7.5 Experimentation with the F6 Function

Following Davis [Dav91], we used 22-bits for representing each variable. An equal number of features was used to cover the domain of each variable ($n=m$). The algorithm was relatively insensitive to the parameter λ and performed well for values of λ greater than 0.2. The value 0.25 for λ was used in the tests. Experiments were performed for varying n (i.e. number of features per variable) to determine how this parameter affects GLS. The values tried for n were 5, 10, 15, 20, 50, and 100. Fifty runs from random solutions (random binary strings) were performed for each value of n considered with the iteration limit set to 10,000 local search improvement cycles. Table 7.1 illustrates the results obtained. The best setting proved to be $n=m=5$. Under this setting, the algorithm succeeded in finding the exact optimal solution (0,0) in 100% of 50 runs. Under all settings, the algorithms found the exact optimum many times.

No. of features	n=m=5	n=m=10	n=m=15	n=m=20	n=m=50	n=m=100
Mean Cost	0.00E+00	4.55E-11	3.19E-10	2.73E-10	1.97E-04	3.21E-04
Best Solution	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Worst Solution	0.00E+00	2.28E-09	2.28E-09	2.28E-09	9.72E-03	9.72E-03
Mean Iterations	2287.32	2566.22	2954.08	3526.9	4132.66	3738.48
Mean Time	2.823333	3.150668	3.634334	4.382333	5.188333	4.654
Mean Funct. Eval.	104958.6	117778.8	135588	161878.4	189675.6	171578.5
Optimal Runs	50	49	43	44	31	22
Total runs	50	50	50	50	50	50

Table 7.1 GLS performance on F6 (Time in CPU seconds on a DEC Alpha 3000/600 175MHz).

This performance further improves if more time is given to the algorithm. For example, in the case (*n=m=100*) where most failures occurred (28 out of 50 runs), we performed the same experiment but this time allowed the algorithm to complete 100,000 local search iterations. The performance of GLS significantly improved and the algorithm found the exact optimum in 50 out of 50 runs (no failures).

The main observation made was that GLS performance degraded as the number of features used increased. More features meant more effort to leave a particular area but also more careful exploration. For this particular function, diversification of search to sample the whole search space proved important to find the global minimum quickly. The distribution of points visited for $n=m=10$ during 10,000 iterations of local search is shown in Figure 7.3. During the particular run that generated Figure 7.3, the optimal solution was found early and after 1965 iterations. Despite that, the algorithm was allowed to continue until 10,000 iterations were completed to get a better picture of the solutions visited by the algorithm. As one can see in Figure 7.3, the algorithm distributed its efforts over the whole of the search space but visited mainly local minima. That is why points in are arranged in concentric cycles around the point (0,0).

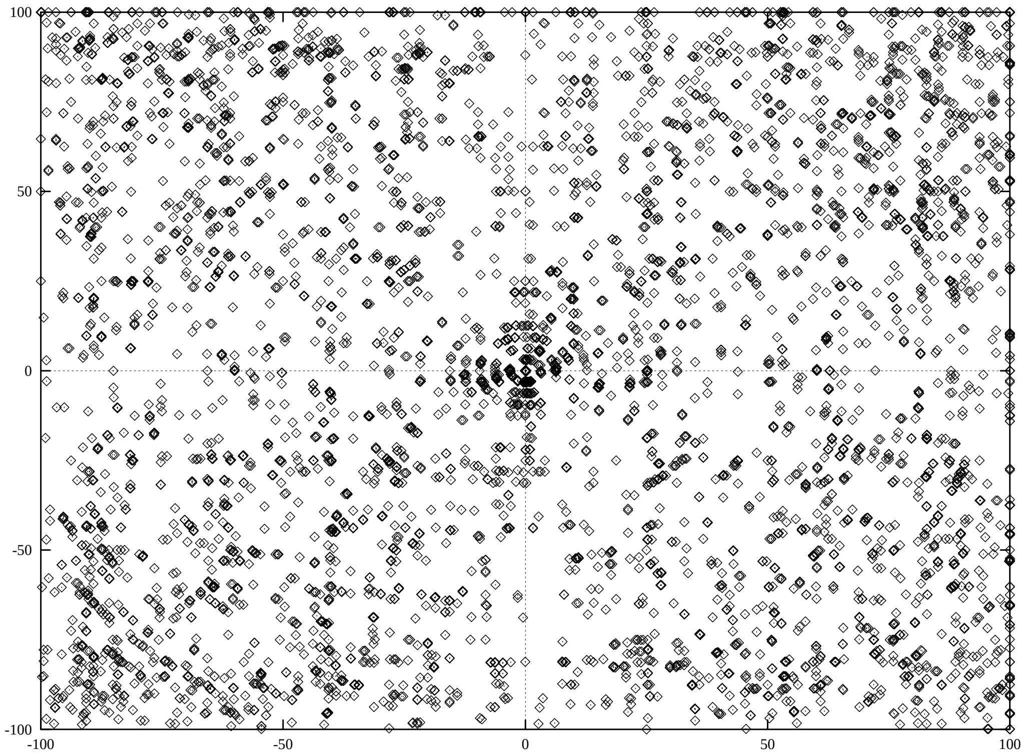

Figure 7.3 All the points visited during the first 10,000 iterations of local search

This is more clearly demonstrated in Figure 7.4 where a 3-D view of the visited points is shown. The shape formed is exactly the bottom part of F6 which suggests that the points are actually local minima in the great majority. Note here, that GLS is exploring

