Barycentric Drawings of Periodic Graphs

Olaf Delgado-Friedrichs

University of Tübingen, WSI Computer Science, Sand 14, 72076 Tübingen, Germany

Abstract. We study barycentric placement of vertices in periodic graphs of dimension 2 or higher. Barycentric placements exist for every connected periodic graph, are unique up to affine transformations, and provide a versatile tool not only in drawing, but also in computation. Example applications include symmetric convex drawing in dimension 2 as well as determining topological types of crystals and computing their ideal symmetry groups.

1 Introduction

Periodic graphs provide convenient models for studying the topology of the interatomic structure of crystal species. Usually, atoms are represented by vertices, while bonds are represented by edges. Some crystals, however, are made from larger building blocks using linkers, in which case it is often more useful to model this meta-structure [OEL+00].

Apart from their relevance for chemistry, periodic graphs occur as the 1-skeleta of periodic tilings [DDH+99] and have applications in VLSI circuit design and integer programming [Orl84,CM91].

Although periodic graphs may be described in a finite way, algorithms for finite graphs do not usually carry over automatically. For example, precise practical methods for determining isomorphism of periodic graphs have been missing up until now, although much searched for by chemists and crystallographers [Kle87].

Several authors [OB92,TRR+97,DDH+99], have constructed many examples of potential crystal structures by means of combinatorially enumerating periodic graphs. In addition to the isomorphism problem, this gives rise to a demand for chemically reasonable 3-dimensional drawing algorithms.

Barycentric placements have been introduced by Tutte [Tut60,Tut63] for constructing convex drawings of "almost" 3-connected planar graphs. Here, an outer face is chosen and realized as an arbitrary convex polygon. Tutte showed that for any such choice a barycentric placement of the remaining vertices uniquely exists. If, moreover, the only pairs of vertices at which the graph can be cut are situated on different sides of the outer polygon, it induces an embedding with convex inner faces and the maximal possible symmetry which is compatible with the outer polygon.

We obtain a general existence and uniqueness result for connected, periodic graphs in general and a convexity result similar to Tutte's for 3-connected, 2-periodic, planar graphs. It should be noted that barycentric drawings have been

employed by crystallographers for quite a while, but the author is not aware of any previous rigorous justification of that practice.

We show further that for a class of graphs which we call *stable*, a number of algorithmic problems can be solved in polynomial time using exact barycentric drawings as a tool.

In [Del01], barycentric drawings for periodic graphs are treated mathematically, with only a brief glance at computational issues. Here, we skip the more intricate mathematical details, while concentrating on the algorithmic and graphical aspects. The next section introduces periodic graphs. Section 3 defines barycentric drawings and gives some basic results. The two-dimensional case is treated in Section 4, and an algorithm for finding canonical forms and symmetries in Section 5. Section 6, finally, discusses some applications.

2   Periodic Graphs

A d-periodic graph is a simple undirected graph on which the group $\mathbb{Z}^d$ of d-dimensional integer vectors acts by automorphisms. The action is to be free, i.e., no vector may map any vertex onto itself, and there may be only finitely many orbits on the set of vertices as well as on the set of edges. Consequently, the set of vertices must be countable and the graph must be locally finite, i.e., only finitely many edges may be incident to any given vertex.

We interpret the result of applying an integer vector to a vertex or edge as a formal translation and write $v + s$ or $e + s$, respectively, where $s = (s_1, \ldots, s_d)$, and $v + (s + t) = (v + s) + t$ and so on.

An isomorphism between periodic graphs $G = (V, E)$ and $G' = (V', E')$ is an ordinary graph isomorphism, say f, which respects the $\mathbb{Z}^d$ action. This means that for each vector $s \in \mathbb{Z}^d$, there is another vector, s', such that $f(v + s) = f(v) + s'$ for any vertex $v \in V$. It is easy to see that the function $f^* \colon \mathbb{Z}^d \to \mathbb{Z}^d$ which maps every vector s to its corresponding s' must be linear and bijective, thus it can be represented by a $d \times d$ unimodular matrix (a matrix with integer entries and determinant 1 or -1).

