A universal set of growth operations for fullerenes

Mahdieh Hasheminezhad [a,1], Herbert Fleischner [b], Brendan D. McKay [c,*]

[a] Department of Computer Science, Faculty of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran
[b] Vienna University of Technology, Institute for Information Systems, Favoritenstrasse 9-11, A-1040 Vienna, Austria
[c] Department of Computer Science, Australian National University, Building 108, Canberra, ACT 0200, Australia

ABSTRACT

We give a simple set of growth operations that suffice to generate all fullerene isomer structures.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

The problem of listing all fullerene isomers with a given number of carbon atoms has received a considerable amount of attention. The first tool was the spiral algorithm [1], which was shown to be incomplete in that some isomers are omitted [2]. Corrective modifications of the spiral algorithm [3] have importance for nomenclature [4], but have not been demonstrated to be efficient for enumeration. Nor have mathematical classification schemes, despite their ingenuity [5]. An alternative technique is the folding net [6], but it has not been shown to produce all isomers. The most successful practical method to date with proven completeness has been the program *fullgen* of Brinkmann and Dress [7,8] that operates by stitching together 'patches' bounded by zigzag (Petrie) paths.

A different approach, the focus of our research, is to construct fullerenes from some simple set of starting isomers by successively applying a growth operation [9] (which we will call an *expansion*) that replaces some fragment of the fullerene by a larger piece. Such transformations are postulated to have physical as well as theoretical reality [10–12]. It is known [13] that no finite set of expansions suffice, so instead we seek an easily-described infinite class of them. Brinkmann et al. [14] gave small growth operations of this nature that suffice to construct all fullerene isomers to at least 200 atoms but fail in the general case. Our aim in the present Letter is to give a set of expansions that is provably complete.

2. The algorithms

As starting points, we consider C_{20} (the dodecahedron), $C_{28}(T_d)$ (index 28:2 in [15]), and the type-(5,0) nanotube fullerenes. There

is exactly one of the latter class with $30 + 10k$ atoms for each $k \geqslant 0$. The smallest is $C_{30}(D_{5h})$ (index 30:1 in [15]); the others have additional rings of hexagons. See Fig. 1.

We will define two families of expansions L_i and $B_{i,j}$ together with a single expansion F. We will establish the following properties.

Algorithm 1. Every fullerene isomer except $C_{28}(T_d)$ can be constructed from C_{20} using expansions of type L, B, and F.

Algorithm 2. Every fullerene isomer, except $C_{28}(T_d)$ and type-(5,0) nanotube fullerenes, can be constructed from C_{20} using expansions of type L and B.

Algorithm 3. Every fullerene isomer with at most 300 atoms, except $C_{28}(T_d)$ and type-(5,0) nanotube fullerenes, can be constructed from C_{20} using expansions of type L.

At the expense of adding an extra expansion type we could generate $C_{28}(T_d)$ from C_{20} as well; for example, removing the central atom of $C_{28}(T_d)$ and suppressing the resulting atoms of valence 2 yields C_{24}. However we have chosen to keep the number of expansion types to a minimum.

We will now formally introduce the expansions: L_i for $i \geqslant 0$ that adds $i + 2$ faces, $B_{i,j}$ for $i,j \geqslant 0$ that adds $i + j + 3$ faces, and F that adds 5 faces. These are defined in accordance with the examples in Fig. 2. In the case of L and B, the transformation consists of inflation of a path between two pentagons. Expansion L_i uses a path of length $2i + 3$ that alternates left and right, while $B_{i,j}$ uses a path of length $2i + 2j + 5$ that alternates left and right except that bends $2i + 2$ and $2i + 3$ have the same orientation. The mirror image of L_i is implicitly included. Though it is not necessarily required for the validity of the expansions, we will show that it suffices to include the cases where all the faces drawn completely in the figure or labelled as f_k or g_k are distinct. However, faces not shown as pentagons or hexagons may be either.

