RESEARCH PAPER
International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009

Solving Facility Layout Problem: Three-level Tabu Search Metaheuristic Approach

Surya Prakash Singh
Xavier Institute of Management Bhubaneswar, Bhubaneswar, India
Email: surya.singh@gmail.com

Abstract— **In this paper an improved tabu search (ITS) based approach is proposed for solving facility layout problem (FLP) which is formulated as quadratic assignment problem (QAP). ITS is an improved version of conventional tabu search technique which incorporates three levels viz. intensification, reconstruction, and solution acceptance. To evaluate the efficacy of the proposed ITS, it is tested for benchmark instances taken from published literature. Also, a comparative analysis of ITS with other meta-heuristic approach is presented. It is found that ITS based approach provide comparative results.**

I. Introduction

Combinatorial optimization problems arising from both practice and theory pose a real challenge and continuously draw an attention of the researchers, practitioners and academicians around the world over last five decades. Many optimization problems belong to the class NP-hard and cannot be solved to optimality within polynomial time. One such combinatorial problem is Facility layout problem (FLP). The FLP is a well researched problem and the determination of facilities in locations is a common problem encountered in manufacturing industry. Ref. [1] was the first to model FLP as Quadratic assignment problem (QAP). The QAP has been shown to be NP-hard as in [2, 3, 4]. Nowadays, the results achieved by applying the best existing exact algorithms (Branch and Bound) are modest: generally problem of size larger than 20 cannot be solved to optimally in reasonable time [5, 6]. Thus, the interest lies in the application of *heuristic* and *meta-heuristic* methods to solve large QAP instances. The heuristic algorithms seek for near-optimal solutions at reasonable computational time but can not guarantee an optimal solution. The heuristic approaches have an essential advantage over exact algorithms: the heuristics usually find high quality solutions much faster than the exact algorithms; this is especially true by solving the large-scale problems. Similarly, meta-heuristic approaches such as simulated annealing (SA), genetic algorithm (GA), TS and ant colony algorithm (ACO) have been also developed and widely applied to solve combinatorial hard optimization problem and FLP specially. Ref. [7, 8, 9, 10, 11, 12] can be referred for meta-heuristic application to solve FLP. A good survey on various approaches to solve FLP can be found from [13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31].

In this paper, the issues related, namely, to the improved strategy for solving the FLP which is a difficult optimization problem is discussed. The focus is on the random search technique i.e. the well-known Tabu Search (TS) method. TS has been proven to be among the most powerful tools for solving various optimization problems. Still, the design of more effective TS for the specific problems is the field of research direction for many practitioners/ academicians. One of the possible extensions over the standard TS paradigm is a so-called ITS is proposed in this paper. ITS method is tested on the well-known NP-hard problem i.e. FLP modelled as QAP. However, after adding the corresponding modifications ITS may be easily applied to other computationally hard problems. Hence, the paper proposes as improved tabu search based approach for solving FLP which is formulated as QAP.

Structure of the paper is organized as follows: the problem is formulated in section II. The conventional TS technique is outlined in section III while the section IV described the proposed ITS based approach for solving FLP. Computational experiments are discussed in section V, and lastly conclusion is given in section VI.

II. Problem Formulation

This section describes the formulation of QAP. Let us assign '*n*' facilities to '*n*' locations with the cost being proportional to the flow between the facilities multiplied with their distances. The objective is to allocate each facility at a location such that total cost is minimized as given in (1). Thus we are given two matrices, the flow matrix f_{ik} and distance matrix d_{jl}. FLP with flow and distance matrix is denoted by QAP introduced by [32] where four-dimensional cost array c_{ijkl} was considered instead of two matrices. In this paper we consider QAP. Equation (2) ensures that only one facility can be located at one location and (3) ensures that each location can only be assigned to one facility. Equation (4) shows that the variables are binary. The QAP in [1] is written as