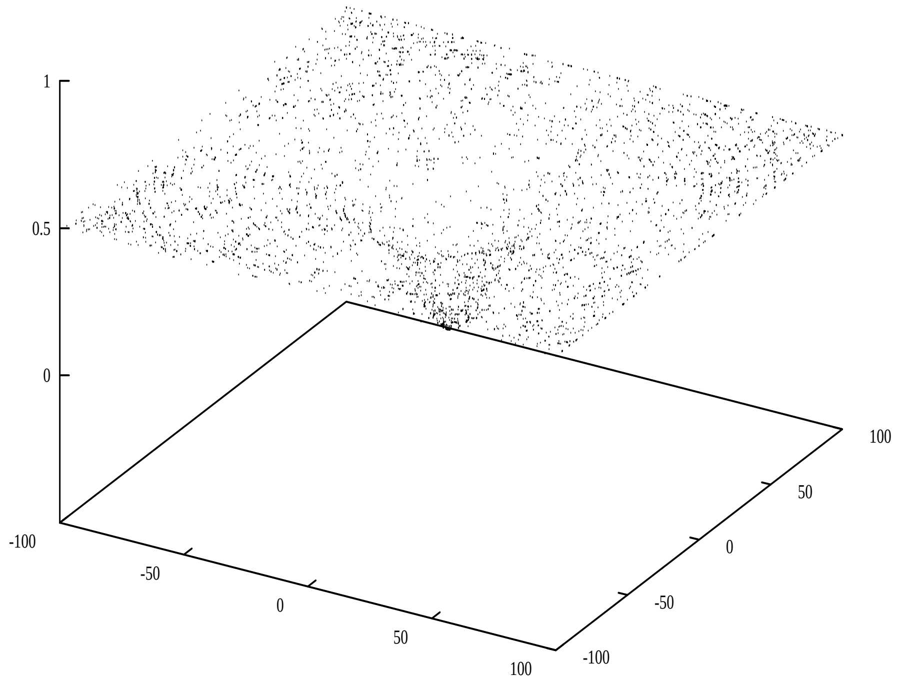

Figure 7.4 3-D View of Figure 7.3

binary space and not numeric space. In general, local minima and their attraction basins in the binary space are different from the local minima and their attraction basins appearing in the numeric space. Because of the symmetrical landscape, the binary encoding used and the structure of the GLS features, the majority of the solutions visited by GLS in the case of F6 have the property of being numeric local minima as illustrated in Figures 1.3 and 1.4. This is not necessarily the case for functions with non-symmetrical landscapes. In these cases, grey encodings (see [BT94] for example) and/or features of different structure may yield better performance than the encoding scheme and features used in this chapter.

7.6 Conclusions

In this chapter, we have shown that GLS has the potential to be utilised in the optimisation of real-valued functions with numerous local minima, which are considered to be difficult for gradient-based methods. The application of GLS to optimise the F6 function, a benchmark for Genetic Algorithms, has been examined. GLS repeatedly located the exact global optimum of the function. The chapter also serves in demonstrating how artificial solution features can be created when no features can be deduced from the structure of the objective function, which adds support to our claim that GLS has wide applications.

Chapter 8

Summary and Conclusions

This study demonstrated the effectiveness and efficiency of the GLS approach to combinatorial optimisation alone or when combined with FLS. We demonstrated that the use of information significantly improves simple local search heuristics transforming them to powerful optimisation algorithms able to compete or even outperform state of the art specialised methods. Furthermore, we demonstrated that the proposed approach is general enough to be applicable to a diversity of problem from the famous TSP and QAP to RFLAP and Workforce Scheduling and even to continuous optimisation problems. In this last chapter, we summarise the research conducted, conclude on GLS and FLS and also discuss the prospects of future research on the subject.

8.1 Summary of the Research Conducted

Guided local search is a novel approach which facilitates the engineering of intelligent search schemes which exploit problem and history information to guide a local search algorithm in a search space. Constraints on solution features are introduced and dynamically manipulated. The objectives of search intensification and diversification are unified in the single objective of distributing the search effort according to information. Various search distribution policies can be implemented. In this study, we examined the case of distributing the search effort according to feature costs either predetermined or evaluated during search.

We demonstrated the effectiveness of the proposed technique in two of the most prominent problems in combinatorial optimisation, the TSP and the QAP. Comparisons conducted with a total of fifteen methods for the TSP and four methods for the QAP showed that the GLS algorithm is better than or at least very competitive to many state of the art algorithms for the problems. Optimal or high quality solutions were consistently found in a variety of instances from the problem libraries TSPLIB and QAPLIB proving the robustness of GLS across these two landmark problems in combinatorial optimisation.

The application of the method to real world problems with various objectives and constraints was also studied in the context of the constrained optimisation problems of Radio Link Frequency Assignment and Workforce Scheduling. GLS was compared with twelve methods for the Radio Link Frequency Assignment Problem and five methods for the Workforce Scheduling problem. These comparisons clearly demonstrated the advantages of using GLS both in terms of solution quality and running times. Solutions found in the benchmark instances of RLFAP and Workforce Scheduling are amongst the best found so far for these problems. The applicability of

GLS to NonConvex optimisation problems was also demonstrated laying the foundations for the development of new methods based on GLS for this very important class of problems.

The technique of FLS was also presented and the benefits from combining it with GLS were studied in the TSP, RLFAP and Workforce Scheduling. The GLS-FLS combination leads to highly efficient variants of GLS which are many times faster than basic GLS without sacrificing solution quality.

