Analysis of energy landscapes of molecular crystal structures employing combinatorial and topological methodologies.

First Year Progress Report

Miloslav Torda

Supervisors:
Prof Yannis Goulermas
Dr Vitaliy Kurlin
Prof Graeme M. Day
Dr Linjiang Chen

Advisors:
Prof Leszek A. Gasieniec
Prof Rahul Savani

Department of Computer Science
University of Liverpool
2. 6. 2019

Contents

1 Motivation

The aim of this project is to aid crystal structure prediction (CSP) computations using only geometry. Present day CSP relies on finding structures with low energies, as those are the ones that are most likely to be stable when synthesized. The search for possible crystal structures involves minimizing energy potentials subject to periodic boundary conditions. The energy formulae vary from simple and fast potential approximations, for example force field methods, to precise but computationally demanding density functional theory. The optimization itself is performed usually with Monte Carlo simulation methods with the help of some theoretical knowledge about a molecule, for example possible space group symmetry configurations that a particular molecule can take. In practice this involves searching for minima on a complex energy landscape. For example in case of Lennard–Jones potential a system of 13 particles has approximately 1509 local minima, 28756 stationary points of index 1 and 88079 stationary points of index 2 [1].

Dense packings tend to have lower energies [2], thus instead of looking for low energy structures we can work with a geometrical representation of a molecule and find dense packed configurations. These dense packed configurations can be then used as a starting position in the usual CSP procedures as opposed to a completely random starting configuration. The estimate is that finding geometrically dense crystals as starting positions could accelerate CSP by two orders of magnitude.

2 Theoretical background

2.1 History of the packing problem

The packing problem is well studied and generally considered to be a hard problem. Stated as David Hilbert's 18th problem:

> *"How can one arrange most densely in space an infinite number of equal solids of given form, e.g. spheres with given radii or regular tetrahedra with given edges (or in prescribed position), that is,how can one so fit them together that the ratio of the filled to the unfilled space may be as great as possible?" [3]*

The first mention of this problem is from 1611 by Johannes Kepler who conjectured that the densest packing of congruent balls is that of a face–centered–cubic latice. This conjecture was proved by Thomas C. Hales [4] in 2005 by the proof of exhaustion.

Since Kepler, the packing problem has been studied extensively. It does seem however that the 18th Hilbert problem is actually a collection of problems. An overview of results to current date can be found in review works of Tóth [5] or Bezdek and Kuperberg [6].

2.2 Problem statement

For the purpose of stating the problem of our interest we will outline the concept of packing and it's two different simplifications, namely lattice packing and periodic packing and their respective densities. We will follow Rogers [7].

Definition. *The system of sets* $S_1,\ S_2, \ldots$ *is said to cover the set* S*, if*

$$S_i \cap S_j = \emptyset \ (i \neq j)$$
$$\bigcup_i S_i \subset S$$

i.e if no two of the sets $S_1,\ S_2, \ldots$ *have any element in common and each element of the sets* $S_1,\ S_2, \ldots$ *belongs to* S*.*

We will identify the set S with the Euclidean space of dimension n i.e. $S = \mathbf{E}^n$. In our specific setting we are interested only in $n = 3$. $S_1,\ S_2, \ldots$ will be a finite collection or countably infinite of translates of rotations of a single set K. By translates of K we mean set of all points $k + a$, where $k \in K$, and a is a fixed point or vector and by rotations we mean set of all points Rk for $k \in K$ and $R \in SO(n) = \{A \in \mathbb{R}^{n\times n} : A^T A = I,\ \det A = 1\}$.

Now we can define packing density. Let $\{a_i\}$ be a sequence of points, $\{R_j : R_j \in SO(n)\}$ a collection of rotations, K a set with finite volume $\mathtt{vol}(K)$, C a cube with the edge length c and system of sets $\mathcal{K} = \{R_j K + a_i\}$ that forms a packing.

Definition. *The density* $\rho(\mathcal{K})$ *of the packing* $\mathcal{K}$ *is*

$$\rho(\mathcal{K}) = \limsup_{c\to\infty} \rho(\mathcal{K}, C)$$

where

$$\rho(\mathcal{K}, C) = \frac{1}{\mathtt{vol}(C)} \sum_{(R_j K + a_i)\cap C \neq \emptyset} \mathtt{vol}(R_j K + a_i).$$

It can be easily proved that $0 \leq \rho(\mathcal{K}) \leq 1$.