Minimize $$f = \sum_{i,j,k,l=1}^{n} f_{ik} \cdot d_{jl} . X_{ij} \cdot X_{kl} \,. \qquad (1)$$

Subject to $$\sum_{j=1}^{n} X_{ij} = 1 \quad \forall\, i = 1,...,n \,. \qquad (2)$$

$$\sum_{i=1}^{n} X_{ij} = 1 \quad \forall j = 1,...,n \,. \qquad (3)$$

$$X_{ij} = (0,1) \quad \forall i, j \;. \qquad (4)$$

We introduce few basic definition related to FLP and is defined by a pair (*S*, *f*), where $S = \{s_1, s_2, ...\}$ is a finite set of feasible solutions (a "solution space") and *f* is an objective function where *f* seeks an optimal solution. For the sake of clarity, let us consider the case where the solutions are permutations of the integers from 1 to *n*, $s_j=\{i, i+1, \ldots, n\}$ where $j=1,\ldots,n!$. For $n=5$, one possible set of feasible solution could be $s_1 = \{3, 4, 2, 5, 1\}$ out of total possible sets s_j, for all *j* which varies from 1 to *n*!.

III. Conventional Tabu Search

The TS originates from the local search technique [33, 34]. However, the TS go beyond the basic structure of the local search approach and enable to escape local optima. TS-based algorithms continue the search even if a locally optimal solution is encountered. Therefore, TS is a process of chains of moves from one local optimum to another. The best local optimum found during this process is regarded as a result of TS. Thus, TS is an extended local search approach. Consequently, it explores much larger part of the solution space when comparing with local search. Hence, TS provides more room for discovering high quality solutions than the local search. The key idea of TS is allowing climbing moves when no improving neighbouring solution exists, i.e. a move is allowed even if a new solution *s′* from the neighbourhood of the current solution *s* is worse than the current one. Naturally, the return to the locally optimal solutions previously visited is to be forbidden to avoid cycling. TS is based on the methodology of prohibitions: some moves are "frozen" (become "tabu") from time to time.

Function: tabu search

// *input*: s° – the initial solution; *output*: s^* – the best solution found; parameter:

h – the tabu list size //

$s \leftarrow s^\circ$; $s^* \leftarrow s^\circ$;

initialize the tabu list *T*;

repeat /continue the main cycle of TS //

given neighbourhood function *N*, tabu list *T*, and aspiration criterion, find the best possible solution

$s' \in N(s)$,

Where $N(s)$ consists of the solutions that are not in the tabu list *T* or satisfy the aspiration criterion;

s ← **s**′; // replace the current solution by new one //

If $f(s) < f(s^*)$ **then** $s^* \leftarrow s$; // save best so far solution //

insert the solution *s* into the tabu list *T*;

if size of (*T*) > *h* **then** remove the "oldest" member of *T*

until termination criterion is satisfied;

return s^*

end.

Figure 1. Basic structure of conventional tabu search.

More formally, the TS algorithm starts from an initial solution s° in *S*. The process is then continued in an iterative way – moving from a solution *s* to a neighbouring one *s′*. At each step of the procedure, a subset of the neighbouring solutions of the current solution is considered, and the move to the solution that improves the objective function value *f* is chosen. Naturally, *s′* must not necessary be better than *s*: if there are no improving moves, the algorithm chooses the one that least increases the objective function (a move is performed to the neighbour *s′* even if $f(s') > f(s)$). In order to eliminate an immediate returning to the solution just visited, the reverse move must be forbidden. This is done by storing the corresponding solution (move) (or its "attribute") in a memory (called a tabu list (*T*)). The tabu list keeps information on the last $h = |T|$ moves which have been done during the search process. Thus, a move from *s* to *s′* is considered as tabu if *s′* (or its "attribute") is contained in *T*. This way of proceeding hinders the algorithm from going back to a solution reached within the last *h* steps. The pseudo-code for the standard (pure) tabu search paradigm is presented in Fig. 1. More details on the fundamentals and principles of TS can be found in [34].