Summarising the contents of the thesis, GLS was presented along with FLS. The method was applied to five combinatorial optimisation problems and compared with 35 algorithms including some of the best heuristic methods for these problems. Variants of almost all the general optimisation methods mentioned in the introduction were compared with GLS in at least one of the problems examined. In particular, GLS was compared with:

- Simulated Annealing on the TSP, RLFAP, and Workforce Scheduling,
- Tabu Search on the TSP, QAP, and RLFAP,
- Genetic Algorithms on the TSP, QAP, Workforce Scheduling, and RLFAP,
- Iterated Local Search on the TSP,
- Repeated Local Search on the TSP and QAP,
- Neural Networks on the RLFAP.

We believe that this is one of the most extensive studies for a newly presented combinatorial optimisation method.

8.2 Concluding Remarks on GLS and FLS

For many years, general heuristics for combinatorial optimisation problems with most prominent examples the methods of Simulated Annealing and Genetic Algorithms heavily relied on randomness to generate good approximate solutions to difficult NP-Hard problems. The introduction and acceptance of Tabu Search by the Operations Research community mainly due to the efforts of Glover, Laguna, Taillard, de Werra, Hertz, Battiti, Tecchioli and others initiated an important new era for heuristic methods where deterministic algorithms exploiting history information started appearing and being used in real world applications.

8.2.1 Guided Local Search

Guided local search proposed in this thesis follows this trend. GLS heavily exploits information (not only the search history) to distribute the search effort in the various regions of the search space. Important structural properties of solutions are captured by solution features. Solutions features are assigned costs and local search is biased to spend its efforts according to these costs. Penalties on features are utilised for that purpose.

When local search settles in a local minimum, the penalties are increased for selected features of the local minimum. By penalising features appearing in local minima, GLS avoids the local minima visited (exploiting historical information) but also diversifies choices for the various structural properties of solutions captured by the solution features. Features of high cost are penalised more times than features of low cost: the diversification process is directed and deterministic rather than undirected and random.

Feature costs contain uncertain information making sometimes speculative assumptions about the desirability of particular structural properties of solutions. Some of these properties could be essential parts of good solutions despite the high cost they may incur on the solution cost. GLS is flexible in such cases by combining search intensification with the continuous diversification process caused by the penalties on feature costs.

8.2.2 The Role of Parameter λ

The task of combining diversification with intensification is accomplished by the regularisation parameter λ which controls the influence of the information on the search process. The local gradients are directing the search process to good solutions undertaking the task of intensification. The parameter λ linearly combines the local gradients with the penalties of GLS blending the two functions of intensification and diversification. If λ is low then GLS is intensifying search slowing down the diversification process. Conversely, if λ is high then the feature costs fully determine the course of local search. For values of λ in the middle of these two extreme cases, an optimal blending of intensification and diversification is achieved. Intensification of search can also be achieved by using penalties of limited duration (see section 4.4.3) or incentives implemented as negative penalties that encourage the use of specific features rather than discourage them as with the penalties in the basic GLS. This last case of incentives has not been explored in our work and it may lead to more advanced schemes for guiding local search.

8.2.3 Fast Local Search

The selective diversification scheme of GLS where particular features are penalised and alternative solutions structures are sought that avoid these features is ideally combined with FLS which limits neighbourhood search to particular parts of the overall solution.

To allow the blending of local gradients with penalties, GLS increases the penalties for features and subsequently invokes local search to remove the penalised features from the solution. Because of λ, local gradients can affect this decision by allowing or not allowing a move to be executed which removes the penalised features. This is an essential part of the operation of GLS and enables the blending of intensification (expressed by the local gradients) and diversification (expressed by the penalties). FLS speeds up this blending allowing a quick test of the local gradients after a penalty increase. The moves which remove the penalised features are checked and if no improving move is found, control immediately returns to GLS which penalises alternative features or the same features depending on the effort already invested in these features as given by the penalties already applied to them.

In general, many penalty cycles may be required before a move is executed out of the local minimum. This should not be viewed as an undesirable situation. It is caused by the uncertainty in the information as captured by the feature costs which makes necessary the testing of the GLS decisions against the local gradients. FLS significantly reduces the computation times required to measure the local gradients in a local minimum allowing far more many penalty modification cycles to be performed by GLS for the same amount of running time.

8.3 Future Research

This thesis offers a first study of GLS and FLS. The method is still in its infancy and future research is required to further develop the method and adapt it to other problems. The use of incentives implemented as negative penalties which encourage the use of specific solution features is one promising direction to be explored. Other potentially interesting research directions include automated tuning of the regularisation parameter, definition of effective termination criteria, and different utility functions for selecting the features penalised.

GLS could also be used to distribute the search effort in other techniques such as Genetic Algorithms. In particular, GLS could be invoked at specific intervals to detect the presence of particular features in a GA population and subsequently diversify or intensify genetic search by applying penalties or incentives on particular features which are considered “bad” or “good” respectively. The GA could be guided to avoid or favour specific features spending its search efforts according to the information which again can be captured in the form of feature costs. The same utility function (Eq. 2.5) could be used by simply replacing the indicator function in Eq. 2.5 with a measure taking values in the interval [0,1] that will reflect how frequently a feature is appearing in the solutions of the population.

Finally, we found it very easy to adapt GLS and FLS to the different problems examined in this thesis something which suggests that it may be possible to built a generic software platform for combinatorial optimisation based on GLS. Although local search is problem dependent, most of the other structures of GLS and also FLS are problem independent. Furthermore, a step by step procedure is usually followed when GLS is applied to a new problem (i.e. identify features, assign costs, etc.) something which makes easier the use of the technique by non-specialists (e.g. software engineers).

References

[AA95] N. Azarmi and W. Abdul-Hameed. Workforce scheduling with constraint logic programming. *BT Technology Journal*, Vol. 13, No. 1, 81-94, 1995.