Arising from the crystallographic applications we will be working only with subset of packings, specifically lattice packings. Let $\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_n}$ be n linearly independent vectors in $\mathbf{E}^n$. The set $\Lambda = \{u_1\mathbf{a_1} + u_2\mathbf{a_2} + \cdots + u_n\mathbf{a_n} \mid u_i \in \mathbb{Z}\}$ is called a lattice.

Definition. *A lattice packing* $\mathcal{K}_L$ *is a system of translates of a given set* K *i.e.*

$$\mathcal{K}_L = \{K + \mathbf{a_i} \mid \mathbf{a_i} \in \Lambda\}$$

if it is a packing into the whole space.

It can be shown that in the case of lattice packing $\mathcal{K}_L$ of a set K the density takes the following form

$$\rho(\mathcal{K}_L) = \frac{\mathtt{vol}(K)}{|\det(\Lambda)|}$$

where $\det(\Lambda)$ is the determinant of the generators of the lattice Λ.

Finally we come to the concept of a periodic packing which is going to be our main area of interest. Let $\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_N}$ be a set of points and let $\mathbf{b_1}, \mathbf{b_2}, \ldots \in \Lambda$ and $R_1, R_2, \ldots, R_N \in SO(n)$.

Definition. *A periodic packing $\mathcal{K}_P$ is a system of translates and rotations of a given set K, that is*

$$\mathcal{K}_P = \{R_i K + \mathbf{a_i} + \mathbf{b_j} \mid i = 1, 2, \ldots, N;\ j = 1, 2, \ldots\}$$

if it is a packing into the whole space.

Similarly to the case of a lattice packing, it can be shown that periodic packing density can be expressed in the following form

$$\rho(\mathcal{K}_P) = \frac{N\mathtt{vol}(K)}{|\det(\Lambda)|} \tag{1}$$

where as previously $\det(\Lambda)$ denotes the determinant of the generators of the lattice Λ.

Given the definitions above, the problem of our interest is called periodic packing problem. Precisely find a periodic packing $\mathcal{K}_{\max}$ of N congruent copies of a given set K such that

$$\mathcal{K}_{\max} = \operatorname*{argmax}_{\mathcal{K}_P} \rho(\mathcal{K}_P).$$

3 Preliminary results

In this section we present two approaches we have already taken in an attempt to solve the periodic packing problem including modelling of molecules. One is a Monte-Carlo molecular dynamics simulation and the second one is a mathematical programming formulation.

3.1 Monte-Carlo molecular dynamics simulations

The first idea was to try a simulated annealing method to find the densest periodic packings. This approach is well explored, for example by Torquato and Jiao [8] or by Damasceno et. al. [9]. Y. Jiao was was kind enough and provided us with a working version of the algorithm from [8] written in C++. Based on the Torquato–Jiao algorithm we implemented our version in Julia programming language [10].

3.1.1 Torquato-Jiao packing algorithm

The input for the algorithm is a representation of the polytop we want to pack by its vertices, edges and faces. Then number of copies of the shape we want to pack N, and initial configuration of the system. The initial configuration is given by the unit cell basis vectors, N coordinates of the centres of the polytops to be packed in the standard basis, and coordinates of the vertices of the N shapes centred and rotated around the origin. The output is in the form similar to the initial configuration input. That is the unit cell basis vectors, N coordinates of the centres of the polytops in the standard basis, and coordinates of the vertices of the N polytops centred and rotated around the origin.

The algorithm implements periodic boundary conditions [11]. The basic idea of the algorithm is at every iteration to take every polytop in the unit cell and attempt to either rotate it or to move it, and then attempt to either shrink or expand the unit cell. The only requirement is that the centres of the polytops are contained in the unit cell and that the polytops do not overlap. From the expression of the periodic packing density (Equation (1)) in can be seen that shrinking of the unit cell is the only thing that actually changes the density and we are in practice trying to find the unit cell with the minimal determinant.

The rotations and motions are performed randomly. We randomly choose if a particle gets rotated or moved and then perform either a rotation around a randomly chosen axis by a small random angle θ, or a motion by a small random displacement Δ. As mentioned before the algorithm allows the unit cell to expand thus allowing the hill climbing in the optimisation procedure. The hill climbing is governed by a simulated annealing schedule i.e

$$p(\mathcal{K}_t \leftarrow \mathcal{K}) = \begin{cases} 1 & \text{if } \Delta\rho > 0 \\ e^{\frac{-\Delta\rho}{T_t}} & \text{if } \Delta\rho \leq 0 \end{cases}$$

where $\Delta\rho = \rho(\mathcal{K}) - \rho(\mathcal{K}_t)$. This simulated annealing procedure was not part of the original Torquato-Jiao algorithm and was added by us.

