Towards a Guided Cooperative Search

Alexandre Le Bouthillier*‡ Teodor G. Crainic*† Peter Kropf*§

*Centre de recherche sur les transports, Université de Montréal
C.P.6128, succursale Centre-ville, Montréal, QC, H3C 3J7, Canada

†Département de management et technologie, Université du Québec à Montréal
C.P. 8888, Succursale Centre-Ville, Montrél, QC, H3C 3P8, Canada
theo@crt.umontreal.ca

‡Département d'informatique et de recherche opérationnelle, Université de Montréal
C.P.6128, succursale Centre-ville, Montréal, QC, H3C 3J7, Canada
alexleb@crt.umontreal.ca

§Institut d'informatique, Université de Neuchâtel
Facultée des sciences, Rue Emile-Argand 11 CH-2007 Neuchâtel, Suisse
peter.kropf@unine.ch

1 Introduction

Parallel computing may significantly strengthen the efficiency and robustness of search methods. These are alos the principal qualities required from good metaheuristics. Three forms of parallelism can be applied to metaheuristics: (i) Division of compute-intensive tasks at a low algorithmic level, (ii) Explicit domain decomposition of the solution or search space, (iii) Multi-thread search. An independent multi-thread search produces the best solution among those found by each independent method. Multi-thread Cooperative Search implements a mechanism that allows information (e.g., solutions) to be exchanged among the search threads thus favoring research in promising directions. Cooperative multi-search also increase the robustness of the global search (Crainic and Toulouse 2003, Toulouse, Thulasiram and Glover 1993, Clearwater, Hogg and Huberman 1992).

Cooperative Search combines the efforts of several independent metaheuristics by using a so-called Solution Warehouse. According to the information stored, the names "adaptive" and "central memory" are also used. This repository receives "good" solutions from the search threads and, on demand and according to their own internal logic, provides them in return with solutions, that may for example help to favorably diversify the search. This simple architecture allows for asynchronous communication and to exchange solutions that influence the search trajectory of each method. Enhanced with simple extraction rules for the returned solutions, cooperation is successfully applied to address a number of difficult combinatorial problems: network design, multi-commodity location-allocation, circuit partitioning and Vehicle Routing Problem with Time Windows (VRPTW).

This paper introduces an enhanced Cooperative Search mechanism that creates new information from partial or complete solutions and that performs global intensification and diversification phases. The proposed framework does not suppose any specific problem structure and may thus be applied to a wide range of combinatorial problems. The paper also reports very good solutions on benchmark problem sets for the VRPTW: 139 best known results and a new best known average on the problems with 200 customers were found.

The paper is organized as follow. Section 2 briefly presents a framework of Cooperative Search. Section 3 introduces the pattern identification mechanism and the global search phases. Section 4 details the implementation of a guided Cooperative Search applied to the VRPTW. Section 5 presents the computational results and analyzes them both from the point of view of solving the VRPTW and from that of the performance of the parallel strategy. Conclusions and perspectives are the subject of the last section.

2 Cooperative Search Framework

A Cooperative Search framework made up of a number of independent processes (threads) of possibly different types communicate through the Solution Warehouse as illustrated in Figure 1. A search thread either heuristically constructs new solutions or improves current ones of the Solution Warehouse by either executing a neighborhood-based search metaheuristic, a population-based metaheuristic (e.g., evolutionary algorithms, scatter search, path relinking) or a post-optimization procedures (e.g., intensive local search). Improving metaheuristics, such as Tabu search, aggressively explore the search space, while population-based methods contribute to increase the diversity of solutions exchanged among the cooperating methods. When the same metaheuristic is used by several search threads, the initial solution and particular setting of a number of important search parameters differentiate each search thread from the others.

The cooperation aspect of the parallelization scheme is achieved through asynchronous exchanges of solutions. Solutions are shared through a Solution Warehouse. In this scheme, whenever a thread desires to send out information (e.g., when a new local optimum is identified), it sends it to the pool. Similarly, when a thread accesses outside information (to diversify the search, for example), it reaches out and receives it from the pool. Communications are initiated exclusively by the individual threads, irrespective of their role as senders or receivers of information.

Two main classes of cooperation mechanisms are found in literature: partial and complete solutions exchanges. Adaptive memory methods (Rochat and Taillard 1995, Taillard et al., Golden et al. 1997) store partial elements of good solutions and combine them to create new complete solutions that are improved by the cooperating threads. Central memory approaches exchange complete elite solutions among neighborhood and population-based metaheuristics (Crainic and Toulouse 2003, Le Bouthillier and Crainic 2005).

