An Unconstrained Quadratic Binary Programming Approach to the Vertex Coloring Problem[1]

by

Gary A. Kochenberger
School of Business, University of Colorado at Denver
Gary.Kochenberger@cudenver.edu

Fred Glover
School of Business, University of Colorado at Boulder
Fred.Glover@Colorado.edu

Bahram Alidaee
Hearin Center for Enterprise Science, University of Mississippi
Balidaee@bus.olemiss.edu

Cesar Rego
Hearin Center for Enterprise Science, University of Mississippi
Crego@bus.olemiss.edu

April, 2003

ABSTRACT:

The vertex coloring problem has been the subject of extensive research for many years. Driven by application potential as well as computational challenge, a variety of methods have been proposed for this difficult class of problems. Recent successes in the use of the unconstrained quadratic programming (UQP) model as a unified framework for modeling and solving combinatorial optimization problems have motivated a new approach to the vertex coloring problem. In this paper we present a UQP approach to this problem and illustrate its attractiveness with preliminary computational experience.

[1] This research was supported in part by ONR grants N000140210151 and N000140010598.

1. Introduction:

The unconstrained quadratic program can be written in the form:

$$\text{UQP: } \min f(x) = xQx$$

where Q is an n by n matrix of constants and x is an n-vector of binary variables. UQP is notable for its ability to represent a significant variety of important problems. The applicability of this representation has been reported in settings such as social psychology (Harary [30]), spin glasses and circuit board layout (Grotschel, et. al. [23] and Palubeckis [43]), financial analysis (Laughunn, [35], McBride and Yormak, [39]), computer aided design (Krarup and Pruzan [34]), traffic management (Gallo et al. [17, Witsgall, [50]), machine scheduling (Alidaee, Kochenberger, and Ahmadian, [1]), cellular radio channel allocation (Chardaire and Sutter [11), molecular conformation (Phillips and Rosen [47]) and the prediction of epileptic seizures (Iasemidus, et. al. [31]. Moreover, many satisfiability problems (Boros and Hammer [25], Boros and Prekopa [8]) as well as combinatorial optimization problems pertaining to graphs such as determining maximum cliques, maximum cuts, maximum vertex packing, minimum coverings, maximum independent sets, and maximum independent weighted sets are known to be capable of being formulated by the UQP problem (see Bourjolly, et. al. [9], Hammer et. al. [25] as well as Pardalos and Rodgers [44,45], and Pardalos and Xue [46]).

The application potential of UQP is yet substantially greater than this, however, due to reformulation methods that enable certain constrained models to be re-cast in the form of UQP. Hammer and Rudeanu [26], Hansen [27], and Hansen et. al. [28] show that any quadratic (or linear) objective in bounded integer variables and constrained by linear equations can be reformulated as a UQP model. Our purpose in this paper is to illustrate how this approach can be effectively employed to model and solve vertex coloring problems. In the section that follows, we present the transformations we use to convert constrained problems into the form of xQx, followed by a brief overview of the tabu search heuristic we use to solve the resulting UQP model. In section 3 we apply the methodology to the k-colorable problem and present our computational experience with a set of standard test problems. This is followed in section 4 with a demonstration of the ability of the UQP formulation to model and solve the optimum cardinality vertex coloring problem, which consists of finding a feasible coloring utilizing the minimum number of colors. Then in section 5 we offer a summary and some conclusions.

2. Motivation and Transformation to xQx:

The goal of this paper is not to propose a new algorithm for coloring problems in the sense of introducing a highly specialized method in a competition for the fastest running time. Rather, our purpose is to propose an alternative framework for solving these problems, enabling them to be handled very effectively by a combined modeling and solution approach that can also be directly applied to many other classes of problems. As we show, our approach is surprisingly effective and robust, even with no special tuning or tailoring for coloring problems. The resulting ability to use general purpose algorithms for this important problem domain (as well as others) provides new incentive for research into improved algorithms that operate within the larger binary quadratic programming framework.