[AFPR93] C. A. Anderson, K. Fraughnaugh, M. Parker, and J. Ryan. Path assignment for call routing: An application of tabu search. *Annals of Operations Research*, 41, 301-312, 1993.

[AK89] E. H. L. Aarts and J. H. M. Korst. *Simulated Annealing and Boltzmann machines*. Wiley, Chichester, 1989.

[Bak93] S. Baker. Applying simulated annealing to the workforce management problem. ISR Technical Report, BT Laboratories, Martlesham Heath, Ipswich, 1993.

[BDST95] J. F. Boyce, C. H. D. Dimitropoulos, G. vom Scheidt, J. G. Taylor. GENET and Tabu Search for combinatorial optimisation problems. *Proceedings of World Congress on Neural Networks*, Washington D.C., INNS Press, 1995.

[Ben92] J. L. Bentley. Fast Algorithms for Geometric Travelling Salesman Problems. *ORSA Journal on Computing*, Vol. 4, 387-411, 1992.

[BFR95] C. Bessiere, E. C. Freuder, and J. Regin. Using Inference to Reduce Arc Consistency Computation. *Proceedings of IJCAI-95*, 592-598, 1995.

[BKR91] R. E. Burkard, S. E. Karisch, and F. Rendl. QAPLIB - A Quadratic Assignment Library. *European Journal of Operational Research*, 55, 115-119, 1991. (Problems and updated version of paper are available by Web at http://fmatbhp1.tu-graz.ac.at/~karisch/qaplib/)

[BSA83] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neurolike adaptive elements that can solve difficult learning problems. *IEEE Transactions on Systems, Man and Cybernetics*, 13, 834-846, 1983.

[BT94] R. Battiti and G. Tecchioli. The Reactive Tabu Search. *ORSA Journal on Computing*, Vol. 6, 126-140, 1994.

[Bur84] R. E. Burkard. Quadratic Assignment Problem. *European Journal of Operational Research*, 15, 283-289, 1984.

[CK95] P. Crescenzi and V. Kann. A compendium of NP optimization problems. Technical report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Università di Roma "La Sapienza", 1995. (also available via Web at : http://www.nada.kth.se/~viggo/problemlist/compendium.html)

[CMMR96] B. Codenotti, G. Manzini, L. Margara and G. Resta. Pertrubation: An Efficient Technique for the Solution of Very Large Instances of the Euclidean TSP. *ORSA Journal on Computing*, Vol. 8, No. 2, 125-133, 1996.

[Con90] D. T. Connoly. An improved annealing scheme for the QAP. *European Journal of Operational Research*, 46, 93-100, 1990.

[Cro58] A. Croes. A Method for Solving Travelling-Salesman Problems. *Operations Research*, 5, 791-812, 1958.

[CS93] J. Chakrapani and J. Skorin-Kapov. Massively parallel tabu search for the quadratic assignment problem. *Annals of Operations Research*, 41, 327-341, 1993.

[Dav91] L. Davis. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1991.

[Dav97] A. Davenport. GENET: a connectionist architecture for constraint satisfaction. PhD Thesis, Department of Computer Science, University of Essex, 1997.

[Dow93] K. A. Dowsland. Simulated Annealing. In Reeves, C. R. (ed.), *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publishing, 20-69, 1993.

[DTWZ94] A. Davenport, E. Tsang, C. J. Wang, and K. Zhu. GENET: A connectionist architecture for solving constraint satisfaction problems by iterative improvement. *Proceedings of AAAI-94*, 325-330, 1994.

[ERR94] A. E. Eiben, P-E. Raua, and Zs. Ruttkay. Solving constraint satisfaction problems using genetic algorithms. *Proceedings of 1st IEEE Conference on Evolutionary Computing*, 543-547, 1994.

[FF94] C. Fleurent and J. A. Ferland. Genetic Hybrids for the Quadratic Assignment Problem. *DIMACS Series in Mathematics and Theoritical Computer Science*, 16, 173-187, 1994.

[FM96] B. Freisleben and P. Merz. A Genetic Local Search Algorithm for Solving the Symmetric and Asymmetric TSP. *Proceedings of IEEE International Conference on Evolutionary Computation*, Nagoya, Japan, 616-621, 1996.

[Fra96] J. Frank. Weighting for Gobot: Learning Heuristics for GSAT. *Proceedings of AAAI-96*, Vol. 1, 338-343,1996.

[FW92] E. C. Freuder and R. J. Wallace. Partial constraint satisfaction. *Artificial Intelligence*, 58, 21-70, 1992.

[GHL94] M. Gendreau, A. Hertz, and G. Laporte. A Tabu Search Heuristic for the Vehicle Routing Problem. *Management Science*, Vol 40, No. 10, 1276-1290, 1994.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A guide to the Theory of NP-Completeness*. W. H. Freeman, San Francisco, 1979.

[GL93] F. Glover and M. Laguna. Tabu Search. In Reeves, C. R. (ed.), *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publishing, 71-141, 1993.

[Glo86] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers Ops Res.*, 5, 533-549, 1986.

[Glo89] F. Glover. Tabu search Part I. *ORSA Journal on Computing*, Vol. 1, 190-206, 1989.

[Glo90] F. Glover. Tabu search Part II. *ORSA Journal on Computing*, Vol. 2, 4-32, 1990.

[Glo94] F. Glover. Tabu Search: improved solution alternatives for real world problems. In Birge & Murty (ed.), *Mathematical Programming: State of the Art*, 64-92, 1994.

[Glo95] F. Glover. Tabu Search Fundamentals and Uses. Graduate School of Business, University of Colorado, Boulder, 1995.