The Torquato-Jiao algorithm is built for simulations of large number of ($N > 1000$) particles and to speed up overlap checking the authors implement near-neighbour lists, where the unit cell is divided into smaller parts and a for every particle a list of the nearest neighbours is stored in the memory. To check for overlap for a given polyhedron it's enough to check it's nearest neighbour. This has some disadvantages in smaller settings ($N < 10$), and because of this disadvantages we do not use near-neighbour lists in our version of the algorithm and at every step we do a complete overlap check including neighbouring unit cells.

The actual overlap check between two particles is done via separating axis theorem which is a consequence of hyperplane separating theorem by Minkowsky [12].

Theorem. *Two convex polyhedra do not intersect if and only if there exists a separating plane which is either parallel to a face of one polyhedron or which is parallel to at least one edge of each polyhedron.*

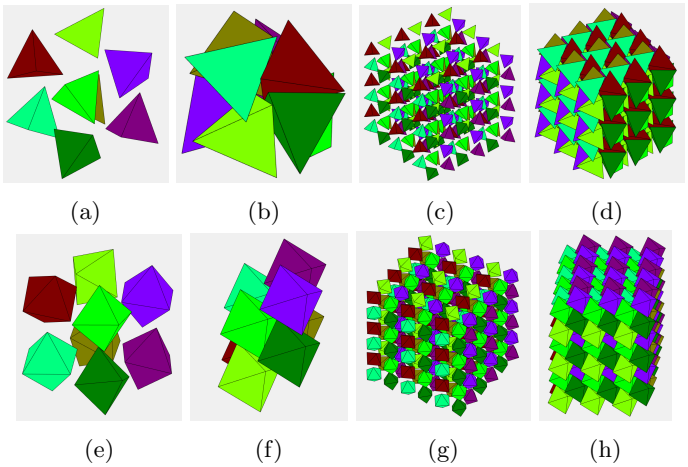


Figure 1: (a) Initial packing of 8 tetrahedra in a unit cell and (b) the output of packing algorithm. (c) As (a) but with 27 unit cells displayed and (d) the output of the packing displayed in 27 unit cells. (e) Initial packing of 8 octahedra in a unit cell and (f) the output of packing algorithm. (g) Initial configuration displayed in 27 unit cells and (h) the output packing displayed in 27 unit cells.

Axis normal to a separating plane is called a separating axis. If the projections of the polyhedra on to the separating axis do not intersect then the polyhedra do no overlap. The use of separating axis theorem limits the use of the algorithm only to convex polyhedra.

We tested our version of the Torquato–Jiao algorithm to pack regular tetrahedra and octahedra. The results from one of the experiments of packing 8 tetrahedra and 8 octahedra in a unit cell are presented in Figure 1. The initial density of the tetrahedra packing was 0.06415 and the output density was 0.67045. Currently the best achieved density of tetrahedra packing is 0.856347 [13] and the best upper bound is 2.6×10^{-24} below 1 [14].

In the case of octahedra the initial density was 0.1667 and the output density was 0.9178. For comparison, Minkowsky showed that the optimal lattice packing of octahedra is 0.9474 [6]. Torquato-Jiao conjecture that the general optimal packing density for centrally symmetric platonic solids is their respective lattice packing [8].

3.1.2 Pentacene modeling and packing

In the next step we moved to applying our packing algorithm to pack molecules of pentacene. One problem is that pentacene is a planar structure see Figure 2a.

To create a 3 dimensional shape around every atom of pentacene we put 14 points placed uniformly on a sphere with the radius 0.5573/2 (Figure 2b). By doing this we created a point cloud, we then computed a convex hull of this