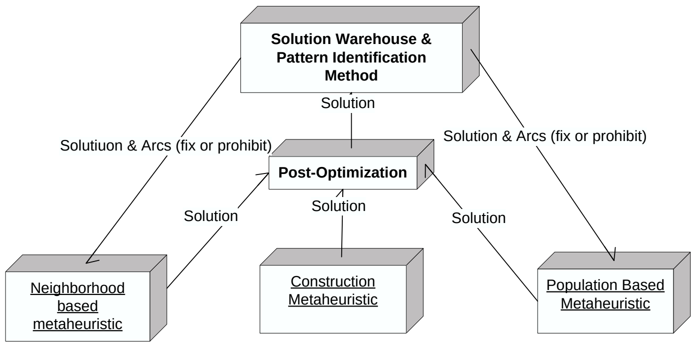

Figure 1: Guided Cooperative Search framework

In a simple Solution Warehouse cooperation scheme (e.g., Le Bouthillier and Crainic 2005), threads share information about their respective good solutions identified so far. When a thread improves the imported solution or when it identifies a new best solution, it sends it to the Solution Warehouse. This scheme is intuitive and simple, and it satisfies the meaningfulness requirement. The selection of the methods involved in Cooperative Search should be oriented towards obtaining: (i) Good quality solutions; (ii) A broad diversity of solutions to facilitate the discovery of promising regions; (iii) The rapid production of intermediate solutions to feed the information exchange mechanisms (iv) A mechanism that combines various solutions to create diversity (v) A mechanism that has the ability to escape local optima.

Independent methods send their improved solutions to the post-optimization algorithms. Theses solutions are considered in-training until they have been post-optimized and sent as adult solutions to the Solution Warehouse. Duplicate solutions received in the Solution Warehouse are eliminated.

All requests for solutions initiated by the independent processes are sent to the Solution Warehouse that responds by sending an adult solution. Solutions are selected randomly according to probabilities biased toward the best ones. The probabilistic ranking is based on the same function used to order solutions in the Solution Warehouse.

The population size in the Solution Warehouse is set relatively to the problem size and the worst results are eliminated as needed. No direct communications take place between processes thus enforcing their independence and the asynchronous mode of exchange. This scheme makes the cooperation design simpler and, eventually, allows modifications of the parallel system by adding new methods or dropping inefficient ones. Moreover, it does not assume any specific problem, which makes it equally relevant for problems where solution components may be easily defined (e.g., the routes in vehicle routing problems) and for problems where such structures are much less apparent (e.g., network design). The goal now is to improve upon this simple cooperating scheme by extracting new knowledge from the information exchanged to yield a more efficient global search.

3 Pattern Identification Mechanism and Global Search

In this section, we introduce a mechanism to extract knowledge from the information exchanged and guide each search method towards promising or unexplored regions of the solution space. It uses a pattern identification mechanism on the solutions present in the Solution Warehouse that is then used to fix or prohibit specific solution attributes (e.g., arcs in network-based problems) for part of the search guided by particular individual metaheuristic. One may therefore constrain the search space of individual metaheuristic in cooperation and thus perform global intensification and diversification phases to guide the exploration of the solution space and control the quality and diversity of the Solution Warehouse population.

The concepts presented in this section may be extended to any combinatorial problem definition and solution attribute but we focus on the class of problems that may be described in terms on inclusion/exclusion of arcs in given networks.

3.1 Pattern definition

Consider the frequency of inclusion of arcs in a given subset of the Solution Warehouse. In particular, this subset may be the entire population, an elite (e.g., with solution in the 10% best), average (between the 10% and 90% best), worst (the last 10%) or solutions from the population obtained from specific metaheuristics. An arc with a high frequency in a given group signals that the metaheuristics participating to the cooperation have often produced solutions that include that arc. When the frequency of inclusion of several arcs is considered, patterns emerge among the solutions of the Solution Warehouse or the specific group examined.

We define a pattern of length n as a subset of arcs of cardinality n = 1, 2, , maximum number of arcs in the problem definition. A frequent (infrequent) pattern relative to a set of arcs is built of arcs with high (low) frequency of appearance in the solutions of the set. High (low) frequency arcs are selected sequentially in decreasing order from the highest (increasing order from the lowest) frequency value.

