. Mandal, A. Sundaresan, and M. Eswaramoorthy, Angew. Chem. **47**, 7685 (2008).
[38] C. Quilliet, C. Zoldesi, C. Riera, A. van Blaaderen, and A. Imhof, Eur. Phys. J. E **27**, 13 (2008).
[39] R. Piazza, T. Bellini, and V. Degiorgio, Phys. Rev. Lett. **71**, 4267 (1993).
[40] Z. Cheng, J. Zhu, P. M. Chaikin, S.-E. Phan, and W. B. Russel, Phys. Rev. E **65**, 041405 (2002).
[41] G. Bryant, S. R. Williams, L. Qian, I. K. Snook, E. Perez, and F. Pincet, Phys. Rev. E **66**, 060501 (2002).
[42] W. C. K. Poon, E. R. Weeks, and C. P. Royall, Soft Matter **8**, 21 (2012).
[43] L. Filion, M. Marechal, B. van Oorschot, D. Pelt, F. Smallenburg, and M. Dijkstra, Phys. Rev. Lett. **103**, 188302 (2009).
[44] E. Bianchi, G. Doppelbauer, L. Filion, M. Dijkstra, and G. Kahl, J. Chem. Phys **136**, 214102 (2012).
[45] J. de Graaf, R. van Roij, and M. Dijkstra, Phys. Rev. Lett. **107**, 155501 (2011).
[46] G. Kuperberg and W. Kuperberg, Discrete Comput. Geom. **5**, 389 (1990).
[47] Y. Limon Duparcmeur, A. Gervois, and J. P. Troadec, J. Phys. I France **5**, 1539 (1995).
[48] U. Betke and M. Henk, Comput. Geom. **16**, 157 (2000).
[49] T. C. Hales and S. P. Ferguson, Discrete Comput. Geom. **36**, 1 (2006).
[50] T. C. Hales, Not. Am. Math. Soc. **55**, 1370 (2008).
[51] R. Ni, A. P. Gantapara, J. de Graaf, R. van Roij, and M. Dijkstra, Soft Matter **8**, 8826 (2012).
[52] See supplementary material at http://dx.doi.org/10.1063/1.4767529 for a database (REPOSITORY.zip) containing crystal structures for 159

(non)convex polyhedral particles and a document visualizing the associated unit cells (unit_cells.pdf).
[53]D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* 2nd ed. (Academic, London, 2002).
[54]N. Metropolis and S. Ulam, J. Am. Stat. Assoc. **44**, 335 (1949).
[55]F. J. Vesely, J. Comput. Phys. **47**, 291 (1982).
[56]R. Najafabadi and S. Yip, Scr. Mater. **17**, 1199 (1983).
[57]GAMMA Research Group at the University of North Carolina, RAPID – Robust and Accurate Polygon Interference Detection, 1997, see http://gamma.cs.unc.edu/OBB/.
[58]G. van den Bergen, FreeSolid, 2004, see http://www.win.tue.nl/~gino/solid/.
[59]GAMMA Research Group at the University of North Carolina, Collision detection algorithms, 2012, see http://gamma.cs.unc.edu/software/#collision.
[60]W. Liu, B. Schmidt, G. Voss, and W. Müller-Wittig, Comput. Phys. Commun. **179**, 634 (2008).
[61]T. Preis, P. Virnau, W. Paul, and J. J. Schneider, J. Comput. Phys. **228**, 4468 (2009).
[62]P. Lu, H. Oki, C. Frey, G. Chamitoff, L. Chiao, E. Fincke, C. Foale, S. Magnus, W. McArthur, D. Tani, P. Whitson, J. Williams, W. Meyer, R. Sicker, B. Au, M. Christiansen, A. Schofield, and D. Weitz, J. Real-Time Image Proc. **5**, 179 (2010).
[63]S. Torquato and Y. Jiao, Nature (London) **460**, 876 (2009).
[64]A. Haji-Akbari, M. Engel, A. S. Keys, X. Zheng, R. G. Petschek, P. Palffy-Muhoray, and S. C. Glotzer, Nature (London) **462**, 773 (2009).