A natural way to encode periodic graphs is the so-called vector or (labelled) orbit graph method [CHK84]: let n be the number of vertex orbits and m the number of edge orbits under the $\mathbb{Z}^d$ action. Choose arbitrary representatives $v_1, \ldots, v_n$ of the vertex orbits. Now each vertex $v \in V$ can be written uniquely as $v_i + s$ for some $i \in \{1, \ldots, n\}$ and some $s \in \mathbb{Z}^d$. For every edge orbit, there is a unique representative $e = (v_i, v_j + t)$, where either $i < j$ or else $i = j$ and the first nonzero entry of t is positive. Specifying all these edge representatives completely characterizes a periodic graph. Consequently, a periodic graph of dimension d with m edge orbits is given as an array of $m \times (d + 2)$ numbers, where the first two entries in each row are indices of vertex representatives, while the remaining d entries form a vector, which we call the shift vector for this edge, and by which the second vertex is to be shifted to obtain the second end of the edge.

Here are a few tiny examples:

$$
\begin{array}{cc}
\begin{array}{cccc} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{array}
& \text{and} &
\begin{array}{cccc} 1 & 2 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{array}
\end{array}
$$

encode the 1-skeleton of a face-to-face tiling by regular squares and the 1-skeleton of a face-to-face tiling by regular hexagons, respectively.

Their 3-dimensional analogues are

$$
\begin{array}{cc}
\begin{array}{ccccc} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{array}
& \text{and} &
\begin{array}{ccccc} 1 & 2 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 \\ 1 & 2 & 0 & 0 & 1 \end{array},
\end{array}
$$

the 1-skeleton of a face-to-face tiling by regular cubes and the atom-bond network of diamond.

A vector representation is naturally interpreted as a finite directed multi-graph with edges labelled by shift vectors, where vertices, edges and their in-cidences represent vertex and edge orbits and their incidences in the original graph, hence the alternative term orbit graph.

For each periodic graph, there are infinitely many choices of vertex represen-tatives. There are, moreover, infinitely many ways of choosing a basis for $\mathbb{Z}^d$, a choice which determines the actual coordinates used in specifying the shift vectors. An obvious improvement is to require that the set of representatives $v_1, \ldots, v_n$, or rather the subgraph induced by them, be connected, if possible. This leaves, up to translations, only finitely many choices for the set of vertex representatives. It is easy to see that a periodic graph is connected if and only if it has a connected set of vertex representatives and for any such set the induced set of shift vectors spans $\mathbb{Z}^d$ [CM91]. It should be possible to formulate addi-tional conditions on the set of shift vectors in order to arrive at a finite set of permissible representations. Note, however, that the set of shift vectors need not contain an integral basis for $\mathbb{Z}^d$. Even in the 1-dimensional case, the "vectors" 2 and 3 form a counterexample.

In the following, we will assume that all periodic graphs under consideration are connected.

3   Barycentric Drawings and Isomorphisms

It is reasonable to require that a drawing of a periodic graph shall display the periodicity. In the following, we will restrict our attention to straight line drawings. Consequently, a drawing is determined by a vertex placement $p\colon \{v_1, \ldots, v_n\} \to \mathbb{R}^d$ and a linear, nonsingular function $p^*\colon \mathbb{Z}^d \to \mathbb{R}^d$, where the extension of p to V is given by $p(v_i + s) := p(v_i) + p^*(s)$.

A barycentric drawing is one which satisfies

$$
p(v) = \frac{1}{|N(v)|} \sum_{w \in N(v)} p(w),
$$

where $N(v)$ denotes the set of all vertices adjacent to v. This definition, quite obviously, is invariant under affine transformations, i.e., if p^* is changed or the whole graph is shifted by some vector, a barycentric drawing remains barycentric.

If p^* and the position of an arbitrary vertex are preassigned, say $p^* = \mathrm{Id}_d$ and $p(v_1) = \mathbf{0}$, there is a unique barycentric drawing satisfying these conditions. Consequently, barycentric drawings always exist for connected, periodic graphs and are unique up to affine transformations.

To see why, consider the "energy function"

$$E(p) := \sum_{vw \in E_0} d(p(v), p(w))^2,$$

where d denotes Euclidean distance and E_0 is a set of representatives for the edge orbits. It is straightforward to see that p is barycentric if and only if p, taken as a point in $\mathbb{R}^{nd}$, is critical for E, i.e., all partial derivatives vanish. Now consider

$$||p||^2 = |p(v_1)|^2 + \ldots + |p(v_n)|^2.$$

Because (V, E) is connected and $p(v_1) = \mathbf{0}$, the maximum edge length can be made arbitrarily large by just taking any p with $||p||^2$ large enough. Now E is a quadratic function, which by the above is non-degenerate in each direction, thus must have a unique critical point, a minimum (c.f. [RG96]).