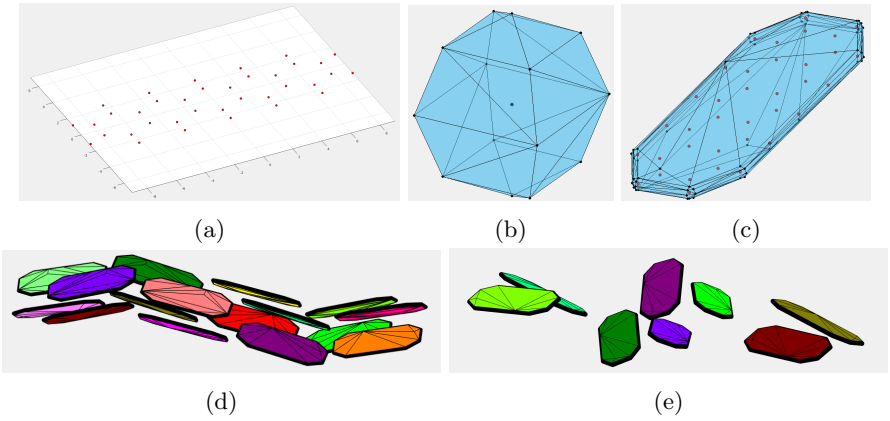


Figure 2: (a) Centers of atoms of pentacene. (b) 14 points placed on a sphere with radius 0.5573/2. (c) Convex hull of the resulting point cloud with centres of pentacene atoms inside (red).(d) A model of the pentacene crystal structure from the CSP dataset. (e) The crystal structure where the minimum distance between two pentacene molecules is attained.

point cloud and finally got a polyhedron defined by a triangulation with 58 vertices, 112 edges, 168 faces (Figure 2c). The volume of the polyhedron is 48.237. The radius of the sphere on which we put the points was computed from a crystal structure prediction dataset by Campbell et.al [15]. The dataset contained 586 pentacene structures. Out of these we have computed minimum euclidean distance between pentacene molecules within every crystal structure (0.5573). An examples of pentacene crystal structures from the mentioned dataset is presented in Figure 2d and the crystal structure with the minimum distance in Figure 2e.

We proceeded to pack the pentacene representation using the Torquato–Jiao algorithm. We packed 1,2,4,8 and 54 pentacene shapes subject to periodic boundary conditions with random initial configurations. The output of the algorithm is presented in Figure 3. In the case of 1 pentacene shape the output density was $\rho \approx 1$ (Figure 3a), in the case of 2 pentacene shapes $\rho \approx 1$ (Figure 3b), in the case of 4 pentacene shapes $\rho = 0.8093$ (Figure 3c), in the case of 8 pentacene shapes $\rho = 0.6940$ (Figure 3d) and in the case of 54 pentacene shapes $\rho = 0.1521$ (Figure 3e). We observe an inverse relationship between numbers of pentacene shapes to be packed end the output of the algorithm. The more shapes the more likely is for the algorithm to converge to a jammed configuration. This is probably due to the flat shape of the pentacene shape.

3.2 Mathematical programming formulation

As another approach we formulated the packing problem as a mathematical optimization problem. Based on Chernov et. al. [16] we built a mathematical

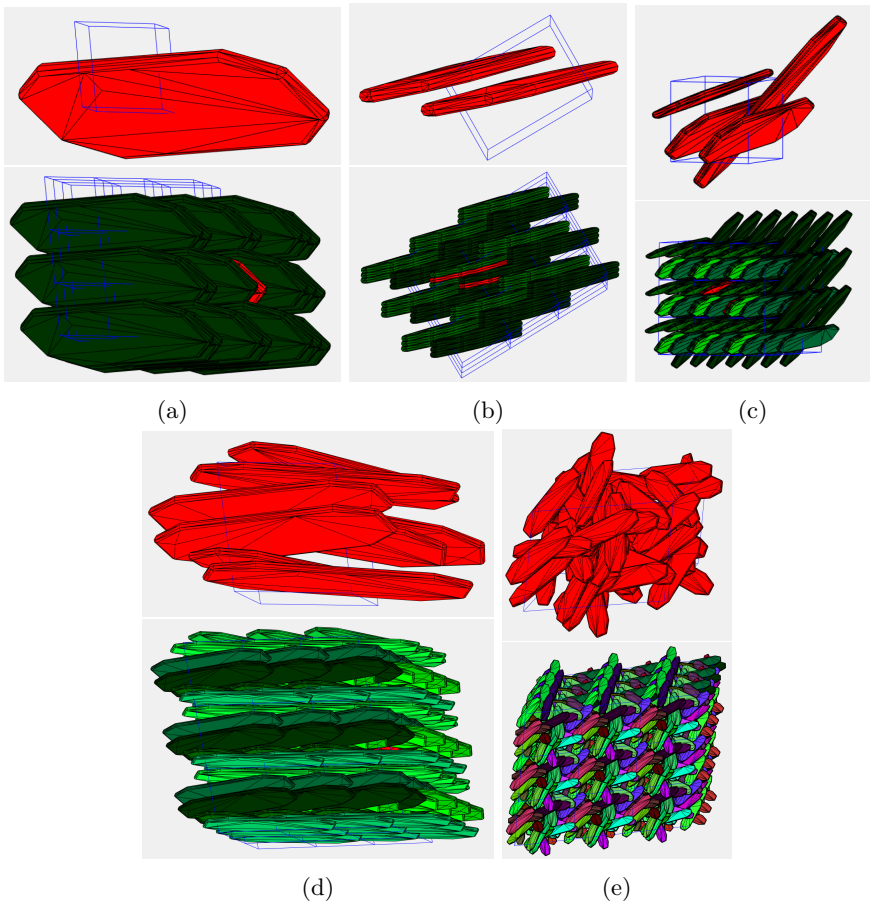


Figure 3: Output configurations of pentacene packing for (a) 1, (b) 2, (c) 4, (d) 8 and (e) 54 pentacenes displayed in a single unit cell and 27 unit cells.

