STUDY ON DIVIDE-AND-CONQUER LOCAL SEARCH HEURISTIC

DAVENDRA KRISHAN

ABSTRACT

We tested our algorithms on artificially generated data sets. We applied our ***algorithm to several data sets with 50, 100, or 150 points. We set t = 100. The data sets w****ere randomly generated from a lattice set of 256 points in two dimensions. For each problem size, 10 different problems were generated. To evaluate different versions of our local search heuristic, we used problems generated from a two-dimensional lattice, as it is easy to compare the quality of the computational results when the original set of points is in two dimensions. In this case, the optimal objective function value is known and equal to zero. Initially, we selected the neighborhood of a point i M, assigned to k N, to be all points in N. That is, every lattice point k N is considered as a possible choice for assigning i M. we show the results for the experiments for this local search heuristic (LS). The frequency column gives the number of solutions (out of 100) that converged to the best solution. In nine of the 10 problems of size 50, LS finds the global*

Key words; *search heuristic, two-dimensional lattice, dimensions.*

INTRODUCTION

In today's business environment, transaction processing with the aid of computers and the use of information technologies such as barcode scanners generate huge volumes of data in operations ranging from retailing to banking to stock trading (Mackinnon and Glick, 1999). Many companies and organizations gather gigabytes or terabytes of business transactions, scientific data, web logs, satellite pictures, and text reports, which are large and complex (Morzy and Zakrzewicz, 2003). In essence, massive databases growing at unprecedented rates are indeed very common today.

Inherent in such data are important insights into the operations they represent. Businesses want to mine retail data to know how to acquire, retain, and increase the profitability and lifetime value of a customer (Cabena et al., 1997). Researchers are developing the tools to mine available data to discover knowledge that facilitate activities such as market research, fraud detection and prevention, the pricing of securities and derivatives, as well as the monitoring of the medical impacts of prescription drugs (Mackinnon and Glick, 1999). Consequently, data mining has become difficult to ignore and hence an area of intense research.

(IJAER) 2013, Vol. No. 5, Issue No. I, January **ISSN: 2231-5152**

Data mining involves the extraction of hidden predictive information from large databases. It is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Data mining tools help predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. Data mining tools can answer business questions that traditionally were too time consuming to resolve. For instance, the type and number of all products in a customer's shopping basket can be recorded and examined, giving insight into the customer's behavior. This enables the shop to draw conclusions for the shop's presentation of its products (Morzy and Zakrzewicz, 2003).

REVIEW OF LITERATURE

Data mining is an interdisciplinary field and utilizes techniques and tools from fields such as machine learning, pattern recognition, statistics, database, and visualization, to address the issue of information extraction or knowledge discovery from complex databases (Cabena et al., 1997; Mackinnon and Glick, 1999). Before the advent of data mining, researchers focused on problems with data sizes that were at most a few hundred to a few thousand cases and had between one and a few dozen variables (Elder and Pregido, 1996). The field emerged when it was realized that traditional decision-support methodologies, which combine simple statistical techniques with executive information systems, could not handle large and complex data sets within the time limits and operational conditions imposed by today's business environment (Cabena et al., 1997). Enterprises must be able to recognize trends early in rapidly changing

MATERIAL AND METHOD

We tested our algorithms on artificially generated data sets. We applied our **algorithm to several data sets with 50, 100, or 150 points. We set t = 100. The data sets w**ere randomly generated from a lattice set of 256 points in two dimensions. For each problem size, 10 different problems were generated. To evaluate different versions of our local search heuristic, we used problems generated from a two-dimensional lattice, as it is easy to compare the quality of the computational results when the original set of points is in two dimensions. In this case, the optimal objective function value is known and equal to zero.

Initially, we selected the neighborhood of a point i M, assigned to k N, to be all points in N. That is, every lattice point k N is considered as a possible choice for assigning i M.