We may select patterns from specific subpopulations (e.g., elite, average, and worst) and compare the rate of appearance of a specific pattern between them. These comparisons form the basis of the guidance mechanism proposed in this paper.

3.2 Comparing pattern-appearance frequencies among subpopulations

We define an *in-pattern* as a pattern that actually appears in at least one solution of the subset of solutions considered, as opposed to a *statistical pattern* that does not appear in any solution of the subset. A statistical pattern is thus only a consequence of the statistical process of accounting for the frequency of individual arcs

Consider an *in-pattern* of length n common to the three sets of elite, average, and worst solution groups. Two meaningful situations can occur with respect to the frequency of appearance of the pattern in these sets as one moves from the elite to the average to the worst subpopulation: The frequency is either increasing or stable, or decreasing. We call the *in-*

pattern unpromising in the first case and *promising* in the latter.

Promising and unpromising patterns may then be used to constrain for a certain time the search space of particular methods in the cooperation and thus induce global intensification and diversification phases.

3.3 Intensification and diversification in global search

Global search intensification and diversification phases may be triggered by fixing (including) or prohibiting (excluding) arcs in the solutions a given metaheuristic explores during a certain period.

Consider an unpromising pattern of length n. To intensify the search around solutions with "good" attributes, one prohibits the arcs defining the pattern. On the other hand, fixing these arcs for a number of iterations will diversify the search relative to the current set of "good" attribute values. Symmetrically, given a promising pattern of length n, one intensifies the search by fixing the arcs in the pattern, while diversification is obtained by prohibiting the arcs in the pattern.

We define the global search as the cumulative search effort of the individual methods. To prevent individual methods from converging too rapidly, we favor diversification by prohibiting arcs during the initial phases of the global search. Later on, we encourage intensification by fixing promising arcs to enforce the exploration of theses promising regions of the solution space. We also vary the length of the patterns as a mean of modulating the intensity of global diversification and intensification phases and thus influence the evolution of the diversity and quality of solutions in the Solution Warehouse.

At the beginning of the search there is not sufficient information gathered as the solution space is insufficiently explored. Therefore, the Solution Warehouse cannot be considered representative of the search of the individual metaheuristic, even when the population is diverse. Consequently, we build initial patterns of increasing length from the average subpopulation only in order to identify more rapidly promising ones. As the search progresses, patterns from elite and worst subpopulations are built as described above. When a statistical pattern is found, we reduce the length of the pattern until an in-pattern is obtained.

The initial framework is enhanced with a process to identify, manage and send patterns to metaheuristic (figure 1). The appropriate pattern and instructions on fixing or prohibiting arcs are sent along with the solution (selected according to the original criteria). With respect to the individual metaheuristics, each one needs to be modified to cope with the instructions relative to fixing or prohibiting arcs.

4 Guided Cooperative Search for the VRPTW

To illustrate the mechanism described in the previous sections, we developed and implemented a guided Cooperative Search method for the Vehicle Routing Problem with Time Windows (VRPTW). The application is described in this section, whereas experimental results are discussed in the following section.

The VRPTW is a well-known combinatorial problem that has been extensively studied and is thus well suited for benchmarking. We address the single depot VRPTW where one is given a set of customers with known positive demands and specific time intervals when service can be provided. Cordeau et al. (2002) review problem variants, formulations, and solution methods for the VRPTW.

Le Bouthillier and Crainic (2005) presented a cooperative parallel method for the VRPTW based on the Solution Warehouse mechanism presented in Section 2. The cooperation involved two Tabu search methods that perform well sequentially, the Unified Tabu (Cordeau, Laporte, and Mercier 2001) and Taburoute (Gendreau, Hertz, and Laporte 1994), two evolutionary algorithms with Order and Edge Recombination Crossover, respectively, as well as a number of post-optimization methods (2-opt, 3-opt, or-opt, and Ejection Chains) that were used to reduce the number of vehicles and the total traveled distance. Four construction algorithms were used to provide initial solutions to the population.

We applied the proposed cooperation framework described in the previous sections to the parallel method of Le Bouthillier and Crainic (2005). In the VRPTW context, the definition of an in-pattern of length n is straightforward. The problem is defined on a network, where an arc corresponds to a possible route between two customers or between a customer and a depot.

Fixing and prohibiting arcs in the solutions explored by the four metaheuristics is straightforward as well. Fixing (including) arcs always leads to a non empty solution space that, at the most extreme, may reduce to a single solution that represents a very (too) long pattern. Prohibiting (excluding) patterns may lead to an empty space of solution when patterns are too long. To avoid both situations and strike a balance between the number of feasible solutions and the size of the constrained solution space, we limit the pattern length to 25% of the size of the problem.