model of periodic packing of N 2-simplices in a unit cell. The choice of packing a 2-simplex was for it's simplicity and easiness of testing the model. We now that the 2-simplex tiles $\mathbb{R}^2$, i.e. $\rho = 1$. The model can be easily extended to an euclidean space of any dimension and any convex polytop.

3.2.1 The model

Let $\Lambda = \{p_1u_1 + p_2u_2 \mid p_1, p_2 \in [0,1]; u_1, u_2 \in \mathbb{R}^2\}$ be a unit cell defined by the set of generators $\overline{\Lambda} = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}$ and $T_0 = \left\{\phi_1 \binom{1}{0} + \phi_2 \binom{0}{1} - \binom{\frac{1}{3}}{\frac{1}{3}} \mid 0 \leq \phi_1 + \phi_2 \leq 1\right\}$ be a 2-simplex centred at the origin. Next we define N copies of T_0 translated by $c_k \in \mathbb{R}^2$ and rotated by $R_k \in SO(2)$

$$T_k = \left\{ \overline{\Lambda} c_k + R_k p_0 \mid p_0 \in T_0 \right\}$$

where $c_k = \binom{x^k}{y^k}$ and $R_k = \begin{pmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{pmatrix}$ for $k = 1, 2, \ldots, N$ and $\theta_k \in [0, 2\pi]$. For the purpose of stating the constraints of the problem in the form of equations we need to define translates $T_k^{i,j}$ of T_k in the neighbouring unit cells

$$T_k^{i,j} = \left\{ \overline{\Lambda} \begin{pmatrix} i \\ j \end{pmatrix} + p_k \mid p_k \in T_k \right\}$$

for $i, j \in \{-2, -1, 0, 1, 2\}$.

The packing problem can be then stated as

$$\min \mathtt{vol}(\Lambda) = \min \det(\overline{\Lambda}) = \min u_{11}u_{22} - u_{12}u_{21} \tag{2}$$

as a function of 10 variables i.e $u_{11}, u_{22}, u_{12}, u_{21}, c_1, c_2, \theta_1, \theta_2$, subject to

$$c_i \in [0, 1] \times [0, 1];\ i = 1, 2, \ldots, N \tag{3}$$

$$\mathtt{Int}(T_k) \cap \mathtt{Int}(T_l^{i,j}) = \emptyset \tag{4}$$

where $\mathtt{Int}(\cdot)$ is the interior of a set, $k, l = 1, 2, \ldots, N$, $i, j = -2, -1, 0, 1, 2$, excluding the case when $l = k, i = 0, j = 0$. The objective function is in fact a homogeneous polynomial of degree 2. The constrains mean that only centres of the triangles have to be contained inside the unit cell and the triangles in the unit cell do not intersect each other and their copies in neighbouring unit cells.

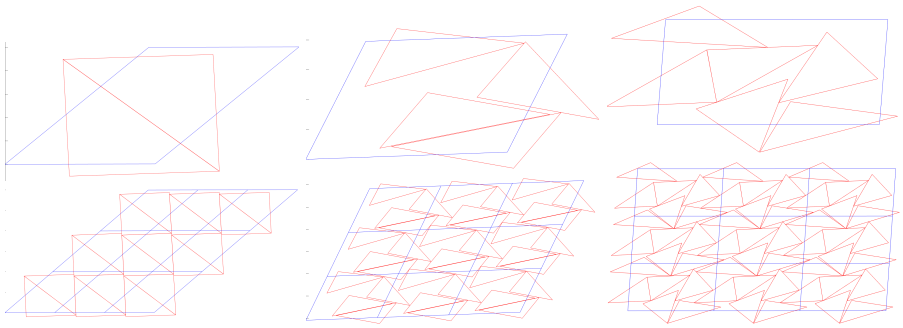
According to the hyperplane separation theorem stated before if two simplexes do no overlap then the separating hyperplane is defined by at least one of the edges of the simplex. Since all the simplexes are rotations and translations of T_0 the possible separating hyperplanes are defined by

$$\alpha_h x + \beta_h y + \gamma_h = 0 \tag{5}$$

for $h = 1, 2, 3$ where $\alpha_h, \beta_h, \gamma_h$ are defined by the edges of T_0.