IV. Improved Tabu Search

TS is a powerful random search optimization tool for NP-hard problem and specially for solving FLP. However, it faces some difficulties which are (1) a huge number of local optima over the solution space, (2) presence of cycles (i.e. repeating sequences) of the search configurations (states), and (3) the phenomenon of so-called "deterministic chaos" (or chaotic attractors). The last one can be characterized by the situation in which "getting stuck" in local optima and cycles are absent but the search is still confined in some "narrow region" of the solution space [35]. So, the search process will visit only a limited part of the solution space: if this portion does not contain the global minimum, it will never be found i.e. a stagnation of the search is said to take place. In order to try to overcome the difficulties mentioned, an essential extension of the standard TS i.e. ITS is proposed.

The main theme of ITS is the concept of intensification and reconstruction (*I* and *R*). The early origin of this concept goes two decades back and since then, various modifications of the basic idea have been proposed, among them are: iterated Lin-Kernighan algorithm [36], combined with the local search [37], TS based on "ruin and recreate" principle [38], and combined with an iterated local search [39, 40]. In general, the working methodology of “*I* and *R*” framework is based mainly on three levels: *intensification*, *reconstruction*, and *accept solution*. A flow chart to present these three factors is given in Fig 2.

A. Level 1: Intensification

This step improves the current solution and is applied to the solution just reconstructed (i.e. the "output" of reconstruction), except the first iteration only at which intensification is applied to the initial solution (refer to Fig. 2). It was revealed by experimentation that there is no need in the expensive runs of the tabu search based improvement procedure. Firstly, the simple tabu search iterations allow saving considerable amount of CPU time. On the other hand, this conventional tabu search in combination with the diversification operators is capable of seeking more near optimal solutions which are better than those obtained by long runs of simple tabu search.

B. Level 2: Reconstruction

It is a kind of reconstruction of the solutions which is responsible for escaping from the current local optimum and moving towards new regions in the solution space. It is important to keep a level of diversification: if it is too high, the resulting algorithm might be quite similar to a pure random multi start and if too low then the process may return to the same solutions on which reconstruction has been applied. It is similar to mutation operation of GA.

Many different reconstruction variants can be devised. For example, one can use the random pair wise interchanges of the i^{th} and $i+1^{th}$ elements in the current solution. The larger the length of the reconstruction level, the stronger will be the effect. In the proposed reconstruction procedure, K% of total ${}^{n}C_{2}$ (total number of pairs that can be exchanged) pair of facilities of the layout solution are exchanged with locations. If K is taken 100%, then the proposed diversification operator exchanges all possible ${}^{n}C_{2}$ pairs of facilities. In this paper ITS is applied for three different values of K i.e. 40%, 60%, and 80%. For n=5 facility problem, 4-pairs (${}^{5}C_{2}$ is equal to 10. So, 40% of 10 will be 4 pair) of facilities interchange there locations on considering K as 40%. In the paper ITS is tested for three different values of K and the best solution out of all is reported in the section V.

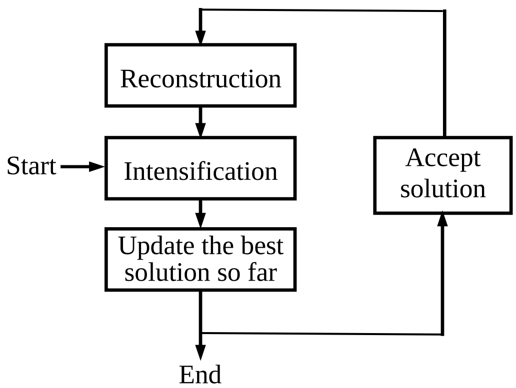

Figure 2. General framework of three-level ITS approach.

