Pergamon

Computers Ops Res. Vol. 21, No. 8, pp. 885–893, 1994

0305-0548/94 $7.00+0.00

A STUDY OF DIVERSIFICATION STRATEGIES FOR THE QUADRATIC ASSIGNMENT PROBLEM

JAMES P. KELLY,† MANUEL LAGUNA‡ and FRED GLOVER§

Graduate School of Business and Administration, Campus Box 419, University of Colorado at Boulder, Boulder, CO 80309-0419, U.S.A.

Scope and Purpose—Heuristic search procedures that aspire to find global optima usually require some type of diversification strategy in order to overcome their myopic perspective of the solution space. Traditionally, randomization has been used to accomplish this diversification. In this paper, we present a deterministic approach to diversification that has proved to be much more powerful than simple randomization. This approach uses the solution method, solution history, and the problem structure to move the search into unexplored regions of the solution space. We believe that the concepts presented here can form the foundation for effective diversification strategies for many types of optimization problems.

Abstract—Diversification strategies can be used to enhance general heuristic search procedures such as tabu search, genetic algorithms, and simulated annealing. These strategies are especially relevant to searches that, starting from a particular point, explore a solution path until new exploitable regions are inaccessible, and a new starting point becomes necessary. To date, no one has studied the effect of applying diversification methods independently of other metastrategic components, to identify their power and limitations. In this paper we develop diversification strategies and apply them to the quadratic assignment problem (QAP). We show that these strategies alone succeed in finding high quality solutions to reasonably large QAP instances reported in the literature. We also describe how our diversification strategies can be easily incorporated within general solution frameworks.

1. INTRODUCTION

Search methods based on local optimization often rely on diversification strategies to increase their effectiveness in exploring the solution space defined by a combinatorial optimization problem. Some of these strategies are designed with the chief purpose of preventing search processes from *cycling*, i.e., from indefinitely executing the same sequence of moves. Others are introduced to impart additional robustness or vigor to the search. In tabu search (TS), for example, the inclusion of long term memory functions is generally regarded as a way of incorporating diversification. Genetic algorithms use randomization in component processes such as combining population elements and applying crossover (as well as occastional mutation), thus providing some diversifying power. Simulated annealing likewise incorporates randomization to make diversification a function of temperature, whose gradual reduction correspondingly diminishes the directional variation in the objective function trajectory of solutions generated. Diversification in GRASP (Greedy Randomized Adaptive Search Procedures) is achieved by means of controlled random sampling. In these procedures parameters are adjusted to lower the probability that the construction phase generates a solution that has already been used as a starting point, while keeping the set of starting solutions within a desired quality level (in terms of the objective function value).

†James P. Kelly is an Assistant Professor of Management Science in the College of Business and Administration and Graduate School of Business Administration at the University of Colorado in Boulder. He received his doctoral degree in Applied Mathematics and Operations Research from the University of Maryland. His current research interests are in the area of heuristic combinatorial optimization. Dr Kelly has published papers on topics such as tabu search and simulated annealing in various journals such as *Operations Research* and the *ORSA Journal on Computing.* Currently, he is attempting to use tabu search to construct neural networks for pattern classification.

‡Manuel Laguna is an Assistant Professor of Operations Management in the College of Business and Administration and Graduate School of Business Administration of the University of Colorado at Boulder. He received master's and doctoral degrees in Operations Research and Industrial Engineering from the University of Texas at Austin. He has done extensive research in the interface between artificial intelligence and operations research to develop solution methods for problems in the areas of production scheduling, telecommunications, and facility layout. Dr Laguna co-edited the Tabu Search volume of Annals of Operations Research.

§Professor Glover's résumé appears earlier in this issue.

Diversification strategies in TS methods are designed and used in a number of different ways, as shown by the following examples from the literature. Gendreau *et al.* [1] designed a TS method for the solution of the maximum clique problem that incorporates a random component for move selection and a secondary tabu list to encode a number of previously visited solutions. The secondary tabu list is used to avoid any move that would lead to a solution visited in the past T iterations, where T is a set to values as large as 150 for solution attempts of 300 iterations. In a parallel TS method for large traveling salesman problems (with the number of cities ranging from 500 to 10,000), Fiechter [2] defines a "high-level" tabu search that operates with super-moves. These super-moves correspond to reallocating key portions of the tour, and therefore modify the current trial solution to such an extent that a re-starting mechanism is not necessary. Another approach to diversification is taken by Woodruff and Spearman [3], who introduce the use of the diversification parameter d. This parameter can be viewed as the reciprocal of a lagrangian multiplier in that "low" values result in nearly infinite costs for constraint violation, while "high" values allow searching through infeasible regions. The diversification parameter is also used to control the amount of randomization in a probabilistic version of tabu search, assigning d a role similar to that taken by temperature in simulated annealing.

