Josef Leydold
Near Graph Products

1 Individual Project's contribution to the CRP

1.1 Aims and Objectives

Graphs have great importance in modeling networks and discrete structures (which themselves may be models for physical or social phenomena), see the project aims in T5. Real world networks, however, often are quite large (e.g., the "internet" to cite the most prominent example). Visualization and analysis of such graphs is a challenging task.

There exists a strong evidence ([6]) that *product graphs* can be a valuable tool for this purpose. Roughly spoken, given a graph G, the aim is to provide several smaller graphs and a rule for the construction of a graph H that is either isomorphic with G or that approximates G, i.e., that is "near" to G. Thus it is possible to describe the geometry of such structures by smaller and often also simpler graphs. A typical example would be the representation of G by the Cartesian product H of several small graphs, with the additional information that G is obtained from H by deletion and/or insertion of several specified edges; see PI Pisanski [7].

This decomposition may, e.g., be essential for graph drawing (see T5) for finding a good presentation of the entire structure, of parts or substructures, or superstructures. In molecular biology this information may be used to gain more insight in the mechanisms of evolution. It can help to detect hierarchies in social networks. It is the aim of this project to provide algorithms that yield the desired structural information.

Clearly this includes the recognition of graph products and similar structures. The recognition of graph products is well advanced, see the forthcoming book by AP Imrich, PI Klavžar, and Richard Hammack [5]. It contains polynomial algorithms for the recognition of many types of products of graphs. Some of the algorithms are very fast. For example, Cartesian products can be recognized in linear time. For other products there exist quasilinear algorithms for some applications that are important for T5 [3], whereas the recognition of strong and direct products is still polynomial of high complexity.

For directed graphs, hypergraphs and similar structures the situation pertaining to their algorithmic recognition is still wide open, despite some early steps, see AP Imrich & PI Stadler [4]. Hence research in the area of fast algorithms for the decomposition of structures with respect to various products is still an objective and will partially be driven by the needs of applications in T5.

If a graph is not a product, but reasonably close to it (small edit distance), it is often called approximate. Recently, algorithms that recognize approximate strong products have been proposed, see [3]. For other products – in particular for the Cartesian and direct product – the field is wide open and there is good chance that the same idea will work equally well. This would enhance the visualization of small graphs (see [7]) as and help for some tasks in T5.

If one has reason to assume that a graph originally had a product structure, but that its representation was significantly disturbed or incomplete (e.g., by some applications in T5) different methods are needed. One might think of imposing a product structure on the given graph, be it for visualization or for the recognition of hidden structural properties. Graphs that we will represent in such a way will still be called *near products*, however. In such cases the type of product may also be unclear originally and may have to be chosen by a reasonable selection process. Once the type as been determined, a representation algorithm will be needed. For this purpose global properties can be used. This includes convexity, numbers of shortest paths between pairs of vertices, diameter, and densities, which were not used for the recognition of products and approximate graph products as described above.

The aim of T2 is the provision of one or more product structures to a given graph that capture essential features. Moreover, we also provide methods that measure of the quality of

the fit of a particular product structure to the given graph. It is intended to implement these algorithms using C/C++ and the boost library (`http://www.boost.org/`) and make them thus available for partner of the CRP as well as to other interested research groups. (Many special purpose computing environments have an interface to load functions written in C/C++.)

1.2 Methodologies

The research will be conducted in close relation to PI Bıyıgoğlu and in particular with AP Imrich from Leoben. He is a renowned specialist in the first two of the methods described below and well versed in the remaining ones. So there is essential overlap with the proposal of AP Imrich.

Exact methods. Implementation and development of algorithms for the recognition of products of graphs. Algorithms for the Cartesian, the strong and the direct product of graphs are collected in the forthcoming book [5]. Some proof of concept implementation (of parts) of the algorithms exist. But they are not complete and need a unified treatment for an implementation using the boost library (or any other computational environment).

Algorithms for several new classes of graphs or new types of products may have to be designed upon request from Part T5. This may be routine or very difficult, even if standard methods are applicable, see, e.g., [2].

Approximate methods. Satisfactory algorithms for approximate strong products have already been developed [3] and implemented for the boost library. It is based on the fact that neighborhoods are subproducts. We expect that it also works for the direct product. For the Cartesian product nothing of the kind exists. We plan to apply a similar approach based on intervals instead of neighborhoods. This looks very promising, but will involve many computational details.

Heuristic methods. For the recognition of near products entirely new methods are needed. We expect to be able to use the fact that layers with respect to the Cartesian product are convex and that vertices in one and the same layer are have minimal numbers of paths between them, for an intelligent guess on how to decompose the given graph. For the strong product similar methods may work, but here the problem is more complex.

Spectral methods. Eigenvalues and vectors of adjacency and Laplacian matrices have been used successfully in heuristics for various problems like graph partitioning, coloring, clustering, graph drawing, and others, see, e.g., Bıyıkoğlu et al. [1]. Spectral plots can also help to detect local symmetric structures in (large) graphs. It is also known that factors of certain Cartesian products can be determined by the first eigenvectors of its Laplacian matrix. Thus we want to investigate the feasibility of spectral methods for finding approximate products.

Statistical methods. In order to make an intelligent guess which product to use we plan to rely on statistical methods. We wish to use the degree distribution, betweenness, diameter and other properties for the selection, see Leskovec et al. [6]. Thus we will make use of advanced statistical methods for which we can use additional expertise that is available at the Institute for Statistics and Mathematics.

Model fitting and quality control. We need methods to estimate the parameters for a particular product structure as well as methods that measures the fit of this model for a given graph. Again statistical methods will be needed. Here we may also may use entropy considerations to choose between different representations.

References

[1] T. Bıyıkoğlu, J. Leydold, and P. F. Stadler. *Laplacian Eigenvectors of Graphs: Perron-Frobenius and Faber-Krahn Type Theorems*, volume 1915 of *Lecture Notes in Mathematics*. Springer Verlag, Heidelberg, 2007.

[2] R. H. Hammack and W. Imrich. On cartesian skeletons of graphs. *Ars Math. Contemp.*, 2: 191–205, 2009.

[3] M. Hellmuth, W. Imrich, W. Klöckl, and P. F. Stadler. Approximate graph products. *European J. Combin.*, 30:1119–1133, 2009.

[4] W. Imrich and P. F. Stadler. A prime factor theorem for a generalized direct product. *Discuss. Math. Graph Theory*, 26:135–140, 2006.

[5] W. Imrich, S. Klavžar, and R. Hammack. *Product Graphs*. Discrete Mathematics and Its Applications. CRC Press, 2nd edition, 2011.

[6] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani. Kronecker graphs: An approach to modeling networks, 2006. `arXiv:0812.4905v2 [stat.ML]`.

[7] T. Pisanski and J. Shawe-Taylor. Characterising graph drawing with eigenvectors. *J. Chem. Inf. Comput. Sci.*, 40:567–571, 2000.