A System for Graph Clustering based on User Hints[1]

Hugo A. D. do Nascimento[2]
Department of Computer Science and S.E.
University of Newcastle, Australia
hadn@cs.usyd.edu.au

Peter Eades[3]
Department of Computer Science
University of Sydney, Australia
peter@cs.usyd.edu.au

Abstract

This paper presents a system for graph clustering where users can visualize the clustering and give "hints" that help a computing method to find better solutions. Hints include a variety of constraints, as well as direct manipulation of the previously computed clustering. The framework is flexible: it can accommodate several kinds of hints, clustering algorithms, and visualization techniques.

Keywords: clustering, user interaction, graph drawing, hints.

Introduction

Graph clustering is an important and difficult optimization problem that arises in Software Engineering (Rayside, Reuss, Hedges and Kontogiannis 2000, Koschke and Eisenbarth 2000, Tzerpos and Holt 2000), VLSI design (Alpert and Kahng 1995), Distributed and Parallel Processing (Kumar, Grama, Gupta and Karypis 1994), and in many other areas. We have developed a system (called HINTS) for graph clustering where users play a strong role. The system follows a framework where a soft computing method for clustering is driven by user's hints. Suggestions provided by the user include a variety of constraints for solutions, as well as direct manipulation of the previously computed clustering. The main resources of our system are:

- Simulated Annealing and hill-climbing algorithms for clustering with support for several constraints;
- a visualization module that shows drawings and various measures of the current clustering;
- an interface for giving hints. Here, we focus on setting constraints (e.g. defining limits on cluster size, on the numbers of clusters, and balance). The user can activate or deactivate these constraints as well as set their relative importance.

HINTS uses soft computing methods like hill-climbing and Simulated Annealing for two reason: (1) they are flexible enough to deal with many constraints, and (2) it allows iterative changes on the solution (which is necessary for user interaction).

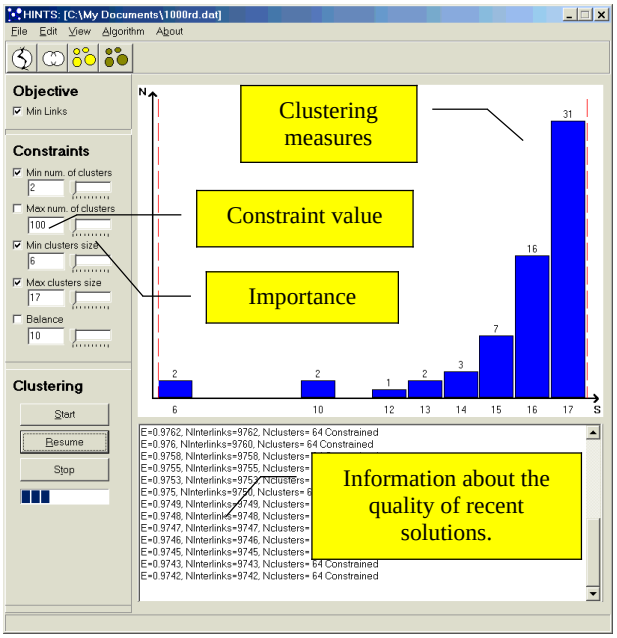
Visualizations Tools

In the system, the clustering algorithms attempt to minimize a weighted function of the constraints and the objective function, representing the quality of a clustering. During the processing the system keeps track of the best solution, which is drawn on the screen using some graph drawing techniques and statistic graphics. The visualization of this solution occurs through three main views. The first is a histogram that shows the number of clusters for each cluster size. The second is a scatter-plot style graphic that shows the size of each cluster. The third is a graph drawing that shows the current clustering. Constraints are also indicated in the views by drawing extra lines or by using different colors. Moreover, the general evolution of the clustering algorithms is shown by continuously updating the visualization. If the screen is not updated for a long time, this means that the system may have fallen into a local minimum and needs user intervention. See a snapshot of the system's interface on the picture below.

---

[2] Lecturer of the Instituto de Informática, UFG-Brazil. Supported by CAPES-Brazil and UFG.

[3] Supported by the Australian Research Council.



Picture 1: HINTS Interface

These visual features offer feedback to the user, so that one can analyse the clustering and identify changes that can drive the optimisation algorithms toward better solutions.

Giving Hints

The system accepts user hints and automatically incorporates them into the optimisation processing. The main types of hints are:

- Constraint modification – the user changes the value of a constraint in order to make it loose or tight, thus helping the system to escape from local minimums. It is also possible to reset the importance of each constraint in order to focus the search on a specific quality aspect (e.g., good balance).
- Manual changes on clusters – the user manually changes the current solution by executing some simple operations like breaking an existing cluster and regrouping its vertices in a different way.
- Algorithm selection – the user can change the clustering algorithm (between a hill-climbing method and a Simulated Annealing in the current version of the system), whenever it gets stuck in a local minimum or it is improving very slowly the current solution.

These hints can be given at any time, as the user sees that part of the clustering has a poor quality or that the clustering method got stuck. Then by giving hints to the system, it is possible to work on those aspects.

Observe that it may also be the case that the users do not know precisely about all constraints to a particular problem at the beginning of the systems' execution; therefore, they will have to dynamically adjust them later on. In this situation we say the hints mechanism also provides a way of adding domain knowledge to the system.

Conclusion

In our system the user plays a strong role by giving hints to a soft computing method for Graph Clustering. Visualization techniques offer feedback to the user about the quality of the clustering as well as some indication of the algorithms' efficiency. This feedback allows the user to decide which type of hints to give next to the system.

There are a number of other hints that have not been included yet into the system such as: defining constraints that group some vertices into a same cluster. However, the currently available facilities already show a very promising path for allowing the user to become a strong part of the optimisation process.

HINTS is one of several attempts to test the hypothesis that a human can assist an optimisation method. Some similar approaches on this direction are described by Andersen *et al.* (2000) and by Lesh, Marks and Patrigname (2000), and they add weight to this hypothesis.

References

ALPERT C. J. and KAHNG A. B. (1995): Recent directions in netlist partitioning: a survey. *Integration: The VLSI Journal*, (19):1-81.

ANDERSEN D., ANDERSEN M., LESH M., MAKS J., MIRTICH B., RATAJCZAC D. and RYALL K (2000): Human guided simple search. To appear in the proceedings of the annual conference of the American Association for Artificial Intelligent.

KOSCHKE R. and EISENBARTH T. (2000): A framework for experimental evaluation of clustering techniques. *Proc. of 8th International Workshop on Program Comprehension*, pages 201-210, IEEE.

KUMAR V., GRAMA A., GUPTA A. and KARYPIS G. (1994): *Introduction to Parallel Computing*. The Bejamin/Cummings Publishing Company, Inc.

LESH N., MARKS J. and PATRIGNAME M. (2000): Interactive Partitioning. Graph Drawing Conference.

RAYSIDE D., REUSS S., HEDGES E. and KONTOGIANNIS K. (2000): The effect of call graph construction algorithms for object-oriented programs on automatic clustering. *Proc. of 8th International Workshop on Program Comprehension*, pages 191-200, IEEE.

TZERPOS V. and HOLT R. C. (2000): On the stability of software clustering algorithms. *Proc. of 8th International Workshop on Program Comprehension*, pages 211-218, IEEE.