[Glo96] F. Glover. Tabu Search and Adaptive Memory Programming - Advances, Applications and Challenges. To appear in: *Interfaces in Computer Science and Operations Research*, Barr, Helgason and Kennington eds., Kluwer Academic Publishers, 1996.

[Gol89] D. E. Goldberg. *Genetic algorithms in search, optimisation, and machine learning*. Addison-Wesley, 1989.

[GTW93] F. Glover, E. Taillard, and D. de Werra. A user's guide to tabu search. *Annals of Operations Research*, 41, 3-28, 1993.

[Hal80] W. K. Hale. Frequency Assignement: Theory and Applications. *Proceedings of the IEEE,* 68, 1497-1514, 1980.

[Han86] P. Hansen. The steepest ascent mildest descent heuristic for combinatorial programming. *Congress on Numerical Methods in Combinatorial Optimisation*, Capri, Italy, 1986.

[Hay94] S. Haykin. Neural Networks. Macmillan, 1994.

[Hen89] P. van Hentenryck. *Constraint satisfaction in logic programming*. MIT Press, 1989.

[HL95] F. S. Hiller and G. Lieberman. *Introduction to Operations Research*. Sixth edition, McGraw-Hill, New York, 1995.

[Hol75] J. H. Holland. *Adaptation in natural and artificial systems*. University of Michigan press, Ann Arbor, MI, 1975.

[Hoo95] J. N. Hooker. Testing Heuristics: we have it all wrong. *Journal of Heuristics*, Vol. 1, No. 1, 33-42, 1995.

[Ing89] A. L. Ingber. Very fast simulated re-annealing. Journal of Mathematical Computer Modelling, Vol. 12 No. 8, 967-973, 1989.

[JAMS89] D. Johnson, C. Aragon, L. McGeoch, and C. Schevon. Optimisation by simulated annealing: an experimental evaluation, part I, graph partitioning. *Operations Research*, 37, 865-892, 1989.

[JAMS91] D. Johnson, C. Aragon, L. McGeoch, and C. Schevon. Optimisation by simulated annealing: an experimental evaluation, part II, graph coloring and number partitioning. *Operations Research*, 39, 378-406, 1991.

[JBMR96] D. Johnson, J. Bentley, L. McGeoch, and E. Rothberg. Near-optimal solutions to very large traveling salesman problems. In preperation. (for experimental results from this work see also [JM95])

[JFM96] M. Jampel and E. Freuder and M. Maher (ed.). Over-Constrained Systems. *Lecture Notes in Computer Science*, 1106, Springer-Verlag, 1996.

[JKS95] Y. Jiang, H. Kautz and B. Selman. Solving problems with hard and soft constraints using a stochastic algorithm for max-sat. *First International Joint Workshop on Artificial Intelligence and Operations Reasearch*, Timberline Lodge, Timberline, Oregon, USA, June, 1995.

[JM95] D. Johnson and L. McGeoch. The Travelling Salesman Problem: A Case Study in Local Optimisation. Manuscript, November, 1995 (appear in *Local Search in Optimization*, Ed. E. H. L. Aarts and J. K. Lenstra, Wiley, Chichester, forthcoming).

[Joh90] D. Johnson. Local Optimisation and the Travelling Salesman Problem. *Proceedings of the 17th Colloquium on Automata Languages and Programming*, Lecture Notes in Computer Science, 443, 446-461, Springer-Verlag, 1990.

[KC93] S. Kota and S. Chiou. Use of Orthogonal Arrays in Mechanism Synthesis. *Mechanical Machine Theory*, Vol. 28, 777-794, 1993.

[KGV83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimisation by Simulated Annealing. *Science*, Vol. 220, 671-680, 1983.

[Kno94] J. Knox. Tabu Search Performance on the Symmetric Travelling Salesman Problem. *Computers Ops Res.*, Vol. 21, No. 8, pp. 867-876, 1994.

[Koo57] B. O. Koopman. The Theory of Search, Part III. The Optimum Distribution of Searching Effort. *Operations Research*, 5, 613-626, 1957.

[LA88] P. J. M. van Laarhoven and E. H. L. Aarts. *Simulated Annealing: Theory and Applications*. Kluwer, Dordrecht, 1988.

[Lap92] G. Laporte. The Travelling Salesman Problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59, 231-247, 1992.

[LG93] M. Laguna and F. Glover. Bandwidth Packing: A Tabu Search Approach. *Management Science*, Vol. 39, No. 4, 492-500, 1993.

[LG93b] M. Laguna and F. Glover. Integrating Target Analysis and Tabu Search for Improved Scheduling Systems. *Expert Systems with Applications*, Vol. 6, 287-297, 1993.

[Lin65] S. Lin. Computer Solutions of the Travelling-Salesman Problem. *Bell Systems Technical Journal*, 44, 2245-2269, 1965.

[LK73] S. Lin and B. W. Kernighan. An effective heuristic algorithm for the travelling salesman problem. *Operations Research*, 21, 498-516, 1973.

[LLKS85] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys (ed.). *The Travelling Salesman Problem: A guided tour in combinatorial optimisation*. John Wiley & Sons, 1985.

[LM86] M. Lundy and A. Mees. Convergence of an annealing algorithm. *Mathematical Programming*, 34, 111-124, 1986.

[LWR95] J. Lever, M. Wallace, and B. Richards. Constraint logic programming for scheduling and planning. *BT Technology Journal*, Vol.13, No.1., 73-80, 1995.

[MGK88] H. Muhlenbein, M. Gorges-Schleuter, and O. Kramer. Evolution Algorithms in combinatorial optimization. *Parallel Computing*, 7, 65-85, 1988.