This does not only show the existence and uniqueness of barycentric drawings, but also points to an aesthetically interesting property, namely that barycentric drawings minimize the sum of squared edge lengths among all periodic drawings with a given p^*.

In the following, we will always use p to denote the barycentric vertex placement with $p^* = \mathrm{Id}_d$ and $p(v_1) = \mathbf{0}$.

We call a graph stable if p is injective, thus vertices can be identified uniquely by their positions. This is a key to lowering the complexity of several computational problems dealing with periodic graphs. Where traversal algorithms are employed, it is usually enough, though, to require local stability, meaning that p need only be injective on $N(v)$ for each vertex v.

For display purposes, p can be conveniently approximated by iterative, force directed methods (c.f [Ead84,FR91]), where all forces are attractive, only exist between neighbors and are proportional in size to the respective edge lengths. In order to identify vertices by their positions, however, the linear system of equations induced by the barycentricity condition has to be solved exactly, i.e., using arbitrary precision rational arithmetic. This is necessary because in principle, vertices positions can get arbitrarily close without being identical.

Next, we claim that an isomorphism between periodic graphs induces an affine mapping between their barycentric drawings. To see this, let f be an isomorphism between graphs (V, E) and (V', E') with barycentric placements p and p', respectively. Define $g^*(s) := p'^*(f^*(p^{*-1}(s)))$ and consider new placements q_1 and q_2 on V, where $q_1(v) := p'(f(v))$ and $q_2(v) = g^*(p(v)) + p'(f(v_1)) - g^*(p(v_1))$. It is then an easy exercise to see that $q_1(v_1) = q_2(v_1) = p'(f(v_1))$ and $q_1(v + s) - q_1(v) = q_2(v + s) - q_2(v) = g^*(p^*(s))$ for any $s \in \mathbb{Z}^d$. Because f is an

isomorphism and all the $\cdot^*$ functions are linear, both q_1 and q_2 are barycentric drawings, so they must be identical.

We state this as our first theorem:

Theorem 1. *If f is an isomorphism between connected periodic graphs (V, E) and (V', E') with barycentric vertex placements p and p', respectively, then there is an affine function φ such that $p'(f(v)) = \varphi(p(v))$ for all $v \in V$.*

Theorem 1 has a nice consequence regarding automorphisms of periodic graphs:

Theorem 2. *If (V, E) is a connected periodic graph, there is a linear function $p^*: \mathbb{Z}^d \to \mathbb{Z}^d$ such that for any barycentric placement satisfying $p(v+s) = p(v) + p^*(s)$ for all $v \in V$ and all $s \in \mathbb{Z}^d$ and any automorphism f of (V, E), there is an isometry φ of $\mathbb{R}^d$ with respect to the Euclidean metric, such that $p(f(v)) = \varphi(p(v))$ for all $v \in V$.*

What this means, essentially, is that the full automorphism group of a periodic graph can be realized as a crystallographic space group. There can be only finitely many conjugacy classes of automorphisms with respect to the $\mathbb{Z}^d$ action. Turning a finite set of affine maps into isometries by redefining the metric, or equivalently, the basis with respect to which they act, is a standard trick in mathematical crystallography, see for example [Sch80].

For stable periodic graphs, every nontrivial automorphism must move some vertex, so the mapping from automorphisms to isometries is injective. Consequently:

Corollary 1. *The automorphism group of any stable, connected, periodic graph is isomorphic to a crystallographic space group.*

For unstable graphs, some nontrivial automorphism may be mapped to the identity. As an example, see Figure 1, where any barycentric placement must map pairs of adjacent white vertices to the same position, so that the automorphism that just flips each such pair induces the identity on $\mathbb{R}^d$.
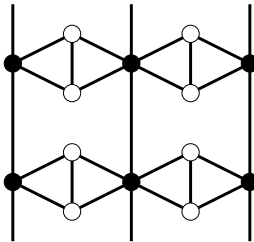

Fig. 1. An unstable 2-periodic graph.

