International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

Genetic – Simulated Annealing Algorithm for Robust Layout in Dynamic Facility Layout Problem

Ganesh M. R.*[1], Dileeplal J.[2], Soni Kuriakose[3]
[1]PG Scholar, [2, 3]Associate Professor
Department of Mechanical Engineering, Mar Athanasius College of Engineering, Kothamangalam, Kerala, India.

***Abstract*— A facility layout design (FLD) problem can be generally introduced as assignment of facilities (departments) to a site such that a set of criterias is satisfied. The robust layout approach for dynamic facility layout problem (DFLP), assumes that rearrangement and production interruption costs are too high and hence, tries to minimize the total material handling costs in all periods using a single layout. In this paper, a genetic – simulated annealing algorithm (GSAA) is suggested for the optimization of robust facility layout in dynamic demand environment.**

***Keywords*— Robust design, facility layout, quadratic assignment problem, genetic algorithm, simulated annealing.**

I. Introduction

The most important duty of managers, engineers and planners after selecting factory location but before implementing operational schemes, is suitable arrangement of equipments or departments by noting statement, goals and strategies by evaluating the most important criteria influencing the different facility layout problems. A facility layout is concerned with the location and arrangement of departments, cells or machines within the cells. Manufacturing companies spend a significant amount of time and money in FLD since the design of a facility layout has a tremendous effect on the operation of the system [1]. As stated by Tompkins et al. [2], the facility planning may include 10–30% of operational cost due to changes. Not only an inappropriate FLD causes rearrangement of existing facilities or/and material handling system, it will also undertake its resulting heavy costs. Therefore, the best work at designing process is to select an optimal FLP under different criteria or objectives, in order to obtain the maximum productivity and profitability.

When the demand is more or less constant with time, static plant layout problem (SPLP) approach is a suitable method for obtaining a good facility layout. But when demand is varying frequently with time, static layout generation approaches may not be efficient in various periods of the planning horizon. Fluctuations in product demand, changes in product mix, introduction of new products, and discontinuation of existing products are all factors that render the current facility layout inefficient and can increase MHC, which might necessitate a change in the layout. Hence dynamic plant layout problem (DPLP) approach is most suitable for the development of layouts [3].

The approaches that have been followed to solve the dynamic facility layout fall into two major categories: adaptive or flexible or agile approach and robust approach. The first approach assumes that layout will accommodate changes from time to time with low rearrangement costs and that the machines can be easily relocated. On the other hand, a robust layout approach assumes that rearrangement costs are too high and hence tries to minimize the total material handling costs in all periods using a single layout. Robust layout approach is one of the methods used for developing layouts for multiple production scenarios of a single period problem and for multi-period problems [4].

The facility layout problem is often formulated as a quadratic assignment problem (QAP), which assigns m departments to n locations while minimizing the MHC. However, QAP is known to be NP-complete, and optimization methods are not capable of solving problems with 15 or more facilities in a reasonable amount of time [5]. Therefore, there is a need for heuristic methods that provide good suboptimal solutions. This paper proposes a genetic - simulated annealing algorithm as a solution methodology for the robust design for DPLP.

II. Literature Review

In recent years many researchers are making efforts to address the DPLP. Various researchers proposed new and improved models and algorithms to solve DPLP. Rosenblatt [3] first developed a model and solution procedure to DPLP with adaptive approach for small size problems. Various reviews of research on the dynamic layout problem are available in [6]-[8]. These papers categorized different algorithms for equal and unequal sized departments, and deterministic and stochastic material flow. Many researchers ([3], [9]-[11]) have modeled the adaptive or flexible or agile layouts that can be easily rearranged to meet the changes in production requirements. Researchers ([3], [9]- [17]) used exact and heuristic methods to solve the DPLPs. Researchers ([9], [12]-[17]) made use of meta-heuristics like simulated annealing, genetic algorithm and ant colony optimization techniques to DPLPs. Recently the hybrid approaches are also attempted in [12] and [16]. Some researchers ([18], [19]) have developed robust layouts for multiple production scenarios in a single period and for multi period. Kouvelis [20] mentioned the importance of robustness for dynamic layout problems and developed an algorithm to generate the robust layouts for the manufacturing systems. Pillai [21] presented a robust approach for forming part families and machine cells, which can handle all the changes in demands and product mixes without any relocations. A genetic algorithm based solution procedure is adopted to solve the problem. Pillai [4] designed a robust model for DPLP and used simulated annealing for solving the robust layout.