[MJPL92] S. Minton, M.D. Johnston, A.B. Philips, and P. Laird. Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58, 161-205, 1992.

[MM93] K. Mak and A. J. Morton. A modified Lin-Kernighan traveling-salesman heuristic. *Operations Research Letters*, 13, 127-132, 1993.

[MMS93] C. Muller, E. H. Magill, and D.G. Smith. Distributed genetic algorithms for resource allocation. Technical Report, Strathclyde University, Glasgow, 1993.

[MO96] O. Martin and S. W. Otto. Combining Simulated Annealing with Local Search Heuristics. In G. Laporte and I. H. Osman (eds.), *Metaheuristics in Combinatorial Optimisation*, *Annals of Operations Research*, Vol. 63, 1996.

[MOF92] O. Martin, S. W. Otto, and E. W. Felten. Large-step Markov chains for the TSP incorporating local search heuristics. *Operations Research Letters*, 11, 219-224, 1992.

[Mor93] P. Morris. The breakout method for escaping from local minima. *Proceedings of AAAI-93*, 40-45, 1993.

[MRRTT53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller. Equation of state calculation by fast computing machines. *Journal of Chem. Phys.*, 21, 1087-1091, 1953.

[MT90] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. Wiley, New York, 1990.

[Osm93] I. H. Osman. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problems. *Annals of Operations Research*, 41, 421-451, 1993.

[Osm95] I. H. Osman. An Introduction to Meta-Heuristics. M. Lawrence and C. Wilson (eds.), In: *Operational Research Tutorial Papers*, Operational Research Society Press, Birmingham, UK, 92-122, 1995.

[PRW93] P. M. Pardalos, F. Rendl, and H. Wolkowicz. The Quadratic Assignment Problem: a survey and recent developments. *DIMACS Series in Discrete Mathematics and Theoritical Computer Science*, Vol. 16, 1-42, 1993.

[PS82] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimisation: Algorithms and Complexity*. Prentice-Hall, 1982.

[RB93] C. R. Reeves and J. E. Beasley. Chapter 1: Introduction. In Reeves, C. (ed.), *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publishing, 1-19, 1993.

[Ree93] C. R. Reeves. Genetic Algorithms. In Reeves, C. (ed.), *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publishing, 151-196, 1993.

[Ree96] C. R. Reeves. Modern Heuristic Techniques. In V. J. Rayward-Smith, I. H. Osman, C. R. Reeves and G. D. Smith (ed.), John Wiley & Sons, *Modern Heuristic Search Methods*, 1-25, 1996.

[Rei91] G. Reinelt. A Travelling Salesman Problem Library. *ORSA Journal on Computing*, 3, 376-384, 1991.

[Rei94] G. Reinelt. The Travelling Salesman: Computational Solutions for TSP Applications. *Lecture Notes in Computer Science*, 840, Springer-Verlag, 1994.

[RLFAP94] The RLFAP instances are available from listserver@saturne.cert.fr. To get more information and the instances send the command "get csp-list CELAR.blurb" to the above list server. The problems are also available via ftp from the Constraints Archive maintained by Michael Jampel (ftp.cs.city.ac.uk:/pub/constraints/csp-benchmarks/celar/).

[RT95] Y. Rochat and E. D. Taillard. Probabilistic diversification and intensification in local search for vehicle routing. *Journal of Heuristics*, 1, 147-167, 1995.

[SCED89] J. D. Schaffer, R. A. Caruana, L. J. Eshelman, and R. Das. A study of control parameters affecting online performance of genetic algorithms for function optimisation. *Proceedings of 3rd Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, 51-60, 1989.

[Sch95] G. vom Scheidt. Extension of GENET to Partial Constraint Satisfaction Problems and Constraint Satisfaction Optimisation Problems. Master's Thesis, Dept. of Mathematics, King's College London, UK, 1995.

[SG96] P. Soriano and M. Gendreau. Diversification Strategies in Tabu Search algorithms for the Maximum Clique Problem. *Annals of Operations Research*, 63, 189-207, 1996.

[Sim89] H. U. Simon. Approximation algorithms for channel assignment in cellular radio networks. *Proc. Fundamentals of Computation Theory*, Lecture Notes in Computer Science 380, Springer-Verlag, 405-416, 1989.

[SK93] B. Selman and H. Kautz. Domain independent versions of GSAT solving large structured satisfiability problems. Proceedings of IJCAI-93, 290-295, 1993.

[Sko90] J. Skorin-Kapov. Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing*, Vol. 2, 33-45, 1990.

[SLM92] B. Selman, H. J. Levesque, and D. G. Mitchell. A new method for solving hard satisfiability problems. *Proceedings of AAAI-92*, 440-446, 1992.

[Sol87] M. M. Solomon. Algorithms for the Vehicle Routing and Scheduling Problem with Time Window Constraints. *Operations Research*, 35, 254 - 265, 1987.

[ST90] M. A. Styblinski and T. S. Tang. Experiments in Nonconvex Optimisation: Stochastic Approximation with Function Smoothing and Simulated Annealing. *Neural Networks*, Vol. 3, 467-483, 1990.

[Sto83] L. D. Stone. The Process of Search Planning: Current Approaches and Continuing Problems. *Operations Research*, 31, 207-233, 1983.

[Tai91] E. Taillard. Robust taboo search for the QAP. *Parallel Computing*, 17, 443-455, 1991.

[Tai96] E. Taillard. Comparison of Iterative Searches for the Quadratic Assignment Problem. *Location Science*, 3, 87-105, 1995.

[TAJ77] A. N. Tikhonov, V. Y. Arsenin, and F. Jonh. *Solutions of Ill-Posed Problems*. John Wiley & Sons, 1977.