In Table 4.2, we show the results for the experiments for this local search heuristic (LS). The frequency column gives the number of solutions (out of 100) that converged to the best solution. In nine of the 10 problems of size 50, LS finds the global optimum. For the 100-point and 150-point problems, LS finds the global optimum in all 20 problems. The average running times are 82.90 seconds, 408.00 seconds, and 797.87 seconds for the 50-point, 100-point, and 150-point problems, respectively. LS allows points to be assigned to all points in the lattice structure. In this case we consider 256 lattice points. As the size of M increased, the running time for LS increased substantially.

To solve the problem of large running times with LS, we propose a divide-and-conquer heuristic to reduce the size of the problem that is solved at each stage. Our divide-and-conquer heuristic has five steps.

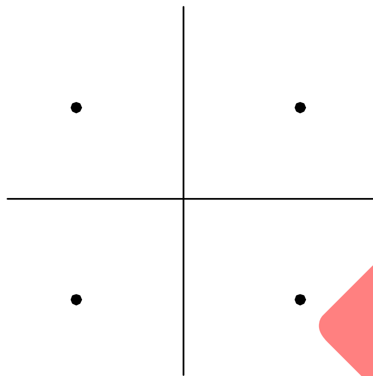

Figure 1 Lattice with 16 points after the four initial points have been subdivided into four additional points each.

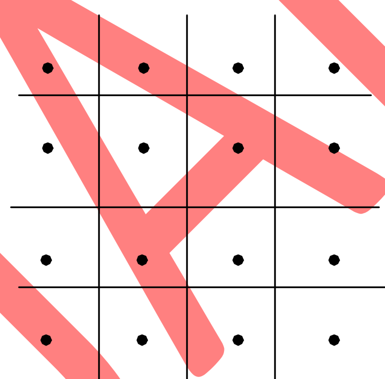

Figure 2

1. Start with a lattice of four points (see Figure 1).

2.Perform local search (as described above) on points in M using these four points; that is, points in M can be assigned to only these four lattice points. When local search terminates, the solutions that have been generated have points divided into four quadrants.
3. Divide each quadrant into four points (see Figure 2).
4. Randomly assign points from each quadrant from the previous assignment to one of the four new points. These solutions are the starting solutions for local search and local search is performed using the new lattice structure; that is, points can be assigned to any of the points in the current lattice

structure in local search.

5. Continue dividing each point into four points and repeat the previous step until a stopping rule is met. We stop at 256 points, unless otherwise specified.

For the divide-and-conquer heuristic, the initial feasible solutions are generated randomly taking into consideration the symmetry elimination constraints. That is, for the initial step of our local search procedure where there are only four lattice points, the symmetry constraints are taken into account and the first point is always assigned to the first lattice point.

We apply our divide-and-conquer local search heuristic (DAC) to the same problem sets we used to test LS. In Table 4.3, we show the results for the experiments for DAC. For all 30 problems, DAC finds the global optimum. In two of the ten 100-point problems and six of the ten 150-point problems, all solutions generated by DAC were optimal (that is, the frequencies were 100%). The average running times for the 50-point problems, the 100-point problems, and the 150-point problems are 41.24 seconds, 96.91 seconds, and 163.14 seconds, respectively.

In seven of 10 problems for the 50-point problems, DAC finds the optimal solution more times than LS. For the 100-point problems, DAC finds the optimal solution more times in nine of the 10 problems. In all 150-point problems, DAC finds the optimal solution with a higher frequency than LS. Also, in the few problems for which LS has a Higher frequency, the difference in the frequency of optimal solutions generated was relatively small. For example, consider problem 4 in the 100-point problems. LS found the optimal solution seven times while DAC finds the optimal solution two times.