We take as our starting point the constrained problem

$$\min x_0 = xQx$$

subject to

$$Ax = b, \quad x \text{ binary}$$

This model accommodates both quadratic and linear objective functions since the linear case results when Q is a diagonal matrix (observing that $x_j^2 = x_j$ when x_j is a 0-1 variable). Problems with inequality constraints can also be put into this form by representing their bounded slack variables by a binary expansion. These constrained quadratic optimization models are converted into equivalent UQP models by adding a quadratic infeasibility penalty function to the objective function in place of explicitly imposing the constraints $Ax = b$.

Specifically, for a positive scalar P, we have

$$\begin{aligned} x_0 &= xQx + P(Ax-b)^t(Ax-b) \\ &= xQx + xDx + c \end{aligned}$$

$$= x\hat{Q}x + c$$

where the matrix D and the additive constant c result directly from the matrix multiplication indicated. Dropping the additive constant, the equivalent unconstrained version of our constrained problem becomes

$$UQP(PEN): \min x\hat{Q}x,\ x\ binary$$

This conversion of a constrained problem to UQP has been known in the literature for many years. (see, for instance, Hammer, et. al. [25], Hammer and Rudeanu [26], and Hansen [27]). From a theoretical standpoint, a suitable value of the penalty scalar P can always be selected so that the optimal solution to UQP(PEN) is the optimal solution to the original constrained problem (Hammer and Rudeanu [26]).

We refer to the preceding general transformation as *transformation #1*. A very important special class of constraints that arise in many applications can be handled by an alternative approach, given below, which we call *transformation #2*.

In particular, consider problems with considerations that isolate two specific alternatives and prohibit both from being chosen. That is, for a given pair of alternatives, one or the other but not both *may* be chosen. If x_j and x_k are binary variables denoting whether or not alternatives j and k are chosen, the standard constraint that allows one choice but precludes both is:

$$x_j + x_k \leq 1$$

Then, for a positive scalar P, adding the penalty function Px_jx_k to the objective function is a simple alternative to imposing the constraint is a traditional manner. This penalty function has sometimes been used by to convert certain optimization problems on graphs (e.g., the maximum clique problem) into an equivalent UQP model (Hammer et. al. [25], Pardalos and Xue [46]). Its potential application, however, goes far beyond these settings as demonstrated in the present paper and in the earlier survey by Kochenberger, Glover, Alidaee and Rego [33]. Note that variable upper bound constraints of the form $x_{ij} \leq y_i$ can be accommodated by transformation # 2 by first replacing the y_i variables by their complement. The opportunity to employ this modeling "trick" in the context of transformation # 2 commonly arises in fixed charge problems and a variety of other settings. Note that both transformations 1 and 2 contain a scalar penalty P. Our experience with a wide variety of problems, consistent with the experience of others, is that frequently P can be chosen much smaller than one would expect. For example, in problems involving a linear objective function and the constraints of transformation #2, P can be chosen as small as the largest objective function coefficient (see Boros and Hammer [6]).

Before illustrating the application of UQP to vertex coloring, we comment on solution procedures for UQP.

2.1 Solving UQP:

UQP has been the focus of a considerable research in recent years, including both exact and heuristic solution approaches. Notable recent studies addressing UQP are those by Williams [49], Pardalos and Rodgers [45], Boros, Hammer and Sun [7], Chardaire and Sutter [11], Billionnet and Sutter [5], Palubeckis [43], Glover, Kochenberger and Alidaee [22], Glover, Kochenberger, Alidaee, and Amini, [20], Alkhamis, Hasan and Ahmed [2], Beasley [4], Lodi, Allemand and Liebling [37], and Amini, Alidaee and Kochenberger [3]. Other promising work is reported by Helmberg and Rendl [29], Katayama, Tani and Narihisa [32], Merz and Freisleben [40,41], Merz and Katayma [42] as well as Glover et. al. [21]. These various studies approach the problem by branch and bound, decomposition, semidefinite programming and cutting planes, tabu search, simulated annealing, evolutionary methods such as genetic algorithms and scatter search, as well as simple one-pass heuristic methods. Each of these approaches exhibits some degree of success and could in principle be utilized to solve problems reformulated as UQP problems. However, the exact methods degrade rapidly with problem size, and have meaningful application to general UQP problems with no more than 100 - 200 variables.