A different form of diversification in TS procedures is implemented through frequency-based long term memory functions. Skorin-Kapov [4] uses an $n \times n$ matrix to record the number of times a pair of objects exchange locations in an adaptation of tabu search to the quadratic assignment problem. The frequencies are weighted to modify the distances between every pair of locations and force the construction of "diverse" solutions during a re-starting phase. In Laguna and Glover [5] frequency counts are used to bias the selection of moves in TS solution states where no improving moves are available. Applied to single machine scheduling, the frequency count is multiplied by a penalty parameter and added to the move value of every non-improving move. Then, the move with the least penalized move value is selected. This strategy successfully avoids long term cycling and allows the procedure to find improved solutions during late stages of the search process. A long term memory function based on move frequencies is also used in Glover and Laguna [6] to encourage non-improving moves with "low" frequency counts. The definition of low frequency is a function of the total number of moves in the candidate list, the maximum tabu list size (since dynamic sizes are used), and the current iteration number.

The purpose of our study is to measure the merit of diversification strategies independently of more general metastrategies. We have selected the quadratic assignment problem (QAP) as a platform to test our ideas. In Section 2, we briefly review the QAP formulation and some relevant characteristics of this problem along with early solution efforts. The diversification strategies developed here are presented in Section 3. Computational experiments are presented in Section 4. Finally, Section 5 gives conclusions and some guidelines for integrating our findings within more powerful search mechanisms, including those of simulated annealing, genetic algorithms, and tabu search.

2. THE QUADRATIC ASSIGNMENT PROBLEM

Quadratic assignment problems are a class of combinatorial optimization problems with a number of interesting practical applications, see e.g. Burckard [7]. The original formulation belongs to Koopmans and Beckmann [8] who used it to model the location of indivisible economic activities. Although many researchers have developed exact and heuristic methods for these problems, in this section we only review those recently developed procedures that are capable of providing optimal or near-optimal solutions to relatively large QAP instances. The QAP can be either formulated as a 0–1 integer programming problem or simply viewed as a permutation problem with the following objective function

$$\text{Minimize} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} f_{ij} d_{\pi(i)\pi(j)}.$$

In this function f_{ij} is the flow between objects i and j, $d_{\pi(i)\pi(j)}$ is the distance between the locations where objects i and j are placed, and n is the total number of objects (or locations). A feasible solution is then given by $\Pi = \{\pi(1), \pi(2), \ldots, \pi(n)\}$, where $\pi(i)$ is the index of the location that contains object i. If the flows are all set to 1 and the term $d_{\pi(n)\pi(1)}$ is added to the objective function, the problem is transformed to a traveling salesman problem. The QAP belongs to the class of NP-hard problems, as shown by Sahni and Gonzalez [9].

In Skorin-Kapov [4] a TS application is used to solve QAPs. The method, called Tabu-Navigation, uses swap moves (i.e., the exchange in the location of two objects) to search the solution space, and a frequency-based long term memory function for a re-starting mechanism. Tabu-Navigation was tested using both relatively small problems (with $n \leqslant 36$) reported in the literature and a newly generated set with the number of objects ranging from 42 to 90. The "navigation" part of the method refers to the user's active role during the search process. Every time a prespecified maximum number of iterations is reached from a given starting point, the user has the choice of stopping the search or re-starting with a new set of parameters. The method then relies on the "expert" choices of the user, who needs to decide on a new size for the short term memory, whether or not to invoke the long term memory, and the type of aspiration criteria to be applied.