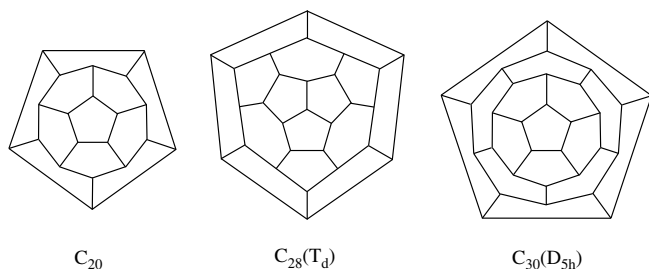By a *reduction* we mean an operation inverse to an expansion.

C$_{20}$ C$_{28}$(T$_d$)   C$_{30}$(D$_{5h}$)

Fig. 1. Basic fullerene isomers.

3. Justification of the algorithms

The main step in the justification of the algorithms consists of identifying the fullerene structures G that cannot be reduced using L, B or F reductions.

We divide the argument according to the value k which is the greatest number of pentagons adjacent to a common pentagon.

If $k = 5$, then either G is C_{20} or else G has a cut of five independent edges with more than a pentagon on each side. The latter only occurs for type-(5,0) nanotube fullerenes [16], which have reductions of type F.

If $k = 4$, then consider a pentagon p which is adjacent to four pentagons $p_1,\ldots,p_4$ and one hexagon h as in Fig. 3a. If f_2 is a hexagon then G has reduction $L(f_2,p_3,p,h)$. (Here, f_2,p_3,p,h is a list of faces corresponding to the 'after' picture for L_0 in Fig. 2. We will index other explicit reductions similarly.) If f_2 is a pentagon then f_1,f_3 are hexagons (since $k = 4$) and $B(f_3,p_4,h,p_1,f_1)$ applies.

Suppose $k = 3$ and let p be a pentagon which is adjacent to three pentagons p_1,p_2,p_3 and two hexagons h_1,h_2. If p_1,p_2,p_3 appear around p consecutively as in Fig. 3b, then f_1 and f_2 are hexagons and G has reduction $L(h_2,p,p_2,f_1)$. Otherwise p_1,p_2,p_3 appear around p as in Fig. 3c. If f_1 is a hexagon, then reduction $L(f_1,p_2,p,h_2)$ applies to G, and if f_3 is a hexagon, then reduction $L(f_3,p_3,p,h_1)$ ap-

plies. Finally, if f_1 and f_3 are pentagons then f_2 must be a hexagon as in Fig. 3d. So G is either C_{28} (T_d), or at least one of $f_4,\ldots,f_9$, say f_4, is a hexagon and G has reduction $L(f_4,f_3,f_2,f_1,h_1)$.

In the case that $1 \leqslant k \leqslant 2$, if there is a pentagon p_1 which is adjacent to exactly one pentagon p_2, then considering the neighbours of p_1 and p_2 as in Fig. 3e, at least one of f_1 and f_2, say f_1, is a hexagon and reduction $L(f_1,p_2,p_1,h_3)$ applies. Otherwise, $k = 2$ and there is a ring of pentagons not adjacent to any other pentagons. If there are three pentagons p_1,p_2,p_3 with a common atom as in Fig. 3f, then reduction $L(h_1,p_1,p_2,h_4)$ applies. So suppose there are no such three pentagons but there is a longer ring R of pentagons. Rings of four pentagons are impossible, and rings of five pentagons only occur surrounding another pentagon [16], so R contains six or more pentagons. If the outside or inside of R consists of a single hexagon, then G has reduction $L(h_1,p_1,h,p_2,h_2)$ as in Fig. 3g. Otherwise, there are two adjacent pentagons p_1,p_2 on R in the configuration of Fig. 3h and reduction $L(h_1,p_1,p_2,h_2)$ applies.

We are left with the case that all the pentagons are isolated. Let π be a path of faces in G such that:

(i) π is as short as possible among all the paths between two pentagons, say p_1,p_2.

(ii) Subject to condition (i), if π is not straight, the segment from p_1 to the first bend is as long as possible. (A path of faces $\ldots f_1 h f_2 \ldots$, where h is a hexagon, is said to have a bend at h if f_1,f_2 are not on opposite sides of h.)

Condition (i) implies that the faces on π, other than p_1 and p_2, are hexagons, and that there are no sharp (60°) bends. It also implies that π (as for all shortest paths) is not self-intersecting. We will show that in fact π is either straight or has a single 120° bend.