For larger problems, heuristic methods are required. Several of the heuristic methods referenced above are reported to perform well on general UQP models with up to a few thousand variables, and the simple one-pass heuristics [7,21] have been usefully employed to address problems with more than 13,000 variables. Two methods we have found to be particularly successful for a wide variety of problems are based on tabu search [18,20] and on the related evolutionary strategy of scatter search [3,19].

Although not pursued by us here, an alternative approach is to solve UQP as a continuous non-linear optimization problem within the unit cube. This allows other heuristic/approximation methods based on continuous optimization methodologies to be applied (see [6], [8], [48]). In the following section we sketch the basis of our tabu search solution approach that was used to produce the computational results presented later in the paper.

2.2 Tabu Search Overview:

Our TS method for UQP is centered around the use of strategic oscillation, which constitutes one of the primary strategies of tabu search. The variant of strategic oscillation we employ may be briefly described in overview as follows.

The method alternates between constructive phases that progressively set variables to 1 (whose steps we call “add moves”) and destructive phases that progressively set variables to 0 (whose steps we call “drops moves”). To control the underlying search process, we use a memory structure that is updated at c*ritical events*, identified by conditions that generate a subclass of locally optimal solutions. Solutions corresponding to critical events are called *critical solutions.*

A parameter *span* is used to indicate the amplitude of oscillation about a critical event. We begin with *span* equal to 1 and gradually increase it to some limiting value. For each value of *span*, a series of alternating constructive and destructive phases is executed before progressing to the next value. At the limiting point, *span* is gradually decreased, allowing again for a series of alternating constructive and destructive phases. When *span* reaches a value of 1, a *complete span cycle* has been completed and the next cycle is launched. The search process is typically allowed to run for a pre-set number of span cycles.

Information stored at critical events is used to influence the search process by penalizing potentially attractive add moves (during a constructive phase) and inducing drop moves (during a destructive phase) associated with assignments of values to variables in recent critical solutions. Cumulative critical event information is used to introduce a subtle long term bias into the search process by means of additional penalties and inducements similar to those discussed above. Other standard elements of tabu search such as short and long term memory structures are also included. A complete description of the framework for the method is given in Glover, Kochenberger, Alidaee and Amini [20].

3. The K-Coloring Problem:

Vertex coloring problems seek to assign colors to nodes of a graph subject to the requirement that adjacent nodes must be assigned different colors. The K-coloring problem attempts to find such a coloring using exactly K colors. This problem is known to be NP-hard.

K-coloring problems can be modeled as satisfiability problems as follows:

Let x_{ij} be 1 if node i is assigned color j, and 0 otherwise.

Since each node must be colored, we have

$$\sum_{j=1}^{K} x_{ij} = 1 \quad i = 1,\ldots, n \tag{1}$$

where n is the number of nodes in the graph. The requirement that adjacent nodes are assigned different colors is handled by imposing the constraints

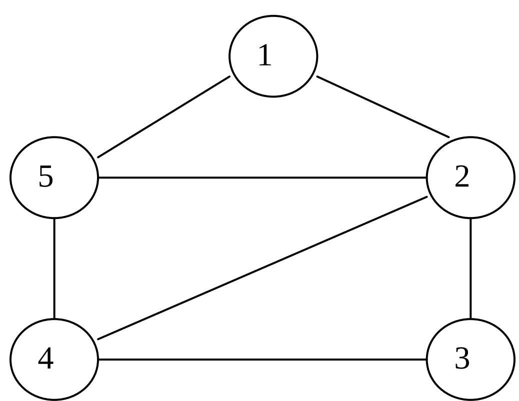
$$x_{ip} + x_{jp} \le 1 \quad p = 1,\ldots, K \tag{2}$$

for all adjacent nodes (i,j) in the graph.

This problem can be re-cast into the form of UQP by using transformation # 1 on the assignment constraints of (1) and transformation #2 on the adjacency constraints of (2). Note that no new variables are required. Since the model of (1) and (2) has no explicit objective function, any positive value for the penalty P will do. The following example gives a concrete illustration of the re-formulation process.

Example: (3-coloring)

Consider the following graph and assume we want to find a feasible coloring of the nodes using 3 colors.