In contrast to the Tabu-Navigation method, Taillard [10] developed a TS procedure with less complexity for the user. Taillard's method also uses swap moves, which seem to be very well suited for QAPs, but incorporates a quick update for the moves in the candidate list at every iteration. This procedure allows the complete evaluation of the swap neighborhood to be performed in $O(n^2)$ time, as shown by Frieze *et al.* [11]. Taillard refers to the method as a "robust" TS implementation, based on the fact that the user only needs to provide a range for the size of the single tabu list. The actual tabu list size is randomly selected from the given range and dynamically changed during the search. Taillard identifies a tabu list size range of $(0.9n, 1.1n)$ as very effective for most of the problems tested. The TS method employs the customary aspiration level criterion of allowing tabu moves to be selected if they lead to solutions better than those previously found. However a second aspiration criterion, incorporating an additional parameter value, is invoked while solving large problems or instances with large flow variance (i.e., individual flows between objects span a large range and are not uniformly distributed). Taillard found this parameter to be "strongly dependent on the problem instance," therefore making his method somewhat less robust. By using 30 randomly generated starting solutions, Taillard was able to improve several of the best known solutions to the set of problems generated in Skorin-Kapov [4].

Dynamic tabu list sizes are also implemented by Skorin-Kapov [12] in the form of moving gaps, an approach proposed by Glover and Hübscher [13]. This improved TS method incorporates intensification and diversification via fixing and freeing objects to and from given locations. As in the case of the Tabu-Navigation, this method also assumes that the user is capable of making intelligent choices for the different parameter values. The new method has the added complexity of requiring the selection of the number of objects to be fixed and the time period for which they will remain fixed. Using this approach in an 11-stage process, Skorin-Kapov was able to find two new current best known solutions to the problems generated in her 1990 paper (for instances with 81 and 90 objects). The procedure involves 9 re-starts of 50,000 iterations each, resulting in a total of 1.2 million iterations for each solution attempt. The study also includes a new set of 6 randomly generated QAP instances with $n=100$, that are subjected only to limited computational experiments.

Another recent study involving QAPs is due to Gavish [14]. A randomized version of his Manifold Search was able to match the best solutions known for the original Skorin-Kapov's problems with up to 72 objects. His method fails in finding the best known solutions to problems with 81 and 90 objects. However, this study reports new best known solutions for the set of problems with 100 objects generated by Skorin-Kapov [12].

In the next section, we describe the diversification strategies developed here that are designed to be used in conjunction with search methods for QAPs.

3. DIVERSIFICATION STRATEGIES AND METHODS

As we have stressed, the heuristic approaches for solving the Quadratic Assignment Problem (QAP) reported in the previous section require a diversification strategy to enable the search to explore new regions of the solution space. Without this diversification, such methods can become localized in a small area of the solution space, eliminating the possibility of finding a global optimum.

To better understand the role of diversification, and the consequences of alternatives for defining what "diversification" means, we present a method for solving the QAP that relies almost entirely on diversification strategies. Although we do not propose that diversification strategies alone can provide the most effective heuristics, we will show that the procedures we devise are competitive with the best known techniques for solving the QAP, with the exception of the highly elaborated tabu search procedure of Skorin-Kapov [12]. Even in this case, we tie the best known solutions for 3 (out of 7) test problems involving up to 90 objects, and obtain solutions close to the best in the remaining cases (e.g., the maximum deviation from the best is 0.05%). More significantly, our approach uses only very rudimentary machinery, and is extremely easy to implement. Its ability to generate such attractive solutions on its own suggests its potential added value as foundation for procedures incorporating more intricate and advanced strategies.

A well-known procedure for finding a solution to the QAP is to generate an initial solution, either randomly or by a constructive process, and then to use a swapping mechanism to exchange objects until no exchanges remain that will improve the objective function. This approach is extremely fast but seldom generates an optimal solution. The method we propose starts in the same way, using a swapping procedure to find a local minimum, and then applies a diversification strategy to incrementally restrict the set of allowed swaps in order to move away from the local minimum. When the current solution is determined to be sufficiently far away from the previous local minimum or if a new best solution is found during this stage, the diversifiction restrictions are lifted and the descending swap procedure is activated once again. When the diversification component is appropriately designed, the heuristic explores large regions of the solution space.