4   Planar Graphs

Tutte's classical results on convex drawings of finite planar graphs [Tut60,Tut63] carry over to planar 2-periodic graphs. In particular, we have the following:

Theorem 3. *For a connected, 2-periodic, planar graph G, the following are equivalent:*

1. *G is 3-connected.*
2. *G has a convex embedding in the plane.*
3. *G has a convex embedding in the plane such that every automorphism of G is realized by a crystallographic symmetry of the embedding.*

Here, as usual, an embedding of a graph is a drawing in which the vertices and interiors of edges are all pairwise disjoint. Whenever the context is clear, we do not distinguish in terminology between vertices or edges and their images in a drawing.

A similar convexity criterion has been given by Thomassen [Tho80] for infinite planar graphs in general, but does not imply anything with respect to symmetries. Obviously, (3) implies (2), and, employing an easy lemma stating that any finite vertex cut in a periodic graph has exactly one infinite component, it is straightforward to see that (2) implies (1).

The implication (1) $\rightarrow$ (3) follows from a result in [Del01] (Theorem 20), namely:

Theorem 4. *The straight line drawing induced by any good, periodic vertex placement of a 2-periodic, 3-connected, planar graph is a convex embedding.*

Here, a vertex placement is called good if every vertex v is placed in the relative interior (c.f. [RG96]) of its neighbor's positions, defined as

$$\left\{ \sum_{w \in N(v)} \lambda_w p(w) \,\middle|\, \sum_{w \in N(v)} \lambda_w = 1; \, \lambda_w > 0 \right\}.$$

The proof, which we are going to sketch here briefly, was largely inspired by Richter-Gebert's treatment [RG96] of the classical Tutte Theorem.

The first step is to show that the neighbors of a given vertex v can never be placed on a common straight line h. For, assume otherwise and let, without loss of generality, h be horizontal. Consider the largest connected subgraph S of G containing v with all vertices placed on h. Then S, by periodicity, can not be the whole of G. Because of the 3-connectedness, and using Menger's theorem, there must be at least 3 vertices a, b and c in S with neighbors not on h. Because the placement is good, each of these has neighbors both above and below h. Using the periodicity again, one can show that those above h are pairwise connected by paths staying above h. Likewise, those below h are connected below h. But then we have 3 independent ways of interconnecting vertices a, b and c, one on, one above and one below h, which establishes a $K_{3,3}$ minor in G.

Consequently, in a good placement, every vertex falls inside the convex hull of its neighbors, which must be non-degenerate. The next step is to prove the theorem for triangulations without degenerate triangles. By summing up angles in two different ways, first around vertices and then inside triangles, and using an Euler characteristic argument, one shows that the triangles are all oriented consistently and that angles sum up to 2π around each vertex. The simple-connectedness of the plane then implies that there can be no overlaps between triangles.

Finally, the theorem is extended to the general case by first triangulating each facial cycle of the given graph without introducing new vertices. This can be done in such a way that the $\mathbb{Z}^d$ images of a facial cycle are triangulated consistently, making the resulting triangulated graph G' periodic again. If p is a good placement for G, it is also a good placement for G'. A perturbation argument is used to show that no degenerate triangles can occur. Consequently, p must induce a straight-line embedding of G' and thus of G. But because p is good, all interior facial angles occurring in the induced embedding of G are less than π, so all the faces are convex.

A nice feature of Theorem 4 is that it allows for some flexibility in the construction of drawings. Tutte's barycentric construction results in aesthetically pleasing pictures, but is also well-known for allowing a rather large variation in edge lengths, making some graph structures hard to recognize. This can be remedied to some extend by using a force-directed drawing method with the barycentric placement as its initial value. Such methods can be adjusted to support a more uniform distribution of edge lengths, for example by using attractive forces proportional to some power larger than 1 of the edge lengths. As long as the process is symmetric and carefully tuned as to keep the result of each iteration good, the final drawing will still be convex and display the full symmetry. The author has first employed this idea to polyhedra graphs, for which an analogue of Theorem 4 holds [RG96]. Figure 2 shows an example.
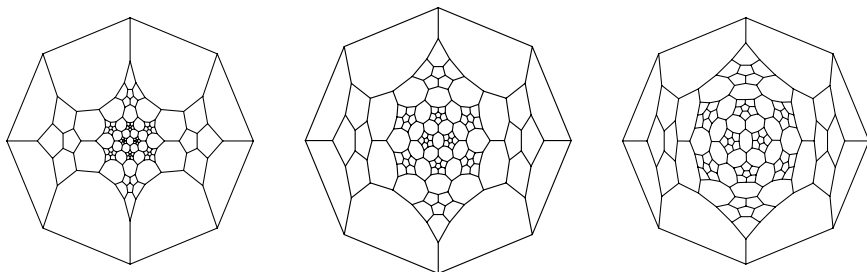

Fig. 2. Convex drawings of a large polyhedron graph. Left: barycentric drawing. Middle and right: variants with attractive forces proportional to the square and fourth power of the edge length.