Our satisfiablity problem is that of finding a solution to:

$$x_{i1} + x_{i2} + x_{i3} = 1 \qquad i = 1,\dots, 5 \qquad (3)$$

$$x_{ip} + x_{jp} \le 1 \qquad p = 1,\dots, 3 \qquad (4)$$

(for all adjacent nodes i and j)

In this traditional form, the model has 15 variables and 26 constraints. To recast this problem into the form of UQP, we use transformation #1 on the equations of (3) and transformation #2 on the inequalities of (4). Arbitrarily choosing the penalty P to be 4, we get the equivalent problem:

$$UQP(Pen): \min x\hat{Q}x$$

where the additive constant is 20 and $\hat{Q}$ is:

$$\hat{Q} = \begin{bmatrix}
-4 & 4 & 4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\
4 & -4 & 4 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 \\
4 & 4 & -4 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\
4 & 0 & 0 & -4 & 4 & 4 & 4 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 \\
0 & 4 & 0 & 4 & -4 & 4 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & 4 & 0 \\
0 & 0 & 4 & 4 & 4 & -4 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & 4 \\
0 & 0 & 0 & 4 & 0 & 0 & -4 & 4 & 4 & 4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 4 & 0 & 4 & -4 & 4 & 0 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 4 & 4 & 4 & -4 & 0 & 0 & 4 & 0 & 0 & 0 \\
0 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & -4 & 4 & 4 & 4 & 0 & 0 \\
0 & 0 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 4 & -4 & 4 & 0 & 4 & 0 \\
0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 4 & 4 & 4 & -4 & 0 & 0 & 4 \\
4 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & -4 & 4 & 4 \\
0 & 4 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 4 & -4 & 4 \\
0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 4 & 4 & -4
\end{bmatrix}$$

Solving this unconstrained model, $x\hat{Q}x$, yields the feasible coloring:

$$x_{11}, x_{22}, x_{33}, x_{41}, x_{53}, = 1 \;\; all \;\; other \;\; x_{ij} = 0 \cdot$$

3.1 Computational Experience:

To test the potential attractiveness of the UQP modeling and solution approach to K-coloring problems, 21 standard test problems from the literature were recast into the form of UQP and solved by our tabu search method. Table 1 gives a description of the problems and presents the results of our computations. All computations were carried out on a 1.7 gigahertz PC.

ID	# Vertices	# Edges	K	# xQx Variables	xQx feasible	xQx Time
Myciel3	11	20	4	44	Yes	< 1 sec
Myciel4	23	71	5	115	Yes	< 1 sec
Myciel5	47	236	6	282	Yes	< 1 sec
Myciel6	95	755	7	665	Yes	< 1 sec
Myciel7	191	2360	8	1528	Yes	< 1 sec
Anna	138	493	11	1518	Yes	47 sec
David	87	406	11	957	Yes	1 min, 13sec
Huck	74	301	11	814	Yes	2 sec
Jean	80	254	10	800	Yes	< 1 sec
Games120	120	638	9	1080	Yes	< 1 sec
Queen5_5	25	160	5	125	Yes	< 1 sec
Queen6_6	36	290	7	252	Yes	< 1 sec
Queen7_7	49	476	7	343	Yes	< 1 sec
Queen8_12	96	1368	12	1162	Yes	< 1 sec
Queen8_8	64	728	9	576	Yes	< 1 sec
Queen9_9	81	2112	10	810	Yes	< 1 sec
Queen10_10	128	3216	12	1200	Yes	3 sec
Le450_5a	450	5714	5	2250	Yes	17 min, 7sec
Le450_5b	450	5734	5	2250	Yes	10 min, 7sec
Le450_5c	450	9803	5	2250	Yes	1 min, 17 sec
Le450_5d	450	9757	5	2250	Yes	47 sec

Table 1: K-coloring test problems from http://mat.gsia.cmu.edu/COLOR/instances.html.

The first four columns of Table 1 indicate the problem identifier along with the size of the graphs and the number of colors (K) to be used. The last three columns give the number of variables involved, whether or not a feasible coloring was found utilizing K colors, and the time our tabu search method took to find a solution. Note that feasible colorings (solutions) were quickly found in all 21 cases. In fact, the solutions shown are, with the exception of the problem Queen10_10 for which an optimal solution (minimum number of colors) has not been reported in the literature, known to be optimal.

