1

TABU SEARCH AND ADAPTIVE MEMORY PROGRAMMING — ADVANCES, APPLICATIONS AND CHALLENGES

Fred Glover

College of Business, CB 419
University of Colorado
Boulder, CO 80309
email: Fred.Glover@colorado.edu

ABSTRACT

Tabu search (TS) has provided advances for solving difficult optimization problems in many domains. At the same time, fundamental TS strategies are often not applied as effectively as they might be, and their underlying rationale is often not completely understood. We examine basic concepts and principles of tabu search, emphasizing those that have sometimes led to applying the label "adaptive memory programming" to this class of methods.

The goal of this paper is to focus on key themes that are given inadequate attention in many treatments of tabu search. We also examine basic TS strategies that provide useful alternatives to procedures often associated with "evolutionary" or "genetic" algorithms. Specific tabu search applications are also summarized to provide a clearer understanding of settings where the method is being used. Finally, we include an Appendix that identifies the elements of tabu search that are most neglected in implementations, and that can significantly improve its performance.

1 INTRODUCTION

Metaheuristic procedures have benefitted from numerous advances in recent years. Developments in new implementations of tabu search (TS) have been especially productive. Most striking are the advances enabling difficult practical problems to be handled with greater effectiveness than previously envisioned possible. At the same time, adaptive memory strategies of tabu search are becoming incorporated into other methods, both significantly modifying the operation of these methods and changing the perspectives on which they are based.

This paper focuses on three main themes. First, in Section 1 we examine the adaptive memory rationale that underlies tabu search, and which has sometimes given this approach the label *adaptive memory programming*. We discuss key precepts that underlie the method and issues that are raised for exploiting them more fully. Next, in Section 2 we examine comparisons and contrasts between specific strategies of tabu search, particularly those based on the *scatter* search model, and currently popular strategies based on models of nature, as represented by approaches that are beginning to be embedded in genetic algorithms (GAs). We show how classical GA procedures and their more modern "evolutionary" counterparts can be improved by strategies for combining solutions that are made available by alternative frameworks. A fuller application of these frameworks that incorporates adaptive memory designs of TS offers a useful alternative to GA based approaches. In Section 3 we present a collection of tabu search vignettes, which give brief descriptions of selected tabu search applications and their outcomes. These vignettes identify a diverse range of settings where tabu search has made useful contributions, and suggest the form of additional applications where similar successes may be anticipated.

Each of these sections is self-contained and independent of the others. Our intent is not to present the detailed workings of tabu search, but to disclose fundamental perspectives and principles, especially those that are often overlooked. To reinforce this theme, we include an Appendix that provides a catalog of some of the most commonly neglected features of tabu search, emphasizing those that can significantly improve the quality of results obtained.

2 TABU SEARCH AND ADAPTIVE MEMORY PROGRAMMING

Tabu search has become the focus of numerous comparative studies and practical applications in recent years. Fruitful discoveries about preferred strategies for solving difficult optimization problems have surfaced as a result.

However, sometimes the nature and implications of these discoveries have not been made entirely clear. The reason for this ambiguity is that tabu search has been presented with two faces in the literature, causing it to be viewed as two different methods — one simpler and one more advanced. The simpler method incorporates a restricted portion of the TS design, and is sometimes used in preliminary analyses to test the performance of a limited subset of its components — usually involving only short term memory. The more advanced method embodies a broader framework that includes longer term memory, with associated intensification and diversification strategies. This second approach, due to its focus on exploiting a collection of strategic memory components, is sometimes referred to as *Adaptive Memory Programming* (AMP).

In fact, in both of its forms (though more strongly on the second) the thrust that gives tabu search its distinctive character is the systematic use of adaptive memory — in contrast to the design of "memoryless" approaches like simulated annealing and genetic algorithms, or "rigid memory" approaches like branch and bound and its various AI cousins such as A* search.

More encompassing forms of tabu search that qualify for the AMP label often prove considerably more effective than the primitive forms. In many situations, attempts to rely predominantly on restricted short term memory is an evident handicap — one that invites comparison to attempts to solve a problem by disengaging a part of the brain. Nevertheless, simplified TS approaches are sometimes surprisingly successful. Since they are also frequently quite easy to implement, these approaches will undoubtedly continue to appear in the literature. However, it is important to be aware that they can be strongly dominated by more complete TS methods. Relevant considerations underlying these differences are sketched in the Appendix.

The remainder of this section will focus on certain issues that underly the orientation of TS/AMP approaches, and on questions that are raised by these issues. (Those who are less interested in "precepts and perspectives" can skip to later sections.)

2.1 Is Memory Really a Good Idea?

Memory would seem unquestionably to be an integral component of any search that deserves to be called "intelligent." Yet, surprisingly, some of the methods widely hailed as innovations in artificial intelligence — as applied to optimization — are largely devoid of memory. The avoidance of memory, or more particularly adaptive memory, is not as unreasonable as might be imagined. Adaptive memory characteristically introduces too many degrees of freedom to be treated in "theorem and proof" developments. Researchers who prefer to restrict consideration to processes (and behavior) that can be characterized by rigorous proofs, which are the lifeblood of academic publishing, must focus their efforts in other directions.

Yet there are more subtle and valid reasons to be wary about the use of memory. Malleable forms of memory entail certain dangers — potential pitfalls that go hand in hand with the ability to provide valuable strategic opportunities. These dangers are the price to be paid for the evolution of "intelligent" mechanisms, including biological mechanisms embodied in a brain.[1]

From an evolutionary standpoint, the emergence of memory may be viewed as posing a challenge comparable to the emergence of oxygen, whose corrosive properties (as evolutionary biologists are fond of telling us) caused considerable destruction until organisms adapted to take advantage of them. Analogous perils may well have been created by the emergence of memory, though today we only see the outcomes that survived and flourished. Characteristically, the blind alleys of poorly designed physical structure are conspicuously imprinted on the fossil record. But the blind alleys of poorly regulated mental adjustment — which may have affected survival in far subtler ways — remain invisible to us.[2]

It is noteworthy, however, that we are memory users whose evolutionary line has survived. Since we tend to endow our problem solving schemes with features that reflect our own disposition, such schemes tend to be protected (at least to a degree) from dangers otherwise presented by adaptive memory. Even so, hastily contrived uses of memory can lead to conspicuously undesirable outcomes.

Accordingly, in order to solve complex problems more effectively, TS/AMP approaches seek to uncover the potential gains of adaptive memory without being caught in the traps of ill-considered memory designs. This leads to a quest for "integrating principles," by which alternative forms of memory are appropriately combined with effective strategies for exploiting them. A novel finding is that such principles are sometimes sufficiently potent to yield effective problem solving behavior with negligible reliance on memory. Over a wide

[1] This perspective invites a reconsideration of popular themes: e.g., if an increase in mental capacities creates such attendant risks, the act of acquiring a "knowledge of good and evil," as in the Garden of Eden story, is an understandable basis for expulsion from a simpler (and safer) existence.

[2] Or perhaps they are more visible than we suspect. Just as we maintain vestiges of physical structure now left behind — a spine still not perfectly adapted to upright walking and joints still not perfectly articulated for intricate grasping — we no doubt maintain vestiges of mental patterns now obsolete, ineffective neural organizations that confine the range of our perceptions and reactions.

range of problem settings, however, strategic use of memory can make dramatic differences in the ability to solve problems.

2.2 Points of Departure.

A starting premise of tabu search is that intelligent inhibition plays a critical role in making effective use of memory. This may be conceived as a reflection of an analogous supposition that appropriate forms of inhibition and restraint correspondingly are essential to survival, although we may not always think of such elements as survival tools. (The usual tenet of our culture is that inhibition represents something which must be overcome, rather than something which can provide important advantages when properly utilized.)

The connotation of the "tabu" term in tabu search carries an implication, as it does in other domains, of rules that are contextual and subject to change. This type of variability can range from narrowly confined interaction to highly complex coordination. The potential intricacy of managing such variation understandably may pose an obstacle to rules that are too rigidly constrained by the quest for mathematical precision, but there is a reverse danger of seeking to handle complexity by the expedient of simplistic rules, particularly those that rely heavily on randomization as a substitute for identifying strategic relationships.

Currently it is fashionable to base the design of search mechanisms on a level of organization represented by primitive organisms. But we may legitimately wonder whether intelligent behavior can be adequately encompassed within physical or biological processes that are distant precursors of the forms of organization embodied in our own brains. If there is value in having the capabilities we call human, then it seems questionable to aspire to mimic something less.

There is of course no reason to limit consideration to forms of intelligence that match our own. Evolution presumably may have honed our skills to handle problems that have typically presented themselves in our surroundings. Our prowess may be less impressive for problems confronted in other settings, including problems created as a result of our own technology. A leading goal of TS/AMP research is to identify memory and strategy combinations that have merit in a wide range of contexts, not restricted to those we have commonly encountered by the accident of history. If this pursuit may yield insights into different types of intelligence, that would be a welcome bonus. Conceivably, by this perspective, the field of memory-based search methods may have something useful to contribute to the field of cognitive behavior. Up to now, the complementary realms of search and psychology have been largely isolated from each other. As findings about the connections between adaptive memory processes and improved problem solving become systematized, however, this situation may change.

2.3 Elements of Adaptive Memory.

Adaptive memory involves an attribute-based focus, and depends intimately on the elements of recency, frequency, influence and logic. This simple catalog disguises a surprising range of alternatives — as becomes apparent when the four basic elements are considered in combination, and differentiated for different attribute classes over varying regions and spans of time. The notion of influence, for example, characteristically refers to changes in quality, structure, feasibility and regionality. The logic classification likewise is not limited to a single dimension, but invites distinctions between “sequential logic” and “event driven logic,” whose alternative forms appropriately give rise to different kinds of memory structures.

A number of key questions arise about the nature and interrelation of these elements, which have important implications for designing search methods. A brief listing of some of these questions follows.

1. Which types of solution attributes can be most effectively exploited by adaptive memory? (What is the impact of different exploitation strategies on selecting neighborhoods for conducting the search?)

2. What types of functions are useful for generating new attributes as combinations of others? (What implications do such functions have for *vocabulary building* methods in tabu search? (See Section 2.))

3. What are relevant measures of influence, as reflected in attribute changes caused by moving from one solution to another? (How can these measures assist in isolating characteristics of past trajectories that are relevant for designing current ones?)

4. What thresholds should be established to identify levels of recency, frequency and influence? (What role should these levels play in determining tabu restrictions and aspiration criteria? How can thresholds be used to provide penalties and inducements for selecting particular moves, and for changing the phase of search?)

5. How should probabilities be designed to take advantage of information provided by evaluative measures and thresholds? (Which search functions and domains should be governed by probabilistic variation and which should preferably be treated deterministically?)

6. How may memory be applied most effectively to coordinate the use of compound neighborhoods? (What forms of memory are most useful for ejection chain strategies, as a basis for concatenating component moves into more elaborate alternatives?)

7. What clustering and pattern classification approaches are best suited to take advantage of the search history? (How can these approaches be coordinated to improve intensification and diversification strategies in tabu search?)

8. Which special adaptations of memory and learning give best results for highly context-specific problems? (Conversely, which "generic" forms are most effective over wide ranges of problems whose structure is not predicated in advance?)

Basic considerations and research directions associated with these questions are examined in the following sections, and in the discussion of commonly neglected aspects of tabu search in the Appendix.

3 MODELS OF NATURE — BEYOND "GENETIC" METAPHORS

One of the most misunderstood aspects of tabu search is the connection between a subset of its strategies and certain approaches embodied in genetic algorithms (GAs). TS researchers have tended sometimes to overlook the part of the adaptive memory focus that is associated with strategies for combining sets of elite solutions. Complementing this, GA researchers have been largely unaware that such a collection of strategies outside their domain exists. This has quite possibly been due to the influence of the "genetic metaphor," which on the one hand has helped to launch a number of useful problem solving ideas, and on the other hand has also sometimes obscured fertile connections to ideas that come from different foundations.

To understand the relevant ties, it is useful to go back in time to examine the origins of the GA framework and of an associated set of notions that became embodied in TS strategies. We will first sketch the original genetic algorithm design, as characterized in Holland (1975). Our description is purposely somewhat loose, to be able to include approaches more general than the specific proposals that accompanied the introduction of GAs. Many variations and changes have come about over the years, as we subsequently observe.

Genetic Algorithm Template

1. Begin with a population of binary vectors.
2. Operate repeatedly on the current generation of vectors, for a selected number of steps, choosing two "parent vectors" at random. Then mate the parents by exchanging certain of their components to produce offspring. (The exchange, called "crossover," was originally designed to reflect the process by which chromosomes exchange components in genetic mating and, in common with the step of selecting parents themselves, was organized to rely heavily on randomization. In addition, a "mutation" operation is occasionally allowed to flip bits at random.)
3. Apply a measure of fitness to decide which offspring survive to become parents for the next generation. When the selected number of matings has been performed for the current generation, return to the start of Step 2 to initiate the mating of the resulting new set of parents.
4. Carry out the mating-and-survival operation of Steps 2 and 3 until the population becomes stable or until a chosen number of iterations has elapsed.

A somewhat different model for combining elements of a population comes from a class of relaxation strategies in mathematical optimization known as surrogate constraint methods. The goal of these approaches is to generate new constraints that capture information not contained in the original problem constraints taken independently, but which is implied by their union. While this may seem somewhat removed from the concerns that motivated the development of genetic algorithms, we will see that some unexpected connections emerge.

The information-capturing focus of the surrogate constraint framework has the aim of developing improved methods for solving difficult optimization problems by means of (a) providing better criteria for choice rules to guide a search for improved solutions, (b) inferring new bounds (constraints with special structures) to limit the space of solutions examined. (For background, see Glover

(1965, 1968, 1975), Greenberg and Pierskalla (1970, 1973), Karwan and Rardin (1976, 1979), and Freville and Plateau (1986, 1993).) Based on these objectives, the generation of new constraints proceeds as follows.

Surrogate Constraint Template

1. Begin with an initial set of problem constraints (chosen to characterize all or a special part of the feasible region for the problem considered).

2. Create a measure of the relative influence of the constraints as basis for combining subsets to generate new constraints. The new (surrogate) constraints, are created from nonnegative linear combinations of other constraints, together with cutting planes inferred from such combinations. (The goal is to determine surrogate constraints that are most effective for guiding the solution process.)

3. Change the way the constraints are combined, based on the problem constraints that are not satisfied by trial solutions generated relative to the surrogate constraints, accounting for the degree to which different source constraints are violated. Then process the resulting new surrogate constraints to introduce additional inferred constraints obtained from bounds and cutting planes. (Weaker surrogate constraints and source constraints that are determined to be redundant are discarded.)

4. Progressively update and modify the surrogate constraints to take advantage of different current stages of the solution method and different regions of the solution space. Repeat the process as long as the solution method for the original problem continues to iterate.

A natural first impression is that the surrogate constraint design is quite unrelated to the GA design, stemming from the fact that the concept of combining constraints seems inherently different from the concept of combining vectors. However, this difference is not as great as it may initially appear. In many types of problem formulations, including those where surrogate constraints were first introduced, constraints are summarized by vectors. More particularly, over time, as the surrogate constraint approach became embedded in both exact and heuristic methods, variations led to the creation of a "primal counterpart" called *scatter search*. The scatter search approach combines solution vectors by rules patterned after those that govern the generation of new constraints, and specifically inherits the strategy of exploiting linear combinations and inference (Glover (1977)).[3]

Accordingly, following the principles that underlie the surrogate constraint design, the scatter search process is organized to (1) capture information not contained separately in the original vectors, (2) take advantage of auxiliary heuristic solution methods (to evaluate the combinations produced and to actively generate new vectors), (3) make dedicated use of strategy instead of randomization to carry out component steps.

We sketch the scatter search approach in its original form and identify some novel connections and contrasts to GA methods. Then we examine extensions that additionally take advantage of the memory-based designs of tabu search.

Scatter Search Procedure.

1. Apply heuristic processes to generate a starting set of solution vectors (trial points). Designate a subset of the best vectors to be reference points. (Subsequent iterations of this step, transferring from Step 3 below, incorporate advanced starting solutions and best solutions from previous history as candidates for the reference points.)

2. Form linear combinations of subsets of the current reference points to create new points. The linear combinations are:

   (a) chosen to produce points both inside and outside the convex region spanned by the reference points.

   (b) modified by generalized rounding processes to yield integer values for integer-constrained vector components.

3. Extract a collection of the best points generated in Step 2 to be used as starting points for a new application of the heuristic processes of Step 1. Repeat these steps until reaching a specified iteration limit.