For intersection check of two simplexes it's enough to check if the vertices of one simplex do not lie inside the other simplex. To formalize this, let $\binom{x_0}{y_0} \in \mathtt{Vert}(T_0)$ where $\mathtt{Vert}(T_0)$ is the set of vertices of T_0. Then the vertices of T_k are defined by $\binom{x_k^{0,0}}{y_k^{0,0}} = \overline{\Lambda} c_k + R_k \binom{x_0}{y_0}$ and vertices of $T_l^{i,j}$ by $\binom{x_l^{i,j}}{y_l^{i,j}} = \overline{\Lambda} \left(c_l + \binom{i}{j} \right) + R_l \binom{x_0}{y_0}$. Let us denote $\binom{\tilde{x}_k^{0,0}}{\tilde{y}_k^{0,0}}$ the vertices of T_k in the coordinate system of $T_l^{i,j}$, that is

$$\begin{pmatrix} \tilde{x}_k^{0,0} \\ \tilde{y}_k^{0,0} \end{pmatrix} = R_l^{-1} \left[\begin{pmatrix} x_k^{0,0} \\ y_k^{0,0} \end{pmatrix} - \overline{\Lambda} \left(c_l + \begin{pmatrix} i \\ j \end{pmatrix} \right) \right]$$



(a) (b) (c)

Figure 4: Output configurations of genetic algorithm for the packing of (a) 2, (b) 4, (c) 6 simplexes in a unit cell.

and $\binom{\tilde{x}_k^{i,j}}{\tilde{y}_k^{i,j}}$ the vertices of $T_l^{i,j}$ in the coordinate system of T_k, precisely

$$\begin{pmatrix} \tilde{x}_l^{i,j} \\ \tilde{y}_l^{i,j} \end{pmatrix} = R_k^{-1} \left[\begin{pmatrix} x_l^{i,j} \\ y_l^{i,j} \end{pmatrix} - \overline{\Lambda} c_k \right].$$

This way we can state an equivalent definition of the intersection constraint (4),

$$\begin{aligned} &\texttt{Int}(T_k) \cap \texttt{Int}(T_l^{i,j}) = \emptyset \\ &\qquad \Leftrightarrow \min_{i,j,k,l} \max\{ \max_{1 \le h \le 3} \min_{\binom{x_0}{y_0} \in \texttt{Vert}(T_0)} v_{h,k}^{0,0}, \max_{1 \le h \le 3} \min_{\binom{x_0}{y_0} \in \texttt{Vert}(T_0)} v_{h,l}^{i,j} \} \ge 0 \end{aligned}$$

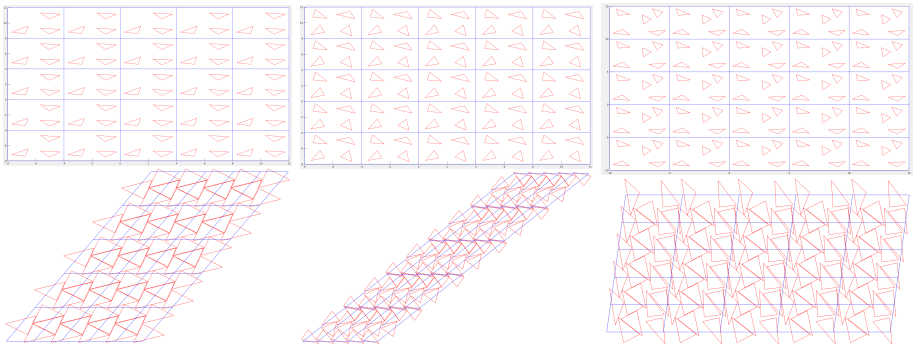
where

$$\begin{aligned} v_{h,k}^{0,0} &= \alpha_h \tilde{x}_k^{0,0} + \beta_h \tilde{y}_k^{0,0} + \gamma_h \\ v_{h,l}^{i,j} &= \alpha_h \tilde{x}_l^{i,j} + \beta_h \tilde{y}_l^{i,j} + \gamma_h \end{aligned}$$

and $\alpha_h, \beta_h, \gamma_h$ are the coefficients of the separating hyperplanes of T_0 defined in Eq. (5). As in the previous formulation of the constraint the case $l = k, i = 0, j = 0$ has to be excluded. The intersection constraint this way becomes a continuous and piecewise differentiable function of 6 variables. It's not difficult to translate this problem to a mixed integer nonlinear programming formulation by introducing artificial integer variables.