Suppose that π has more than one bend, with the first two bends occurring at hexagons h_1 and h_2. If these bends have the same orientation (both left or both right), then the walk π' defined as in Fig. 4a is shorter than π, which violates (i). If instead the two bends have opposite orientation, then the path π'' defined as in
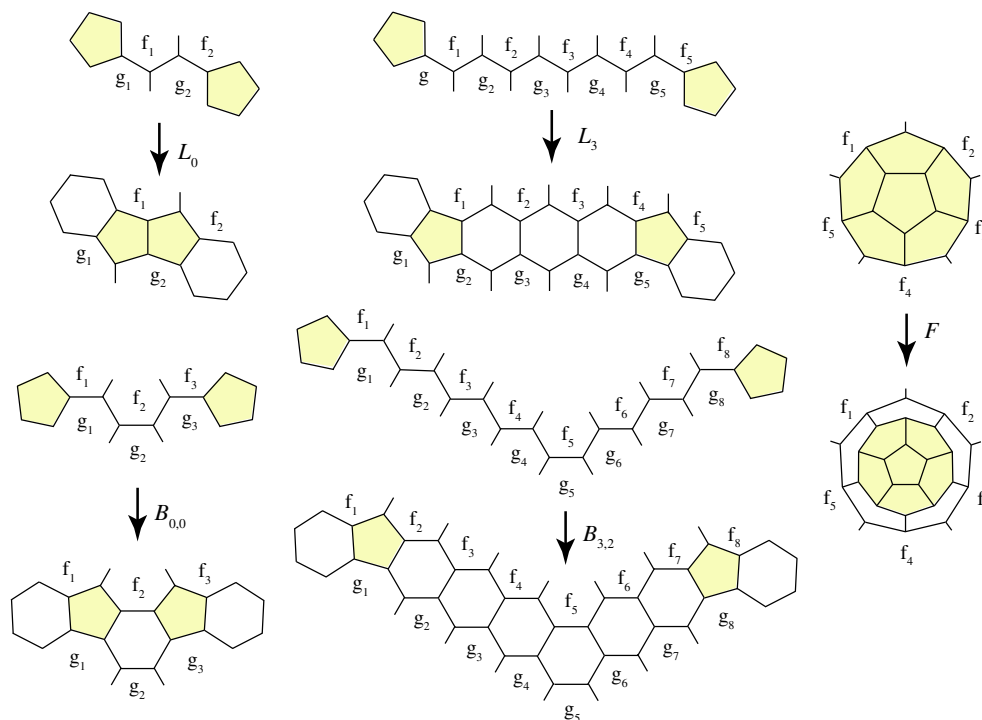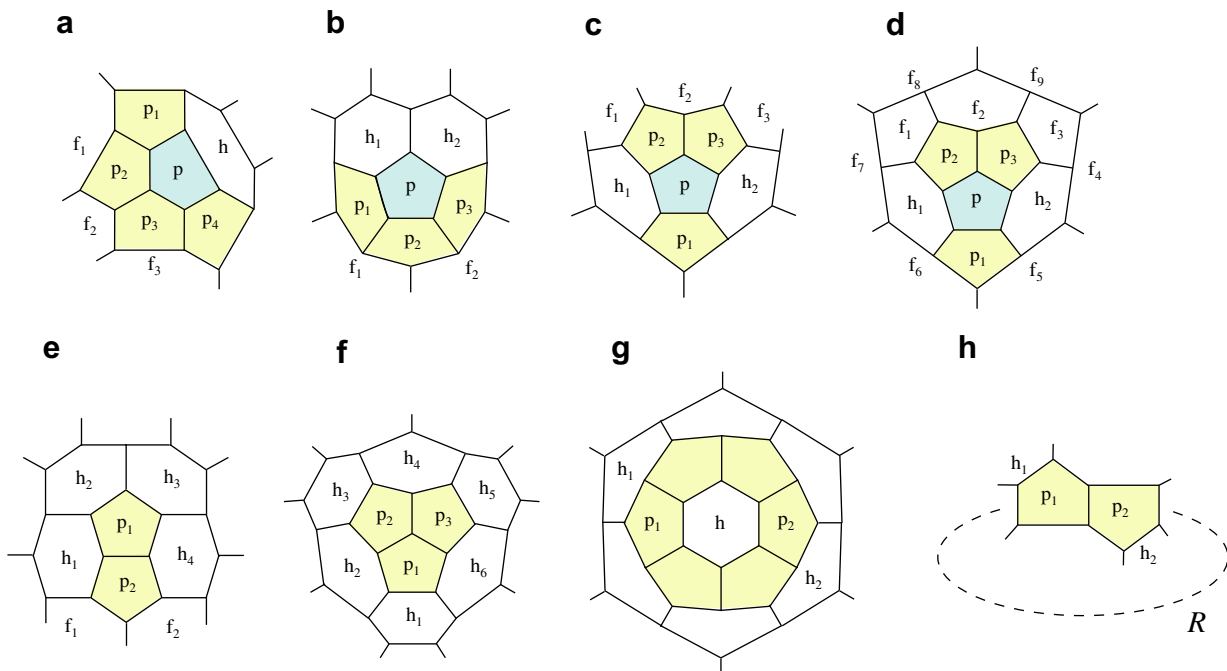