Recall from the discussion of transformation # 1 that the transformation process produces and additive constant, c, in addition to the quadratic function, xQx. Since an optimal solution would have a net objective function value of zero for the problems considered here, our solution procedure was run for a pre-determined number of span cycles (1000) or until xQx was equal to –c, whichever occurred first for each problem. In all cases, as indicated in the table, we were able to find solutions with xQx = –c before hitting the span termination limit. If, for some problem, we were to terminate with a result other than –c, it would most likely mean that the problem is infeasible. Given the heuristic nature of our solution procedure, however, this conclusion cannot be taken with certainty. In this respect, our method offers no shortcuts in establishing irrefutably that infeasibility exists.

Significantly, from a computational standpoint, the size of our model (as embodied in the Q matrix) for the K-coloring problem is determined by the number of vertices and K—not by the number of edges in the graph. This suggests that the approach may well be attractive for both sparse and dense graphs.

4. General Vertex Coloring:

The previous section illustrated how xQx could be utilized to model and solve K-coloring problems. This approach can be used to find the minimum number of colors required for a feasible coloring by examining a sequence of K-coloring problems where K starts at some large value and is decreased by 1 in each subsequent problem until it is no longer possible to find a feasible coloring.

An alternative to the foregoing *sequential* approach is to pose the problem at the outset as a more general version of the vertex coloring problem that seeks to simultaneously determine both the minimum number of colors needed and a corresponding feasible coloring. In this section we illustrate how this problem can also be modeled and solved using the UQP framework.

Let K_max be the maximum number of colors to be considered and n be the number of vertices in the graph. As before, let x_{ij} be 1 if node i is assigned color j, and 0 otherwise. In addition, let y_j be 1 if color j is used in a feasible coloring, 0 otherwise.

Then the formulation is

$$\min \sum_{j=1}^{K\_\max} y_j \qquad (5)$$

$$st$$

$$\sum_{j=1}^{K\_\max} x_{ij} = 1 \quad i = 1,\dots, n \qquad (6)$$

$$x_{ip} + x_{jp} \le 1 \;\; \textit{for each edge } (i, j) \textit{ and color } p \qquad (7)$$

$$x_{ip} \le y_p \;\; \textit{for each vertex } i \textit{ and color } p \qquad (8)$$

$$x, y \;\; \textit{binary}$$

This model can be re-cast into an equivalent UQP form as follows. The first two sets of constraints, as in the development of the previous section, can be accommodated by transformation #1 and transformation # 2 respectively. The variable upper bound constraints of (8) can be accommodated by transformation # 2 by first replacing each “y” variable by its complement. The completed transformation process yields an equivalent model of the desired form

$$\min x\hat{Q}x$$

As in the case considered in section 3, this equivalent unconstrained problem is obtained without the introduction of additional variables.

4.1 Computational Experience:

To illustrate the application of UQP to this more general form of the coloring problem, 8 of the smaller problems considered in section 3 were first reformulated as problems of the form of (5) – (8) above and then re-cast in the form of UQP. Table 2 presents the results of the computations.

ID	# nodes	# edges	K_max	# xQx variables	xQx solution	Opt Solution
Jean	80	254	12	972	10	10
David	87	406	12	1056	11	11
Huck	74	301	14	1050	11	11
Mycie13	11	20	8	96	4	4
Mycie14	23	71	10	240	5	5
Mycie15	47	236	10	480	6	6
Mycie16	95	755	10	960	7	7
Queen5_5	25	160	10	260	5	5

Table 2: Results corresponding to the general coloring formulation

Table 2 identifies the problems solved along with the maximum number of colors allowed, the size of the resulting UQP model, and the results obtained. For example, for problem "Jean", a maximum of 12 colors were allowed yielding a UQP model with 972 variables. Solving the UQP version of Jean gave a solution with 10 colors which is known to be optimal. As shown in Table 2, the UQP equivalent formulation led to optimal solutions for each of the 8 problems. The UQP equivalent models for the general formulation considered here, (5) – (8), are larger than the corresponding dimensions displayed in Table 1 due to the extra colors that are considered by the general model. This observation motivates an interest in developing effective heuristics for generating appropriate (small) values for K_max, enabling larger instances to be solved in reasonable computational time. Such efforts are part of our on-going research.