3.1. *A tenet of diversification*

It is appropriate to provide a word of background about the orientation underlying our approach. Often there appears to be a hidden assumption that diversification is somehow tantamount to randomization. Certainly the introduction of a random element to achieve a diversifying effect is a widespread theme among search procedures, and is fundamental to the operation of simulated annealing and genetic algorithms. From an abstract standpoint, there is clearly nothing wrong with equating randomization and diversification, but to the extent that diversity connotes differences among elements of a set, and to the extent that establishing such differences is relevant to an effective search strategy, then the popular use of randomization is at best a convenient proxy (and at worst a haphazard substitute) for something quite different.

Accordingly, the diversification approach presented here is completely deterministic, with no reliance on a randomizing component. Such a component sometimes can be construed as a replacement for memory, and we provide two extremely simple memory devices to establish a more systematic diversifying process. Following a theme introduced in tabu search (but at a more rudimentary level than customarily applied), one is a recency-based memory that achieves a first order form of diversification, and the other is a frequency-based memory that achieves a second order form of diversification. To start the procedure, we need only to generate an initial solution. The manner in which this is done, and the ways that the two memory structures carry out the diversification process, are detailed in the following sections.

3.2. *Generating a starting solution*

The initial solution is constructed by solving a linear assignment problem which assigns objects to locations to minimize a linear cost function. We generate the cost coefficients c_{ij} of this function to be lower bounds on the costs of assigning object i to location j, thus providing a lower bound on the optimal solution as well as a starting solution.

To determine the cost coefficients for the linear assignment problem, define the flow matrix as $\mathbf{F}=(\mathbf{f}_{ij})=(\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_i, \ldots, \mathbf{f}_n)$ and the distance matrix as $\mathbf{D}=(\mathbf{d}_{ij})=(\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_n)$ where $\mathbf{f}_i$ and $\mathbf{d}_j$ are vectors of n nonnegative integers. Column $\mathbf{f}_i$ consists of the flows between object i and the n objects in the problem. Similarly, column $\mathbf{d}_j$ consists of the distances between location j and the n locations in the problem. The cost of assigning object i to location j can be bounded from below by $\mathbf{c}_{ij}=\mathbf{f}'_i * \mathbf{d}'^T_j$, where $\mathbf{f}'_i$ and $\mathbf{d}'_j$ are vectors of $n-1$ nonnegative integers. $\mathbf{f}'_i$ is formed by removing the ith element (zero) from $\mathbf{f}_i$ and reordering the remaining elements in ascending order. Likewise, $\mathbf{d}'_j$ is formed by removing the jth element (zero) from $\mathbf{d}_j$ and reordering the remaining elements in descending order.

3.3. *First order diversification*

Following the initial construction, and the succeeding series of pair-wise exchanges that lead to a local optimum, the algorithm enters the first order diversification stage. This stage in our approach is designed to take the search to a solution that is "maximally diverse" with respect to the local minimum most recently visited. Our notion of diversity in the recency-based context employs two concepts. One is a form of distance concept that characterizes two solutions as being increasingly diverse as their separation increases, where we define separation to be the minimum number of moves to get from one to the other. (We assume the available moves are symmetric, i.e., bidirectional, as in the case of the swap moves employed.)

The second concept is coupled with the first as a means of expressing the "difficulty' of getting from one solution to the next, which we associate with an estimate of the probability that the second solution will be encountered in the set of solutions equally separated from the first. If solutions are distributed so that those with the smallest objective function define a diminishing tail, then the probability of encountering such solutions is relative small. Since we are using a greedy descent procedure to find local minima, the probability of traversing between two solutions, separated by a fixed number of swaps, decreases with the difference between their objective function values. (On the other hand, if good solutions can be encountered with a high probability, then a diversification component is less relevant. The treatment of these solutions as "improbable" is not a liability in case they are more easily found.)

Thus, specifically, starting from a local optimum, we conceive another solution to be diverse in relation to this point if it is distantly separated (relative to the minimum number of moves to reach it) and has an objective function value close to or better than that of the first. Then we anticipate that the probability of reaching the second point from the first is small, particularly by the application of random moves that take the same number of steps.

In a fuller sense, diversity must depend on the relation between multiple solutions, and not just on an association between separate pairs. However, this consideration is also treated to some extent by the ideas at hand. That is, given a low probability of getting from one point to another, the likelihood of returning to the region of an earlier point is also diminished. (Nevertheless, we treat this consideration more fully by means of the second order diversificatin process.) Proceeding from this foundation, our first order diversification strategy takes the following straightforward form.