Fig. 2. Expansions.

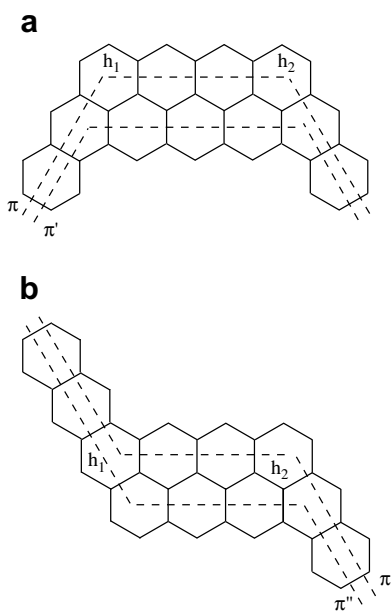**Fig. 3.** Configurations appearing in the proof.

Fig. 4. Illustration of the case that all pentagons are isolated.

Fig. 4b has the same length but a longer first segment. Therefore, π has at most one bend.

If π has no bends then an L reduction applies to G and otherwise a B reduction applies. The uniqueness of the faces neighbouring π, as specified in Section 2, follows from property (i) together with the fact that fullerenes do not have cuts of four independent edges [17].

The foregoing establishes that all fullerene structures other than C_{20} and $C_{28}(T_d)$ are reducible by an L, B or F reduction. To complete the justification of Algorithm 1, we can easily check that every fullerene formed by applying a single expansion to $C_{28}(T_d)$ can be reduced to C_{20} without $C_{28}(T_d)$ appearing as an intermediate step.

To establish the correctness of Algorithm 2, we start by recalling that reduction F can only be applied in the case of a type-(5,0) nanotube fullerene, as noted above in case $k = 5$. Furthermore, each L or B expansion of a type-(5,0) nanotube fullerene G can be reduced by L or B to a fullerene which is not of type (5,0).

For Algorithm 3, first note that in the case there are adjacent pentagons we applied a B reduction only in the case of Fig. 3a with f_2 a pentagon and f_1,f_3 hexagons. Such fullerenes are type-(3,3) nanotubes [16] with the same configuration of pentagons occurring in each cap (including the 0th member of this nanotube sequence, C_{26}). In that case there is an L reduction which extends from one cap to the other. Therefore, only L reductions are required for adjacent pentagons. If all the pentagons are isolated, B reductions might be required but this does not occur until quite a large size. We have not established when this first happens, but examination of all IPR (isolated pentagon) fullerenes with up to 300 atoms using *fullgen* [7] found no examples. The smallest we know of is a fullerene of 1340 atoms found by Brinkmann et al. [14], which has reduction $B_{1,6}$ but no reductions of type L. Note that there are probably in excess of 10^{11} fullerene isomers with at most 300 atoms, so Algorithm 3 is likely to satisfy any practical purpose.

