communications in mathematical
match
and in computer chemistry

no. 36, October 1997

ISSN 0340-6253

MATCDY (36) 233-237 (1997)

CaGe – a Virtual Environment for Studying Some Special Classes of Large Molecules

Gunnar Brinkmann, Olaf Delgado Friedrichs, Andreas Dress and Thomas Harmuth
Fakultät für Mathematik
Universität Bielefeld
D 33501 Bielefeld Germany
Email: gunnar(delgado,dress,harmuth)@mathematik.uni-bielefeld.de

Since the DENDRAL project [1] started in 1965, computer generation of chemical structures has become an important tool in chemical research, and it is applied for structure elucidation in science as well as in industry. Of course not only computers, but algorithms and programs too, have evolved since then, and nowadays the most powerful structure generator appears to be MOLGEN [2].

At present, efficient generation programs like MOLGEN can generate tens of thousands of structures per second. Nevertheless, not every task can be solved by general purpose programs. A standard example is the task of generating all those 3-valent molecular structures with 60 atoms which can be *embedded* into the sphere in such a way that all faces are pentagons or hexagons and no two pentagons share an edge. It is known that there is exactly one such structure, namely the structure of the famous *buckyball* C_{60}; yet, generating all 3-valent structures and filtering them for those with the required properties would require millions of years even with the most efficient structure generation programs. Another example is $C_{10}H_8$. For this compound, there are 488 125 isomers. Yet, if we knew that we are searching for a planar polycyclic hydrocarbon, there is just one possibility. In this case, generating all possibilities and filtering for the suitable ones would still be possible. Yet, for planar polycyclic $C_{400}H_{50}$, it would be completely impossible – in spite of the fact that there are "only" 12 683 planar polycyclic isomers. Consequently, additional requirements need to be included already into the *generation* process in such cases. So, we need algorithms developed especially for the particular class of structures in question.

CaGe has been designed to provide a user interface for some such programs. All of these programs deal with *planar* structures – that is with structures than can be *embedded* into the 2-dimensional sphere. At present, CaGe comprises four generation programs:

Fullerenes: Here the user can choose the number of atoms and whether he wants all fullerenes or only the IPR-fullerenes (that is, fullerenes that satisfy the *isolated-pentagon rule*) to be generated. The generation can also be restricted to certain symmetry groups – though at the moment this is only implemented as a filter, so it is very inefficient for most symmetry groups.

Nanotubes: Running around a cap of a nanotube and following this path in the hexagonal lattice, we get a vector that is characteristic for the tube. For the nanotube program, the two components of this vector – called *perimeter* and *shift* – and the length of the tube are required as input. Furthermore, the user can optionally restrict the program to generate only IPR-tubes or to generate only halftubes with a pregiven number of hexagonal rings.

Hydrocarbons: Here the number of C atoms, H atoms and pentagons (at most 5) must be given. The program generates all planar polycyclic hydrocarbons corresponding to these numbers, with all non-pentagons being hexagons. The user can also restrict the program to structures obeying the isolated-pentagon rule, strictly peri-condensed structures or structures with an upper bound for the number of successive carbon atoms in the boundary without a hydrogen bond.

Cubic planar graphs: For this program, the user must give the number of atoms and the face sizes that are allowed. The generation can also be restricted to 1- 2- or 3-connected graphs, but this is implemented as a filter only. Optionally, the dual structures (triangulations) are generated.

The structures generated by these programs can be visualized in 3 dimensions or in 2 dimensions (as Schlegel diagrams) and can be written to a file or to a pipe, so that other programs can use the structures for additional analyses. Examples for windows displaying the 2D and 3D output are given in Figures 1(a) and (b). To display the 3D output, we use the program *Rasmol* [3], a public domain program for visualizing molecular structure.

For these visualizations, we took advantage of the specific properties of our structures, which allowed us to develop a very efficient and accurate embedding routine. Consequently, the resulting embeddings can be used not only for visualization, but also as a starting point for molecular dynamics programs.

Two principal requirements have guided the design and implementation of CaGe: ease of usage and fastness of responses. Though for advanced use of the program (e.g. piping structures into programs developed by the user) it is necessary to read some pages of the manual, it is also possible to use the program by just following the instructions in the windows that are displayed on the screen.