C. Level 3: Accept Solution

In the process of selection of candidate solution for the diversification operator, two main alternatives exist: a) *exploitation* and b) *exploration*. Exploitation is achieved by choosing only the currently best local optimum i.e. the best so far solution while in exploration, a variety of policies can be used. For exploration, each locally optimized solution can be considered as a potential candidate for diversification. Also, generation of a new solution from scratch is possible as an extreme case. A so-called "where you are" strategy is worth mentioning: in this case, every new local optimum irrespective of solution quality is accepted for the reconstruction process. In the proposed ITS based heuristic approach exploitation strategy is used to for candidate selection under diversification operator.

The typical flow of the ITS is as follows. It is initiated by an improvement of an initial solution by means of the conventional TS. As a result, the first optimized solution is achieved. Further, a given solution undergoes reconstruction, and a new solution, say $s^{\sim}$, is obtained. The goal of reconstruction is not to destroy the current solution absolutely. On the contrary, it is desirable that the resulting solution inherits some properties from previous local optimum, since parts of this optimum may be close to the ones of the globally optimal solution. The solution $s^{\sim}$ serves as an input for the subsequent TS procedure, which starts immediately after the perturbation process, is finished. The ITS returns new optimized solution $s^{\bullet}$, which, (or some other local optimum) in turn, is reconstructed, and so on. A new better solution (s^{*}) found during the iterative process is saved in a memory (as potential resulting solution of ITS). This continues until a stopping criterion is met. The ITS algorithm is shown in Fig. 3.

Function improved tabu search

//input:s°–initial solution; output: s^{*}–best solution //
$s^{\bullet} \leftarrow$ Tabu Search(s°); // improve the initial solution s° by TS, get the resulting solution $s^{\bullet}$ //
$s \leftarrow s^{\bullet}$; $s^{*} \leftarrow s^{\bullet}$;

repeat //continue the cycle of the iterated tabu search //
$s \leftarrow$ Acceptance of solution ($s^{\bullet}$,s, ...); // select solution for reconstruction //
$s^{\sim} \leftarrow$ Reconstruction(s); //reconstruct the selected solution, obtain a new solution $s^{\sim}$ //
$s^{\bullet} \leftarrow$ Tabu Search($s^{\sim}$);//improve the solution $s^{\sim}$ by TS, get the resulting solution $s^{\bullet}$ //
if $f(s^{\bullet}) < f(s^{*})$ **then** $s^{*} \leftarrow s^{\bullet}$ // save the best so far solution (as a possible result of ITS) //
until termination criterion is satisfied ;
return s^{*}

end.

Figure 3. Architecture of improved tab search.

V. COMPUTATIONAL EXPERIMENTS

The proposed ITS is tested on instances available in [41]. Code of robust tabu search (RoTS) for QAP is available at www.seas.upenn.edu/qaplib/codes.html which is modified as per the requirement needed in the paper. Table 1 and table 2 presents the results obtained from ITS and compared with the best known solution (BKS) available in literature. Table 1 presents solutions of instances Chrxxx, Rouxx, and Scrxx while table 2 presents the result of Escxxx, Hadxx, and Tailxxx. ITS based approach is also compared with other tabu search based algorithms such as R_OTS algorithm [42] and reactive tabu search (R_eTS) algorithm [36] for selected instances of type Taixxx. The comparative results of ITS for Taixxx are also shown in tables 3 and table 4 (in table 3, the results for the randomly generated instances are given, whereas in table 4, the results for the real-life like instances are presented–instances of this type are generated to resemble a distribution of real problems). For the random instances, the results are inferior than real-life like instances. This is an indication that the random instances are much more difficult to solve and still remain a real challenge for researchers and practitioners. Regarding the real-life like instances, they are relatively easy for many heuristics, the ITS algorithm too. The results of ITS may be improved even more by accurate tuning of the control parameters such as deciding appropriate value of K in reconstruction and by changing the total number of iterations.

TABLE I.

