The Graph Grammar Library - a generic framework for chemical graph rewrite systems

Martin Mann[1], Heinz Ekker[2], and Christoph Flamm[2]

[1] Bioinformatics, Institut for Computer Science, University of Freiburg,
Georges-Köhler-Allee 106, 79106 Freiburg, Germany,
mmann@informatik.uni-freiburg.de
[2] Institute for Theoretical Chemistry, University of Vienna, Währingerstrasse 17,
1090 Vienna, Austria, xtof@tbi.univie.ac.at

Graph rewrite systems are powerful tools to model and study complex problems in various fields of research [7]. Their successful application to chemical reaction modelling on a molecular level was shown [1, 2, 6] but no appropriate and simple system is available at the moment [8]. The Graph Grammar Library (GGL), presented in this contribution and more extensively in [4], fills this gap and provides feature-rich functionality especially for chemical transformation.

The GGL implements a simple generic Double Push Out approach for general graph rewrite systems [7] on labeled undirected graphs. The object oriented C++ framework focuses on a high level of modularity as well as high performance, using state-of-the-art algorithms and data structures, and comes with extensive end user and API documentation. Central modules (e.g. graph matching, match handling, graph storage) are combined via simple interfaces, which enables an easy combining to tackle the problem at hand.

The large GGL chemistry module enables extensive and detailed studies of chemical systems. It well meets the requirements and abilities envisioned by Yadav et al. [8] for such chemical rewrite systems. Here, molecules are represented as vertex and edge labeled undirected graphs while chemical reactions are described by according graph grammar rules, see Fig. 1. Such a graph grammar is a generating system for the explicit construction of an entire chemical space,

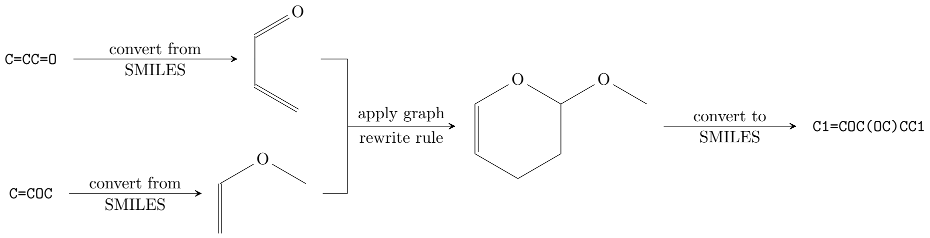

Fig. 1. Illustration of the basic steps to convert the educt molecules acolein and methyl vinyl ether via a Diels-Alder reaction [4] to the cyclic product molecule. Physico-chemical properties for the molecules, such as free energies (ΔG), or for the reaction, e.g. reaction rates, can be estimated either by using GGL built-in functionality or via calls to the OpenBabel chemistry toolkit [5].

i.e. all molecules reachable from the initial molecules by iterative reaction applications. An extensive system of wildcards, degree and adjacency constraints, and negative application conditions (NAC), such as the non-existence of edges, makes it easy to formulate very specific graph transformation rules by modulating their context dependent matching behaviour. Rules are encoded using the Graph Modelling Language (GML) easily understood and used by non-expert users. The molecule graphs produced by the graph grammar encoded chemical reactions have to pass extensive sanity checks and e.g. arromaticity correction to ensure the production of proper molecules only.

Besides the efficient handling of chemical transformation the GGL offers advanced cheminformatics algorithms. Among them are methods for the estimation of reaction rates or the free energies of molecules, the generation of canonical SMILES (a popular line notation for molecules) or chemical ring or aromaticity perception. Furthermore the entire functionality of the popular chemical toolbox Open Babel [5] can be harnessed from within the GGL via the implementation of a bi-directional interface for the exchange of chemical graphs. All these features are used within the GGL-based `toyChem` tool part of the library that enables the expansion and visualization of reaction networks given some initial molecules and a set of chemical reaction rewrite rules.

The graph grammar based simulation of chemical reactions offered by the GGL is a powerful tool for extensive cheminformatics studies on a molecular level and it already provides rewrite rules for all enzymes listed in the KEGG LIGAND database [3]. The GGL is freely available at

http://www.tbi.univie.ac.at/software/GGL

For a full description of all GGL features please refer to [4] available at http://arxiv.org/abs/1304.1356

References

1. Benkö, G., Flamm, C., Stadler, P.F.: A graph-based toy model of chemistry. J Chem. Inf. and Comp. Sci. 43(4), 1085–1093 (2003)
2. Flamm, C., Ullrich, A., Ekker, H., Mann, M., Hoegerl, D., Rohrschneider, M., Sauer, S., Scheuermann, G., Klemm, K., Hofacker, I.L., Stadler, P.F.: Evolution of metabolic networks: A computational framework. J Syst. Chem. 1(1), 4 (2010)
3. Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., Tanabe, M.: KEGG for integration and interpretation of large-scale molecular data sets. Nuc. Acids Res. (2011)
4. Mann, M., Ekker, H., Flamm, C.: The graph grammar library - a generic framework for chemical graph rewrite systems. arXiv http://arxiv.org/abs/1304.1356 (2013)
5. O'Boyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T., Hutchison, G.R.: Open Babel: An open chemical toolbox. J. Cheminf. 3(1), 33+ (2011)
6. Rosselló, F., Valiente, G.: Chemical graphs, chemical reaction graphs, and chemical graph transformation. Electron. Notes Theor. Comput. Sci. 127, 157–166 (2005)
7. Rozenberg, G. (ed.): Handbook of graph grammars and computing by graph transformation: volume I. foundations. World Scientific Publishing Co., Inc. (1997)
8. Yadav, M.K., Kelley, B.P., Silverman, S.M.: The potential of a chemical graph transformation system. In: ICGT. LNCS, vol. 3256, pp. 83–95. Springer (2004)