3.2.2 The experiments

We tested our simplex packing model first by using Matlab's Global optimization toolbox genetic algorithm and next by using Matlab's Optimization toolbox nonlinear constrained solver. Using the genetic algorithm we tested instances



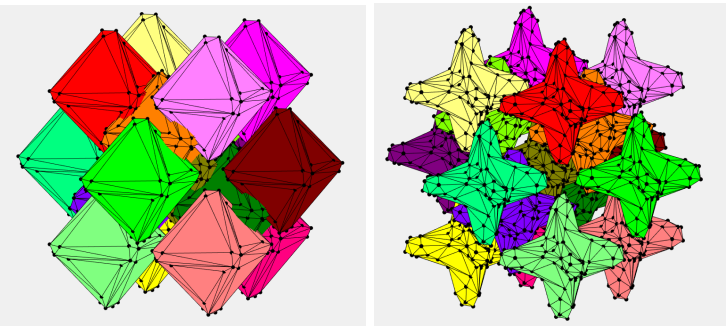
(a) (b) (c)

Figure 5: Best solutions found using nonlinear constrained optimization for (a) 3, (b) 4, (c) 4 simplexes in a unit cell. For each case initial (upper image) and output (lower image) configurations are displayed

with 2,4 and 6 simplexes in a unit cell. Results from the experiments are presented in Figure 4. The only case when the algorithm found an optimal solution was in the case of packing 2-simplexes (Figure 4a). In case of 4 simplexes the output density was 0.6715 (Figure 4b) and in the case of 6 simplexes 0.7534.

The experiments using nonlinear constrained optimization were performed for packing of 2,3,4,5 simplexes in a unit cell. Matlab's optimization toolbox nonlinear constrained solver uses interior point method to find optimal solutions. We provided the solver with the gradient of the objective function and since the intersection constraint is nondifferentiable we used the BFGS algorithm for Hessian approximation of the constraints that is already built in the nonlinear constrained optimization solver. For every setting we performed 100 runs with random initial rotations.

Results of the best solutions are presented in Figure 5. The setting with 2 simplexes generated although optimal solutions but with degenerate unit cells i.e $\det(\overline{\Lambda}) \approx 0$. Because of this we do not present this setting. In the case packing 3 simplexes, 39 runs exited with a feasible solution. Mean density in the feasible solution cases was 0.5827 and variance 0.0197. The best case had density 0.8118 (Figure 5a). In the 4 simplex setting out of 100 runs 20 resulted in feasible solutions. Mean density in the feasible solution cases was 0.5850 and variance 0.0127. The best case had density 0.7959 (Figure 5b). Only 9 out of 100 runs exited with a feasible solution in the 5 simplex case. Mean density in the feasible solution cases was 0.5637 and variance 0.0111. The best case had density 0.7246 (Figure 5c).



(a) (b)

Figure 6: (a) convex hull and (b) α-shape of an organic cage molecule crystal structure.

4 Future work

From the preliminary results presented, neither approach works particularly well. The Monte–Carlo algorithm seems works for simple polyhedra, for example octahedra, but struggles with more complicated shapes, as is the case of pentacene representations where the packing configurations are obvious. The experiments with the mathematical programming formulation of the packing problem (Equation (2)) suggest similar conclusions. In both, genetic algorithm and nonlinear constrained solver applications, the algorithms struggle even in a simple tasks of packing simplexes. More over in case of the nonlinear constrained optimization the solutions depend on the initial configuration, where adding more simplexes to the problem decreases the chance of even finding a feasible solution. These observations suggest a complicated energy landscape (if we use the language of energy minimization) with many local minima separated by high energy barriers. Because of this a more analytical approach needs to be taken, although it has been clear for a long time that the packing problem can not be solved entirely analytically and combination of analytical and computational methods need to be employed.

An alternative approach to periodic packing problem is to glue together shapes in the unit cell and this way to convert a periodic packing problem to a lattice packing problem. This idea was used for example by Chen et. al. [13] where the authors found the densest known packings of tetrahedra by gluing together two tetrahedra and creating tetrahedra dimers.

The gluing is going to be done by attaching congruent faces of the shapes. This way we encounter a combinatorial problem, since for example the pentacene representation has 168 faces (Figure 2c) it would be beneficial to trim the possible glued shapes that we need to find dense lattice packing for.