Two particular features of the scatter search proposal, which will be elaborated later, deserve mention. The use of clustering strategies is suggested for selecting subsets of points in Step 2, which allows different blends of intensification and

[3] The motivation for this development goes beyond the goal of producing a primal analog for the surrogate constraint approach. In situations where surrogate constraint relaxations yield a duality gap, the natural response is to combine elite solutions that "rim" this gap as a basis for exploiting information that may be contained in their union.

diversification by generating new points "within clusters" and "across clusters." Also, solutions selected in Step 3 as starting points for re-applying heuristic processes are not required to be feasible, since heuristics proposed to accompany scatter search include those capable of starting from an infeasible solution.

In sum, scatter search is founded on the following premises.

(P1) Useful information about the form (or location) of optimal solutions is typically contained in a (sufficiently diverse) collection of elite solutions[4].

(P2) When solutions are "combined" as a strategy for exploiting such information, it is important to provide for combinations that can extrapolate beyond the regions spanned by the solutions considered, and further to incorporate heuristic processes to map combined solutions into new points.

(P3) Taking account of multiple solutions simultaneously, as a foundation for creating combinations, enhances the opportunity to exploit information contained in the union of elite solutions.

The fact that the heuristic processes of scatter search (as referred to in *(P2)*) are not restricted to a single uniform design, but represent a varied collection of procedures, affords strategic possibilities[5] whose implications are examined in the Appendix.

The table on the following page traces the links between the conceptions underlying scatter search and conceptions that have been introduced over time as amendments to the GA framework.

These innovations in the GA domain, which have subsequently been incorporated in a wide range of studies, are variously considered to be advances or heresies according to whether they are viewed from liberal or traditional

[4] Useful information may also be contained in bad solutions. However, such solutions are usually much more numerous and varied than good ones, and consequently there is less advantage in trying to make use of them. On the other hand, valuable information can be contained in trajectories from bad solutions to good solutions (or from good solutions to other good solutions), and such trajectory-based information is one of the elements that tabu search seeks to exploit.

[5] This theme also shares a link with the original surrogate constraint proposal, where heuristics for surrogate relaxations are introduced to improve the application of exact solution methods. In combination, the heuristics are used to generate strengthened surrogate constraints and, iteratively applied, to generate trial solutions for integer programming problems.

Scatter Search Features (1977) Incorporated into Non-Traditional GA Approaches

- Introduction of "flexible crossover operations." (Scatter search combinations include all possibilities generated by the early GA crossover operations, and also include all possibilities embedded in the more advanced "uniform" and "Bernoulli" crossovers (Ackley (1987), Spears and DeJong (1991)). Path relinking descendants of scatter search provide further possibilities, noted subsequently.)

- Use of heuristic methods to improve solutions generated from processes for combining vectors (Muhlenbein et al. (1988), Ulder et al. (1991)), (Whitley, Gordon and Mathias (1994)).

- Exploitation of vector representations that are not restricted to binary representations (Davis (1989), Eschelman and Schaffer (1992)).

- Introduction of special cases of linear combinations for operating on continuous vectors (Davis (1989), Wright (1990), Bäck et al. (1991), Michalewicz and Janikow (1991)).

- Use of combinations of more than two parents simultaneously to produce offspring (Eiben et al. (1994), Mühlenbein and Voight (1996)).

- Introduction of strategies that subdivide the population into different groupings (Mühlenbein and Schlierkamp-Voosen (1994)).

perspectives. Significantly, their origins are somewhat diffuse, rather than integrated within a single framework[6]. It is clear that a number of the elements of the scatter search approach remain outside of the changes brought about by these proposals. A simple example is the approach of introducing adaptive rounding processes for mapping fractional components into integers. There also has conspicuously been no GA counterpart to the use of clustering to create strategic groupings of points, nor (as a result) to the notion of combining points according to distinctions between membership in different clusters. (The closest approximation to this has been the use of "island populations" that evolve separately, but without concern for analyzing or subdividing populations based on inference and clustering. The relevance of such matters, and of related conditional analyses, also is discussed in the Appendix.)

The most important distinction, however, is the link between scatter search and the theme of exploiting history. The prescriptions for combining solutions within scatter search are part of a larger design for taking advantage of information about characteristics of previously generated solutions to guide current search. In retrospect, it is perhaps not surprising that such a design should share an intimate association with the surrogate constraint framework, with its emphasis on extracting and coordinating information across different solution phases. This orientation, which takes account of elements such as the recency, frequency and quality of particular value assignments, has become the foundation of notions incorporated within tabu search. (The same reference on surrogate constraint strategies that is the starting point for scatter search is also often cited as a source of early TS conceptions.) By this means, the link between tabu search and so-called "evolutionary" approaches becomes apparent[7].

---

[6] The "press" for the GA approach suggests by contrast that it is not subject to such variation, but represents a manifestation of immutable natural law. An amusing quote from the January 16, 1996 issue of the Wall Street Journal is illustrative: "Three billion years of evolution can't be wrong," [according to a genetic algorithm pioneer].... "It's the most powerful algorithm there is."

[7] The term *evolutionary* has undergone an interesting evolution of its own. By a novel turn, the term "mutation" in the GA terminology has become reinterpreted to refer to any form of change, including the purposeful change produced by a heuristic process. As a result, all methods that apply heuristics to multiple solutions, whether or not they incorporate strategies for combining solutions, are now considered kindred to genetic algorithms, and the enlarged collection is labeled "evolutionary methods." This terminology accordingly has acquired the distinction of embracing nearly every kind of method conceivable.

Scatter Search Extensions
■ Tabu search memory is used to select current reference points from a historical pool (Glover (1989, 1994a)).
■ Tabu search intensification and diversification strategies guide the generation of new points (Mulvey (1995), Zenios (1996), Fleurent et al. (1996)).
■ Solutions generated as "vector combinations" are further improved by explicit tabu search guidance (Trafalis and Al-Harkan (1995), Kelly, Laguna and Glover (1996), Fleurent et al. (1996)).
■ Directional rounding processes focus the search for feasible zero-one solutions allowing them to be mapped into convex subregions of hyperplanes produced by valid cutting plane inequalities (Glover (1995a)).
■ Neural network learning is applied to filter out promising and unpromising points for further examination, and pattern analysis is used to predict the location of promising new solutions (Kelly, Laguna and Glover (1996)).
■ Mixed integer programming models generate sets of diversified points, and yield refined procedures for mapping infeasible points into feasible points (Kelly, Laguna and Glover (1996)).
■ Structured combinations of points take the role of linear combinations, to expand the range of alternatives generated (Glover (1994a)).

3.1 Modern Forms and Applications of Scatter Search

Recent implementations of scatter search (cited below) have taken advantage of the implicit learning capabilities provided by the tabu search framework, leading to refined methods for determining reference points and for generating new points. Current scatter search versions have also introduced more sophisticated mechanisms to map fractional values into integer values. This work has produced new theorems about searches over spaces of zero-one integer variables. Special models have also been developed to allow both heuristic and exact methods to transform infeasible trial points into feasible points. Finally, scatter search has been generalized to produce a broader of methods called *path relinking* methods, which offer a wide range of mechanisms for creating productive combinations of reference solutions. A brief summary of some of these developments appears in the table on the following page.

Implementation of various components of these extensions have provided advances for solving general nonlinear mixed discrete optimization problems with both linear and nonlinear constraints.

Path Relinking

From a spatial orientation, the process of generating linear combinations of a set of reference points may be characterized as generating paths between and beyond these points (where points on such paths also serve as sources for generating additional paths). This leads to a broader conception of the meaning of creating *combinations* of points: by natural extension, we may conceive such combinations to arise by generating paths between and beyond selected points in neighborhood space, rather than in Euclidean space (Glover (1989, 1994b)).

The character of such paths is easily specified by reference to attribute-based memory, as used in tabu search. In particular, it is only necessary to select moves in a neighborhood space that perform the following simple function: upon starting from an *initiating solution*, the moves must progressively introduce attributes contributed by a *guiding solution* (or reduce the distance between attributes of the initiating and guiding solutions). The process invites variation by interchanging the roles of the initiating and guiding solutions, inducing each to move simultaneously toward the other as a way of generating combinations. (When the goal for combining the solutions can be expressed as an optimization model, algorithmic processes may appropriately be incorporated to generate the moves, as in the case of vocabulary building approaches subsequently described. Related approaches are also described in Aggarwal, Orlin and Tai (1996) and Balas and Niehaus (1996).)

Multiparent path generation possibilities emerge by considering the combined attributes provided by a set of guiding solutions, where these attributes are weighted to determine which moves are given higher priority. The generation of such paths in neighborhood space characteristically "relinks" previous points in ways not achieved in the previous search history, hence giving the approach its name of *path relinking*.

Neighborhoods for this process may differ from those used in other phases of search. For example, they may be chosen to *tunnel through* infeasible regions that may be avoided by other neighborhoods. Such possibilities arise because feasible guiding points can be coordinated to assure that the process will reenter the feasible region, without danger of becoming "lost."

The scope of strategies made available by path relinking is significantly affected by the fact that the term *neighborhood* has a broader meaning in tabu search than it typically receives in the popular literature on search methods. Often, the neighborhood terminology refers solely to methods that progressively transform one solution into another. Such neighborhoods are called *transition neighborhoods* in tabu search, and are considered as merely one component of a collection of neighborhoods that also include *constructive* and *destructive* neighborhoods[8].

In addition, tabu search characteristically endows such a collection of neighborhoods with the ability to operate in regions beyond those visited by standard procedures for generating solutions. Strategic oscillation approaches in TS, for example, include variations that build solutions beyond the point of "complete construction," and more generally introduce complementary constructive and destructive processes that go beyond standard boundaries in both directions. By selecting neighborhoods that are relevant to a given problem setting, drawing on this expanded interpretation of a neighborhood, path relinking automatically provides solution combination procedures that are appropriate for specific contexts. This is in noteworthy contrast to the situation encountered with genetic algorithms, where each new class of problems initiates a search for "new crossovers," in order to overcome the limitations of classical models. (The possibility of creating combinations that exploit information from the problem

[8] This distinction seems either "superficial" or "perverse," according to different perspectives, yet disregarding it has led to a variety of misconceptions in the literature, and to designs for solution methods that are inappropriately limited.

setting is also sometimes still overlooked in GA approaches, due to the lingering preference for models that are "context free.")

Vocabulary Building

Recent discoveries point to a feature of the path relinking model that deserves special consideration, involving a connection to the tabu search strategy of *vocabulary building*. The basic idea of this approach is to identify meaningful fragments of solutions, rather than focusing solely on full vectors, as a basis for generating combinations. A pool of such fragments is progressively enriched and assembled to create larger fragments, until ultimately producing complete trial solutions. In some settings these fragments can be integrated into full solutions by means of optimization models (Glover (1992), Glover and Laguna (1993)). Procedures using this design have been developed with highly successful outcomes by Rochat and Taillard (1995) and Kelly and Xu (1995). Approaches that heuristically assemble fragments have also been successfully implemented by Taillard et al. (1995) and Lopez, Carter and Gendreau (1996).

Vocabulary building effectively may be conceived as an instance of path relinking. There are two key objectives: (1) to identify a good collection of reference points, in this case consisting of "partial solutions" (which include the fragments), and (2) to identify paths in neighborhood space that will unite components of these partial solutions, with suitable attendant modifications, to produce complete solutions. (Again it is important to keep in mind that neighborhood spaces include constructive and destructive spaces, as well as transition spaces. Attributes of different partial solutions may be imperfectly compatible, and hence the synthesis of such partial solutions can benefit from multiple or "compound" transformations to create effective linkages.) As a special instance, of course, solution fragments may be united by linear combinations, as in scatter search.

The crucial element of adaptive memory that permeates these alternative modes of combination, and binds them to other tabu search strategies, affords challenging opportunities for research into the nature and meaning of "intelligent combinations." To the extent that the evolutionary label is coming to be applied to increasingly broad domains, researchers who have previously adopted a GA perspective may find a shared interest in exploring the connections that emerge from complementary elements of the TS framework.

4 TS/AMP VIGNETTES

This final section provides a collection of "vignettes" that briefly summarize applications of tabu search in a variety of settings. These vignettes are edited versions of reports by researchers and practitioners who are responsible for the applications[9].

4.1 Protein Conformation Lattice Model Using Tabu Search

The determination of the three-dimensional structure of a protein from a given sequence of amino acids is one of the most challenging unsolved problems in the science of molecular biology. There have been many computer models designed to solve the protein folding problem. All computer models, though employing different types of energy minimization, can be expressed as the global optimization of a non-convex potential energy function. The basic difficult in solving these models is the existence of multiple local minimizers. Recently, there have been various approaches used to solve these models arising from protein folding.

Lattice models have been used by many researchers to describe the protein folding mechanism. This is motivated from two aspects of research interests. On one side, scientists (with practical insight) are hoping to use some lattice structures to obtain initial solutions of protein conformations, with the assumption that an optimal or near-optimal native state can be obtained by relaxing the monomers around the lattice structure. On the other side, acknowledging that we do not understand the mechanism of protein folding, scientists are seeking to understand protein conformations by using lattice models.

Pardalos, Liu and Xue (1995) design an algorithm for a class of lattice models using tabu search and test their approach with a chain of 27 monomers. The algorithm was developed in C and tested using the same data for a fundamental test set published by the American Mathematical Society. Among the protein sequences tested, only a few sequences fail to match the best results previously

[9] A debt of gratitude is owed to the individuals whose contributions have made this summary possible. (Deficiencies in describing their work are solely due to the current editing, and should not be interpreted to reflect shortcomings of the original reports. In a number of cases, the work cited presents only a small sampling of significant contributions by the authors referenced.)

reported. In all other cases (from a set of 200 examined) the tabu search method obtains results as good or better than the previous best.

4.2 Optimization of B-ISDN Telecommunication Networks

Costamagna, Fanni and Giacinto (1995) develop a Tabu Search algorithm for topological optimization of broad band communication networks whose structure is based on a single exchange and on a number of multiplexing centers.

The topology of the network is represented by an undirected graph G(N,A). The set of nodes N represents locations of both existing or possible multiplexers, and location of users. The set of arcs A represents the possible communication links that may be used to connect the users, the multiplexers and the exchanges among them (constituting a cable conduit graph). This graph contains all the information about the area in which the network must be built.

The design problem consists of choosing a spanning tree T of G that connects all the users to the multiplexers through the distribution network, and the multiplexers to the exchange through the transport network, allowing the overall cost of the plant be minimized.

An empirical study was performed comparing TS with three other methods: a Simulated Annealing (SA) method, a genetic algorithm (GA) and a heuristic "Add & Drop" procedure, in terms of computational time and cost. The TS approach reached better configurations in a time equal to or lower than that required by other algorithms. The work has been supported by Marconi, Genova, Italy. Thesis awards have been also granted by S.I.P. S.p.A. (now Telecom Italia), and the Italian public telephone company, which also provided cost data and evaluation of results.

4.3 Tabu Search for Location Analysis: The P-Median Problem

Many location analysis problems are prime targets for tabu search techniques. One such binary decision problem, the p-median problem, can be stated as follows: given a graph G = (V,E), the goal to find a set of nodes, S, of size p, such that the weighted sum of the distances from the remaining nodes (those of V-S) to the set S is minimized. Rolland, Schilling and Current (1995) provide a tabu search algorithm that utilizes single node transfers with a simple short-term memory structure, and a strategic oscillation scheme that allows the procedure to search through an infeasible solution space. Long-term memory structures are used to penalize moves that occur inordinately often. The resulting tabu search procedure outperformed all known heuristics both with respect to solution quality and computational effort.

4.4 Scheduling in Manufacturing Systems

Effective scheduling in manufacturing systems leads to the reduction of manufacturing costs (inventory costs, labor costs, etc.) and improves the operational efficiency of management. The most frequently used and extensively studied problems in the literature are (A) the job shop problem and (B) the flow shop problem. In addition, a basic model for a broad family of cases called flexible flow line scheduling problems is given by the problem known as (C) the flow shop problem with parallel machines. Industrial applications arise in computer systems, telecommunication networks, and the chemical and polymer industries.

Nowicki and Smutnicki (1993, 1994, 1995) have developed effective tabu search methods for problems A, B, and C to optimize the makespan criterion. These algorithms employ a classical insertion neighborhood which is significantly reduced by a candidate list strategy for removing useless moves, in order to concentrate on "the most promising part" of the neighborhood.

The proposed algorithms employ a short-term memory tabu list which stores attributes of visited solutions, represented by selected pairs of adjacent jobs on a machine. Linked intensification and diversification occurs by storing the best solutions collected during the search on a list of limited length. An extended sequence of unproductive steps triggers a 'back jump' on the search trajectory to the nearest elite solution, which is recovered together with its associated search history as a basis for re-initiating the search.