SOLUTION OF CHRXXX, HADXX, ROUXX AND SCRXX SERIES.

	BKS	ITS	% Dev.
Chr12a	9552	9552	0
Chr12b	9792	9792	0
Chr12c	11156	11156	0
Chr15a	9896	9896	0
Chr15b	7990	7990	0
Chr15c	9504	9504	0
Chr18a	11098	11098	0
Chr18b	1534	1534	0
Chr20a	2192	2192	0
Chr20b	2298	2298	0
Chr22a	14142	14142	0
Chr22b	6156	6156	0
Chr25a	6194	6194	0
Rou12	235528	235528	0
Rou15	354210	354210	0
Rou20	725522	726020	0.06
Scr12	31410	31410	0
Scr15	51140	51140	0
Scr20	110030	110030	0

TABLE II.

SOLUTION OF ESCXXX, HADXXX AND TAIXXX

	OS	ITS	Dev.
Esc16a	68	68	0
Esc16b	292	292	0
Esc16c	160	160	0
Esc16d	16	16	0
Esc16e	28	28	0
Esc16f	0	0	0
Esc16g	26	26	0
Esc16h	996	996	0
Esc16i	14	14	0
Esc16j	8	8	0
Had12	1652	1652	0
Had14	2724	2724	0
Had16	3720	3720	0
Had18	6922	6922	0
Had20	5358	5358	0
Tai12a	224416	224416	0
Tai12b	39464925	39464925	0
Tai15a	388241	388241	0
Tai15b	51765268	51765268	0.47
Tai17a	491812	491812	0

TABLE III.

SOLUTION OF TAIXXa

	BKV	RoTS	ReTS	ITS
Tai20	703482	703482	703482	703482
Tai25	1167256	1167256	1171324	1167256
Tai30	1818146	1818146	1818248	1818146
Tai35	2422002	2422002	2422640	2422002
Tai40	3139370	3139484	3139370	3139370
Tai50	4941410	4941480	4941520	4941410
Tai60	7205962	7208572	7208634	7205962
Tai80	13546960	13557864	13558086	13547068

TABLE IV.

SOLUTION OF TAIXXb (REAL-LIFE LIKE INSTANCES).

	BKS	R_oTS	R_eTS	ITS
tai20	122455319	122455319	122456213	122455319
tai25	344355646	344355646	344356316	344355646
tai30	637117113	637117113	637118125	637117113
tai35	283315445	283315445	283316315	283315445
tai40	637250948	637250948	637251254	637250948
tai50	458821517	458821517	458822595	458821517
tai60	608215054	608215054	608216234	608215054
tai80	818415043	818415064	818416357	818415043

VI. CONCLUSIONS

In this paper, a TS based approach is proposed to solve FLP which is named as ITS. The novelty of ITS approach is the incorporation of the special kind of solution reconstruction into the classical TS paradigm. ITS can be seen as a "reconstruct and improve" principle based optimization policy. It is distinguished, in principle, for two main components: reconstruction and intensification. During the first step, an existing solution is reconstructed in a proper way. In the second step, the local improvement based on the traditional TS procedure is applied to the solution just "ruined"; hopefully, the new improved solution is better than the solutions obtained in the previous iterations. By repeating these phases many times, one tries to seek for near-optimal feasible solutions and this is the main part of the ITS. On the basis of the proposed framework, two variants of ITS were designed, which were applied to the QAP. The results obtained from the experiments demonstrate promising performance of the proposed approach and also ITS appears to be superior to the conventional TS algorithms, as well as other meta-heuristic algorithms. As the results obtained for the QAP which show promising efficiency of the ITS, it may be worthy applying the proposed versions of the ITS method to other well-known combinatorial optimization problems which are difficult to solve.