A more general approach of relaxing the barycentric drawing would define an appropriate penalty function and use optimization methods such as simulated annealing or downhill simplex to search for a minimum. In Figure 3 we show three drawings of the same periodic graph, all derived from variations and relaxations of the barycenter construction.
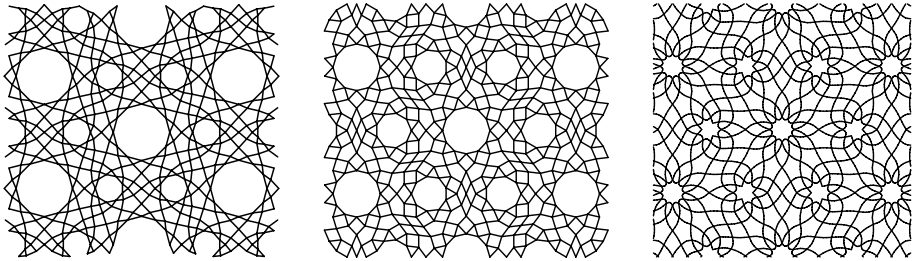

Fig. 3. Three drawings of a periodic planar graph, inspired by a Moorish pattern. Left: barycentric drawing. Middle: relaxed barycentric drawing with equal edge lengths. Right: vertices on mirror planes were spaced evenly, then those inside the fundamental triangle were placed using a barycentric drawing of the second derived subdivision.

5   Symmetries and Canonical Forms

Isomorphism of two structures can be tested in linear time when these are given in a canonical form, i.e., a representation that uniquely depends on the isomorphism type. Therefore, an efficient canonical forms algorithm is highly desirable.

A vector representation of a periodic graph as in Section 2 can be converted into a unique linear string of numbers by first sorting the rows lexicographically and then concatenating them into a single row. Our approach towards canonical forms is to efficiently construct a small characteristic set of vector representations and pick a minimal one with respect to the lexicographic order of their corresponding strings.

A straightforward way to obtain a finite set is by picking vertex representatives as they occur in, say, a breadth first traversal of the graph. To make the resulting set characteristic, every possible order of picking vertices has to be executed, which may easily lead to an exponentially large set of representations.

We use barycentric placements to arrive at a smaller set of traversals. Let p be a barycentric placement and recall that a graph is called locally stable if for each vertex v the function p is injective on the set $N(v)$ of v's neighbors. Then a traversal must pick these in lexicographical order of their coordinates as given by p, with respect to some appropriate basis. Because p is unique up to affine transformations, there is a unique breadth first traversal of this type for every choice of start vertex and coordinate basis.

It remains to pick a characteristic set of pairs, each consisting of a vertex and a basis. The term characteristic means that these must depend only on the

isomorphism type of G, not on its original representation. A natural choice is an ordered d-tuple of edge vectors. There are $2m$ of these in a graph with m edge orbits, thus $(2m)^d$ potential bases, which can be readily determined from a given vector representation. Of course, only linearly independent tuples can be used, but by the d-periodicity and connectedness of G, at least one such exists. As a start vertex for the traversal, the first vertex of the first directed edge is used.

If during the traversal, a vertex is encountered which belongs to a new orbit, it is made the representative of this orbit and an edge with trivial shift vector is added to the new representation. If the orbit has already been seen, an edge with a non-trivial shift vector is added. These shift vectors are expressed in the basis used for the traversal, which is determined by the original tuple of edge vectors and in general is not an integral basis of the translation lattice. Thus, in the end, a basis for the integral span of the shift vectors must be computed in a deterministic way and the shift vectors expressed in terms of that basis. The complete algorithm is shown as Algorithm 5 below.

It is straightforward to extend the algorithm to computing the automorphisms as well. Indeed, whenever two d-tuples of directed edges lead to the same representation, an automorphism of the periodic graph G has been found and all automorphisms must occur in this way.