CONCLUSION

Our experiments indicate that DAC generates better results with much smaller running times than LS. Furthermore, DAC finds the optimal solution with a greater frequency than LS. Thus, the probability that DAC finds the optimal solution is likely to be greater than for than LS, especially if the procedure involves fewer starting solutions. However, for a 150-point problem with a 256-point lattice, DAC has an average running time of 163 seconds. For larger data sets with more lattice points, the running time may become very large. This is because DAC allows points to be assigned to any of the lattice

REFERENCES

E. Aarts and J. K. Lenstra (editors), Local Search in Combinatorial Optimization, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, West Sussex, England, 1997

W. P. Adams and T. Johnson. Improved linear programming based lower bounds for the quadratic assignment problem, in Quadratic Assignment and Related Problems, P. Pardalos and H. Wolkowicz (editors), DIMACS Series in Discrete Mathematics and Theoretical Computer Science 16, 43 - 75, 1994.

R. K. Ahuja, J. B. Orlin, and A. Tiwari, A greedy genetic algorithm for the quadratic assignment problem, Computers & Operations Research 27, 917 - 934, 2000.

M. Berry and G. Linoff, Mastering Data Mining, John Wiley & Sons, New York, 2000.

M. Berry and G. Linoff, Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, John Wiley & Sons, New York, 2004.

M. S. Bazaraa and H. D. Sherali, Benders' partitioning scheme applied to a new formulation of the quadratic assignment problem, Naval Research Logistics Quarterly 27, 29 - 41,1980.

I. Borg and P. Groenen, Modern Multidimensional Scaling: Theory and Applications, Springer, New York, 1997.

R.E. Burkard and F. Rendl, A thermodynamically motivated simulation procedure for combinatorial optimization problems, European Journal of Operational Research 17, 169 - 174, 1984.

R.E. Burkard, Quadratic assignment problems, European Journal of Operational Research 15, 283 - 289, 1984.

P. Cabena, P. Hadjinian, R. Stadler, J. Verhees, and A. Zanasi, Discovering Data Mining: From Concept to Implementation, Prentice Hall, Upper Saddle River, New Jersey, 1997.

E. Cela, The Quadratic Assignment Problem: Theory and Algorithms, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

N. Christofides and E. Benavent, An exact algorithm for the quadratic assignment problem, Oper. Res. 37, 760 - 768, 1989.

E. Condon, B. Golden, S. Lele, S. Raghavan, and E. Wasil, "A visualization method based on adjacency data," Decision Support Systems 33, 349 - 362, 2002.

E. Condon, B. Golden, and E. Wasil, "Visualizing group decisions in the analytic hierarchy process," Computers & Operations Research 30, 1435 - 1445, 2003.

K.C. Cox, S. G. Eick, G. J. Wills, and R. J. Brachman, "Visual data mining: recognizing telephone calling fraud," Data Mining and Knowledge Discovery 1, 225 - 231, 1997.

L. Davis, Handbook of Genetic Algorithm, Van Nostrand Rienhold, New York, 1991.

G. Deboeck and T. Kohonen (editors), Visual Explorations in Finance, Springer, London, 1998.

U. Dorndorf and E. Pesch. Data Warehouses, in Handbook on Data Management in Information Systems, J. Blazewicz, W. Kubiak, T. Morzy, and M. Rusinkiewicz (editors), International Handbooks on Information Systems, 387 - 430, Springer, New York, 2003.

J. Elder and D. Pregibon, A statistical perspective on knowledge discovery in databases, in Advances in Knowledge Discovery and Data Mining, U.M. Fayyad et al. (editors), 83 - 116, AAAI/MIT Press, California, 1996.

G. Finke, R.E. Burkard, and F. Rendl, Quadratic assignment problems, Annals of Discrete Mathematics 31, 61 - 82, 1987.

C. Fleurent and J.A. Ferland, Genetic hybrids for the quadratic assignment problem, in Quadratic Assignment and Related Problems, P. Pardalos and H. Wolkowicz (editors), DIMACS Series in Discrete Mathematics and Theoretical Computer Science 16, 173 - 187, 1994.

C. Fleurent and J.A. Ferland, Genetic and hybrid algorithms for graph coloring, Annals of Operations Research 63, 437 - 461, 1996.