Denote the most recent local minimum by the permutation $\Pi_{MIN}=\{\pi_{MIN}(1), \pi_{MIN}(2), \ldots, \pi_{MIN}(n)\}$ and the current solution by permutation $\Pi_{CUR}=\{\pi_{CUR}(1), \pi_{CUR}(2), \ldots, \pi_{CUR}(n)\}$. Consider all swaps, $\pi_{CUR}(x)\leftrightarrow\pi_{CUR}(y)$ such that $\pi_{CUR}(x)=\pi_{MIN}(x)$ or $\pi_{CUR}(y)=\pi_{MIN}(y)$. Swaps of this type will always increase the separation from the local minimum. Then, from among the swaps in the indicated category, we choose one that degrades the objective function the least (or improves it the most if improving moves are available). This move may increase, decrease, or keep the objective value constant.

Diversifying moves are made until no moves exist that belong to the indicated set. At that point the algorithm switches back to selecting improving exchanges until reaching a local optimum, and then the procedure is repeated. However, if a new best solution is found during first order diversification, then the algorithm switches immediately to the improving phase.

3.4. *Second order diversification*

The first order diversification component, as noted, only directly serves the goal of characterizing diversity over larger collections of solutions. Thus in addition to maintaining the simple recency-based memory embodied in the vector Π_{MIN}, we keep a frequency-based memory in a matrix $\mathbf{M}$, where m_{ij} counts the number of times that object i occupies location j in the local minima encountered throughout the search history. A recency-based memory is only concerned with remembering the characteristics of the most recent occurrence of a particular event (e.g., the composition of the most recently found local minimum), whereas a frequency-based memory records these characteristics in an accumulative fashion at every occurrence of the event. Our frequency-based memory is extremely simple to maintain and update (i.e., by referring to each Π_{MIN} vector employed in the recency-based memory). By keeping track of the total number m^* of local optima encountered, we may infer from $\mathbf{M}$ the matrix $\mathbf{M}^*$ whose entries $m^*_{ij}=m_{ij}/m^*$ represent the relative number of times an item i occurs in a location j over the set of local optima.

Although more sophisticated uses of **M** and **M*** are possible, we apply them in the second order diversification process in just two alternative ways:

(1) Re-starting. Allow each first order diversification phase to be terminated by a selected cutoff rule, and generate a new starting solution by solving a linear assignment problem with cost coefficients given by $c_{ij}=m_{ij}$. Once this new solution is generated, the first order diversification resumes until again meeting the conditions of the cutoff rule.

(2) Periodic Second-Order Evaluation. After a selected number of local optima are generated during the first order phase, the next sequence of moves (away from the last local optimum) replaces the objective function evaluation with an evaluation that minimizes $m_{iy}+m_{kx}$, or more generally a convex combination of this term and $\mathrm{Max}(m_{iy}, m_{kx})$, where $i=\Pi_{\mathrm{CUR}}(x)$ and $k=\Pi_{\mathrm{CUR}}(y)$. (An incremental version of this rule replaces m_{iy} with $m_{iy}-m_{ky}$ and m_{kx} with $m_{kx}-m_{ix}$, thus favoring exchanges that move objects out of high-frequency locations or into locations that have corresponding low frequencies.) The sequence of moves is continued by the same rules applied in the first order diversification phase, and after visiting a chosen number of local optima (one, in this study) the process then reverts to the standard first order process, initiating the sequence of phases once again.

Although we have not tested it, we note that a penalty variant of (2) results by replacing **M** with **M***, whereon, the contribution of the *second-order evaluation* can be weighted by a penalty factor p and added to the objective function evaluation used in the first order process. (Consideration of such a variant is motivated by the success of penalty methods for diversification in the approaches cited in earlier sections.)

To allow the simplest types of implementations, however, we have elected to avoid variants or advanced calibration alternatives for such rules, and instead investigate only a small number of choices that are context independent (i.e., that do not make use of information about distributions). Details of these choices and their outcomes are given in the following sections.