In making the transformation to xQx, a penalty value of P = 20 was used in each case. All problems were solved by our tabu search procedure with each problem allowed to run for a total of 300 Span cycles. The largest of the problems took less than 3 minutes to complete the 300 cycles.

5. Summary and Conclusion:

In this paper we demonstrated how the vertex coloring problem, in both its K_coloring and general form, can be effectively modeled and solved as an unconstrained quadratic program. Computational results with our tabu search heuristic indicate that this approach is not only effective but very competitive with special purpose methods designed for these problems. Table 1 in particular showcases how robust our approach is in terms of finding best known solutions over a diverse set of problems. Comparisons in the literature of competing methods (see for instance Di Blas, et al [14] and Mehrotra and Trick [38]) indicate that no single method dominates all others in terms of solution quality and computational time for the problems considered here. Many of the problems examined in Table 1 are part of the DIMACS Challenge problem set and are known to be difficult. Some of the best known and well established methods for graph coloring (such as DSATUR and LPCOLOR [38]) are challenged by certain instances considered here (e.g., the "queen" problems and the "myciel" problems) and were unable to find best known solutions in all cases. By contrast, our general modeling and solution approach proved able to find best known solutions in all cases and to do so in reasonable computational times. Comparisons with other methods reported in the literature also indicate that our approach compares favorably on those problems that have more commonly been tackled by other methods (see for instance Coudert [13] and Lewandowski and Condon [36]).

Some very recent additions to the literature contain papers describing new procedures for vertex coloring problems that hold considerable promise in terms of solving much larger instances than those considered here. These methods, specially crafted for coloring problems, have raised the bar in terms of what is possible and what other researchers will come to use as benchmarks. The works by Dorne and Hao [15], Galinier and Hao [16], and Chiarandini and Stutzle [12] are particularly promising in this regard. The model we present here for coloring problems could be employed as an alternative representation of such problems. Whether or not adopting such a perspective proves to be competitive from a computational point of view depends on the continued improvement in solution methodologies for the UQP model that would be required to efficiently solve UQP instances with upwards of 20,000 variables or more. We are currently working on such a solution capability and will report on its comparative performance with the latest coloring methods in future papers.

The reformulation approach described in this paper can in principle be applied to any linearly constrained quadratic and linear programs in bounded integer variables. Our experience with other classes of problems, similar to our experience reported here, is that the approach works very well and provides a unifying modeling and solution framework for general combinatorial problems. As research continues to lead to algorithmic improvements for solving UQP, this approach will increasingly become an attractive alternative to conventional modeling and solution procedures.

References

1. Alidaee, B, G. Kochenberger, and A. Ahmadian, "0-1 Quadratic Programming Approach for the Optimal Solution of Two Scheduling Problems," *International Journal of Systems Science*, 25, 401-408, 1994.

2. Alkhamis, T. M., M. Hasan, and M. A. Ahmed, "Simulated Annealing for the Unconstrained Binary Quadratic Pseudo-Boolean Function," *European Journal of Operational Research*, 108, (1998), 641-652.

3. Amini, M., B. Alidaee, G. Kochenberger, "A Scatter Search Approach to Unconstrained Quadratic Binary Programs," *New Methods in Optimization*, Corne, Dorigo, and Glover, eds., McGraw-Hill, pp. 317 – 330, 1999.

4. Beasley, J. E.,"Heuristic Algorithms for the Unconstrained Binary Quadratic Programming Problem, Working Paper, Imperial College, 1999.

5. Billionet, A. and A. Sutter, "Minimization of a Quadratic Pseudo-Boolean Function." *European Journal of OR,* 78 pp. 106-115, (1994).

6. Boros.E. and P. Hammer, "Pseudo-Boolean Optimization," *Discrete Applied Mathematics*, 123(1-3), (2002) pp. 155-225.

7. Boros, E, P. Hammer, and X, Sun, "The DDT Method for Quadratic 0-1 Minimization," RUTCOR Research Center, RRR 39-89, 1989.