The runtime analysis of the Algorithm is complicated by the fact that exact rational numbers have to be used in all the calculations involving vertex positions. These arise from the solution of a linear system of equations $AX = B$, where A is an $n \times n$ integer matrix, B is an $n \times d$ integer matrix and X is an $n \times d$ rational matrix. Here, the i-row of A corresponds, of course, to the barycenter condition for the i-th vertex representative. It is not hard to show that the logarithm of the numerators and denominators of entries in X is $O(n \log n)$ if the entries in A and B are assumed to be constant in size, say, representable by machine words. In practice, we may then also assume that n is representable by a machine word, so the above boils down to $O(n)$. It turns out that the maximal representation size of any rational number occurring during the execution of the algorithm rises to $O(dn)$.

Ignoring the complexity of arithmetic for the moment, the main loop of Algorithm 5 runs in time $O(dm^{d+1} \log D)$, where m, once again, is the number of edge representatives and D is the maximal degree of a vertex. Clearly, the main loop is executed $(2m)^d$ times and the time spent per iteration is dominated by sorting the transformed shift vectors at each step of the traversal and accumulates to $O(dm \log D)$. Everything else, including the preparations before the execution of the main loop, takes time $O(dm)$.

It remains to consider the final steps after the main loop. In order to find a basis for the integral span of the shift vectors, we need to triangulate the matrix formed by these using column operations which only add integer multiples of columns to other columns. The problem here is that it may take several steps to eliminate a single entry, which may lead to an overall doubly exponential growth in some of the other entries. Clearly, by multiplying with the least common denominator of all entries, we can reduce the original problem to that of

Algorithm 5. Canonical Form.
Input

 - A list E of edge representatives of the form (v, w, s) with $v, w \in \mathbb{N}$ and $s \in \mathbb{Z}^{d \times 1}$ (a d-dimensional column vector), representing a connected, locally stable, d-periodic graph G.
 - A barycentric vertex placement p for G.

Output

 - A list C of edge representatives, constituting a canonical form for G.

Set $A = [\,]$.
for (v, w, s) in E:
    Append $(v, w, (p_w + s) - p_v)$ and $(w, v, p_v - (p_w + s))$ to A.
for $v = 1$ to n:
    Let $S_v = [\,(w, s) \mid (v, w, s) \in A\,]$.

for each ordered d-tuple $((v_1, w_1, s_1), \ldots, (v_d, w_d, s_d))$ from A:
    Let $v = v_1$.
    Let B be the $d \times d$ matrix with columns $s_1, \ldots, s_d$.
    if B is singular:
        Skip to the next loop iteration.
    Set up an empty FIFO queue Q and enter v.
    Set $I_v = 1$, $q_v = \mathbf{0}$ and $k = 2$.
    Set $N = [\,]$.

    while Q is not empty:
        Pop v from Q.
        Set $T = [\,(w, B^{-1}s) \mid (w, s) \in S_v\,]$ and sort by second components.
        for (w, s) in T:
            if I_w is undefined:
                Set $I_w = k$, $q_w = q_v + s$ and increase k.
                Enter w into Q.
            Set $t = q_v + s - q_w$.
            if $I_v < I_w$ or ($I_v \equiv I_w$ and $t > \mathbf{0}$):
                Append (I_v, I_w, t) to N.

    if C is undefined or $N < C$:
        Set $C = N$.

Let B be a deterministic basis for the integral span of $[\,s \mid (v, w, s) \in C\,]$.
for (v, w, s) in C:
    Set $t = B^{-1}s$.
    if $v \equiv w$ and $t < \mathbf{0}$:
        Set $t = -t$.
    Replace (v, w, s) by (v, w, t) in C.
Sort and return C.

triangulating a matrix over the integers. Bachem and Kannan [BK79] were the first to present an polynomial time algorithm for this latter problem.

We conclude that, even after taking into account the extra cost involved in using precise rational number arithmetic, canonical forms for stable, connected, periodic graphs can be computed in polynomial time (assuming the dimension is taken to be constant) using barycentric vertex placement, and that two such graphs, given by their canonical forms, can be tested for isomorphism in time $O(md)$.