[65]E. R. Chen, M. Engel, and S. C. Glotzer, Discrete Comput. Geom. **44**, 253 (2010).
[66]U. Agarwal and F. A. Escobedo, Nature Mater. **10**, 230 (2011).
[67]A. Haji-Akbari, M. Engel, and S. C. Glotzer, J. Chem. Phys. **135**, 194101 (2011).
[68]P. F. Damasceno, M. Engel, and S. C. Glotzer, ACS Nano **6**, 609 (2012).
[69]P. F. Damasceno, M. Engel, and S. C. Glotzer, Science **337**, 453 (2012).
[70]W. Qi, J. de Graaf, F. Qiao, S. Marras, L. Manna, and M. Dijkstra, Nano Lett. **12**, 5299 (2012).
[71]S. Atkinson, Y. Jiao, and S. Torquato, Phys. Rev. E **86**, 031302 (2012).
[72]A. Ostaszewski, *Advanced Mathematical Methods*, 1st ed. (Cambridge University Press, Cambridge, 1991).
[73]D. Eberly, Intersection of convex objects: The method of separating axes, 2008, see http://www.geometrictools.com/.
[74]F. Alonso-Marroquín and Y. Wang, Granular Matter **11**, 317 (2009).
[75]S. A. Galindo-Torres and D. M. Pedroso, Phys. Rev. E **81**, 061303 (2010).
[76]B. S. John and F. A. Escobedo, J. Phys. Chem. B **109**, 23008 (2005).
[77]T. D. Nguyen, E. Jankowski, and S. C. Glotzer, ACS Nano **5**, 8892 (2011).
[78]C. Avendano and F. A. Escobedo, Soft Matter **8**, 4675 (2012).
[79]Y. Jiao, F. H. Stillinger, and S. Torquato, Phys. Rev. Lett. **100**, 245504 (2008).
[80]M. Marechal, R. J. Kortschot, A. F. Demirörs, A. Imhof, and M. Dijkstra, Nano Lett. **10**, 1907 (2010).
[81]C. De Michele, e-print arXiv:0903.1608v2.
[82]S. Gottschalk, M. C. Lin, and D. Manocha, "OBBTree: A hierarchical structure for rapid interference detection," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, *SIGGRAPH '96* (Association for Computing Machinery, New York, 1996), p. 171.
[83]J. de Graaf, M. Dijkstra, and R. van Roij, Phys. Rev. E **80**, 051405 (2009).
[84]J. de Graaf, M. Dijkstra, and R. van Roij, J. Chem. Phys. **132**, 164902 (2010).
[85]P. Ewald, Ann. Phys. **369**, 253 (1921).
[86]H. T. Stokes, D. M. Hatch, and B. J. Campbell, FindSym software package, 2007, see http://stokes.byu.edu/isotropy.html.
[87]T. Hahn, *International Tables for Crystallography: Volume A, Space-Group Symmetry*, 5th ed. (Springer, Berlin, 2002).
[88]Center for Computational Materials Science, Database of molecular prototypes corresponding to the 230 existing space groups, 2008, see http://cst-www.nrl.navy.mil/lattice/index.html.
[89]S. Torquato and Y. Jiao, Phys. Rev. E **80**, 041104 (2009).
[90]S. Torquato and Y. Jiao, Phys. Rev. E **81**, 041310 (2010).
[91]Y. Jiao and S. Torquato, J. Chem. Phys. **135**, 151101 (2011).
[92]S. Gravel and V. Elser, Phys. Rev. E **78**, 036706 (2008).
[93]Y. Kallus, V. Elser, and S. Gravel, Discrete Comput. Geom. **44**, 245 (2010).
[94]L. Assoud and R. Messina, Phys. Rev. E **83**, 036113 (2011).
[95]A.-P. Hynninen, L. Filion, and M. Dijkstra, J. Chem. Phys. **131**, 064902 (2009).
[96]M. Gardner, *The Colossal Book of Mathematics: Classic Puzzles, Paradoxes, and Problems*, 1st ed. (Norton, New York, 2001).
[97]S. Torquato and Y. Jiao, Phys. Rev. E **86**, 011102 (2012).