Implementations made on a PC are able to improve significantly the best known solution found by other algorithms. Computation times are only a few minutes for instances of A&B problems containing 10,000 operations, and for instances of C problems containing 3,000 operations. An extensive comparative study shows the significant superiority of TS over other approaches including iterative improvement, genetic search, simulated annealing, threshold accepting, constraint satisfaction, neural networks, and other local search methods (see Vaessens, Aarts and Lenstra (1995)).

4.5 Tabu Search for Designing Optical Telecommunication Networks

The increased complexity and globalization of today's world has been accompanied by the emergence of optical networks as flexible, fast, efficient, and reliable media for transferring information. In this domain, congestion minimization presents one of the main challenges of telecommunication network design. The problem often includes the goals of devising efficient routing and management techniques in case of network failures. An improved approach for minimizing congestion in optical networks, based on tabu search, has been developed by Skorin-Kapov and Labourdette (1995).

Further motivation for this work arises because changing traffic conditions create a need for fast algorithms to re-arrange logical connections. Algorithms that quickly obtain very high quality solutions are mandatory in order to optimize the use of network with respect to a given criterion. The goal is to find the logical connection diagram and routing of flow which minimizes the maximum congestion on a link. (This goal also effectively increases the relative capacity of the network.)

The tabu search approach for this problem has generated improved solutions for data sets established to provide comparative benchmarks. The approach not only improves on previous results, but has been calibrated by Skorin-Kapov and Labourdette to identify performance characteristics for different parameter values, and on different patterns of input data. The outcomes also yield guidelines for solving larger problems.

4.6 Automated Guided Vehicle Systems Flowpath Design Applications

Automated Guided Vehicle Systems (AGVSs) have been of great interest to industry for the last two decades. The number of applications of these systems has increased to a point where AGVSs are considered to be a basic concept in material handling. Although initial applications of AGVSs were generally limited to warehouses, in recent years an increasing number of applications in manufacturing systems have been reported.

Chiang and Kouvelis (1994a) address the flowpath design issue of AGVSs. The authors concentrate on the design of unidirectional flowpaths (where vehicles are restricted to travel only on one direction along a given segment of the flowpath), and develop different versions of simulated annealing and tabu search algorithms for the design of unidirectional AGVSs. Extensive computational results indicate that a tabu search implementation with the use of a frequency based memory structure dominates all tested heuristics in terms of solution quality, with an impressive average performance over 45 test problems of less than 0.85% deviation from optimality.

4.7 Graph Theory: Uniform Graph Partitioning

The uniform graph partitioning problem may be described as follows. Given a graph G = (V,E), where $|V| = m = 2n$, we seek a partition of V into two node sets V1 and V2 such that V = V1 + V2 and $|V1| = |V2| = n$. The goal is to identify such a partition that minimizes the sum of the cost of edges (i,j) where $i \in V1$ and $j \in V2$. Rolland, Pirkul and Glover (1995) have developed a tabu search algorithm for this problem that outperformed all other heuristics tested (including the reported best versions of simulated annealing and the Kernighan-Lin approach), both with respect to solution quality and computational requirements. A key element of the tabu search was a strategic oscillation that drove the search through infeasible solution configurations, causing the cardinality of the node sets to grow and shrink in coordinated waves. (Additional recent important tabu search developments for this problem are provided by Dell'Amico and Maffioli (1996).)

4.8 Tabu Search for Audit Scheduling

Scheduling problems often become increasingly difficult as they acquire greater realism. Audit scheduling adds complexities to traditional scheduling. Characteristically, in such problems the processing units (the auditors):

1. have unequal processing times

2. are not all fit (educated) for all jobs
3. are not always available (yet must work at least a minimum amount, and at most a maximum amount of time)
4. are movable (at a cost), and can transfer between various projects: hence introducing sequence dependent setup costs and times

Dodin, Elimam, and Rolland (1995) develop a tabu search procedure that utilizes traditional dispatching rules (such as forward loading) with the short- and long-term memory structures of tabu search. The tabu search intensifies the search using short-term memory, and diversifies the search using controlled dispatching, long-term memory and candidate lists. Computational tests show the tabu search has approach produces schedules superior to those obtained by heuristics traditionally applied to these problems.

4.9 Mapping Tasks to Processors to Minimize Communication Time in a Multiprocessor System

Connectionist machines are attracting widespread attention for their value as an embodiment of massively parallel computer architecture. This is particularly true for solving combinatorial optimization problems arising in a variety of engineering applications. At the same time, the goal of designing and implementing a connectionist machine as effectively as possible introduces challenging optimization problems.

An important problem is to minimize the communication time required by a connectionist machine. Communication time often is a substantial determinant of overall cost and efficiency. In a significant class of applications, such as finite element analysis, the communication pattern is static. The memory locations defining the source and destinations of messages do not change in these applications, but only the communicated data varies. Improved designs for allocating processors to chips according to the structure of their communication pattern offer considerable potential for savings in cost and time.

Chakrapani and Skorin-Kapov (1995) have developed an effective tabu search method for the problem of mapping tasks to processors to minimize communication time in a multiprocessor system. The method incorporates a parallel processing implementation which includes tabu search memory and guidance mechanisms for iteratively selecting pairs of tasks and swapping their processor assignments. The implementation employs two levels of parallelism. First, the candidate tasks to be swapped are identified in parallel. Second, more than one pair of tasks are swapped in a single iteration. This strategy is designed to operate with efficient approximations that allow inaccurate (incomplete) information for evaluating moves. The authors propose a diversification strategy which makes the search robust under these circumstances. Due to its robust parallel implementation, the algorithm can be used to develop heuristics for "quasi-dynamic" communication patterns, in which the task graph changes slowly with time.

4.10 Multiprocessor Task Scheduling in Parallel Programs

When parallel application programs are executed on MIMD machines, the parallel portion of the application can be speeded up according to the number of processors allocated to it. In a homogeneous architecture, where all processors are identical, the sequential portion of the application will have to be executed in one of the processors, considerably degrading the execution time. In a heterogenous structure, where a faster processor, responsible for executing the serial portion of the parallel application and is tightly coupled to other smaller processors, higher performance may be achieved. The procedure of assigning tasks to processors (task scheduling) is more complex in the heterogeneous case, where the processors have distinct processing speeds.

Porto and Ribeiro (1995a) have applied the tabu search metaheuristic to the task scheduling problem in a heterogeneous multiprocessor environment under precedence constraints. A series of different tabu search parameters and strategies were studied side-by-side with a variety of task precedence graphs (topology, number of tasks, serial fraction, service demand of each task) and system configurations (number of processors, architecture heterogeneity measured by the processor power ratio). The algorithm showed itself to be very robust and effective, systematically improving by approximately 25% the makespan of the solutions obtained by the best greedy algorithm used to provide an initial solution.

4.11 Probabilistic Diversification and Intensification in Local Search for Vehicle Routing

Rochat and Taillard (1995) develop a probabilistic TS technique to diversify, intensify and parallelize almost any local search for almost any VRP. This technique makes the local search more robust since it converges more often solutions whose quality is close to that of the best known solution. This technique has several advantages: First, it is relatively easy to design a local search that locally finds good tours, but it is hard to design a search that finds good tours for all customers simultaneously; the proposed technique makes it possible to overcome this difficulty and to design a fairly robust method more easily. Second, this technique may be applied to several types of VRPs, for example those including the following constraints:

- Time windows for the customer deliveries.
- Differentiated vehicles (cost of use per kilometer, volume capacity, carrying capacity).
- Constraints on the tours (maximum length, driver breaks, customers that cannot be reached by any vehicle).
- Backhauls.
- Multiple depots.

Third, this technique may easily be parallelized with an arbitrary number of processors (not depending on problem size).

The Rochat and Taillard approach exploits two primary perspectives, as follows.

The first comes from probabilistic tabu search, which is founded on the idea of translating information generated by the search history, coupled with current measures of attractiveness, into evaluations that are monotonically mapped into probabilities of selection. Operating in a neighborhood framework, the approach then successively selects among available alternatives according to a probability assignment that is strongly biased to favor the choice of higher evaluations.

The second main perspective that underlies the R&T approach derives from one of the most basic types of intensification strategies. The heart of this approach lies in generating solutions by reference to the notions of strongly determined and consistent variables.

It is shown that efficient first level tabu searches for vehicle routing problems may be significantly improved with this technique. Moreover, the solutions produced by this technique may often be improved by a post-optimization technique presented in this paper too, which embodies an effective means for applying a vocabulary building strategy in this context. The solutions of nearly 40 problem instances of the literature have been improved. This technique may also be applied to other local searches or other VRPs.

4.12 Optimization of Electromagnetic Structures With Tabu Search

Fanni, Giacinto and Marchesi (1996) develop a Tabu Search strategy to optimize the design of a magnet for Magnetic Resonance Imaging (MRI). This is an important biomedical device whose optimal design is sought by many companies, such as general Electric, Siemens and Oxford Instruments. Among different magnetic structures, MRI magnet systems are a tough benchmark for optimization procedures.

The goal of the problem considered was to design coils to yield a 'homogeneous' magnetic field in a fixed region, according to an appropriate function. For each coil the position and thickness have to be determined. To apply a TS based method, Fanni, Giacinto and Marchesi discretize the range of variation of each variable dividing it in sub-ranges, yielding a finite alphabet. The neighborhood of a solution consists of all the configurations obtained by considering all the possible symbols for each variable, keeping the others constant. Short term memory prevents repetitions of configurations of the coils (in terms of symbols of the finite alphabet). In non improving phases, frequency based memory also penalizes choices of moves that drive toward configurations often visited. Finally, local minimization using golden search is applied after the choice of the move. Computational tests show the TS approach performs more efficiently than an SA and a GA approach, and requires less than half as much solution time.

4.13 Multiprocessor Task Scheduling Using Parallel Tabu Search

Porto and Ribeiro (1995b) have designed and implemented parallelization tabu search strategies for the multiprocessor task scheduling problem. Parallelization relies exclusively on the decomposition of the solution space exploration. Four different parallel strategies were proposed and implemented on a 32-processor IBM SP1 parallel machine running PVM for varying problem sizes and number of processors: the master-slave model, with two different schemes for improved load balancing, and the single-program-multiple-data model (SPMD), with single-token and multiple-token message passing schemes. These two basic models mainly differ in the way information is exchanged between parallel tasks at the end of each iteration of the tabu search. The computational results confirmed the high adaptability of the TS algorithm to parallelization, showing that communication is not a burden to achieving almost ideal efficiency in the majority of the test problems. The task scheduling problem considered in this study is characterized by very large neighborhood structures that are costly to explore. However, the speedup achieved through simple parallelization techniques made possible a less restricted neighborhood search, which not only reduced computation time but produced better solutions for several test problems.

4.14 Tabu Search Applied to the Quadratic Assignment Problem (QAP), and Implementations for Connection Machines

The Quadratic Assignment Problem (QAP) is a classical NP-hard problem arising in many applications involving, for example, facility layout or VLSI design. Skorin-Kapov (1990, 1994) solves the quadratic assignment problem sub-optimally using the so-called Tabu-Navigation procedure, obtaining improved outcomes over previous results. In the process of designing an efficient massively parallel algorithm for the QAP, Chakrapani and Skorin-Kapov (1992) first generalized the connectionist model proposed by Aarts and Korst (1989) for the Traveling Salesman Problem (TSP) to solve the QAP. This was the first study replacing simulated annealing with (deterministic) tabu search in a connectionist model. This was also the first study involving dynamically changing connection strengths for such problems. In a subsequent paper Chakrapani and Skorin-Kapov (1993a) developed a massively parallel tabu search algorithm and implemented it on the Connection Machine. The careful implementation on Connection Machine, a massively parallel computer architecture, proved to be extremely suitable: provided enough processors, the computational time grows with O(log n).

4.15 Facility Layout in Manufacturing

The design of the facility layout of a manufacturing system is critically important for its effective utilization: 20 to 50 percent of the total operating expenses in manufacturing are attributed to material handling and layout related costs. Use of effective methods for facilities planning can reduce these costs often by as much as 30 percent, and sometimes more. In general, the facility layout problem has been formulated as a quadratic assignment problem (QAP). The QAP is to find the optimal assignment of n candidate facilities (departments, machines, workstations) to n candidate sites, for the goal of minimizing the total layout costs (which includes the material handling cost, expressed as the product of workflow and travel distance, and a fixed cost associated with locating a facility at a specific location).

Chiang and Kouvelis (1994b) provide a new implementation of the tabu search metaheuristic to solve the QAP, with particular emphasis on facility layout problems, utilizing recency-based and long term memory structures, dynamic tabu size strategies, and intensification and diversification strategies. The tabu search algorithm quickly converges from an arbitrarily generated random solution. Computational experiments, including statistical analysis and library analysis, strongly support the superiority of the C & K tabu search heuristic compared to other procedures for this facility layout application.

4.16 Quadratic Semi-Assignment and Mass Transit Applications

The quadratic semi-assignment problem (QSAP) is related to the quadratic assignment problem by the requirement of assigning a set of n objects to any of m locations. The QSAP differs by allowing each location to be assigned none, one, or even more than one object, whereas the QAP requires a one-one mapping of objects to locations (m = n).

Voss (1992), Domschke et al. (1992) and Daduna and Voss (1995) develop dynamic tabu search approaches for the QSAP, in a series of applications for modeling and solving a schedule synchronization problem in a mass transit system, where the goal is to minimize the total transfer waiting times of passengers. Both from an economic and a social standpoint, reducing passenger waiting time is a major issue in the operation of mass transit systems.

The outcomes of the tabu search applications show that better schedules are produced than those obtained by previous approaches, which were based on simulated annealing. The Daduna and Voss study also reports the successful incorporation of the tabu search schedule synchronization procedures into an overall solution approach for changing a public mass transit system by introducing new bus lines. The resulting advances include the ability to perform sensitivity analysis more effectively, disclosing that small changes, appropriately determined, can create large improvements in both cost and quality of service.

4.17 Reactive Tabu Search in Combinatorial Optimization

Reactive Tabu Search (RTS) as developed by Battiti and Tecchioli (1992, 1994b), has been applied to a considerable range of optimization problems. Combinatorial problems studied with this approach include:

- Quadratic Assignment Problems
- N-K Models (derived from biological inspiration),
- 0-1 Knapsack and Multi-Knapsack Problems,
- Max-Clique Problems
- Biquadratic Assignment Problems

In many cases the results obtained with alternative competitive heuristics have been duplicated with low computational complexity, and without intensive parameter and algorithm tuning. In some cases (e.g., in the Max-Clique and Biquadratic assignment problems) significantly better results have been obtained. A comparison of RTS with alternative heuristics (Repeated Local Minima Search, Simulated Annealing, Genetic Algorithms and Mean Field Neural

Networks) is presented in B&T (1995c) and a comparison with Simulated Annealing on QAP tasks is contained in B&T (1994c), disclosing the effectiveness of the RTS approach relative to these alternative procedures.

4.18 Asynchronous Parallel Tabu Search for Integrated Circuit Design

The logical test of integrated circuits is one of the main phases of their design and fabrication. The pseudo-exhaustive approach for the logical test of integrated circuits consists in partitioning the original circuit to be tested into non-overlapping subcircuits with a small, bounded number of input gates, which are then exhaustively tested in parallel. Andreatta and Ribeiro (1994) developed an approximate algorithm for the problem of partitioning integrated combinational circuits, based on the tabu search metaheuristic. The circuits are modelled as directed acyclic graphs. The proposed algorithm contains several original features, including reduced neighborhoods; complex moves (similar to an ejection chain strategy); a multicriteria cost function and the use of a bin-packing heuristic as a post-optimization step. Computational results were compared with those obtained by the best algorithm previously published in the literature, with significant improvements. The average reduction rates have been on the order of 30% for the number of subcircuits in the partition, and of the order of 40% for the number of cuts required.

4.19 Asynchronous Multithread Tabu Search Variants

The use of alternative types of move attributes for the formation of the tabu lists, and multiple strategies for obtaining initial solutions, can very often enhance the quality of solutions obtained in TS approaches. As shown by Aiex, Martins, Ribeiro and Rodriguez (1996) the combination of different initial solution procedures with different types of move attributes may be usefully integrated in an asynchronous multithread procedure, in which each processor runs a tabu search algorithm with a different pair of initial solution procedure and move attributes.

Each time one of the search threads finds a new local optimum, it writes this solution in a pool of elite solutions, which is kept by a master processor in charge of search coordination. When a search thread is not able to improve its local best solution, it accesses the pool of solutions and randomly chooses one of them to restart the search. Global stopping criteria are used. Aiex et al. implement this asynchronous strategy for the circuit partitioning problem, using 10 processors, where one of them is the master in charge of the search coordination, and the other processors run nine different strategy combinations. The results obtained by this multithread version of the tabu search algorithm yield much better solutions than those obtained by the sequential tabu search algorithm, within very reasonable computational times. This multithread search procedure was implemented using two different parallel programming tools, PVM and Linda, also leading to comparative results concerning these tools. (Reviews of the parallel TS literature and evaluations of several parallel programming tools have been recently published by Martins, Ribeiro and Rodriguez (1996) and Toulouse, Crainic and Gendreau (1996). See also the Appendix for a discussion of strategies for exploiting multiple choice rules and neighborhoods.)