REFERENCES

[1] T.C. Koopmans, and M. Beckman, “Assignment problems and the location of economic activities”, *Econometrica*, Vol. 25, pp. 53-76, 1957.
[2] S. Sahni, and T. Gonzalez, “P-complete approximation problems”, *J. of ACM*, Vol. 23, pp. 555–565, 1976.
[3] M.R. Garey, and D.S. Johnson, “Computers and Intractability: A guide to the theory of NP-Completeness”, Freeman, San Francisco, USA, 1979.
[4] R.E. Burkard, and K.H. Stratmann, “Numerical investigations on quadratic assignment problems”, *Naval Res. Log. Quar.*, Vol. 25, pp. 129-148, 1978.
[5] Y. Li, P.M. Pardalos, K.G. Ramakrishnan, and M. Resende, “Lower bounds for the quadratic assignment problem”, *Annals Oper. Res.*, Vol. 50, pp. 387-410, 1994.
[6] A. Diponegoro, and B.R. Sarker, “Machine assignment in a nonlinear multi-product flowline”, *J. Oper. Res. Soc.*, Vol. 54 No 5, pp. 472-489, 2003.
[7] R.E. Burkard, and F. Rendl, “A thermodynamically motivated simulation procedure for combinatorial optimization problems”, *Eur. J. Oper. Res.*, Vol. 17, pp. 169-174, 1984.
[8] M.R. Wilhelm, and T.L. Ward, “Solving quadratic assignment problems by simulated annealing”, *IIE Trans.*, Vol. 19, pp. 107-119, 1987.
[9] D.T. Connolly, “An improved annealing scheme for the QAP”, *Eur. J. Oper. Res.*, Vol. 46, pp. 93-100, 1990.
[10] C. Fleurent, and J.A. Ferland, “Genetic hybrids for the quadratic assignment problem”, in Pardalos P.M., Wolkowicz H. (Ed), Quadratic Assignment and Related Problems. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, AMS, Providence, Vol 16, pp. 173–188, 1994.
[11] A. Misevicius, “A tabu search algorithm for the quadratic assignment problem”, *Comput. Opt. Appl.*, Vol. 30, pp. 95–111, 2005.
[12] S.P. Singh, and R.R.K. Sharma, “Two level Simulated Annealing based approach to Solve Facility Layout Problem”, *Int. J. Prod. Res.*, Vol. 46, No. 13, pp. 3563-3582, 2008.
[13] J.M. Moore, “Computer aided facilities design: An International Survey”, *Int. J. Prod. Res.*, Vol 12 No 1, pp. 21-44, 1974.
[14] R. Battiti, and G. Tecchiolli, “Simulated Annealing and Tabu Search in the Long Run: A Comparison on QAP Tasks”, *Computers and Mathematics with applications*, Vol. 28, No. 6, 1-8, 1994.
[15] R.L. Francis, and J.A. White, “*Facility layout and locations: An analytical approach*”, Prentice hall, Englewood Cliffs.
[16] C. O'Brien, and S.E.Z. Abdel Barr, “An interactive approach to computer aided facility layout”, *Int. J. Prod. Res.*, Vol. 18, No. 2, pp. 201-211, 1980.
[17] J.A. Tompkins, and J.A. White, “*Facilities planning*”, Wiley, New York, 1994.
[18] A. Kusiak, and S.S. Heragu, “The facility layout problem”, *Eur. J. Oper. Res.*, Vol. 29, pp. 229-251, 1987.
[19] S.S. Heragu, “Modeling the machine layout problem”, Proceedings of the 12th Computers and industrial engineering Conference, March 12-14, 1988.
[20] S.S. Heragu, “Recent models and techniques for solving the layout problem”, *Eur. J. Oper. Res.*, Vol. 57, pp. 136-144, 1992.