4. EXPERIMENTAL DESIGN

For an initial test of the diversification method, which is the basis for the first set of outcomes reported in this paper, we have coupled the first order diversification phase only with the *re-starting* component of the second order phase. A cut-off rule is selected for the first order phase that gives it a dominant role in the overall procedure, transferring to the second order re-start only if there is evidence of potential repetition in the first order process. Specifically, during the first order phase we record the objective function values at each local minimum, and if a sequence of objective values of length at least 5 occurs twice in succession, then the second order phase takes over and the method is re-started. The first order diversification phase with the re-starting component by itself yields remarkably good outcomes, as we now show.

A FORTRAN 77 implementation of the method was used to validate the usefulness of our diversification strategies. For all of our experimentation (performed on a DecStation 5000/200), we have used the randomly generated set of problems found in Skorin-Kapov [4, 12]. Our primary goal is to show that the proposed diversification scheme alone represents a competitive method for the solution of QAPs. Furthermore, we contend that the method proposed here is considerably less complex than those discussed in Skorin-Kapov [4, 12] and Taillard [10]. Skorin-Kapov's problems allow us to directly compare the quality of the solutions obtained in four previous research efforts with the solutions found by our approach. Table 1 shows the best solutions found by each of the methods reviewed in Section 2 and the ones obtained using our diversification procedure. An asterisk in this table denotes that the method from the corresponding column was able to improve upon or match the best known solution to a particular problem instance. Solutions to the 100-object instances are not available for the first two TS methods, i.e., Skorin-Kapov [4] and Taillard [10]. Our results in Table 1 were obtained by allowing the diversification procedure to perform a total of 1 million iterations, which roughly corresponds to the amount of computational effort employed in finding the solutions reported in the other studies.

Table 1 shows the competitiveness of our method in terms of solution quality. For problems with up to 90 objects, our approach is able to match 3 of the best known solutions. Our procedure outperforms Gavish's method in problems of sizes 81 and 90. and Taillard's TS procedure in the problem with 90 items. Our solutions to problems of size 49 and 72 are inferior to those found by the rest of the procedures with the exception of the ones reported in Skorin-Kapov [4], however Skorin-Kapov's results were obtained with considerably less computational effort.

For the set of 6 problems with 100 objects, our approach succeeds in finding 5 new best known solutions (for problems A, C, D, E and F). The solutions reported for these problems in Skorin-Kapov [12] were found performing only approx. 1000 iterations, and therefore they cannot be directly compared with either manifold search or our method. Partial comparison, however, is possible by limiting the total number of iterations that our procedure is allowed to perform. The result of such experiment is reported in form of a plot in Fig. 1. This figure shows the change on the percentage deviation from best of the average objective function value over all 100-item problems, during the first 2500 iterations. The horizontal line represents the percentage deviation from the best known solutions achieved by Skorin-Kapov's method. Figure 1 shows that our method is capable of achieving the same level of deviation from best as Skorin-Kapov's TS procedure, in only 1750 iterations.

Table 1. Best solutions found by different solution approaches

Problem size	Tabu search Skorin-Kapov [4]	Tabu search Taillard [10]	Tabu search Skorin-Kapov [12]	Manifold search Gavish [14]	Kelly, Laguna & Glover
42	7932	7906*	7906*	7906*	7906*
49	11,768	11,693*	11,693*	11,693*	11,699
56	17,368	17,229*	17,229*	17,229*	17,229*
64	24,480	24,249*	24,249*	24,249*	24,249*
72	33,378	33,128*	33,128*	33,128*	33,134
81	45,889	45,514	45,504*	45,554	45,517
90	58.180	57,781	57,771*	57,789	57,778
100-A	n/a	n/a	76,691	76,097	76,048*
100-B	n/a	n/a	77,534	77,051*	77,080
100-C	n/a	n/a	74,832	74,056	73,947*
100-D	n/a	n/a	75,348	75,107	74,943*
100-E	n/a	n/a	75,315	74,894	74,881*
100-F	n/a	n/a	75,336	74,719	74,554*

*Best known solution.
n/a = not available.