4.20 Tabu Search for Two Dimensional Irregular Cutting

The two-dimensional cutting problem is an optimization problem in which two-dimensional elements of arbitrary specified shapes are to be cut out of a rectangular material. The objective is to determine a cutting pattern that will minimize the amount of material used. The importance of the problem is growing due to its relation to packing, loading and partitioning, which have applications in multiple branches of industry.

Błażewicz and Walkowiak (1995) apply three variants of tabu search to this problem, extending earlier work of Błażewicz, Hawryluk and Walkowiak (1993). The first approach is a simple short term memory version of tabu search that incorporates a tabu list, an aspiration function and a single criterion for optimization. The next variant introduces a new type of evaluating function which combines several criteria to be optimized, together with an associated tabu condition. In the last version a probabilistic approach is used which translates the evaluation criteria into probabilities of selection. An exact algorithm for the subproblem of finding the placement of the polygon is incorporated to enhance the quality of solutions. The final version is also embedded in a parallel version of the algorithm by taking into account various tabu method parallelization schemes and geometric features of the algorithm and problem space.

Extensive computational comparisons disclose that all variants of the tabu search method tested, but most particularly the advanced ones, obtain significantly improved patterns for cutting layouts. In addition, the parallel implementation demonstrates the ability of the method to be exploited highly effectively in a multiprocessor environment.

4.21 Solving the Vehicle Routing Problem with Time Windows

In the face of today's global competition, the need for more efficient logistical planning has become a pressing issue for most manufacturing and distribution concerns. Barnes and Carlton (1995) present a reactive tabu search (RTS) approach to the vehicle routing problem with time windows (VRPTW). The VRPTW considered has available, at a single depot, m identical vehicles with a specified cargo capacity. Each of n nonidentical customers require a specified volume of cargo which must be delivered within a specified contiguous interval of time. Each customer must be visited exactly once and the objective is to find the feasible set of vehicle routes that minimize the total travel time.

The B & C study furnishes a brief review of the most recent literature associated with the VRPTW, presents their RTS algorithm and gives computational results for the algorithm when applied to a widely used benchmark test set of vehicle routing problems with time windows due to Solomon. The results were produced without any attempt at "tuning," and were obtained in a small fraction of the time required by current exact techniques. The proposed algorithm does not suffer from the computational limitations of exact approaches, which are unable to successfully attack larger problems. The algorithm experienced no difficulty in obtaining solutions to all 56 Solomon problems for the 100-customer sets.

4.22 The Damper Placement Problem on Space Truss Structures

NASA has conducted a set of laboratory experiments investigating the control of space structures. To facilitate these experiments, a large, flexible structure was assembled from truss elements and antenna support members and dubbed the Controls-structures-interactions Evolutionary Model (CEM). The CEM was designed to simulate characteristics of a large earth-observation platform and was dynamically tested in the NASA Langley Space Structures laboratory.

The overall structural motion of a flexible truss structure can be reduced by the use of structural dampers that both sense and dissipate vibrations. Kincaid and Berger (1993) develop a tabu search method for the problem of locating these dampers. The goal is to assure that vibrations arising from the control or operation of the structure and its payloads, or by cyclic thermal expansion and contraction of the space structure, can be damped as effectively as possible.

Given a strain energy matrix with rows indexed on the modes and the columns indexed on the truss members, the problem can be expressed as that of finding a set of p columns such that the smallest row sum, over the p columns, is maximized. The TS approach obtained high quality solutions, as verified by comparisons with designs previously proposed and also with upper bounds provided by the optimum value of an LP relaxation. Outcomes from the study led NASA engineers to reconsider their design assumptions, and in consequence to change the rigidity of support arms for the truss structure.

4.23 Mixed Integer, Multi-stage Stochastic Programming

A very large class of problems is characterized by a multi-stage decision process where the future is uncertain and some decisions are constrained to take on values of either zero or one (as in the decision of whether or not to open a facility at a particular location). Although some mathematical theory exists for such problems, no general purpose algorithms have been available to address them. Løkketangen and Woodruff (1996) introduce the notion of integer convergence for progressive hedging, and provide the first implementation of general purpose methods for finding good solutions to multi-stage, stochastic mixed integer (0,1) programming problems. The solution method makes use of Rockafellar and Wets' progressive hedging algorithm that averages solutions rather than data, and then applies a tabu search algorithm to obtain solutions to the induced quadratic, (0,1) mixed-integer sub-problems. Computational experiments verify the effectiveness of the new method across a range of problem instances. The software that the authors have developed reads standard (SMPS) data files.

4.24 Tabu Search for the Fixed Charge Transportation Problem

In a fixed charge transportation problem, a fixed ("all or none") cost is incurred whenever a route in the transportation network is used. Goods transported along that route are additionally subject to a unit variable cost. The underlying model is also applicable to plant and warehouse location problems, purchase/lease problems, and personnel hiring problems.

A tabu search approach was developed for this problem by Sun et al. (1995) using recency based and frequency based memories, two strategies for each of the intermediate and long term memory processes, and a network based implementation of the simplex method as the local search method. A computational comparison was performed to evaluate the performance of this approach on randomly generated problems of different sizes and of different ranges of magnitude of fixed costs relative to variable costs. Objective function values and CPU time were used as criteria to compare the performance of this procedure with that previously proposed methods consisting of an exact solution algorithm and a heuristic procedure.

The tabu search procedure obtained optimal and near-optimal solutions much faster than the exact solution algorithm for simple problems, and thoroughly dominated the exact algorithm for more complex problems. For example, in a set of 15 randomly generated problems in the class studied, restricting the problem size, the tabu search procedure found the optimal solutions for 12 problems, and the objective function value of the worst solution to the remaining 3 problems was less than 0.06% higher than that of the optimal solution. The exact solution procedure used an average of 5888 CPU seconds for these problems, while the tabu search procedure used an average of just 1.63 seconds. As problem size increased or as fixed costs became high relative to variable costs, the solution time for the exact algorithm became inordinate. Compared to the alternative heuristic approach, statistical results showed that the tabu search procedure found comparable solutions at least as good for very small and easy test problems, and found significantly better solutions for all other problems. For the small problems, the solution times used by both heuristics were similar, while for larger problems and for problems with higher fixed relative to variable costs, the tabu search procedure was 3 to 4 times faster than the competing heuristic.

4.25 Sub-symbolic Machine Learning (Neural Networks)

While derivative-based methods for training from examples have been used with success in many contexts (error backpropagation is an example in the field of neural networks), they are applicable only to differentiable performance functions and are not always appropriate in the presence of local minima. In addition, the calculation of derivatives is expensive and error-prone, especially if special-purpose VLSI hardware is used. Battiti and Tecchiolli (1995b) use a significantly different approach: the task is transformed into a combinatorial optimization problem (the points of the search space are binary strings), and solved with a reactive tabu search algorithm. To speed up the neighborhood evaluation phase a stochastic sampling of the neighborhood is adopted and a "smart" iterative scheme is used to compute the changes in the performance function caused by changing a single weight. The RTS approach escapes rapidly from local minima, it is applicable to non-differentiable and even discontinuous functions and it is very robust with respect to the choice of the initial configuration. In addition, by fine-tuning the number of bits for each parameter one can decrease the size of the search space, increase the expected generalization and realize cost-effective VLSI.

4.26 Vehicle Routing Problem With Time Windows Applications

The vehicle routing problem with time windows (VRPTW) can be used to model many real-world problems and has recently been the subject of intensive research. Applications of the VRPTW include bank deliveries, postal deliveries, industrial refuse collection, national franchise restaurant deliveries, school bus routing, and security patrol services.

Chiang and Russell (1995) have developed a reactive tabu search metaheuristic for the VRPTW that dynamically varies the size of the list of forbidden moves (in order to avoid cycles as well as an overly constrained search path). The method incorporates intensification and diversification strategies to achieve higher quality solutions. C & R also developed simulated annealing metaheuristics which achieve solutions that compare favorably with previously reported results. Computational tests of problems from the literature as well as of large-scale real-world problems show that several new best known solutions are achieved by both the tabu search and simulated annealing approaches.

However, tabu search outperforms simulated annealing in solution quality. The tabu search approach is especially effective in reducing fleet size requirements for routing problems constrained by time windows.

4.27 The Polymer Straightening Problem

Polymer chemists at NASA-Langley Research Center are interested in the crystallization of high-performance aromatic polyimides. The value of these polymers lies in their thermal stability, strength, and toughness. Aromatic polyimides are used to build high-performance carbon fiber composites for structural components in aircraft and spacecraft, which depends on their crystallization. A key problem is to determine a priori if there exists a conformation for which a given aromatic polyimide crystallizes.

The role of the optimizer in this application is to determine if a straight line conformation for a given polyimide exists among all possible combinations of allowable (minimum energy) torsion angles for the rotable bonds. The total number of combinations may be quite large, easily containing as many as 100 million possible conformations.

Kincaid, Martin and Hinkley (1995) develop a simple tabu search procedure to find a conformation that maximizes the cosine of the angle between the first bond and the projection of the last bond over all allowable conformations. The method was applied to the analysis of three polyimides of interest to NASA Langley Research Center, and succeeded in finding the optimal conformation in all three cases. (Normally, a research chemist could require as much as three years to perform such an analysis.) The ultimate goal of this research is to provide a technique that will serve as an aid to chemists in deciding what conformations are most likely to result in crystallizable structures when produced in a laboratory.

4.28 Tabu Search for Portfolio Management

An important issue in portfolio management is how to measure and handle risk. The challenge is to solve large problems that typically cannot be attacked with non-linear programming methods. Rolland (1996) develops a tabu search method that handles real valued decision variables by discretizing the problem space in 1% and 0.1% increments, and by further incorporating a greedy search to adjust the decision variables to real numbers “finer” than the 0.1% accuracy level. The moves alter their focus to comply with minimum variance and the target return. The approach identified optimal solutions to all the random and real-world problems it was tested on. Computation time was also very modest, even for large problems with 100 or more assets. This approach recently has been extended by Rolland and Johnson (1996) to be able to handle skewness computations and targets.

4.29 Modeling Generalized Capacity Requirements in Production Planning

Production planning problems that arise in real world applications are typically attended by capacity requirements. The incorporation of generalized capacity effects such as economies and diseconomies of scope and the learning-curve effect gives rise to a capacity-consumption function that is nonlinear in the tasks assigned to each facility. The resulting models for facility planning and loading decisions often involve nonlinear optimization problems in which some or all of the decision variables are integer-valued. The combined conditions of nonlinearity and discreetness makes these problems exceedingly difficult to solve.

Mazzola and Schantz (1995a) consider the resource allocation of a single facility under capacity-based economies and diseconomies of scope, and develop two models for this problem. In the first (more general) model the capacity of a single facility is considered to be a general function of the subset of tasks selected to be produced. In the second model the capacity is assumed to be consumed as a function of the number of tasks assigned to the facility. These can be viewed as single-facility production loading models that capture economies and diseconomies of scope within the production planning framework. The problems arising within each of these models generalize both the 0-1 knapsack problem and the 0-1 collapsing knapsack problem. Tabu search heuristics and branch-and-bound algorithms are defined for each model. Computational testing shows the tabu search heuristics to be extremely effective in obtaining high-quality solutions to these problems, including the more difficult problems that exhibit a high degree of nonlinear behavior.

This study is extended in Mazzola and Schantz (1995b) to the multiple-facility setting, focusing on under capacity-based economies (and diseconomies) of scope (MFLS), including applications of MFLS in hierarchical production planning, group technology, and professional services. MFLS is formulated as a nonlinear 0-1 mixed-integer programming problem which generalizes many well known and widely applicable optimization problems, such as the generalized assignment problem and the capacitated facility location problem. A tabu-search heuristic and a branch-and-bound algorithm are developed for MFLS, and computational testing of the solution procedures is discussed. Once again, tabu search proves to be an effective approach for heuristically solving MFLS, and is reported to be a powerful tool for capturing complex capacity requirements.

4.30 Continuous Optimization

A simple benchmark on a function with many suboptimal local minima is considered in Battiti and Tecchiolli (1994b), where a straightforward discretization of the domain is used. A novel algorithm for the global optimization of functions (C-RTS) is presented in Battiti and Tecchiolli (1995a), in which a combinatorial optimization method cooperates with a stochastic local minimizer. The combinatorial optimization component, based on reactive tabu search, locates the most promising boxes, where starting points for the local minimizer are generated. In order to cover a wide spectrum of possible applications with no user intervention, the method is designed with adaptive mechanisms: in addition to the reactive adaptation of the prohibition period, the box size is adapted to the local structure of the function to be optimized (boxes are larger in "flat" regions, smaller in regions with a "rough" structure).

4.31 Chunking in Tabu Search

Chunking — grouping basic units of information to create higher level units — is a critical component of human intelligence. The tendency for people to group information was described in the celebrated 1956 paper "The Magic Number Seven: Plus or Minus Two" by G.A. Mitchell. Human problem solvers, when faced with a hard problem, often proceed by linking and integrating features they perceive as related and germane to the solution process. However, people often prefer to organize problem data and problem solving methods in a hierarchical fashion. When possible, they decompose the problem into subproblems and solve those. When the problem cannot be decomposed, a common strategy is to form groupings of solution attributes so that the search space can be reduced and higher level relationships can be discovered or exploited.

Woodruff (1995, 1996) identifies special types of chunking to enhance tabu search memory structures, producing improved problem solving ability and giving useful supporting information for decision makers. Although the proposals are most natural in the context of the tabu search paradigm, they can also be employed in genetic algorithms and simulated annealing by omitting links to memory based constructions. This work outlines theory and proposals for learning about chunks and using them. Computational experience to date is briefly summarized to support the contention that chunking can be an important part of effective optimization algorithms.

4.32 Production Planning with Work-Force Learning

Mazzola, Neebe, and Rump (1995) consider a production planning problem in which the work-force productivity for each product depends on the amount of previous production of the product. This change in productivity is captured in the corresponding resource coefficients that occur in the work-force requirement constraint for each time period.

The model includes a learning effect in each period that depends on the level of production of each product in the preceding period. The corresponding work-force coefficient can increase, decrease, or remain the same, representing a forgetting, learning, or status-quo production effect. The resulting problem is modeled as a mixed-integer programming problem. In addition to establishing problem complexity and defining a branch-and-bound algorithm for solving the problem to optimality, this paper also examines heuristics for the problem. A forward pass, linear programming-based heuristic previously defined in the literature is examined and shown to produce arbitrarily bad solutions for this problem. The paper then proposes a new tabu search heuristic for the problem. Extensive computational experiments with the solution procedures establishes the effectiveness of the tabu-search approach in this problem setting.

The TS approaches of this study are concluded to provide an ability to handle new levels of modeling complexity, allowing for closer approximation of real-world phenomena. The findings indicate that this is particularly true for problems involving complex, nonlinear behavior involving discrete decision variables.

4.33 Tabu Search Applied to Hub-and-Spoke Communication Networks

The location of hub facilities is an important issue arising in the design of communication networks. Applications include: traffic networks (airline passengers flow and parcel delivery networks), as well as telecommunication networks (location of digital switching offices for Digital Data Service (DDS) networks, location of base stations for wireless networks).

In general, determining optimal locations of hub nodes and allocations of non-hub nodes to those hubs is an NP-hard combinatorial problem. For a widely used benchmark set of problems (the Civil Aeronautics Board (CAB) data set) efficient tabu search algorithms and lower bounds for a class of uncapacitated multiple and single allocation p-hub median problems have recently been developed which notably improve on results previously obtained. (Attempts to solve the problems of the CAB data set have been undertaken in more than 70 research papers.)

Skorin-Kapov and Skorin-Kapov (1994) have developed an efficient tabu search heuristic for the single allocation p-hub median problem, which models the situation when n nodes can interact only via a set of fully interconnected hubs. The hubs are uncapacitated, and their number is initially prescribed. Using the amount of flow and the cost per unit of flow between any two nodes in a network, one has to decide on the location of hubs, and on the allocation of each non-hub node to one of the hubs. The problem can be formulated as a quadratic integer program with a nonconvex objective function. The new tabu search approach, in addition to being efficient, obtains a number of new best solutions for the CAB data set.

Skorin-Kapov et al. (1995) provide a novel way to further take advantage of these improved heuristic outcomes by using high quality heuristic solutions to derive lower bounds. Accompanying this, they provide tight linear programming relaxations for the hub location and some other relevant uncapacitated p-hub median problems that allow the CAB data set to be solved to optimality for the first time. By this means, they verified that the solutions obtained by the tabu search approach were in fact optimal for all of the test problems.