[TBGGP95] E. Taillard, P. Badeau, M. Gendreau, F. Guertin, and J.Y. Potvin. A tabu search heuristic for the vehicle routing problem with soft time windows. Publication CRT-95-66, Centre de recherche sur les transports, Université de Montréal, 1995.

[THL95] S. Tiourine, C. Hurkens, and J. K. Lenstra. An overview of algorithmic approaches to frequency assignment problems. CALMA Symposium, Scheveningen, November, 1995. (available via ftp at ftp://ftp.win.tue.nl/pub/techreports/CALMA/index.html)

[Thr92] S. B. Thrun. Efficient exploration in reinforcement learning. Technical report CMU-CS-92-102, School of Computer Science, Carnegie-Mellon University, 1992.

[Tsa93] E. Tsang. *Foundations of Constraint Satisfaction*. Academic Press, 1993.

[TV97] E. Tsang, and C. Voudouris. Fast local search and guided local search and their application to British Telecom's workforce scheduling problem. *Operations Research Letters*, Vol. 20, No. 3, 119-127, 1997.

[VT94] C. Voudouris and E. Tsang. Tunnelling Algorithm for Partial CSPs and Combinatorial Optimisation Problems. Technical Report CSM-213, Department of Computer Science, University of Essex, Colchester, UK, 1994.

[VT95a] C. Voudouris and E. Tsang. Guided Local Search. Technical Report CSM-247, Department of Computer Science, University of Essex, Colchester, UK, August, 1995.

[VT95b] C. Voudouris and E. Tsang. Function Optimization using Guided Local Search. Technical Report CSM-249, Department of Computer Science, University of Essex, Colchester, UK, September, 1995.

[VT96] C. Voudouris and E. Tsang. Partial Constraint Satisfaction Problems and Guided Local Search. *Proceedings of 2nd Int. Conf. on Practical Application of Constraint Technology* (PACT'96), London, April, 337-356, 1996.

[WF93] R. J. Wallace and E. C. Freuder. Conjunctive Width Heuristics for Maximal Constraint Satisfaction. *Proceedings of AAAI-93*, 762-768, 1993.

[WF96] R. Wallace and E. Freuder. Heuristic Methods for Over-Constrained Constraint Satisfaction Problems. *Lecture Notes in Computer Science*, 1106, Springer-Verlag, 207-216, 1996.

[WT91] C. J. Wang and E. Tsang. Solving constraint satisfaction problems using neural-networks. *Proceedings of IEE Second International Conference on Artificial Neural Networks*, 295-299, 1991.

[WT94] T. Warwick and E. Tsang. Using a genetic algorithm to tackle the processors configuration problem. *Symposium on Applied Computing*, 217-221, 1994.

[WZ93] D. L. Woodruff and E. Zemel. Hashing vectors for tabu search. *Annals of Operations Research*, 41, 123-137, 1993.

[XCG95] J. Xu, S. Y. Chiu, and F. Glover. A Probabilistic Tabu Search Heuristic for the Telecommunications Network Design. Graduate School of Business, University of Colorado, Boulder (to appear in Journal of Combinatorial Optimization), 1995.

[XCG96] J. Xu, S. Y. Chiu, and F. Glover. Fine-Tuning a Tabu Search Algorithm with Statistical Tests. Graduate School of Business, University of Colorado, Boulder, 1996.

[XK96] J. Xu and J. Kelly. A new network flow-based tabu search heuristic for the vehicle routing problem. Graduate School of Business, University of Colorado, Boulder (to appear in Transportation Science), 1996.

[ZD95] M. Zachariasen and M. Dam. Tabu Search on the Geometric Travelling Salesman Problem. In I. H. Osman and J. P. Kelly (ed.), *Meta-heuristics: Theory and Applications*, Kluwer, Boston, 571-587, 1996.

Appendix A

Results on the Travelling Salesman Problem

The set of problems used in the evaluation of the Repeated Local Search, Guided Local Search and Iterated Local Search (using the Double Bridge move) variants on the TSP included 20 problems from 48 to 1002 cities all from TSPLIB (see Chapter 3 for details on these techniques). For each variant tested, 10 runs were performed from random solutions and 5 minutes of CPU time were allocated to each algorithm in each run on a DEC Alpha 3000/600 (175MHz) machine. To measure the success of the variants, we considered the percentage excess above the optimal solution as in Eq. 3.5. For GLS variants, the normalised lambda parameter a was provided as input and λ was determined after the first local minimum using Eq. 3.6. For GLS variants using 2-Opt, a was set to a = 1/6 while the GLS variants based on 3-Opt used the slightly lower value a = 1/8 and the LK variants the even lower value a = 1/10. Results for GLS are shown in Table A.1.

Iterated Local Search was using the Double Bridge move. No simulated annealing was used which is roughly equivalent to the Large-Markov Chains Methods with temperature T set to 0. Results for Iterated Local Search are shown in Table A.2.

Finally, Repeated Local Search was restarting from a random solution whenever local search was reaching a local minimum. Results for Repeated Local Search are shown in Table A.3. The names of the variants were formed according to the following convention:

<meta-heuristic>-<local search type>-<neighbourhood type>.