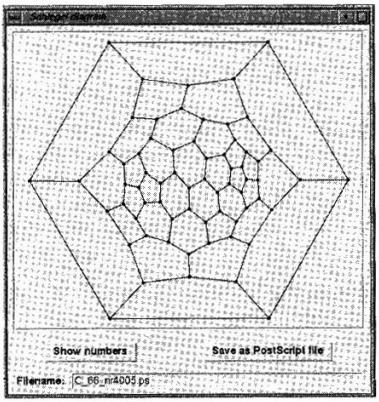
The development of *CaGe* is still at its beginning, and improvements regarding the generation and embedding programs as well as the user interface are in progress. Nevertheless, first versions can be obtained by E-mail from the authors. CaGe is free for scientific use and can be installed on any Unix platform. The authors will be grateful for any hints and suggestions as to how to improve the program.

formula	structures	CPU Linux Pentium 133 Mhz
C_{20}	1	0 sec
C_{30}	3	0 sec
C_{40}	40	0.2 sec
C_{50}	271	2 sec
C_{60}	1 812	14.1 sec
C_{60} ipr	1	0.3 sec
C_{70}	8 149	70 sec
C_{70} ipr	1	1.4 sec
C_{80}	31 924	284 sec
C_{80} ipr	7	5.8 sec
C_{90}	99 918	961 sec
C_{90} ipr	46	20 sec
C_{100}	285 914	3 017 sec
C_{100} ipr	450	63 sec
C_{110} ipr	2 355	189 sec
C_{120} ipr	10 774	558 sec
C_{130} ipr	39 393	1 550 sec

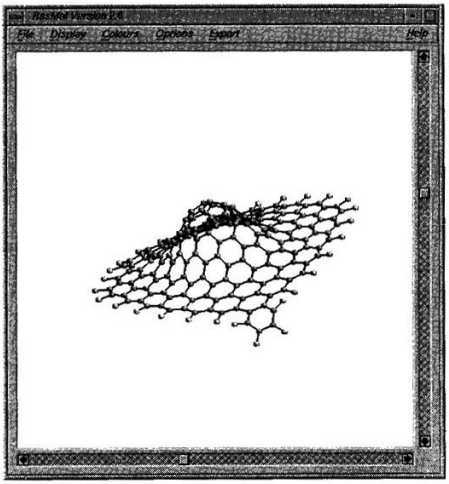
Table 1: Sample running times for fullerenes

formula, pentagons	structures	CPU Linux Pentium 133 Mhz
$C_{20}H_{12}$, 2	56	0 sec
$C_{26}H_{12}$, 4	1 553	0.3 sec
$C_{28}H_{16}$, 1	570	0.1 sec
$C_{30}H_{14}$, 1	111	0.1 sec
$C_{30}H_{16}$, 1	892	0.1 sec
$C_{30}H_{16}$, 2	4 950	0.7 sec
$C_{34}H_{18}$, 2	31 444	4.1 sec
$C_{40}H_{22}$, 1	69 163	8.8 sec
$C_{50}H_{16}$, 2	1 913	7.4 sec
$C_{60}H_{20}$, 1	15 946	31 sec
$C_{60}H_{32}$, 2	392 153 165	33.9 h
$C_{79}H_{19}$, 2	852	308 sec
$C_{150}H_{28}$, 1	120	2.3 sec
$C_{180}H_{16}$, 5	17 661	73 sec
$C_{303}H_{31}$, 3	2 599	94 sec
$C_{400}H_{50}$, 0	12 683	7.5 sec
$C_{401}H_{41}$, 2	31 676	1 917 sec

Table 2: Sample running times for planar polycyclic hydrocarbons


Show numbers
Save as PostScript file
Filename:
C_66_nr4005.ps

(a)


File
Display
Colours
Options
Export
Help

(b)

Figure 1

References

[1] R.K. Lindsay, B.G. Buchanan, E.A. Feigenbaum, and J. Lederberg. *Artificial Intelligence for Organic Chemistry: The Dendral Project.* McGraw-Hill, New York, 1980.

[2] R. Grund, A. Kerber, and R. Laue. MOLGEN ein Computeralgebrasystem für die Konstruktion molekularer Graphen. *match 27*, pages 87–131, 1992.

[3] R. Sayle. Rasmol – a molecular visualisation program http://www.umass.edu/microbio/rasmol.