By exploiting the LP solution, and the best known heuristic solutions derived from tabu search, integrality was quickly achieved by adding a partial set of integrality constraints. This result, in turn, proved the optimality of the best known tabu search heuristic solutions. Moreover, it provided a new approach of using the best known heuristic solution as a guidance in adding a partial set of integrality constraints to achieve integer solutions for those instances whose linear programming relaxations resulted with fractional solutions.

4.34 VLSI Systems with Learning Capabilities

In contrast to the exhaustive design of systems for pattern recognition, control, and vector quantization, an appealing possibility consists of specifying a general architecture, whose parameters are then tuned through Machine Learning (ML). ML becomes a combinatorial task if the parameters assume a discrete set of values: a reactive tabu search algorithm developed by Battiti et al. (1994a, 1994b) permits the training of these systems with low number of bits per weight, low computational accuracy, no local minima "trapping", and limited sensitivity to the initial conditions.

A board with the TOTEM chip used for Machine Learning applications — A project for IRST aims at developing special-purpose VLSI modules to be used as components of fully autonomous massively-parallel systems for real-time adaptive applications. Because of the intense use of parallelism at the chip and system level and the limited precision used, the obtained performance is competitive with that of state-of-the-art supercomputers (at a much lower cost), while a high degree of flexibility is maintained through the use of combinatorial algorithms. In particular, neural nets can be realized. In contrast to many "emulation" approaches, the developed VLSI completely reflects the combinatorial structure used in the learning algorithms. The first chip of the project (TOTEM, partially funded by INFN and EU (Esprit project MInOSS) and designed at IRST achieves a performance of more than one billion multiply-accumulate operations. Applications considered are in the area of pattern recognition (Optical Character Recognition), events "triggering" in High Energy Physics [A+], control of non-linear systems (Battiti and Tecchiolli (1995c), compression of EEG signals [B+]. Test boards for ISA, PCI or VME buses with software and technical documentation are available at IRST.

4.35 A Tabu Search Approach for Routing and Distribution

Rochat and Semet (1994) consider a real-life vehicle routing and distribution problem that occurs in a major Swiss company producing pet food and flour. In contrast with usual hypothetical problems, a large variety of restrictions must be considered. The main constraints involve the accessibility and the time windows at customers, the carrying capacities of vehicles, the total duration of routes and the drivers' breaks. The optimization problem for the transport plan consists in elaborating a set of routes that minimizes the total travel distance while satisfying the indicated constraints.

The general scheme used to solve this real-life VRP first applies a straightforward construction procedure to generate an initial solution which provides a starting point for the tabu search procedure. The key features of the TS approach consist of a constraint relaxation strategy for diversification and an intensification strategy. The relaxation of constraints makes it possible to expand the solution space, diversifying the search by examining infeasible solutions as well as feasible ones. The intensification strategy plays a complementary role, and leads the search to visit solutions close to the best solution found so far by rendering some routes tabu. Computational results show that the TS method yields solutions dominating those of the constructive heuristic even when the total number of iterations is small. Thus, good solutions are obtained in a reasonable amount of CPU time. Moreover, the study shows that embedding the TS algorithm in decision support software can be particularly useful. The fact that the TS approach generates multiple solutions that have approximately the same length of routes as the best makes it possible to propose several transport plans to the user. Comparisons of the solutions produced with the routes actually covered by the company disclose that the total distance traveled is reduced significantly by these solutions.

4.36 Tabu Search for Scheduling a Flow-Line Manufacturing Cell

Effective scheduling of flow-lines for manufacturing cells improves the operational efficiency of manufacturing processes, leading to reductions in setup costs, labor costs, tooling and inventory costs. This leads to further reductions in throughput times and a corresponding increase in the shipment of on-time deliveries. A tabu search method for this problem has been proposed and successfully implemented by Skorin-Kapov and Vakharia (1993).

A manufacturing cell consists of a group of similar machines located in close proximity to one another and dedicated to the manufacture of a specific number of part families. Part families consist of a set of jobs with similar processing requirements. In this context, a feasible schedule S consists of a sequence of part families and a sequence of jobs within each family in a manufacturing cell. The tabu search heuristic of Skorin-Kapov and Vakharia efficiently schedules a pure flow-line manufacturing cell under varying parameter conditions (given F families, M machines and N(f) jobs in family f).

A collection of alternative tabu search strategies (designed to test different aspects of tabu search) was compared against state-of-the-art simulated annealing heuristic that was tailored to solve this problem. Results from testing multiple data sets with alternative ratios of family set up times to job processing times showed the clear superiority of tabu search for these scheduling problems.

4.37 A Hybrid Scatter Genetic Tabu Approach for Continuous Global Optimization

A hybrid scatter genetic tabu search approach (HSGT) is proposed by Trafalis and Al-Harkan (1995) to solve an unconstrained continuous nonlinear global optimization problem. This approach combines the characteristics of the following metaheuristics: scatter search (SS), genetic algorithms (GAs), and tabu search (TS). The proposed approach has been tested against a simulated annealing (SA) algorithm and a modified version of a hybrid scatter genetic search approach (HSG) by optimizing twenty-one well known test functions. From the computational results, the HSGT approach proved to be quite effective in identifying the global optimum solution which makes the HSGT approach a promising approach to solve the general nonlinear optimization problem. In the hundred runs performed for each of the twenty-one functions, the HSGT approach performed better than the HSG and the SA approaches, except for one function. Also, the HSGT approach converged to a near global optimum in CPU times ranged between 1.3 seconds and 19.55 seconds. The algorithm was implemented in a GATEWAY 2000 (Pentium, P5-90) computer using the Microsoft FORTRAN PowerStation version 4.

4.38 Active Structural Acoustic Control

Active structural acoustic control is a method in which the control inputs used to reduce interior noise are applied directly to a vibrating structural acoustic system. The ultimate goal is to use active acoustic control to decrease the interior noise in propeller driven jet aircraft. Kincaid (1995) studies the instance of this problem in which the objective consists of damping noise generated by a single exterior source in the interior of a cylinder.

The model requires a determination of the force inputs and sites for piezoelectric actuators so that (1) the interior noise is effectively damped; (2) the level of vibration of the cylinder shell is not increased; and (3) the power requirements needed to drive the actuators are not excessive.

A tabu search approach was developed to determine the best set of actuator sites to meet the three specified objectives. Experiments confirmed that the TS procedure is able to uncover better solutions than those selected based solely upon engineering judgement. In addition, the high quality solutions generated by tabu search, when minimizing interior noise, do not further excite the cylinder shell. Thus, it was possible to meet objective (2) without imposing an additional constraint or forming a multi-objective performance measure. The TS solutions also led to identifying natural groupings that require fewer control channels and that permit a simpler control system.

4.39 Tabu Search for Automatic Graph Drawing

Graphs are commonly used as a basic modeling tool in areas such as project management, production scheduling, line balancing, business process reengineering, and software visualization. Drawings of graphs are called maps and their value for modeling and analysis is widely heralded in various fields of the economic, social and computational sciences.

The main quality desired for maps is readability: A map is readable if its meaning is easily captured by the way it is drawn. It is extremely difficult to make a readable map by hand of a graph that represents a real system, even when the graph size is relatively small. Therefore, an automatic procedure for drawing graphs by computer is indispensable for generating readable maps quickly.

An important problem in the area of graph drawing is to minimize arc crossings in a hierarchical digraph. It is customary to draw a hierarchical digraph by placing the vertices on a set of equally spaced horizontal or vertical lines called layers and then drawing the arcs as straight-line segments.

Valls, Marti and Lino (1995) provide a Tabu Thresholding (TT) approach for the problem of minimizing the number of arc crossings in a 2-layer hierarchical digraph (a bipartite digraph). The procedure combines elements of probabilistic tabu search, candidate list strategies and thresholds, yielding a simplified implementation of basic TS ideas that does not make explicit use of memory structures.

The computational study has been carried out on a set of 250 randomly generated problems of varying sizes and densities. The TT algorithm has been compared with the Greedy Switching (GS) algorithm and the Splitting (S) algorithm, which are reported to be the best methods in the literature previously available for graph drawing. Outcomes from the TT methods are also compared with the optimal solutions for the test problems, in those cases where the problems are small enough to permit them to be solved by state-of-the-art exact methods. Results show that in the 130 test problems where an optimum is available, the TT solution is optimal. Moreover in each of the 250 examples tested, the TT solution and generally is superior (and never inferior) to the best of those given by the GS and S heuristics.

4.40 Bipartite Graph Drawing with Tabu Search

Marti (1995) has developed a TS heuristic for the problem of minimizing the number of arc crossings in a bipartite graph. To perform an aggressive search for the global optimum, the author has considered intensification, diversification, influential moves and strategic oscillation elements of tabu search.

The procedure has three different search states: normal, influential and opposite, and oscillates among them according to the search history. In each state there are two alternately applied phases, an intensification phase and a diversification phase. The moves defined in the intensification are based on a positioning function while those defined in the diversification are based on permuting consecutive vertices. The use of different moves reinforces the non-monotonic search strategy. The criteria for differentiating between “improving” and “disimproving” moves within the oscillation strategy are not limited to the objective function evaluation, but consider factors of move influence, as determined by context and the search history.

Two variants of the general TS procedure are developed and compared with the GS, SP, BC and SM methods and with the Tabu Thresholding (TT) heuristic. The computational results show that both of these TS variants perform better than the other methods, closely followed by the TT procedure, which in turn also performs better than the methods remaining.

4.41 Layered Graph Drawing

Laguna, Marti and Valls (1995) propose a TS algorithm for the general k-layer graph drawing problem (k greater than 2). Existing solution methods for this problem are based on simple ordering rules for single layers that may lead to inferior drawings. The Tabu Search implementation consists of an intensification phase that seeks local optimal orderings of layers using an insertion move, and two levels of diversification. The first level of diversification is a strategy for selecting layers for intensification, while the second one escapes local optimality by means of switching moves. The authors utilize two different termination criteria (TABU1 and TABU2).

Computational testing was performed on a set of 200 randomly generated instances, including graphs with up to 571 vertices and 2,241 arcs. Comparisons were performed with procedures that have shown to be effective for arc crossing minimization, i.e., the barycentric and the semi media methods with switching (BC+SW, SM+SW).

The results of the experiments show that in terms of solution quality the procedures are ranked in the order TABU2, TABU1, BC+SW, and SM+SW. In terms of computational time, the tabu search version TABU1 is quite competitive with the procedures based on simple ordering rules plus switching, in spite of yielding significantly better outcomes. This allows TABU1 to be considered as a powerful procedure for real-time drawing (e.g., drawing on a computer screen). When still higher quality drawings are important, at the cost of additional computational time, TABU2 obtains solutions with fewer arc crossings with a maximum running time of 209 seconds.

4.42 Finding a Good Forest Harvest Schedule Satisfying Green-Up Constraints

A distinct change in public attitude toward the environment has led to demands for a forest management paradigm shift from one of a dominant timber use to one in which forests are managed for multiple values. Among the most important implications of such a shift in policy on forest management was the introduction of adjacency or green-up constraints. Green-up constraints are imposed to avoid large clear cut areas. For example, in British Columbia legislation requires that a block of forest which has been harvested must reach a mean tree height of three meters before any adjacent block can be harvested.

One result of these green-up constraints is that the forest harvesting scheduling problem, which previously was often formulated as a linear program, becomes combinatorial in nature. Current harvest scheduling codes, like FORPLAN, Timber RAM and others, are unable to generate harvest schedules satisfying the green-up constraints. Brumelle et al. (1996) formulate forest harvesting problems with green-up constraints arising in the Tangier watershed in British Columbia as multicriteria discrete optimization problems.

The study considers two harvest scheduling problems associated with the Tangier watershed. The small problem focuses on a 219 cut-block subset of the watershed located at the southern end. The northern boundary coincides with that of the proposed Serenity Peaks Wilderness Area. The larger problem included all 491 cut-blocks comprising the entire watershed, which will be appropriate should the proposed park not eventuate.

Tabu search is used to investigate the trade-offs between different criteria, which were chosen as the total volume of lumber cut, the period to period deviation from even-flow of lumber during a harvest rotation and adjacency violations. The tabu search methodology easily obtained good solutions to these problems, and was shown to be much superior to a biased random search method which is cited as one of the most effective methods to obtain good schedules satisfying green-up constraints. In fact, the tabu search method generates schedules which harvest more timber than the upper bound of the confidence interval suggested by previous empirical and algorithmic analysis.

APPENDIX A

THE MOST NEGLECTED TABU SEARCH STRATEGIES — PROMISING AVENUES FOR FUTURE RESEARCH

This appendix briefly reviews several key strategies in tabu search that are often neglected (especially in beginning studies), but which are important for producing the best results.

Our purpose is to call attention to the relevance of particular elements that may be missed in a lengthier or more formal exposition. In addition, observations about useful directions for future research are included.

A comment regarding implementation: first steps do not have to include the most sophisticated variants of the ideas described below, but the difference between "some inclusion" and "no inclusion" can be significant. Implementations that incorporate simple instances of these ideas will often disclose the manner in which refined implementations can lead to improved performance.

A.1 CANDIDATE LIST STRATEGIES

Efficiency and quality can be greatly affected by using intelligent procedures for isolating effective candidate moves, rather than trying to evaluate every possible move in a current neighborhood of alternatives. This is particularly true when such a neighborhood is large or expensive to examine. The gains to be achieved by using candidate lists have been widely documented, yet many TS studies overlook their relevance.

Careful organization in applying candidate lists, as by saving evaluations from previous iterations and updating them efficiently, can also be valuable for reducing overall effort. Time saved in these ways allows a chance to devote more time to higher level features of the search.

While the basic theme of candidate lists is straightforward, there are some subtleties in the ways candidate list strategies may be used. Considerable benefit can result by being aware of fundamental candidate list approaches, such as the *Subdivision Strategy*, the *Aspiration Plus Strategy*, the *Elite Candidate*

List Strategy, the *Bounded Change Strategy* and the *Sequential Fan Strategy* (see, for example, Glover (1995b)).

An effective integration of a candidate list strategy with the rest of a tabu search method will typically benefit by using TS memory designs to facilitate functions to be performed by the candidate lists. This applies especially to the use of frequency based memory. A major mistake of some TS implementations, whether or not they make use of candidate lists, is to consider only the use of recency based memory. Frequency based memory — which itself takes different forms in intensification phases and diversification phases — can not only have a dramatic impact on the performance of the search in general but also can often yield gains in the design of candidate list procedures.

A.2 PROBABILISTIC TABU SEARCH

Several studies have suggested the value of a probabilistic version of TS, where evaluations (including reference to tabu status) are translated into probabilities of selection, strongly skewed to favor higher evaluations. Findings from such studies support the notion that probabilities may partly substitute for certain functions of memory (hence reduce the amount of memory needed) but also suggest that probabilities may have a role in counteracting "noise" in the evaluations.

In well designed TS implementations, the gains of probabilistic TS over deterministic TS are chiefly in accelerating the rate at which good solutions are discovered in earlier stages of search. Overall, some settings appear more exploitable by probabilistic TS and others appear more exploitable by deterministic TS. The most effective forms of each type of approach depend strongly on identifying and implementing TS strategies of the type described below.

A.3 INTENSIFICATION APPROACHES

Intensification strategies, which are based on recording and exploiting elite solutions or, characteristically, specific features of these solutions, have proved very useful in a variety of applications. Some of the relevant forms of such strategies and considerations for implementing them are as follows.

(1) The simplest intensification approach is the strategy of recovering elite solutions in some order, each time the search progress slows, and then using these solutions as a basis for re-initiating the search. The list of solutions that are candidates to be recovered is generally limited in size, often in the range of 20 to 40 (although in parallel processing applications the number is characteristically somewhat larger). The size chosen for the list in serial TS applications also corresponds roughly to the number of solution recoveries anticipated to be done during the search, and so may be less or more depending on the setting. When an elite solution is recovered from the list, it is removed, and new elite solutions are allowed to replace less attractive previous solutions — usually dropping the worst of the current list members. However, if a new elite solution is highly similar to a solution presently recorded, instead of replacing the current worst solution, the new solution will compete directly with its similar counterpart to determine which solution is saved.

This approach has been applied very effectively in job shop and flow shop scheduling, in vehicle routing, and in telecommunication design problems. One of the best approaches for scheduling applications (see Section 3) keeps the old TS memory associated with the solution, but makes sure the first new move away from this solution goes to a different neighbor than the one visited after encountering this solution the first time. Another effective variant does not bother to save the old TS memory, but uses a probabilistic TS choice design.