III. PROBLEM DESCRIPTION

The robust approach to dynamic layout problem involves development of a layout for the expected flow between facilities or expected demand scenario of the various periods. This layout is applied in all the periods. Thus, the entire planning horizon uses a single layout even though the demand or flow between facilities is different in different periods of the planning horizon.

Quadratic assignment model of robust approach is developed as in Pillai [4] and the following equations represent this model. In this model, a layout is developed for an average scenario and this layout is used in every period without relocation of facilities in any period of planning horizon. In this model the computational effort required to solve the dynamic layout problem is same as that of the static layout problem. The total MHC of the planning horizon is determined by applying the layout of the expected scenario to every period of the planning horizon.

Minimize $Z = \sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{r=1}^{n}\sum_{s=1}^{n} WE_{jk}R_{rs}x_{jr}x_{ks}$

Subjected to

$\sum_{j=1}^{n} x_{jr} = 1 \quad k=1,\ldots,n$

$\sum_{k=1}^{n} x_{jr} = 1 \quad k=1,\ldots,n$

$x_{jk} = (0,1)$ for all j and k

where WE_{jk} average part flow weight from j to k, and R_{jk} is the rectilinear distance between j and k

For P number of periods, and D_{pi} demand at each period, average demand DE_i is

$DE_i = \sum_{i=1}^{P} D_{pi}/P$

$WE_{jk} = \sum_{i=1}^{N} DE_i / B_{ijk}$ where B_{ijk} is number of parts i per transportation when transported from facility j to facility k

Total material handling cost (TMHC) is calculated as follows

$f_{pjk} = \sum_{i=1}^{N} D_{pi} / B_{ijk}$

$MHC_P = \omega * \sum_{j=1}^{n}\sum_{k=1}^{n} R_{jk}f_{pjk}$

$TMHC = \sum_{p=1}^{P} MHC_p$

$R_{jk} = |X_j - X_k| + |Y_j - Y_k|$

where MHC_p is material handling cost in each period and ω is the cost per unit part movement.

IV. PROPOSED GENETIC – SIMULATED ANNEALING ALGORITHM (GSAA)

Genetic algorithms are adaptive methods, which may be used to solve search and optimization problems [23]. They are based on the genetic process of biological organisms. Over many generations, natural populations evolve according to the principles of natural selection, i.e. survival of the fittest, first clearly stated by Charles Darwin in The Origin of Species. By mimicking this process, genetic algorithms are able to evolve solutions to real world problems, if they have been suitably encoded. The procedures of GA can be summarized as chromosome representation (encoding scheme) of a solution, an initial population, an evaluation function for rating solutions in terms of their fitness, genetic operators (reproduction, crossover, and mutation) that modify the genetic composition of offspring for the next generation and a termination rule [24]. Genetic algorithm is very powerful for searching larger regions of the solution space globally.

Simulated annealing is a stochastic approach for solving combinatorial optimization problems, in which the basic idea comes from the annealing process of solids. In this process, a solid is heated until it melts, and then the temperature of the solid is slowly decreased (according to an annealing schedule) until the solid reaches the lowest energy state or the ground state. Simulated annealing is very powerful for searching local regions of the solution space exhaustively via stochastic hill climbing. Simulated annealing also has the solution refining capability [25]. By combining global crossover operator of genetic algorithm and the local hill-climbing of simulated annealing, this study proposes a hybrid optimization algorithm, named as genetic-simulated annealing algorithm.

The flow chart shown shows the working of the GSAA. The algorithm starts with initialization of parameters. The initial population consists of randomly generated chromosomes. Each chromosome consists of randomly selected facility from the set of alternative facilities for each location and a gene denotes a facility to be arranged in one location without replication. The reproduction operator in the genetic algorithm module consists of genetic operations like selection, crossover and mutation. Roulette wheel selection is used as the selection operator. Crossover probability and mutation probability decides whether the chromosome should undergo these operations. Single point crossover is used for crossover. If infeasible chromosomes are produced during crossover, they are made feasible by omitting repeated departments and by inserting the missing departments. Swap mutation is used for mutation. Then the best solution is fed to the SA module for solution refinement. This improved solution is fed to the GA for next iteration till the termination condition is satisfied.