Problem	No.Cities	Mean Excess (%) over 10 runs					
		GLS-FI-LK	GLS-FI-3Opt	GLS-FI-2Opt	GLS-FLS-LK	GLS-FLS-3Opt	GLS-FLS-2Opt
att48	48	0	0	0	0	0	0
eil76	76	0	0	0	0	0	0
kroA100	100	0	0	0	0	0	0
bier127	127	0.218207	0.116586	0.019699	0.206625	0.002198	0
kroA150	150	0.029784	0.084075	0.000754	0.001508	0.001131	0
u159	159	0	0.460551	0.225285	0	0	0
kroA200	200	0.436189	0.526083	0.257083	0.088872	0.00681	0
gr202	202	0.732321	0.406375	0.309512	0.252988	0.011703	0
gr229	229	0.392788	0.468195	0.381644	0.152969	0.015007	0.004309
gil262	262	0.328007	0.723297	0.428932	0.084104	0.046257	0.004205
lin318	318	1.00264	1.74284	1.33884	0.583407	0.129197	0.02641
gr431	431	1.69438	2.71862	2.34071	0.563665	0.134003	0.023919
pcb442	442	0.966363	0.80783	1.36634	0.38816	0.038403	0.044311
att532	532	1.04746	2.28599	2.52871	0.386116	0.224662	0.089937
u574	574	1.36892	2.81263	3.66807	0.580951	0.278824	0.141444
rat575	575	0.806142	1.77174	2.25011	0.287908	0.171268	0.098922
gr666	666	1.66056	4.38707	6.00476	0.855251	0.497863	0.206279
u724	724	1.02505	2.25101	3.03054	0.61298	0.336674	0.168218
rat783	783	0.897116	2.24052	3.36929	0.511015	0.285033	0.161254
pr1002	1002	1.97877	3.31969	5.54336	1.04229	0.945357	0.620626
Average Excess		**0.729235**	**1.356155**	**1.653182**	**0.32994**	**0.15622**	**0.079492**

Table A.1 Results for GLS on the TSP.

Problem	No.Cities	Mean Excess (%) over 10 runs					
		DB-FI-LK	DB-FI-3Opt	DB-FI-2Opt	DB-FLS-LK	DB-FLS-3Opt	DB-FLS-2Opt
att48	48	0	0	0	0	0	0
eil76	76	0	0	0	0	0	0
kroA100	100	0	0	0	0	0	0
bier127	127	0	0	0	0	0	0
kroA150	150	0	0.001508	0.003393	0	0	0
u159	159	0	0	0	0	0	0
kroA200	200	0	0.077295	0.10113	0	0.004767	0.075252
gr202	202	0.009213	0.088396	0.457171	0	0.155129	0.257719
gr229	229	0.014116	0.157576	0.382387	0.004755	0.064115	0.124515
gil262	262	0.016821	0.20185	0.626577	0	0.075694	0.475189
lin318	318	0.255776	0.719027	1.14588	0.240786	0.279093	0.3519
gr431	431	0.332703	0.94403	2.13495	0.222386	0.394192	0.615294
pcb442	442	0.066367	0.368861	1.8961	0.081728	0.309977	0.684745
att532	532	0.225023	1.03554	2.64971	0.08163	0.270534	0.422957
u574	574	0.114348	1.20038	2.94269	0.092399	0.404823	0.553042
rat575	575	0.13731	1.15016	3.75904	0.097446	0.445888	0.649638
gr666	666	0.418878	1.25178	3.27054	0.175874	0.359528	0.816489
u724	724	0.356955	1.43617	3.94106	0.166547	0.367693	0.627535
rat783	783	0.240745	1.79764	5.00454	0.153305	0.516693	0.744947
pr1002	1002	1.04742	2.05625	5.19902	0.446332	0.872049	1.05727
Average Excess		**0.161784**	**0.624323**	**1.675709**	**0.088159**	**0.226009**	**0.372825**

Table A.2 Results for Iterated Local Search on the TSP.

Problem	No.Cities	Mean Excess (%) over 10 runs					
		REP-FI-LK	REP-FI-3Opt	REP-FI-2Opt	REP-FLS-LK	REP-FLS-3Opt	REP-FLS-2Opt
att48	48	0	0	0	0	0	0
eil76	76	0	0	1.35688	0	0	1.48699
kroA100	100	0	0.39564	0.222254	0	0.225543	0.215205
bier127	127	0.030098	0.403696	1.19629	0.027899	0.370386	1.29513
kroA150	150	0.002262	0.8317	2.00912	0.002262	0.8038	2.01553
u159	159	0	0.30038	1.62619	0	0.265447	2.05894
kroA200	200	0.024517	1.00688	3.30768	0.004767	0.922092	3.23583
gr202	202	0.141434	1.22958	3.58591	0.129731	1.19995	3.68352
gr229	229	0.097695	1.36774	3.40129	0.094427	1.27301	3.56443
gil262	262	0.054668	1.3709	5.12195	0.054668	1.2868	5.77796
lin318	318	0.629565	2.17992	4.37936	0.636703	2.022676	4.9128
gr431	431	0.679641	2.07801	5.33877	0.665232	2.20915	5.97495
pcb442	442	0.48525	1.77636	6.65012	0.516956	1.72417	7.19544
att532	532	0.530232	2.29033	6.28368	0.579354	2.29141	7.13899
u574	574	0.738382	2.91397	7.46674	0.703157	2.6934	8.4788
rat575	575	0.807618	2.69895	7.69231	0.887347	2.70781	8.61066
gr666	666	0.837619	3.18259	8.14712	0.847811	2.97203	9.94096
u724	724	0.933667	2.90551	7.76903	1.0241	2.87473	8.83202
rat783	783	1.00045	3.2864	8.46468	1.06518	3.39882	9.38792
pr1002	1002	1.5046	3.50511	8.62028	1.39138	3.59138	10.5847
Average Excess		**0.424885**	**1.686183**	**4.631983**	**0.431549**	**1.64163**	**5.219539**

Table A.3 Results for Repeated Local Search on the TSP.