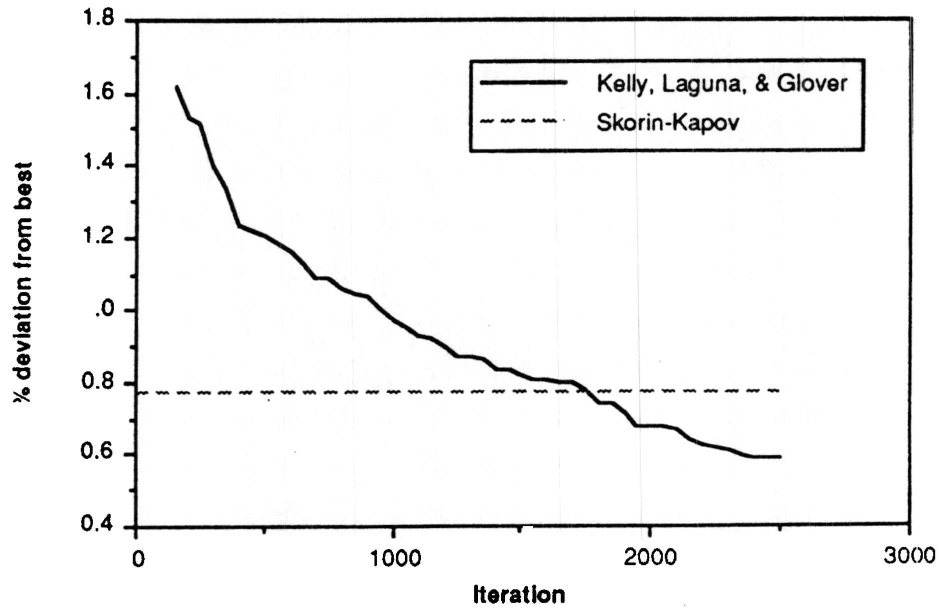

Fig. 1. Percentage deviation from best of the average objective function value over all 100-item problems.

The TS procedures described in Taillard [10] and Skorin-Kapov [12] incorporate probabilistic elements. Taillard's approach uses randomization to generate initial trial solutions, and to select tabu list sizes (within a specified range) during the search. Skorin-Kapov's method adds a random component to the "moving gap" strategy to gradually reduce the size of the initial tabu list. When a random element is part of a procedure, the solutions obtained by repeated applications of the method may vary. Therefore for these solution methods, it is important to compare the average percentage deviation from the best known solutions. Table 2 contains the average percentage deivation values for the methods developed in Taillard [10], Skorin-Kapov [12], and here. The best results reported in Gavish [14] are obtained by a random variant of manifold search, but the average percentage deviations are not available.

The average percentage deviations in Table 2 were calculated using the best solutions obtained after n^2 iterations from every random re-start of the TS method in Taillard [10], and for every long-term memory re-start of the Skorin-Kapov's [4] procedure. For our procedure, average percentage deviations are not required. Instead, the values reported in Table 2 are the exact deviations of the best solutions found after n^2 iterations. These exact deviations show the ability of our approach to obtain solutions that are better, in 4 out of 7 cases, than the estimated expected value of the solutions generated by the methods with random components.

The final set of experiments consist of comparing our results in Table 1 with the results obtained by coupling the first order diversification phase with the *periodic second-order evaluation* (applied when a potential cycle is detected). Additionally, we compare these results with the popular form of diversification that consists of finding local optima from random starting points. The outcomes of these experiments are reported in Table 3. The method with *periodic second-order evaluation* was not used for problems with $n > 81$, since the first order diversification does not cycle within 1 million iterations. This method is able to match the best known solutions to the 42 and 49 object problems. This simple random approach to diversification is clearly inferior in terms of solution quality.

The results in Table 3 show the merit of introducing diversification into a search mechanism by means of a systematic set of rules (or strategies). The common approach of achieving diversification by incorporating a random element is dominated by the deterministic approaches.

Table 2. Average percentage deviation from the best known solutions after n^2 iterations

	Problem size						
Approach	42	49	56	64	72	81	90
Taillard [10]	0.40	0.20	0.50	0.40	0.40	0.40	0.40
Skorin-Kapov [12]	0.26	0.50	0.29	0.28	0.39	0.36	0.34
Kelly, Laguna and Glover	0.00	0.55	0.84	0.25	0.13	0.80	0.15