8. Boros, E. and A. Prekopa, "Probabilistic Bounds and Algorithms for the Maximum Satisfiability Problem," *Annals of OR*, 21 (1989), pp. 109-126.

9. Bourjolly, J.M., P. Gill, G. Laporte, and H. Mercure, "A Quadratic 0/1 Optimization Algorithm for the Maximum Clique and Stable Set Problems," Working paper, University of Montreal, (1994).

10. Campos, V., R. Marti, and M. Laguna, "Intensification and Diversification with Elite Tabu Search Solutions for the Linear Ordering Problem," Working Paper, University of Colorado at Boulder (2000).

11. Chardaire, P, and A. Sutter, "A Decomposition Method for Quadratic Zero-One Programming," *Management Science*, 41:4, 704-712, 1994.

12. Chiarandini, M. and T. Stutzle, "An Application of Iterated Local Search to Graph Coloring," *Proceedings of the Computational Symposium on Graph Coloring and its Generalizations*, (Eds.) D. S. Johnson, A. Mehrotra, and M. Trick, pp. 112-125, 2002.

13. Coudert, O. "Exact coloring of Real Life Graphs is Easy," *Proceedings of the 34th Design Automata Conference*, NY, 1997, 121-126.

14. Di Blas, A. Jagota and R. Hughey, "Energy function-Based Approaches to graph Coloring," *IEEE Transactions on Neural Networks, 13, 81-91*, (2002).

15. Dorne, R. and J. Hao, "Tabu Search for Graph Coloring, t-Coloring and set t-Coloring," In: *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, (Eds.) S. Voss, S. Martello, I. Osman, and C. Roucairol. Kluwer Academic Publisher, Boston, 1999.

16. Galinier, P. and J. Hao, "Hybrid Evolutionary Algorithms for Graph Coloring," *Journal of Combinatorial Optimization*, 3(4), pp. 37`9-397, 1999.

17. Gallo, G, P. Hammer, and B. Simeone, "Quadratic Knapsack Problems," *Mathematical Programming*, 12, 132-149, 1980.

18. Glover, F, and M. Laguna, "Tabu Search," Kluwer Academic Publishers, 1997.

19. Glover, F., "A Template for Scatter Search and Path Relinking," School of Business, University of Colorado, Technical Report, February 1998.

20. Glover, F., G. Kochenberger, B. Alidaee, and M.M. Amini, "Tabu with Search Critical Event Memory: An Enhanced Application for Binary Quadratic Programs," In: *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, (Eds.) S. Voss, S. Martello, I. Osman, and C. Roucairol. Kluwer Academic Publisher, Boston, 1999.

21. Glover, F., B. Alidaee, C. Rego, and G. Kochenberger, "One-Pass Heuristics for Large-Scale Unconstrained Binary Quadratic Programs," *EJOR 137,pp. 272-287*, 2002.

22. Glover, F., G. Kochenberger., and B. Alidaee, "Adaptive Memory Tabu Search for Binary Quadratic Programs," *Management Science*, 44:3, 336-345, 1998.

23. Grotschel, M., M. Junger, and G. Reinelt, "An Application of Combinatorial Optimization to Statistical Physics and Circuit Layout Design," *Operations Research*, Vol.36, #3, May-June, (1988), pp. 493-513.

24. Hammer, P, E. Boros, and X. Sun, "On Quadratic Unconstrained Binary Optimization," *INFORMS National Meeting,* Seattle, October, 1998.

25. Hammer, P., P. Hansen, and B. Simone, "Upper Planes of Quadratic 0/1 Functions and Stability in Graphs," Nonlinear Programming 4, O. Mangasarian, R. Meyer, and S. Robinson (eds), Academic Press, New York, (1981), pp. 395-414.

26. Hammer, P., and S. Rudeanu, *Boolean Methods in Operations Research,* Springer-Verlag, New York, 1968.

27. Hansen, P.B., "Methods of Nonlinear 0-1 Programming," *Annals Discrete Math*, vol. 5, pp.53-70, 1979.

28. Hansen, P, B. Jaumard., and V. Mathon, "Constrained Nonlinear 0-1 Programming," *INFORMS Journal on Computing*, 5:2, 97-119, 1993.