The wall clock time allocated to the cooperative method is divided into four phases: Two phases of diversification at the beginning to broaden the search, followed by two intensification phases to focus the search around promising regions. The four phases proceed as follows:

- Phase I. Build unpromising in-patterns of frequent arcs in the average subpopulation and prohibit them in the independent metaheuristics;
- Phase II. Prohibit arcs from frequent unpromising in-patterns from the worst subpopulation;
- Phase III. Work with the average subpopulation and fix arcs from frequent promising in-patterns;
- Phase IV. Build frequent promising in-patterns from the elite sub-population and fix the arcs for the meta-heuristic searches.

Pattern lengths are explored in decreasing length in the first two phases and in increasing length in the last two phases, by increments of 1 unit.

5 Computational experiments

The experimentation has a dual objective. On one hand, we aim to compare the guided Cooperative Search to the simple version and to the best performing methods proposed in the literature for the VRPTW (see list of methods and authors at http://www.top.sintef.no as of may 2005) and, thus to validate our claim that the proposed method offers competitive performance in terms of both solution quality and computational effort. On the other hand, we also aim to evaluate the impact of guiding the search towards or away from specific patterns and performing diversification and intensification to control the entropy of the population. Tests have been carried out on the 56 test problems of 100 customers proposed by Solomon and the extended set produced by Homberger and Gehring (1999) with 300 problem instances that vary from 200 to 1000 customers.

For the Taburoute and the Unified Tabu, we use the parameter settings indicated in the original papers (Gendreau, Hertz, and Laporte 1994, Cordeau, Laporte, and Mercier 2001). Solutions in the Solution Warehouse are sorted, first by the number of vehicles, second by a weighted sum, C(p), of attributes: the total time required to serve all customers, the associated total distance and total waiting time at customers, and the sum of the slack left in each time window: C(p)=W1*totalTime + W2*totalDistance + W3*totalWait + W4*totalSlack; Parameters W1 to W4 were set to 1 in all the reported experiments. In-pattern lengths were fixed to at most 25% of the problem size. The four global phases that prohibit or fix arcs have each been allocated 1/4 of the total wall-clock execution time.

In previous research (Le Bouthillier and Crainic 2005), Cooperative Search was found to provide faster results of equivalent or better quality than each of its independent searches. We therefore compare only the simple and guided parallel Cooperative Search with other methods. The entry LC03 and LCK05 in table 1 refers to the simple Cooperative Search and the Guided Cooperative Search respectively.

Runs of 12 min wall-clock time were performed on 5 processors by the cooperative metaheuristics for each of the 100 city problems. Longer running times, equal to those reported by Homberger and Gehring (1999) were allowed for the larger problem instances. These times go up to 50 min wall-clock time for the 1000 customers problem.

Problem	BVH		MB		LC03		LCK05		HG	
	CNV	CTD	CNV	CTD	CNV	CTD	CNV	CTD	CNV	CTD
100 total	**405**	**57 272**	N/A	N/A	407	57412	405	57 360	405	57715
200 total	697	171 715	694	168 573	694	173 062	**694**	**169959**	694	173313
400 total	1 393	410 112	1 389	390 386	1390	410 330	1389	396612	**1388**	**409764**
600 total	2 091	858 040	2 082	796 172	2088	840 583	2086	809494	**2076**	**851681**
800 total	2 778	1 469 790	2 765	1 361 586	2766	1 475 436	2761	1443400	**2755**	**1479802**
1000 total	3 468	2 266 959	3 446	2 078 110	3451	2 225 367	3442	2133645	**3439**	**2236583**
Total	10 832	5 233 888	N/A	N/A	10 796	5 182 188	10 777	5 010 470	**10 757**	**5 208 858**

Table 1: CNV and CTD for 100-1000 customers

Table 1 shows the cumulative number of vehicles (CNV) and distance (CTD) rounded up for the standard Solomon problems and for extended set of problems. Best results are shown in bold face (see list of authors abbreviation and references on http://www.top.sintef.no as of may 2005).

For the problem of 100 customers, we found 24 of the best known results and report the best known Cumulative Number of Vehicle. The guided Cooperative Search (LCK05) reports a new best average for the R1 problem class. For C1 and C2 problem classes, almost all methods found the best known average number of vehicles and the total distance. For the other classes of problems, we found the best known number of vehicles in all instances and at most 0.16% of increase in distance when compared to the bests methods in the litterature.