Table 3. Best solutions found by diversification approaches

Problem size	Re-start second order component	Periodic second order evaluation	Random re-starting
42	7906	7906	7935
	11,699	11,693	11,737
56	17,229	17,235	17,310
64	24,249	24,250	24,425
72	33,134	33,137	33,356
81	45,517	45,517	45,698
90	57,778	—	58,239
100-A	76,048	—	76,481
100-B	77,080	—	77,396
100-C	73,947	—	74,344
100-D	74,943	—	75,211
100-E	74,881	—	75,060
100-F	74,554	—	75,119

5. CONCLUSIONS AND APPLICATIONS

The operational procedure we have applied to implement our concept of first order diversity may be perceived as corresponding to the execution of a "stringent and unforgetting" application of tabu search, in the sense that the restriction imposed on the choice of swaps is equivalent to a tabu restriction that disallows the exchange of objects i and j if both are currently placed in locations different from the ones they occupied in the most recent local minimum. This restriction is not lifted until either no more moves are possible or a solution is found that is better than the best visited so far. Given the determination of this restricted set, our implementation of first order diversification may be interpreted in the context of simulated annealing as assigning a zero probability of selection to all those moves that qualify as members. Our procedure can readily be embedded in simulated annealing by resorting the normal probability evaluations once the diversification stage terminates. Similarly, it can be embedded within a genetic algorithm approach by starting from a selected parent solution and generating new solutions from the diversification approach as candidates to enrich the gene pool. Embedding our procedure in tabu search is straightforward, by activating the approach at a point where the progress of the TS method otherwise begins to diminish (e.g., initiating the diversification procedure relative to some subset of best solutions generated since the last time it was applied).

The straightforward nature and ease of implementation of the diversification approach, and the attractive outcomes it obtains on its own, suggest the potential value of integrating this procedure with other methods.

Acknowledgements—The authors would like to thank Jadranka Skorin-Kapov for making her test problems available. This research was supported in part by the Joint Air Force Office of Scientific Research and Office of Naval Research Contract No. F49620-90-C-0033 at the University of Colorado.

Authors' note—During the course of this study we have become aware of contemporaneous new studies by Chakrapani and Skorin-Kapov, Taillard, and Gavish which extend their previous designs to yield improved performance. We again stress that progressive refinement ultimately must provide the best results, and that the power of the straightforward ideas presented here suggests the value of their role in such refinement.

REFERENCES

1. M. Gendreau, P. Soriano and L. Salvail, Solving the maximum clique problem using a tabu search approach. *Ann. Ops Res.* **41**, 385–404 (1993).
2. C.-N. Fiechter, A parallel taboo search algorithm for large travelling salesman problem. Report ORWP 90/1, Département de Mathématiques, Ecole Polytechnique Fédérale de Lausanne (1990).
3. D. Woodruff and M. L. Spearman, A framework for diversification in tabu search. Technical Report, Department of Industrial Engineering and Management Sciences, Northwestern University (1990).
4. J. Skorin-Kapov, Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing* **2**, 33–45 (1990).
5. M. Laguna and F. Glover, Integrating target analysis and tabu search for improved scheduling systems. *Experts Systems Applic.* **6**, 287–297. (1993).
6. F. Gover and M. Laguna, Bandwidth packing: a tabu search approach. *Mgmt Sci.* **39**, 492–500.
7. R. E. Burckard, Quadratic assignment problems. *European Journal of Research* **15**, 283–289 (1984).
8. T. C. Koopmans and M. J. Beckmann, Assignment problems and the location of economic activities. *Econometrica* **25**, 53–76 (1957).
9. S. Sahni and T. Gonzalez, P-complete approximation problems. *J. Assoc. Comput. Mach.* **23**, 555–565 (1976).
10. E. Taillard, Robust taboo search for the quadratic assignment problem. *Parallel Comput.* **17**, 443 (1991).
11. A. M. Frieze, J. Yadegar, S. El-Horbaty and D. Parkinson, Algorithms for assignment problems on an array processor. *Parallel Computing* **15**, 151–262 (1989).
12. J. Skorin-Kapov, Extensions of a tabu search adaptation to the quadratic assignment problem. *Computers Ops Res.* **21**, 855–865 (1994).
13. R. Hübscher and F. Glover, Applying tabu search with influential diversification to multiprocessor scheduling. *Computers Ops Res.* **21**, 877–884 (1994).
14. B. Gavish, Manifold search techniques applied to the quadratic assignment problem. Technical Report, Owen Graduate School of Management, Vanderbilt University (1991).