29. Helmberg, C. and F. Rendl, "Solving Quadratic (0,1)-Problems by Semidefinite Programs and Cutting Planes," *Mathematical Programming*, 82, pp. 291-315, (1998).

30. Harary, F, "On the Notion of balanced of a signed Graph," *Michigan Mathematical Journal*, 2, 143-146, 1953/54.

31. Iasemidis, L. D., D. S. Shiau, J.C. Sackellares, and P. Pardalos, "Transition to Epileptic Seizures: Optimization," *DIMACS Series in Discrete Math and Theoretical Computer Science,* Vol. 55, (2000), pp. 55-73.

32. Katayama, K., M. Tani, and H. Narihisa, "Solving Large Binary Quadratic Programming Problems by an Effective Genetic Local Search Algorithm," In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'00).* Morgan Kaufmann, 2000. (to appear).

33. Kochenberger, G., F. Glover, B. Alidaee, and C. Rego, "Applications of the Unconstrained Quadratic Binary Program," University of Colorado Working Paper, 2002.

34. Krarup, J, and A. Pruzan, “Computer Aided Layout Design,” *Mathematical Programming Study*, 9, 75-94, 1978.

35. Laughunn, D.J, “Quadratic Binary programming,” *Operations Research*, 14, 454-461, 1970.

36. Lewandowski, G. and A. Condon, “Experiments with Parallel Graph Coloring Heuristic and Applications of Graph Coloring,” *DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Providence, RI,* American Mathematical Society, Vol.26, 335-358, (1996).

37. Lodi, A., K. Allemand, and T. M. Liebling, “An Evolutionary Heuristic for Quadratic 0-1 Programming,” Technical Report OR-97-12, D.E.I.S., University of Bologna, 1997.

38. Mehrotra, A. and M. Trick, ‘A Column Generation Approach for Graph Coloring,” Working Paper, Carnegie Mellon University, (1995).

39. McBride, R.D., and J.S. Yormack, “An Implicit Enumeration Algorithm for Quadratic Integer Programming,” *Management Science*, 26, 282-296, 1980.

40. Merz, P. and B. Freisleben, “Genetic Algorithms for Binary Quadratic Programming,” *Proceedings of the 1999 International Genetic and Evolutionary Computation Conference (GECCO’99),* pp. 417-424, Morgan Kaufmann, 1999.

41. Merz, P. and B. Freisleben, “Greedy and Local Search Heuristics for the Unconstrained Binary Quadratic Programming Problem,” *Journal of Heuristics*, Vol. 8, no. 2, pp.197-213, 2002.

42. Merz, P. and K. Katayama, “A Hybrid Evolutionary Local Search Approach for the Unconstrained Binary Quadratic Program,” *Bio Systems* (in press), 2003.

43. Palubeckis, G. “A Heuristic-Branch and Bound Algorithm for the Unconstrained Quadratic Zero-One Programming Problem,” *Computing,* pp. 284-301, (1995).

44. Pardalos, P, and G.P. Rodgers, “Computational Aspects of a Branch and Bound Algorithm for Quadratic Zero-one Programming,” *Computing*, 45, 131-144, 1990.

45. Pardalos, P, and G.P. Rodgers, “A Branch and Bound Algorithm for Maximum Clique problem,” *Computers & OR*, 19, 363-375, 1992.

46. Pardalos, P, and J. Xue, “The Maximum Clique Problem,” *The Journal of Global Optimization,* 4, 301-328, 1994.

47. Phillips, A.T., and J.B. Rosen, “A Quadratic Assignment Formulation of the Molecular Conformation Problem,” *The Journal of Global Optimization*, 4, 229-241, 1994.

48. Rosenberg, I., “0/1 Optimization and Non-Linear Programming,” *Revue Francaise d’Automatique, d’Informatique et de Recherche Operationnelle,* (S\’erie Bleue) 2, (1972), pp. 95-97.

49. Williams, A. C., “Quadratic 0-1 Programming Using the Roof Duality with Computational Results,” Rutcor Research Report 8-85, Rutgers University, New Brunswick, NJ., 1985.

50. Witsgall, C., “Mathematical Methods of site Selection for Electronic System (EMS),” NBS Internal Report, 1975.