The most common strategy is to go through the list from best to worst, but in some cases it has worked even better to go through the list in the other direction. In this approach, it appears effective to allow two passes of the list. On the first pass, when a new elite solution is found that falls below the quality of the solution currently recovered, but which is still better than the worst already examined on the list, the method still adds the new solution to the list and displaces the worst solution. Then a second pass, after reaching the top of the list, recovers any added solutions not previously recovered.

(2) The other primary intensification strategy is to examine elite solutions to determine the frequency in which particular solution attributes occur (where the frequency is typically weighted by the quality of the solutions in which the attributes are found).

This strategy was originally formulated in the context of identifying "consistent" and "strongly determined" variables — where, loosely speaking, consistent variables are those more frequently found in elite solutions, while strongly determined variables are those that would cause the greatest disruption by changing their values (as sometimes approximately measured by weighting the frequencies based on solution quality). The idea is to isolate the variables that qualify as more consistent and strongly determined (according to varying thresholds), and then to generate new solutions that give these variables their "preferred values." This can be done either by rebuilding new solutions in a multistart approach or by modifying the choice rules of an ongoing solution effort to favor the inclusion of these value assignments.

Keeping track of the frequency that elite solutions include particular attributes (such as edges of tours, assignments of elements to positions, narrow ranges of values taken on by variables, etc.) and then favoring the inclusion of the highest frequency elements, effectively allows the search to concentrate on finding the best supporting uses and values of other elements. A simple variant is to "lock in" a small subset of the most attractive attributes (value assignments) — allowing this subset to change over time or on different passes.

A Relevant Concern: In the approach that starts from a current (good) solution, and tries to bring in favored elements, it is important to introduce an element that yields a best outcome from among the current contenders. If an attractive alternative move shows up during this process, which does not involve bringing in one of these elements, aspiration criteria may determine whether such a move should be taken instead. Under circumstances where the outcome of such a move appears sufficiently promising, the approach may be discontinued and allowed to enter an improving phase (reflecting a decision that enough intensification has been applied, and it is time to return to searching by customary means).

Intensification of this form makes it possible to determine what percent of "good attributes" from prior solutions should be included in the solution currently generated. It also gives information about which subsets of these attributes should go together, since it is preferable not to choose attributes during this process that cause the solution to deteriorate compared to other choices. This type of intensification strategy has proved highly effective in the settings of vehicle routing and zero-one mixed integer optimization.

(3) Memory and Intensification: It is clearly somewhat more dangerous to hold elements "in" solution than to hold them "out" (considering that a solution normally is composed of a small fraction of available elements — as where a tree contains only a fraction of the edges of a graph). However, there is an important exception, previously intimated. As part of a longer term intensification strategy, elements may be selected very judiciously to be "locked in" on the basis of having occurred with high frequency in the best solutions found. In that case, choosing different mutually compatible (and mutually reinforc-

ing) sets to lock in can be quite helpful. This creates a *combinatorial implosion* effect (opposite to a combinatorial explosion effect) that shrinks the solution space to a point where best solutions over the reduced space are likely to be found more readily.

The key to this type of intensification strategy naturally is to select an appropriate set of elements to lock in, but the chances appear empirically to be quite high that some subset of those with high frequencies in earlier best solutions will be correct. Varying the subsets selected gives a significant likelihood of picking a good one. (More than one subset can be correct, because different subsets can still be part of the same complete set.) Aspiration criteria make it possible to drop elements that are supposedly locked in, to give this approach more flexibility.

(4) Relevance of Clustering for Intensification: A search process over a complex space is likely to produce clusters of elite solutions, where one group of solutions gives high frequencies for one set of attributes and another group gives high frequencies for a different set. It is important to recognize this situation when it arises. Otherwise there is a danger that an intensification strategy may try to compel a solution to include attributes that work against each other. This is particularly true in a strategy that seeks to generate a solution by incorporating a collection of attributes "all at once," rather than using a step by step evaluation process that is reapplied at each move through a neighborhood space. (Stepping through a neighborhood has the disadvantage of being slower, but may compensate by being more selective. Experimentation to determine the circumstances under which each of these alternative intensification approaches may be preferable would be quite valuable.)

A strategy that incorporates a block of attributes together may yield benefits by varying both the size and composition of the subsets of high frequency "attractive" attributes, even if these attributes are derived from solutions that lie in a common cluster, since the truly best solutions may not include them all. (An example of the relevance of clustering and related conditional analysis is provided in Topic VI of the Appendix.)

A.4 DIVERSIFICATION APPROACHES

Diversification processes in tabu search are sometimes applied in ways that limit their effectiveness, due to overlooking the fact that diversification is not just "random" or "impulsive," but depends on a purposeful blend of memory and strategy. As noted in Section 2, recency and frequency based memory are both relevant for diversification, and stem in part from proposals to use such ideas in surrogate constraint procedures. In this setting, the impetus is not simply to achieve diversification, but to derive appropriate weights in order to assure that evaluations will lead to solutions that satisfy required conditions. Accordingly, it is important to account for elements such as how often, to what extent, and how recently, particular constraints have been violated, in order to determine weights that produce more effective valuations.

The implicit *learning effects* that underlie such uses of recency, frequency and influence are analogous to those that motivate the procedures used for diversification (and intensification) in tabu search. Early strategic oscillation approaches exploited this principle by driving the search to various depths outside (and inside) feasibility boundaries, and then employing evaluations and directional search to move toward preferred regions.

In the same way that these early strategies bring diversification and intensification together as part of a continuously modulated process, it is important to stress that these two elements should be interwoven in general. A common mistake in many TS implementations is to apply diversification without regard for intensification. "Pure" diversification strategies are appropriate for truly long term strategies, but over the intermediate term, diversification is generally more effective if it is applied by heeding information that is also incorporated in intensification strategies. In fact, intensification by itself can sometimes cause a form of diversification, because intensifying over part of the space allows a broader search of the rest of the space. A few relevant concerns are as follows.

(1) Diversification and intensification links: A simple and natural diversification approach is to keep track of the frequency that attributes occur in non-elite solutions, and then to periodically discourage the incorporation of attributes that have modest to high frequencies (giving greater penalties to larger frequencies). The reference to non-elite solutions tends to avoid penalizing attributes that would be encouraged by an intensification strategy.

More generally, for a "first level" balance, an Intermediate Term Memory matrix may be used, where the high frequency items in elite solutions are not penalized by the long term values, but may even be encouraged. The tradeoffs involved in establishing the degree of encouragement, or the degree of reducing the penalties, represents an area where a small amount of preliminary testing can be valuable. This applies as well to picking thresholds to identify high frequency items. (Simple guesses about appropriate parameter values can often yield benefits, and tests of such initial guesses can build an understanding that leads to increasingly effective strategies.)

By extension, if an element has never or rarely been in a solution generated, then it should be given a higher evaluation for being incorporated in a diversification approach if it was "almost chosen" in the past but didn't make the grade. This observation has not been widely heeded, but is not difficult to implement, and is relevant to intensification strategies as well.

(2) Implicit conflict and the importance of interactions: Current evaluations also should not be disregarded while diversification influences are activated. Otherwise, a diversification process may bring elements together that conflict with each other, make it harder rather than easier to find improved solutions.

For example, a design that gives high penalties to a wide range of elements, without considering interactions, may drive the solution to avoid good combinations of elements. Consequently, diversification should be carried out for a limited number of steps, accompanied by watching for and sidestepping situations where indiscriminately applying penalties would create incompatibilities or severe deterioration of quality. To repeat the theme: even in diversification, attention to quality is important. And as in "medical remedies," sometimes small doses are better than large ones.

(3) An approach called "Reactive Tabu Search" (RTS) developed by Battiti and Tecchiolli (1992, 1994b) deserves consideration as a way to achieve a useful blend of intensification and diversification. RTS incorporates hashing in a highly effective manner to generate attributes that are very nearly able to differentiate among distinct solutions (that is, very few solutions contain the same hashed attribute). Accompanying this, B&T use an automated tabu tenure, which begins with the value of 1 (preventing a hashed attribute from being reinstated if this attribute gives the "signature" of the solution visited on the immediately preceding step). This tenure is then increased if examination shows the method is possibly cycling, as indicated by periodically generating solutions that produce the same hashed attribute.

The tabu tenure, which is the same for all attributes, is increased exponentially when repetitions are encountered, and decreased gradually when repetitions disappear. Under circumstances where the search nevertheless encounters an excessive number of repetitions within a given span (i.e., where a moving frequency measure exceeds a certain threshold), a diversification step is activated, which consists of making a number of random moves proportional to a moving average of the cycle length.

The reported successes of this approach invite further investigations of its underlying ideas and related variants. As a potential bases for generating such variants, attributes created by hashing may be viewed as fine grain attributes, which give them the ability to distinguish among different solutions. By contrast, "standard" solution attributes, which are the raw material for hashing, may be viewed as coarse grain attributes, since each may be contained in (and hence provide a signature for) many different solutions. Experience has shown that tabu restrictions based on coarse grain attributes are often advantageous for giving increased vigor to the search. (There can exist a variety of ways of defining and exploiting attributes, particularly at coarser levels, which complicates the issue somewhat.) This raises the question of when particular degrees of granularity are more effective than others.

It seems reasonable to suspect that fine grain attributes may yield greater benefits if they are activated in the vicinity of elite solutions, thereby allowing the search to scour "high quality terrain" more minutely. Of course, this effect may also be achieved by reducing tabu tenures for coarse grain attributes — or basing tabu restrictions on attribute conjunctions — and using more specialized aspiration criteria. Closer scouring of critical regions can also be brought about by using strongly focused candidate list strategies, such as a sequential fan candidate list strategy. (Empirical comparisons of such alternatives to hashing clearly would be of interest.)

However, another type of alternative to hashing also exists, which is to create new attributes by processes that are not so uniform as hashing. A potential drawback of hashing is its inability to distinguish the relative importance (and appropriate influence) of the attributes that it seeks to map into others that are fine grained. A potential way to overcome this drawback is to make use of vocabulary building (Section 2) and of conditional analysis (Topic VI of the Appendix). See Voss (1993), Carlton and Barnes (1995a, 1995b) and Woodruff (1995, 1996) for useful related observations.

(4) Ejection chain approaches: Ejection chain methods provide an implicit blending of diversification and intensification by generating compound moves out of simpler components. Such approaches have provided breakthroughs in handling certain types of tough problems, particularly those related to optimization over graphs (see for example, Dorndorf and Pesch (1994), Laguna et al. (1995), Pesch and Glover (1995), Rego and Roucairol (1996), Rego (1996a, 1996b)). TS memory structures can be used at two levels with ejection chains, both at a simple internal level which operates primarily as a bookkeeping function to avoid duplicate patterns (as complex moves are woven from simpler ones), and at an external level that guides the successively generated com-

pound moves to go beyond conditions of local optimality. So far ejection chain studies have chiefly focused on internal as opposed to external levels of control, and new discoveries may be expected by broadening this focus. In addition, opportunities exist in many settings for applications of ejection chains where such forms of compound neighborhoods have so far remained uninvestigated.

A.5 STRATEGIC OSCILLATION

A considerable amount has been written on strategic oscillation and its advantages. However, one of the uses of this approach that is frequently overlooked involves the idea of oscillating among alternative choice rules and neighborhoods. An important aspect of strategic oscillation is the fact that there naturally arise different types of moves and choice rules that are appropriate for negotiating different regions and different directions of search. Thus, for example, there are many constructive methods in graph and scheduling problems, but strategic oscillation further leads to the creation of complementary "destructive methods" which can operate together with their constructive counterparts. Different criteria emerge as relevant for selecting a move to take on a constructive step versus one to take on a destructive step. Similarly, different criteria apply according to whether moves are chosen within a feasible region or outside a feasible region (and whether the search is moving toward or away from a feasibility boundary).

The variation among moves and evaluations introduces an inherent vitality into the search that provides one of the sources underlying the success of strategic oscillation approaches. This reinforces the motivation to apply strategic oscillation to the choice of moves and evaluation criteria themselves, selecting moves from a pool of possibilities according to rules for transitioning from one choice to another. In general, instead of picking a single rule, a process of invoking multiple rules provides a range of alternatives that run all the way from "strong diversification" to "strong intensification."

This form of oscillation has much greater scope than may at first be apparent, because it invokes the possibility of simultaneously integrating decision rules and neighborhoods, rather than only visiting them in a strategically determined sequence. To understand the considerations involved, it is useful to trace a brief history of approaches that offer prototypes for such designs.

Among the first of such approaches, an early study by Crowston et al. (1963) examined ways to generate improved decision rules for job shop scheduling, using a strategy of combining various standard local decision rules such as least-waiting- time, first-come/first-serve, longest-time-remaining-to-completion, and so forth. The decision rules were combined in two ways: (1) by choosing probabilistically from the set of component rules at each decision point, and (2) by integrating the rules into a "parametric rule" rather than by alternating among them. The probabilistic combination approach began by giving equal probabilities to selecting the rules, and then revised the probabilities of selection according to the quality of the solution produced — increasing the probability of applying a particular rule at a particular juncture if the current resulting schedule turned out to be one of higher than usual quality. The parametric rule approach first re-expressed the component rules to give them a common metric, and then formulated a single composite rule as a weighted combination of the components. The weight parameters were then systematically varied to find a preferred combination, following the philosophy that it is more meaningful to heed the input of different rules simultaneously, rather than to alternately give priority to one or another input in isolation from the rest. (A bow and arrow analogy was postulated, to suggest that it is better to simultaneously account for factors such as wind direction, distance from the target, length of the draw, and so forth, than to simply alternate among heeding these factors.) Both of these two types of approaches improved over using any single decision rule, and the parametric rule approach proved especially effective.

The probabilistic and parametric approaches were later unified and extended (Glover and McMillan (1986)) to allow both rules and moves (hence neighborhoods) to be integrated by means of a "generalized voting" concept. In applying this concept, the probabilistic element governs aspects of timing while the parametric element allows different rules to vote on an outcome implicitly, by restructuring and joining the rules into a master rule. These forms of integration can naturally be extended by making use of strategic oscillation and path relinking approaches. Details of such extensions, based on generating structured combinations of solutions, appear in Glover (1994a).

Such concepts are beginning to find counterparts in investigations being launched by the computer science community. The "agent" terminology is being invoked in such applications to characterize different choice mechanisms and neighborhoods as representing different agents. Relying on this representation, different agents then are assigned to work on (or "attack") the problem serially or in parallel. The CS community has begun to look upon this as a significant innovation — unaware of the literature where such ideas were introduced a decade or more ago — and the potential richness and variation of these ideas still seems not to be fully recognized. (For example, there have not yet been any studies that consider the idea of "strategically sequencing" rules and neighborhoods, let alone those that envision the notion of parametric integration. The further incorporation of adaptive memory structures to enhance the application of such concepts also lies somewhat outside the purview of most current CS proposals.) At the same time, TS research has neglected to conduct empirical investigations of the broader possibilities. This is clearly an area that deserves fuller study.

A.6 CLUSTERING AND CONDITIONAL ANALYSIS

To reinforce the theme of identifying opportunities for future research, we provide an illustration to clarify the relevance of clustering and conditional analysis, particularly as a basis for intensification and diversification strategies in tabu research.

An Example: Suppose 40 elite solutions have been saved during the search, and each solution is characterized as a vector $\mathbf{x}$ of zero-one variables x_j, for $j \in N = \{1, \ldots, n\}$. Assume the variables that receive positive values in at least one of the elite solutions are indexed x_1 to x_{30}. (Commonly in such circumstances, n may be expected to be somewhat larger than the number of positive valued variables, e.g., in this case, reasonable values may be $n = 100$ or 1000.)

For simplicity, we restrict attention to a simple weighted measure of consistency which is given by the frequency that the variables x_1 to x_{30} receive the value 1 in these elite solutions. (We temporarily disregard weightings based on solution quality and other aspects of "strongly determined" assignments.) Specifically, assume the frequency measures are as shown in the following table:

Variables $x_j = 1$	Number of Solutions
x_1 to x_{10}	24
x_{16} to x_{20}	21
x_{21} to x_{25}	17
x_{26} to x_{30}	12

Since each of x_1 to x_{15} receives a value of 1 in 24 of the 40 solutions, these variables tie for giving "most frequent" assignments. An intensification strategy that favors the inclusion of some number of such assignments would give equal bias to introducing each of x_1 to x_{15} at the value 1. (Such a bias would typically be administrated either by creating modified evaluations or by incorporating probabilities based on such evaluations.)

To illustrate the relevance of clustering, suppose the collection of 40 elite solutions can be partitioned into two subsets of 20 solutions each, whose characteristics are summarized in the following table.