6   Applications

The methods discussed here have been implemented in the programming language Python. The program, called *Systre*, is available from the author upon request. It contains a library for dealing with crystallographic groups in Dimension 3, following the conventions of the so-called International Tables [Hah83]. Because crystal structures often display fewer translational symmetries than possible, an algorithm for finding additional potential translations has been incorporated as well. These can be recognized immediately as translational symmetries of the barycentric drawing. Systre computes both the ideal symmetry group and
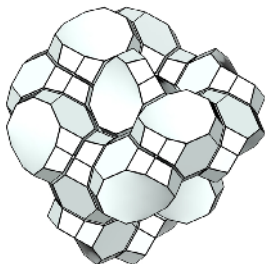

Fig. 4. A periodic tiling of 3-dimensional space with some non-planar faces.

canonical form of a given periodic graph. The latter can be used as a database index. Graphs are specified by vector representations or, alternatively, by atom positions and unit cell parameters. Systre then tries to deduce the bonds. A relaxed drawing, aiming for uniform edge lengths and a large total volume can be produced, using the barycentric drawing as a starting point.

In Section 4 we discussed barycentric placements for periodic, planar graphs and thus periodic tilings of the plane. Tilings of 3-dimensional space can be drawn similarly: first the 1-skeleton is embedded, using barycentric placement and force-models. Non-planar faces are then realized by first triangulating them, then barycentrically placing the inner vertices. This leads to nice, smooth-looking curved surfaces with the full symmetry of their respective boundaries. Figure 4 shows an example.

References

[BK79]   A. Bachem and R. Kannan. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM Journal Computing*, 8:499–507, 1979.

[CHK84]   S. J. Chung, Th. Hahn, and W.E. Klee. Nomenclature and generation of three-periodic nets: the vector method. *Acta Cryst.*, A40:42–50, 1984.

[CM91]   Edith Cohen and Nimrod Megiddo. Recognizing properties of periodic graphs. In *Applied geometry and discrete mathematics*, volume 4 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 135–146. Amer. Math. Soc., Providence, RI, 1991.

[DDH+99]   O. Delgado Friedrichs, A.W.M. Dress, D.H. Huson, J. Klinowski, and A.L. Mackay. Systematic enumeration of crystalline networks. *Nature*, 400:644–647, 1999.

[Del01]   O. Delgado-Friedrichs. Equilibrium placement of periodic graphs and tilings. submitted, 2001.

[Ead84]   P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[FR91]   T. Fruchterman and E. Reingold. Graph drawing by force-directed placement. *Software—Practice and Experience*, 21(11):1129–1164, 1991.

[Hah83]   T. Hahn, editor. *International Tables for Crystallography*, volume A. D. Reidel Publishing Company, Dordrecht, Boston, 1983.

[Kle87]   W.E. Klee. The topology of crystal structures. *Z. Kristallogr.*, 179:67–76, 1987.

[OB92]   M. O'Keeffe and N. E. Brese. Uninodal 4-connected 3d nets. I. Nets without 3- or 4-rings. *Acta Cryst.*, A48:663–669, 1992.

[OEL+00]   M. O'Keeffe, M. Eddaoudi, Hailian Li, T. Reineke, and O. M. Yaghi. Frameworks for extended solids: Geometrical design principles. *J. Solid State Chem.*, 152(1):3–20, 2000.

[Orl84]   James B. Orlin. Some problems on dynamic/periodic graphs. In *Progress in combinatorial optimization (Waterloo, Ont., 1982)*, pages 273–293. Academic Press, Toronto, ON, 1984.

[RG96]   Jürgen Richter-Gebert. *Realization Spaces of Polytopes*. Springer Verlag, Berlin, 1996.

[Sch80]   Rolf L. E. Schwarzenberger. *n-dimensional crystallography*, volume 41 of *Research Notes in Mathematics*. Pitman (Advanced Publishing Program), Boston, Mass., 1980.

[Tho80]   C. Thomassen. Planarity and duality of finite and infinite graphs. *Journal of Combinatorial Theory,* Series B, 29:244–271, 1980.

[TRR+97]   M.M.J. Treacy, K.H. Randall, S. Rao, J.A. Perry, and D.J. Chadi. Enumeration of periodic tetrahedral frameworks. *Z. Krist.*, 212:768–791, 1997.

[Tut60]   W.T. Tutte. Convex representations of graphs. *Proc. London Math. Soc.* (3), 10:304–320, 1960.

[Tut63]   W.T. Tutte. How to draw a graph. *Proc. London Math. Soc.*, 13:743–767, 1963.