We obtain a new best known Cumulative Total Distance value for problems of size 200 customers. For all the problem classes, the difference to the other methods is less than 0.19% in terms of the cumulative number of vehicles. In all cases, we improved upon the simple Cooperative Search. We report a slightly higher total number of vehicles compared to **HG** but we report a reduction in the total cumulative distance by 3.96% for all classes.

6 Conclusion and perspective

We presented an enhanced Cooperative Search mechanism that creates new information from exchanged solutions and thus performs global intensification and diversification phases. The proposed framework does not suppose any specific problem structure, even though we illustrated the methodology in the context of problems defined by the inclusion/exclusion of arcs in particular networks. We also applied this methodology to the VRPTW and report very good solutions on the extended benchmark problem sets: 139 best known results found and the best known average on the problems of size 200 customers. The Guided Cooperative Search produced is more robust in term of solution value than its individual metaheuristics or the simple Cooperative Search.

Experimental results showed that pattern identification method yields good information to guide the global search. Patterns of attributes may be constructed independently of particular solution structures and applied to a wide range of combinatorial problems. Patterns of attributes could then be used in various sequential and parallel methods to orient the search in promising regions and out of unpromising region in diversification/intensification phases.

Cooperative Search is quite simple to implement and the pattern identification method provides an easy mechanism to constrain the search within promising regions of the search space or away from unpromising regions. Very good quality solutions were found in linear speed up by combining good off-the-shelf methods without any particular parameter tuning. The quality of the individual methods influences the global search quality. Improved solutions may be found, however, by generating new information from the frequency of pattern appearance in the best solutions visited and by using the new information to guide the global search.

7 Acknowledgements

We are grateful to Jean-François Cordeau for the generous offer of sharing his Unified Tabu search code and to Michel Gendreau for his ideas with the original version of Taburoute. This research has been partially supported by the NSERC through its Discovery Grants program.

References

Clearwater, S.H. and Hogg, T. and Huberman, B.A. (1992). Cooperative Problem Solving. In Huberman, B., editor, *Computation: The Micro and the Macro View*, pages 33–70. World Scientific.

Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M. M., and Soumis, F. (2002). The VRP with Time Windows. In Toth, P. and Vigo, D., editors, *The Vehicle Routing Problem*, SIAM, Monographs on Discrete Mathematics and Applications, chapter 7, pages 157–193. SIAM, Philadelphia, PA.

Cordeau, J.-F., Laporte, G., and Mercier, A. (2001). A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows. *Journal of the Operational Research Society*, 52:928–936.

Crainic, T.G. and Toulouse, M. (2003). Parallel Strategies for Meta-heuristics. In F. Glover and G. Kochenberger, editors, *State-of-the-Art Handbook in Metaheuristics*, pages 475–513. Kluwer Academic Publishers, Norwell, MA.

Gendreau, M., Hertz, A., and Laporte, G. (1994). A Tabu Search Heuristic for the Vehicle Routing Problem. *Management Science*, 40:1276–1290.

Golden, B.L., Laporte, G., and Taillard, É.D. (1997). An adaptive memory heuristic for a class of vehicle routing problems with minmax objective. *Computers & Operations Research*.

Homberger, J. and Gehring, H. (1999). Two Evolutionary Metaheuristics for the Vehicle Routing Problem with Time Windows. *INFOR*, 37:297–318.

Le Bouthillier, A. and Crainic, T.G. (2005). A Cooperative Parallel Meta-Heuristic for the Vehicle Routing Problem with Time Windows. *Computers & Operations Research*, 32(7):1685–1708.

Rochat, Y. and Taillard, É.D. (1995). Probabilistic Diversification and Intensification in Local Search for Vehicle Routing. *Journal of Heuristics*, 1(1):147–167.

Taillard, É.D., Gambardella, L.M., Gendreau, M., and M., Potvin, J.-Y. (2001). Adaptive Memory Programming: A Unified View of Meta-Heuristics. *European Journal of Operational Research*.

Toulouse, M., Thulasiraman, K., and Glover, F. (1999). Multi-Level Cooperative Search. In Amestoy, P., Berger, P., Daydé, M., Duff, I., Frayssé, V., Giraud, L., and Ruiz, D., editors, *5th International Euro-Par Parallel Processing Conference*, volume 1685 of *Lecture Notes in Computer Science*, pages 533–542. Springer-Verlag, Berlin.