Subset 1 (20 solutions)		Subset 2 (20 solutions)	
Variables $x_j = 1$	No. of Solutions	Variables $x_j = 1$	No. of Solutions
x_{11} to x_{15}	20	x_{16} to x_{20}	20
x_{21} to x_{25}	16	x_6 to x_{10}	16
x_1 to x_5	12	x_1 to x_5	12
x_6 to x_{10}	8	x_{26} to x_{30}	8
x_{26} to x_{30}	4	x_{11} to x_{15}	4
x_{16} to x_{20}	1	x_{21} to x_{25}	1

A very different picture now emerges. The variables x_1 to x_{15} no longer appear to deserve equal status as "most favored" variables. Treating them with equal status may be a useful source of diversification, as opposed to intensification, but the clustered data provide more useful information for diversification concerns as well. In short, clustering gives a relevant contextual basis for determining the variables (and combinations of variables) that should be given special treatment.

Conditional Relationships

To go a step beyond the level of differentiation provided by cluster analysis, it is useful to sharpen the focus by referring explicitly to interactions among variables. Such interactions can often be identified in a very straightforward way, and indeed can form a basis for more effective clustering. In many types of problems, the number of value assignments (or the number of "critical attributes") needed to specify a solution is relatively small compared to the total number of problem variables. (For example, in routing, distribution and telecommunication applications, the number of links contained in feasible constructions is typically a small fraction of those contained in the underlying graph.) Using a 0-1 variable representation of possibilities, it is not unreasonable in such cases to create a *cross reference* matrix, which identifies variables (or coded attributes) that simultaneously receive a value of 1 in a specific collection of elite solutions.

To illustrate, suppose the index set $P = \{1, \ldots, p\}$ identifies the variables x_j that receive a value of 1 in at least r solutions from the collection of elite solutions under consideration. (Apart from other strategic considerations, the parameter r can also be used to control the size of p, since larger values of r result in smaller values of p.)

Then we create a $p \times p$ symmetric matrix M whose entries m_{ij} identify the number of solutions in which x_i and x_j are both 1. (Thus, row M_i of M represents the sum of the solution vectors in which $x_i = 1$, restricted to components x_j for $j \in P$.) The value m_{ii} identifies the total number of elite solutions in which $x_i = 1$, and the value m_{ij}/m_{ii} represents the "conditional probability" that $x_j = 1$ in this subset of solutions. Because p can be controlled to be of modest size, as by the choice of r and the number of solutions admitted to the elite set, the matrix M is not generally highly expensive to create or maintain.

By means of the conditional probability interpretation, the entries of M give a basis for a variety of analyses and choice rules for incorporating preferred attributes into new solutions. Once an assignment $x_j = 1$ is made in a solution currently under consideration (which may be either partly or completely constructed), an updated conditional matrix M can be created by restricting attention to elite solution vectors for which $x_j = 1$. (Restricted updates of this form can also be used for look-ahead purposes.) Weighted versions of M, whose entries additionally reflect the quality of solutions in which specific assignments occur, likewise can be used.

Critical event memory (Glover (1995c), Glover and Kochenberger (1996)), provides a convenient mechanism to maintain appropriate variation when conditional influences are taken into account. The "critical solutions" associated with such memory in the present case are simply those constituting a selected subset of elite solutions. Frequency measures for value assignments can be obtained by summing these solution vectors for problems with 0-1 representations and the critical event control mechanisms can then assure assignments are chosen to generate solutions that differ from those of previous elite solutions.

Conditional analysis, independent of such memory structures, can also be a useful foundation for generating solution fragments to be exploited by vocabulary building processes, as discussed in Section 2.

A.7 REFERENT-DOMAIN OPTIMIZATION

Referent-domain optimization is based on introducing one or more optimization models to strategically restructure the problem or neighborhood, accompanied by auxiliary heuristic or algorithmic process to map the solutions back to the original problem space. The optimization models are characteristically devised to embody selected heuristic goals (e.g., of intensification, diversification or both), within the context of particular classes of problems.

There are several ways to control the problem environment as a basis for applying referent-domain optimization. A natural control method is to limit the structure and range of parameters that define a neighborhood (or the rules used to navigate through a neighborhood), and to create an optimization model that operates under these restricted conditions.

Example 1: The use of specially constructed neighborhoods (and aggregations or partitions of integer variables) permits the application of mixed integer programming (MIP) models to identify the best options from all moves of depth at most k (or from associated collections of at most k variables). When k is sufficiently small, such MIP models can be quite tractable, and produce moves considerably more powerful than those provided by lower level heuristics.

Example 2: In problems with graph-related structures, the imposition of directionality or non-looping conditions gives a basis for devising generalized shortest path (or dynamic programming) models to generate moves that are optimal over a significant subclass of possibilities. This type of approach gives rise to a combinatorial leverage phenomenon, where a low order effort (e.g., linear or quadratic) can yield solutions that dominate exponential numbers of alternatives. (See, e.g., Glover (1992), Rego and Roucairol (1996).)

Example 3: A broadly applicable control strategy, similar to that of a relaxation procedure, but more flexible, is to create a proxy model that "resembles" the original problem of interest, and which is easier to solve. Such an approach must be accompanied with a method to transform the solution to the proxy model into a trial solution for the original problem. A version of such an approach, which also can be used to induce special structure into the proxy model, can be patterned after layered surrogate/lagrangian decomposition strategies for mixed integer optimization.

Referent-domain optimization can also be applied in conjunction with the *target analysis* learning approach, to provide a basis for creating more effective solution strategies (Glover and Laguna (1992), Glover (1995b), Løkketangen and Glover (1996)). In this case, a first stage learning model, based on controlled solution attempts, identifies a set of desired properties of good solutions, together with *target solutions* (or *target regions*) that embody these properties. Then a second stage model is devised to generate neighborhoods and choice rules to take advantage of the outcomes of the learning model. Useful strategic possibilities are created by basing these two models on a proxy model for referent-domain optimization, to structure the outcomes so that they may be treated by one of the control methods indicated in the foregoing examples.

REFERENCES

[1] Aarts, E.H.L. and J.H.M. Korst (1989). "Boltzmann Machines for Travelling Salesman Problems," *European Journal of Operational Research*, 39,79-95.

[2] Ackley, D. (1987). *A Connectionist Model for Genetic Hillclimbing*, Kluwer, Dordrecht. Academic Publishers.

[3] Aggarwal, C.C., J.B. Orlin, and R.P. Tai (1996). "Optimized Crossover for the Independent Set Problem," MIT Technical Report, to appear in *Operations Research*.

[4] Aiex, R.M., S.L. Martins, C.C. Ribeiro and N.R. Rodriguez (1996). "Asynchronous parallel strategies for tabu search applied to the partitioning of VLSI circuits," Research report, Department of Computer Science, Catholic University of Rio de Janeiro.

[5] Al-Mahmeed, A.S. (1996). "Tabu Search Combination and Integration," In *Meta- Heurisitcs: Theory and Applications*. pp. 319-330, I.H. Osman & J.P. Kelly (eds.), Kluwer Academic Publishers.

[6] Andreatta, A.A. and C.C. Ribeiro (1994). "A Graph Partitioning Heuristic for the Parallel Pseudo-exhaustive Logical Test of VLSI Combinational Circuits," *Annals of Operations Research* 50, 1-36.

[7] Bäck, T., F. Hoffmeister and H. Schwefel (1991). "A Survey of Evolution Strategies," In *Proceedings of the Fourth International Conference on Genetic Algorithms*, eds. R. Belew and L. Booker, pp. 2-9. Morgan Kaufmann, San Mateo, CA.

[8] Balas, E., and W. Niehaus (1996). "Optimized Crossover-Based Genetic Algorithms for the Maximum Cardinality and Maximum Weight Clique Problems," Management Science Research Report #MSRR-613, submitted to *Journal of Heuristics*.

[9] Barnes, J. Wesley (1993). "Solving the Multiple-Machine Weighted Flow Time Problem Using Tabu Search," *IIE Transactions*, Vol 25, Number 2.

[10] Barnes, J.W. and W.B. Carlton (1995). "Solving the Vehicle Routing Problem with Time Windows Using Reactive Tabu Search," presented at the Fall INFORMS Conference in New Orleans, Louisiana, Tuesday, October 31, 1995, in review by the *IIE Transactions*.

[11] Battiti, R., P. Lee, A. Sartori, and G. Tecchiolli (1994a). "Totem: A Digital Processor for Neural Networks and Reactive Tabu Search," In *Fourth International Conference on Microelectronics for Neural Networks and Fuzzy Systems*, MICRONEURO 94, pages 17-25.

[12] Battiti, R. P. Lee, A. Sartori, and G. Tecchiolli (1994b). "Combinatorial Optimization for Neural Nets: RTS Algorithm and Silicon," Technical Report no. 9406-04, IRST, Trento, IT.

[13] Battiti, R. and G.Tecchiolli (1992). "Parallel Based Search for Combinatorial Optimization: Genetic Algorithms and Tabu Search," *Microprocessor and Microsystems*, 16:351-367.

[14] Battiti, R. and G. Tecchiolli (1994b). "The Reactive Tabu Search," *ORSA Journal on Computing*, 6(2):126-140.

[15] Battiti, R. and G. Tecchiolli (1994c). "Simulated Annealing and Tabu Search in the Long Run: A Comparison on QAP Tasks," *Computer and Mathematics with Applications*, 28(6):1-8.

[16] Battiti, R. and G. Tecchiolli (1995a). "The Continuous Reactive Tabu Search: Blending Combinatorial Optimization and Stochastic Search for Global Optimization," *Annals of Operations Research* (in press).

[17] Battiti, R. and G. Tecchiolli (1995b). "Training Neural Nets with the Reactive Tabu Search," *IEEE Transactions on Neural Networks*, 6(5):1185-1200

[18] Battiti, R. and G. Tecchiolli (1995c). "Local Search with Memory: Benchmarking RTS," *Operations Research Spektrum*, 17(2/3):67-86.

[19] Beltran, H.F. and D. Skorin-Kapov (1994). "On Minimum Cost Isolated Failure Immune Networks," *Telecommunication Systems*, 3, 183-200.

[20] Błażewicz, J., P. Hawryluk and R. Walkowiak (1993). "Using a Tabu Search Approach for Solving the Two-dimensional Irregular Cutting Problem," *Annals of Operations Research*, 41, 313-325.

[21] Błażewicz, J. and R. Walkowiak (1995). "A Local Search Approach for Two-dimensional Irregular Cutting," *OR Spectrum*, 17, 93-98.

[22] Brumelle, S., D. Granot, M. Halme, and I. Vertinsky (1996). "A Tabu Search Algorithm for Finding a Good Forest Harvest Schedule Satisfying Green-up Constraints," Forest Economics and Policy Analysis Research Unit The University of British Columbia Vancouver, B.C., Canada

[23] Carlton, William B. and J. Wesley Barnes (1995). "A Note on Hashing Functions and Tabu Search Algorithms," To appear in the *European Journal of Operational Research.*

[24] Carlton, William B. and J. Wesley Barnes (1996). "Solving the Traveling Salesman Problem with Time Windows Using Tabu Search," to appear in *IIE Transactions.*

[25] Chakrapani, J. and J. Skorin-Kapov (1992). "Connectionist Approaches to the Quadratic Assignment Problem," *Computers and Operations Research*, 19(3/4), 287-295.

[26] Chakrapani, J. and J. Skorin-Kapov (1993). "Connection Machine Implementation of a Tabu Search Algorithm for the Traveling Salesman Problem," *Journal of Computing and Information Technology*, 1(1), 29-36.

[27] Chakrapani, J. and J. Skorin-Kapov (1993a). "Massively Parallel Tabu Search for the Quadratic Assignment Problem," *Annals of Operations Research*, 41, 327-341.

[28] Chakrapani, J. and J. Skorin-Kapov (1995). *Mapping Tasks to Processors to Minimize Communication Time in a Multiprocessor System, The Impact of Emerging Technologies of Computer Science and Operations Research*, Kluwer Academic Publishers, 45-64.

[29] Chiang, W.-C. and P. Kouvelis (1994a). "Simulated Annealing and Tabu Search Approaches for Unidirectional Flowpath Design for Automated Guided Vehicle Systems," *Annals of Operation Research*, Vol. 50.

[30] Chiang, W.-C. and P. Kouvelis (1994b). "An Improved Tabu Search Heuristic for Solving Facility Layout Design Problems," *International Journal of Production Research* (forthcoming).

[31] Chiang, W.-C. and R. Russell (1995). "A Reactive Tabu Search Metahuristic for the Vehicle Routing Problem with Time Windows," submitted to *ORSA Journal on Computing*, Working Paper, Department of Quantitative Methods and Management Information Systems.

[32] Costamagna, F., A. Fanni and G. Giacinto (1995). "Tabu Search for the Optimization of B-ISDN Telecommunication Networks," Tech. Report, Dept. of Electrical Eng., University of Cagliari, no. 60.

[33] Crainic, T.G., M. Toulouse and M. Gendreau (1993a). "A Study of Synchronous Parallelization Strategies for Tabu Search," Publication 934, Centre de recherche sur les transports, Universite de Montreal, 1993.

[34] Crainic, T.G., M. Toulouse, and M. Gendreau (1993b). "Appraisal of Asynchronous Parallelization Approaches for Tabu Search Algorithms," Publication 935, Center de recherche sur les transports, Universite de Montreal, 1993.

[35] Crowston, W.B., F. Glover, G.L. Thompson and J.D. Trawick (1963). "Probabilistic and Parametrick Learning Combinations of Local Job Shop Scheduling Rules," *ONR Research Memorandum No. 117*, GSIA, Carnegie-Mellon University, Pittsburg, PA.

[36] Daduna, J.R., and S. Voss (1995). "Practical Experiences in Schedule Synchronization," *Lecture Notes in Economics and Mathematical Systems*, 430, 39-55.

[37] Dammeyer, F., and S. Voss (1993). "Dynamic Tabu List Management Using the Reverse Elimination Method," *Annals of Operations Research*, 41, 31-46.

[38] Davis, L. (1989). "Adapting Operator Probabilities in Genetic Algorithms," In *Proceedings of the Third International Conference on Genetic Algorithms*, pp. 61- 69, Morgan Kaufmann, San Mateo, CA.

[39] Dell'Amico, M. and M. Trubian (1993). "Applying Tabu Search to the Job-Shop Scheduling Problem," *Annals of Operations Research*, Vol. 41, 231-252.

[40] Dell'Amico, M. and Francesco Maffioli (1996). "A New Tabu Search Approach to the 0-1 Equicut Problem," In *Meta-Heuristics: Theory & Applications*, pp. 361-378, I.H. Osman & J.P. Kelly (eds.), Kluwer Academic Publishers.

[41] Dodin, B., A.A. Elimam and E. Rolland (1996). "Tabu Search in Audit Scheduling," submitted to the special issue of *European Journal of Operation Research* on Tabu Search.

[42] Domschke, D., P. Forst and S. Voss (1992). "Tabu Search Techniques for the Quadratic Semi-assignment Problem," In: G. Fandel, T. Gulledge and A. Jones (eds.), *New Directions for Operations Research in Manufacturing*, Springer, Berlin, 389 - 405.

[43] Dorndorf, U. and E. Pesch (1994). "Fast Clustering Algorithms," *ORSA Journal on Computing*, 6, 141-153.

[44] Eiben, A.E., P.-E. Raue and Zs. Ruttkay (1994). "Genetic Algorithms with Multi-Parent Recombination," In Y. Davidor, H.-P Schwefel, and R. Manner (Eds.) *Proceedings of the Third International Conference on Parallel Problem Solving from Nature (PPSN)* (pp. 78-87). New York: Springer-Verlag.

[45] Eschelman, L.J. and J.D. Schaffer (1992). "Real-Coded Genetic Algorithms and Interval-Schemata," Technical Report, Phillips Laboratories.

[46] Fanni, A., G. Giacinto and M. Marchesi (1996). "Tabu Search for Continuous Optimization of Electromagnetic Structures," submitted to *Int. Workshop on Optimization and Inverse Problems in Electromagnetism*, Brno, Czech Republic, June.

[47] Fleurent, C., F. Glover, P. Michelon and Z. Valli (1996). "A Scatter Search Approach for Unconstrained Continuous Optimization," *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pp. 643-648.

[48] Fréville, A. and G. Plateau (1986). "Heuristics and Reduction Methods for Multiple Constraint 0-1 Linear Programming Problems," *European Journal of Operational Research*, 24, 206-215.

[49] Fréville, A. and G. Plateau (1993). "An Exact Search for the Solution of the Surrogate Dual of the 0-1 Bidimensional Knapsack Problem," *European Journal of Operational Research*, 68, 413-421.

[50] Glover, F. (1965). "A Multiphase-dual Algorithm for the Zero-one Integer Programming Problem," *Operations Research*, 13, 879-919.

[51] Glover, F. (1968). "Surrogate Constraints," *Operations Research*, 16, 741-749.