Table 1
Counts of IPR fullerenes with 214–300 atoms

214	36173081	236	117166528	258	331516984	280	842498880
216	40536922	238	129476607	260	362302637	282	912274538
218	45278722	240	142960479	262	395600325	284	987874095
220	50651799	242	157402781	264	431894257	286	1068507786
222	56463948	244	173577766	266	470256442	288	1156161305
224	62887775	246	190809628	268	512858451	290	1247686188
226	69995887	248	209715141	270	557745669	292	1348832359
228	77831323	250	230272559	272	606668511	294	1454359796
230	86238206	252	252745513	274	659140287	296	1568768503
232	95758929	254	276599784	276	716217919	298	1690214821
234	105965373	256	303235792	278	776165187	300	1821766836

Since the numbers of IPR fullerenes have not previously been published beyond 214 atoms [7], we provide this information in Table 1.

4. Concluding remarks

Our algorithms can be used in conjunction with the method of McKay [18] to produce a generator of non-isomorphic fullerenes. Briefly the method works as follows. For each fullerene *G*, one expansion is attempted from each equivalence class of expansions under the symmetry group of *G*. If the new larger fullerene is *H*, then *H* is accepted if the reduction inverse to the expansion by which *H* was constructed is equivalent under the symmetry group of *H* to a 'canonical' reduction of *H*; otherwise it is rejected. The essential algorithmic requirements are computation of symmetry groups and canonical labelling, which are both easy to do in linear time using a depth-first search starting at the pentagons. In addition, since there are only 12 pentagons, the number of reductions that can apply to any fullerene is bounded. By [18] Theorem 3, this means that the set of all fullerene isomers of at most *n* atoms, without isomorphs, can be found in $O(n)$ time per isomer. See Brinkmann and McKay [19] for further description and examples of this technique. Finally, we wish to thank Gunnar Brinkmann for helpful advice.

References

[1] D.E. Manolopoulos, J.C. May, S.E. Down, Chem. Phys. Lett. 181 (1991) 105.
[2] D.E. Manolopoulos, P.W. Fowler, Chem. Phys. Lett. 204 (1993) 1.
[3] P.W. Fowler, T. Pisanski, A. Graovac, J. Zerovnik, DIMACS Discrete Math. Theoretical Comput. Sci. 51 (2000) 175.
[4] P.W. Fowler, D. Horspool, W. Myrvold, Chem. Europ. J. 13 (2007) 2208.
[5] J.E. Graver, SIAM J. Disc. Math. 17 (2004) 596.
[6] M. Yoshida, E. Ōsawa, Bull. Chem. Soc. Jpn. 68 (1995) 2073.
[7] G. Brinkmann, A.W.M. Dress, J. Algorithms 23 (1997) 345. Program at <http://cs.anu.edu.au/~bdm/plantri>.
[8] G. Brinkmann, O. Delgado Friedrichs, A. Dress, T. Harmuth, MATCH Commun. Math. Comput. Chem. 36 (1997) 233. Program at <http://www.mathematik.uni-bielefeld.de/~CaGe>.
[9] G. Brinkmann, P.W. Fowler, J. Chem. Inf. Comput. Sci. 43 (2003) 1837.
[10] X. Liu, D.J. Klein, T.G. Schmalz, W.A. Seitz, J. Comp. Chem. 12 (1991) 1252.
[11] Y. Xia, Y. Mu, Y. Xing, R. Wang, C. Tan, L. Mei, Phys. Rev. B 57 (1998) 14950.
[12] Yu.E. Lozovik, A.M. Popov, Phys.–Uspekhi 40 (1997) 717.
[13] G. Brinkmann, J.E. Graver, C. Justus, J. Math. Chem., in press, doi:10.1007/s10910-008-9403-6. ISSN 0259-9791 (Print) 1572-8897 (Online).
[14] G. Brinkmann, D. Franceus, P.W. Fowler, J.E. Graver, Chem. Phys. Lett. 428 (2006) 386.
[15] P.W. Fowler, D. Manolopoulos, An Atlas of Fullerenes, Oxford University Press, 1995.
[16] F. Kardoš, R. Škrekovaski, J. Math. Chem. 44 (2008) 121.
[17] T. Došlić, J. Math. Chem. 33 (2003) 103.
[18] B.D. McKay, J. Algorithms 26 (1998) 306.
[19] G. Brinkmann, B.D. McKay, MATCH Commun. Math. Comput. Chem. 58 (2007) 323.