[21] S.S. Heragu, “*Facilities Design*”, PWS Publishing Co., Boston, MA, 1997.
[22] M.J. Rosenblatt, and H.L. Lee, “A Robustness Approach to Facilities Design”, *Int. J. Prod. Res.*, Vol. 25, pp. 479-486, 1987.
[23] S.S. Heragu, and A. Kusiak, “Machine layout problem in flexible manufacturing systems”, *Oper. Res.*, Vol. 36, No. 2, pp. 258-268, 1988.
[24] S.S. Heragu, and A. Kusiak, “Machine Layout: An optimization and knowledge based approach”, *Int. J. Prod. Res.*, Vol. 28, No. 4, pp. 615-635, 1989.
[25] R.D. Meller, and K.Y. Gau, “The facility layout problem: recent and emerging trends and perspectives”, *J. Manuf. Sys.*, Vol. 15, pp. 351-366, 1996a.
[26] R.D. Meller, and K.Y. Gau, “Facility layout objective functions and robust layouts”, *Int. J. Prod. Res.*, Vol. 34, No. 10, pp. 2727-2742, 1996b.
[27] C.J. Malmborg, “Empirical approximation of volume distance distributions for line layout problems”, *Int. J. Prod. Res.*, Vol. 37, No. 2, pp. 375-392, 1999.
[28] J.S. Kochhar, and S.S. Heragu, “Facility Layout Design in a Changing Environment”, *Int. J. Prod. Res.*, Vol. 37, pp. 2429-2446, 1999.
[29] S.P. Singh, and R.R.K. Sharma, “A survey on various approaches to solve facility layout problem”, *Int. J. Adv. Manuf. Tech.*, Vol. 30, pp. 425-433, 2006.
[30] S.P. Singh, and R.R.K. Sharma, “Solving single and Multi-period facility layout problem: few SA and GA based heuristics”, Ph.D. thesis, Indian Institute of Technology Kanpur, INDIA, 2007, unpublished.
[31] E.M. Loiola, N.M.M. Abreu, P.O. Boarenture-Netto, P. Hahn, and T. Querido, “An analytical survey for the Quadratic assignment problem”, *Eur. J. Oper. Res.*, Vol. 176, pp. 657-690, 2007.
[32] E.L. Lawler, “The Quadratic Assignment Problem”, *Manag. Sci.*, Vol. 9, No. 4, 586-599, 1963.
[33] F. Glover, “Tabu search: part I”, *ORSA J. Comput.*, Vol. 1, pp. 190–206, 1989.
[34] F. Glover, “Tabu search: part II”, *ORSA J. Comput.*, Vol. 2, pp. 4–32, 1990.
[35] E. Glover, and M. Laguna, “*Tabu Search*”, Kluwer, Dordrecht, 1997.
[36] R. Battiti, and G. Tecchioli, “The reactive tabu search”, *ORSA J. Comput.*, Vol. 6, pp. 126–140, 1994.
[37] D.S. Johnson, “Local optimization and the travelling salesman problem”, *Lec. Notes Comp. Sci.*, Vol. 443, Springer, Berlin, pp. 446–461, 1990.
[38] O. Martin, and S.W. Otto, “Combining simulated annealing with local search heuristics”, *Annals Oper. Res.*, Vol. 63, pp. 57–75, 1996.
[39] G. Schrimpf, J. Schneider, H. Stamm-Wilbrandt, and G. Dueck, “Record breaking optimization results using the ruin and recreate principle”, *J. Comput. Phy.*, Vol. 159, pp. 139–171, 2000.
[40] H.R. Lourenco, O.R. Martin, and T. Stützle, “*Iterated local search*”, in F. Glover, and G. Kochenberger, (Ed.), Handbook of Metaheuristics, Kluwer, Norwell, pp. 321–353, 2002.
[41] R.E. Burkard, S. Karisch, and F. Rendl, 1997. “QAPLIB–a quadratic assignment problem library”, *J. Global Opt.*, Vol.10, pp. 391–403, 1997.
[42] E. Taillard, “Robust taboo search for the QAP”, *Parallel Computing*, Vol. 17, pp. 443–455, 1991.