[52] Glover, F. (1975). "Surrogate Constraint Duality in Mathematical Programming," *Operations Research*, 23, 434-451.

[53] Glover, F. (1977). "Heuristics for Integer Programming Using Surrogate Constraints," *Decision Sciences*, 8, 156-166.

[54] Glover, F. (1989). "Tabu Search, Part I," *ORSA Journal on Computing*, 1, 190-206.

[55] Glover, F. (1992). "Ejection Chains, Reference Structures and Alternating Path Methods for Traveling Salesman Problems," University of Colorado. Shortened version published in *Discrete Applied Mathematics*, 1996, 65, 223-253.

[56] Glover, F. (1994a). "Tabu Search for Nonlinear and Parametric Optimization (with Links to Genetic Algorithms)," *Discrete Applied Mathematics*, 49, 231-255.

[57] Glover, F. (1994b). "Genetic Algorithms and Scatter Search: Unsuspected Potentials," *Statistics and Computing*, 4, 131-140.

[58] Glover, F. (1995a). "Scatter Search and Star-Paths: Beyond the Genetic Metaphor," *OR Spektrum*, 17:125-137.

[59] Glover, F. (1995b). "Tabu Search Fundamentals and Uses." Graduate School of Business, University of Colorado, condensed version published in *Mathematical Programming: State of the Art*, 1994, Birge & Murty, eds., 64-92.

[60] Glover, F. (1995c) "Tabu Thresholding: Improved Search by Nonmonotonic Trajectories," *ORSA Journal on Computing*, Vol. 7, No. 4, 426-442.

[61] Glover, F. and G. Kochenberger (1996). "Critical Event Tabu Search for Multidimensional Knapsack Problems," In *Meta-Heuristics: Theory & Applications*, I.H. Osman and J.P. Kelly (eds.), 407-427, Kluwer Academic Publishers, Norwell, MA.

[62] Glover, F. and M. Laguna (1993). "Tabu Search," In *Modern Heuristic Techniques for Combinatorial Problems.* ed. C. Reeves, pp. 70-141. Blackwell Scientific Publishing, Oxford.

[63] Glover, F. and C. McMillan (1986). "The General Employee Scheduling Problem: An Integration of Management Science and Artificial Intelligence," *Computers and Operations Research*, 15:5, 563-593.

[64] Greenberg, H.J. and W.P. Pierskalla (1970). "Surrogate Mathematical Programs," *Operations Research*, 18, 924-939.

[65] Greenberg, H.J. and W.P. Pierskalla (1973). "Quasi-conjugate Functions and Surrogate Duality," *Cahiers du Centre d'Etudes de Recherche Operationelle*, 15, 437-448.

[66] Hansen, P., B. Jaumard and Da Silva (1992). "Average Linkage Divisive Hierarchical Clustering," to appear in *Journal of Classification.*

[67] Hansen, P., B. Jaumard and M. Poggi di Aragao (1992). "Mixed Integer Column Generation Algorithms and the Probabilistic Maximum Satisfiability Problem," *Proceedings of the 2nd Integer Programming and Combinatorial Optimization Conference*, Carnegie Mellon.

[68] Hertz, A., B. Jaumard, and M. Poggi di Aragao (1991). "Topology of Local Optima for the K-Coloring Problem," to appear in *Discrete Applied Mathematics.*

[69] Jaumard, B., P. Hansen, and M. Poggi di Aragao (1991). "Column Generation Methods of Probabilistic Logic," *ORSA Journal on Computing*, 3, 135-148.

[70] Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, MI.

[71] Karwan, M.H. and R.L. Rardin (1976). "Surrogate Dual Multiplier Search Procedures in Integer Programming," School of Industrial Systems Engineering, *Report Series No. J-77-13*, Georgia Institute of Technology.

[72] Karwan, M.H. and R.L. Rardin (1979). "Some Relationships Between Lagrangean and Surrogate Duality in Integer Programming," *Mathematical Programming*, 17, 230-334.

[73] Kelly, J.P., M. Laguna and F. Glover (1996). "Tabu Search and Scatter Search for Optimizing Simulation," School of Business, University of Colorado, Presented at the national INFORMS Meeting, Washington, D.C.).

[74] Kelly, J.P. and J. Xu (1995). "Tabu Search and Vocabulary Building for Routing Problems," Graduate School of Business Administration, University of Colorado at Boulder.

[75] Kincaid, R.K., (1995). "Actuator Placement for Active Sound and Vibration Control of Cylinders," *NASA ASEE Report.*

[76] Kincaid, R.K. and R.T. Berger (1993). "The Damper Placement Problem on Space Truss Structures," *Location Science*, Vol. 1, No. 3, pp. 219-234.

[77] Kincaid, R.K., A.D. Martin, and J.A. Hinkley (1995). "Heuristic Search for the Polymer Straightening Problem," *Computational Polymer Science*, Vol. 5, pp. 1-5.

[78] Laguna, M., J.P. Kelly, J.L. Gonzalez-Velarde and F. Glover (1995). "Tabu Search for the Multilevel Generalized Assignment Problem," *European Journal of Operations Research*, 82, 176-189.

[79] Laguna M., R. Marti and V. Valls (1995). "Arc Crossing Minimization in Hierarchical Digraphs with Tabu Search," Tech Report, Graduate School of Business, University of Colorado, Campus Box 419, Boulder, CO 80309.

[80] Lee, P., A. Sartori, G. Tecchiolli and A. Zorat (1995). "A Parallel Processor for Neural Networks," 1995 Symposium on VLSI Circuits - Digest of Technical Papers, 8-10 June, 1995, Kyoto, *IEEE CAT. No. 95 CH 35780*, p. 81-82.

[81] Løkketangen, A. and F. Glover (1996). "Surrogate Constraint Methods with Simple Learning for Satisfiability Problems," *Proceedings of the DIMACS workshop on Satisfiability Problems: Theory and Applications*, D-Z. Du, J. Gu and P. Pardalos, editors.

[82] Løkketangen, A. and D. L. Woddruff, (1996). "Progressive Hedging and Tabu Search Applied to Mixed Integer (0,1) Multi-Stage Stochastic Programming," Forthcoming in *Journal of Heuristics*.

[83] Lopez, L., M.W. Carter and M. Gendreau (1996). "The Hot Strip Mill Production Scheduling Problem: A Tabu Search Approach," Centre de recherche sur les transports, Universite de Montreal.

[84] Marti, R. (1996). "An Aggressive Search Procedure for the Bipartite Drawing Problem," In *Meta-Heuristics: Theory & Applications*, 97-113, I.H. Osman and J.P. Kelly (eds.), Kluwer Academic Publishers.

[85] Martins, S.L., C.C. Ribeiro and N.R. Rodriguez (1996). "Parallel Programming Tools for Distributed Memory Environments," (in Portuguese), *Research report MCC-01/96*, Department of Computer Science, Catholic University of Rio de Janeiro.

[86] Mazzola, J.B. and R.H. Schantz (1995a). "Single-facility Resource Allocation Under Capacity-based Economies and Diseconomies of Scope," *Management Science*, 41, 669-689.

[87] Mazzola, J.B. and R.H. Schantz (1995b). "Multiple-Facility Loading Under Capacity-Based Economies of Scope," Working Paper, Fuqua School of Business, Duke University, August.

[88] Mazzola, J.B., A.W. Neebe, and C.M. Rump (1995). "Multiproduct Production Planning in the Presence of Work-Force Learning," Working Paper, Fuqua School of Business, Duke University, August.

[89] Michalewicz, Z. and C. Janikow (1991). "Genetic Algorithms for Numerical Optimization," *Statistics and Computing*, 1, 75-91.

[90] Mühlenbein, H., M. Gorges-Schleuter and O. Kremer (1988). "Evolution Algorithms in Combinatorial Optimization," *Parallel Computing*, 7, 65-88.

[91] Mühlenbein, H. and D. Schlierkamp-Voosen (1994). "The Science of Breeding and its Application to the Breeder Genetic Algorithm," *Evolutionary Computation* 1, 335-360.

[92] Mühlenbein, H. and H.-M. Voigt (1996). "Gene Pool Recombination in Genetic Algorithms," In *Meta-Heurisitics: Theory & Applications*, I.H. Osman and J.P. Kelly, eds., 53-62, Kluwer Academic Publishers.

[93] Mulvey, J. (1995). "Generating Scenarios for the Towers Perrin Investment Systems," Technical Report SOR Report, Princeton University, Department of Civil Engineering and Operations Research, Princeton University, Princeton, NJ, to appear *Interfaces*.

[94] Nowicki E. and Smutnicki C., (1993). "A Fast Taboo Search Algorithm for the Job Shop," Report ICT PRE 8/93, Technical University of Wroclaw. *Management Science*. (in print), 1996.

[95] Nowicki, E. and C. Smutnicki (1994). "A Fast Taboo Search Algorithm for the Flow Shop," Report ICT PRE 8/94, Technical University of Wroclaw. *European J. Oper. Res.* (in print), 1996.

[96] Nowicki, E. and C. Smutnicki (1995). "The Flow Shop with Parallel Machines: A Tabu Search Approach," Report ICT PRE 30/95, Technical University of Wroclaw.

[97] Pardalos, P.M., X. Liu and G. Xue (1995). "Protein Conformation of Lattice Using Tabu Search," Unpublished.

[98] Pesch, E. and F. Glover (1995). "TSP Ejection Chains," Graduate School of Business, University of Colorado, Boulder, to appear in *Discrete Applied Mathematics*.

[99] Porto, S.C.S. and C.C. Ribeiro (1995a). "A Tabu Search Approach to Task Scheduling on Heterogeneous Processors Under Precedence Constraints," *International Journal of High-Speed Computing* 7, 45-71.

[100] Porto, S.C.S. and C.C. Ribeiro (1995b). "Parallel Tabu Search Message-passing Synchronous Strategies for Task Scheduling Under Precedence Constraints," to appear in *Journal of Heuristics.*

[101] Potvin, J. Y., T. Kervahut, B. L. Garcia and J. M. Rousseau (1996). "The Vehicle Routing Problem with Time Windows - Part I: Tabu Search," *INFORMS Journal on Computing* Vol 8, No. 2, 158-164.

[102] Rego, C. (1996a). "Relaxed Tours and Path Ejections for the Traveling Salesman Problems," Universidade Portucalense, Porto, Portugal.

[103] Rego, C. (1996b). "A Subpath Ejection Method for the Vehicle Routing Problem," Universidade Portucalense, Porto, Portugal.

[104] Rego, C. and C. Roucairol (1996). "A Parallel Tabu Search Algorithm Using Ejection Chains for the Vehicle Routing Problem," In *Meta-Heuristics: Theory & Applications*, 661-675, I.H. Osman and J.P. Kelly, (eds.), Kluwer Academic Publishers.

[105] Rochat, Y. and F. Semet (1994). "A Tabu Search Approach for Delivering Pet Food and Flour in Switzerland," *J. Opl. Res. Soc.*, Vol 45, No. 11 1233-1246.

[106] Rochat, Y. and E. Taillard (1995). "Probabilistic Diversification and Intensification in Local Search for Vehicle Routing," *Journal of Heuristics*, Vol. 1, No. 1, 147-167.

[107] Rolland, E. (1995). "A Tabu Search Method for Constrained Real-Number Search: Applications to Portfolio Selection," Working paper, The A. Gary Anderson Graduate School of Management, University of California, Riverside.

[108] Rolland, E. and H. Johnson (1996). "Skewness and the Mean-Variance Frontier: A Tabu Search Approach," Working paper, The A. Gary Anderson Graduate School of Management, University of California, Riverside.

[109] Rolland, E., H. Pirkul and F. Glover (1995). "Tabu Search for Graph Partitioning," in *Annals of Operations Research*, Special issue on Metaheuristics in Optimization.

[110] Rolland, E., D.A. Schilling and J.R. Current (1996). "The P-Median Problem: Efficient Solutions using Tabu Search," to appear in *European Journal of Operational Research*.

[111] Skorin-Kapov, J. (1990). "Tabu Search Applied to the Quadratic Assignment Problem," *ORSA Journal on Computing*, Vol.2, No.1, 33-45.

[112] Skorin-Kapov, J. (1994). "Extensions of a Tabu Search Adaptation to the Quadratic Assignment Problem," *Computers and Operations Research*, 21(8), 855-865.

[113] Skorin-Kapov, J. and J.-F. Labourdette (1995). "On Minimum Congestion in Logically Rearrangeable Multihop Lightwave Networks," forthcoming in *Journal of Heuristics*.

[114] Skorin-Kapov, D., J. Skorin-Kapov and M. O'Kelly (1995). "Tight Linear Programming Relaxations of p-Hub Median problems," (forthcoming in) *European Journal of Operational Research*.

[115] Skorin-Kapov, D. and J. Skorin-Kapov (1994). "On Tabu Search for the Location of Interacting Hub Facilities," *European Journal on Operational Research*, 73(3), 501-508.

[116] Skorin-Kapov, J. and A. Vakharia (1993). "Scheduling a Flow-Line Manufacturing Cell: A Tabu Search Approach," *International Journal of Production Research*, 31(7), 1721-1734.

[117] Spears, W.M. and K.A. DeJong (1991). "On the Virtues of Uniform Crossover," In *4th International Conference on Genetic Algorithms*, La Jolla, CA.

[118] Sun, M., J.E. Aronson, P.G. McKeown and D. Drinka (1995). "A Tabu Search Heuristic Procedure for the Fixed Charge Transportation Problem," Working Paper No. 95-414, Terry College of Business, The University of Georgia, May.

[119] Taillard, E., P. Badeau, M. Gendreau, F. Guertin and J-Y. Potvin (1995). "A New Neighborhood Structure for the Vehicle Routing Problem with Time Windows," Centre de Recherche sur les Transports, Publication CRT-95-66.

[120] Taillard, E. (1993). "Parallel Iterative Search Methods for Vehicle Routing Problems," *Networks*, Vol. 23, 661-673.

[121] Trafalis, T., and I. Al-Harkan (1995). "A Continuous Scatter Search Approach for Global Optimization," Extended Abstract in: *Conference in Applied Mathematical Programming and Modeling (APMOD'95)*, London, UK, 1995.

[122] Toulouse, Michael, Teodor G. Crainic and Michel Gendreau (1996). "Communication Issues in Designing Cooperative Multi-Thread Parallel Searchers" In *Meta- Heuristics: Theory & Applications*, pp. 503-522, I.H. Osman and J.P. Kelly (eds.), Kluwer Academic Publishers.

[123] Ulder, N.L.J., E. Pech, P.J.M. van Laarhoven, H.J. Bandelt and E.H.L. Aarts (1991). "Genetic Local Search Algorithm for the Traveling Salesman Problem," In *Parallel Problem Solving from Nature*, eds. R. Maenner and H.P. Schwefel, pp. 109-116. Springer-Verlag, Berlin.

[124] Vaessens, R.J.M, E.H.L. Aarts and J.K. Lenstra (1995). "Job Shop Scheduling by Local Search," *Memorandum COSOR 94-05*, Eidhoven University of Technology.

[125] Valls V., R. Marti and P. Lino (1995). "A Tabu Thresholding Algorithm for Arc Crossing Minimization in Bipartite Graphs," *Annals of Operations Research*, Vol. 60, *Metaheuristics in Combinatorial Optimization*.

[126] Voss, S. (1992). "Network Design Formulations in Schedule Synchronization," *Lecture Notes in Economics and Mathematical Systems*, 386, 137-152.

[127] Voss, S. (1993). "Tabu Search: Applications and Prospects," In: D.-Z. Du und P.M. Pardalos (Hrsg.), *Network Optimization Problems: Algorithms, Applications and Complexity*, World Scientific, Singapore, 333 - 353.

[128] Voss, S. (1994). "Concepts for Parallel Tabu Search," *Technische Hochschule Dormstadt*, Germany.

[129] Whitley, D., V.S. Gordon and K. Mathias (1994). "Lamarckian Evolution, the Baldwin Effect and Function Optimization," in *Proceedings of the Parallel Problem Solving from Nature*, 3, pp. 6-15. New York: Springer-Verlag.

[130] Whitley, D. (1993). *Foundations of Genetic Algorithms 2*, Morgan Kaufmann.

[131] Woodruff, D. (1996). "Proposals for Chunking and Tabu Search," Submitted to *European Journal of Operational Research*.

[132] Wright, A.H. (1990). "Genetic Algorithms for Real Parameter Optimization," In *Foundations of Genetic Algorithms*, ed. G. Rawlins, pp. 205-218. Morgan Kaufmann, Los Altos, CA.

[133] Zenios, S. (1996). "Dynamic Financial Modeling and Optimizing the Design of Financial Products," Presented at the National INFORMS Meeting, Washington, D.C., May 1996.