Second part of the gluing method is to find optimal lattice packings of glued shapes. By gluing together shapes we are very likely going to get noncovex shapes. This way we are presented with a problem of lattice packing of noncon-

vex sets. To our knowledge this area is not very well explored.

To find optimal packings of nonconvex sets is important for us also from another perspective. With the application of packing problems to crystal structure prediction we want to move beyond packing pentacene. With the use of α-shapes [17] we can create geometric representations of molecules, possibly nonconvex, that can be packed. An example is presented in Figure 6 where the convex hull (Figure 6a) and α-shape (Figure 6b) for some α of an organic cage molecule crystal by Jones et. al. [18] are displayed. The convex hulls of the molecules overlap but the α-shapes do not and so we need to work with α–shapes which are non-convex.

One of the useful theoretical challenges is to find a good upper bound for packings of nonconvex sets to guide us as to how close are we to an optimum packing. Another challenge that needs to be addressed concerns the packing of pentacene. We need to complicate the geometric model. Pentacene has four main packing types (see Campbell et. al. [15]) arising from intermolecular interactions. These interactions need to be integrated into the packing model for the model to be more plausible.

References

[1] D. Wales, Energy Landscapes: Applications to Clusters, Biomolecules and Glasses, Cambridge University Press, 2004.

[2] G. M. Day and A. I. Cooper, Energy–Structure–Function Maps: Cartography for Materials Discovery, Advanced Materials, vol. 30, pages 1704944, 2018.

[3] D. Hilbert, Mathematical problems, Bull. Amer. Math. Soc., vol. 8, pages 437-479, 1902.

[4] T. C. Hales, A proof of the Kepler conjecture, Annals of Mathematics, vol. 162, pages 1065-1185, 2005.

[5] G. F. Tóth, Packing and covering, Handbook of Discrete and Computational Geometry 3rd ed., CRC Press, 2017.

[6] A. Bezdek and W. Kuperberg, Dense Packing of Space with Various Convex Solids, Geometry — Intuitive, Discrete, and Convex: A Tribute to László Fejes Tóth", Springer Berlin Heidelberg, 2013.

[7] C.A. Rogers, Packing and Covering, Cambridge University Press, 1964.

[8] S. Torquato and Y. Jiao, Dense packings of polyhedra: Platonic and Archimedean solids, Phys. Rev. E, vol. 80, pages 041104, 2009.

[9] P. F. Damasceno, M. Engel, S. C. Glotzer, Predictive Self-Assembly of Polyhedra into Complex Structures, Science, vol. 337, pages 453-457, 2012.

[10] J. Bezanson, A. Edelman, S. Karpinski, V. B Shah, Julia: A fresh approach to numerical computing, SIAM review, vol. 59, pages 65–98, 2017.

[11] T. Schlick, Molecular Modeling and Simulation: An Interdisciplinary Guide, Springer: New York, 2002.

[12] E. Huber and W. Barth, Surface-to-surface intersection with complete and guaranteed results, DEVELOPMENTS IN RELIABLE COMPUTING Springer Science & Business Media, 2013.

[13] E. R. Chen, M. Engel, S. C. Glotzer, Dense Crystalline Dimer Packings of Regular Tetrahedra, Discrete & Computational Geometry, vol. 44, pages 253280, 2010.

[14] S. Gravel, V. Elser, Y. Kallus, Upper Bound on the Packing Density of Regular Tetrahedra and Octahedra, Discrete & Computational Geometry, vol. 46, pages 799818, 2010.

[15] J. E. Campbell, J. Yanga, G. M. Day, Predicted energy–structure–function maps for the evaluation of small molecule organic semiconductors, Journal of Materials Chemistry C, vol. 5, pages 7574-7584, 2017.

[16] N. Chernov, Yu. Stoyan, T.Romanova, Mathematical model and efficient algorithms for object packing problem, Computational Geometry, vol. 43, pages 535-553, 2010.

[17] H. Edelsbrunner, D. Kirkpatrick, R. Seidel, On the shape of a set of points in the plane, IEEE Transactions on Information Theory, vol. 29, pages 551 - 559, 1983.

[18] J. T. A. Jones, T. Hasell, X. Wu, J. Bacsa, K. E. Jelfs, M. Schmidtmann, S. Y. Chong, D. J. Adams, A. Trewin, F. Schiffman, F. Cora, B. Slater, A. Steiner, G. M. Day, A. I. Cooper, Modular and predictable assembly of porous organic molecular crystals, Nature, vol. 474, pages 367–371, 2011.