The above proposed genetic-simulated annealing algorithm is coded in MATLAB. The parameters setting of the GA pre-test were referred to the related literatures [24]. The population size (PS) was set to [30, 50, 100]; the cross over probability (P_{CR}) was set to [0.4, 0.6, 08]; the mutation probability (P_{MR}) was set to [0.01, 0.05, 0.1]. The termination criteria for the algorithm are set when optimal value reaches or the solution remains unimproved for 300 generations [21].

The optimum result was generated when PS=30, P_{CR}=0.6, and P_{MR}=0.05. For the simulated annealing module the initial temperature was set to 90, cooling ratio to be 0.98, termination condition or final temperature to be 3, Metropolis criterion was selected to govern the acceptance or rejection of configuration changes as suggested in Pillai [4].

The flow chart for the above mentioned algorithm is shown in the Fig. 1 and 2.

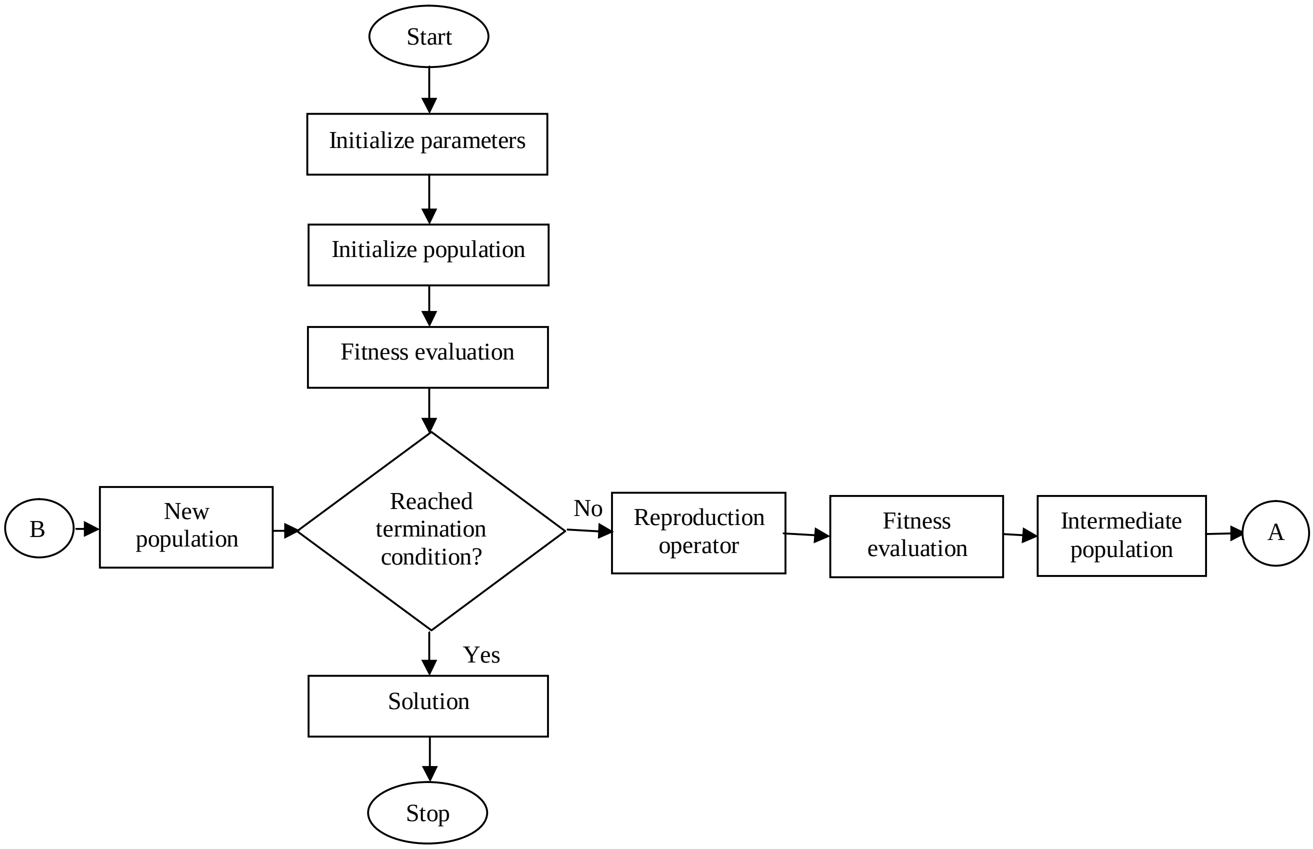

Fig. 1 Flow chart for GSAA: GA module

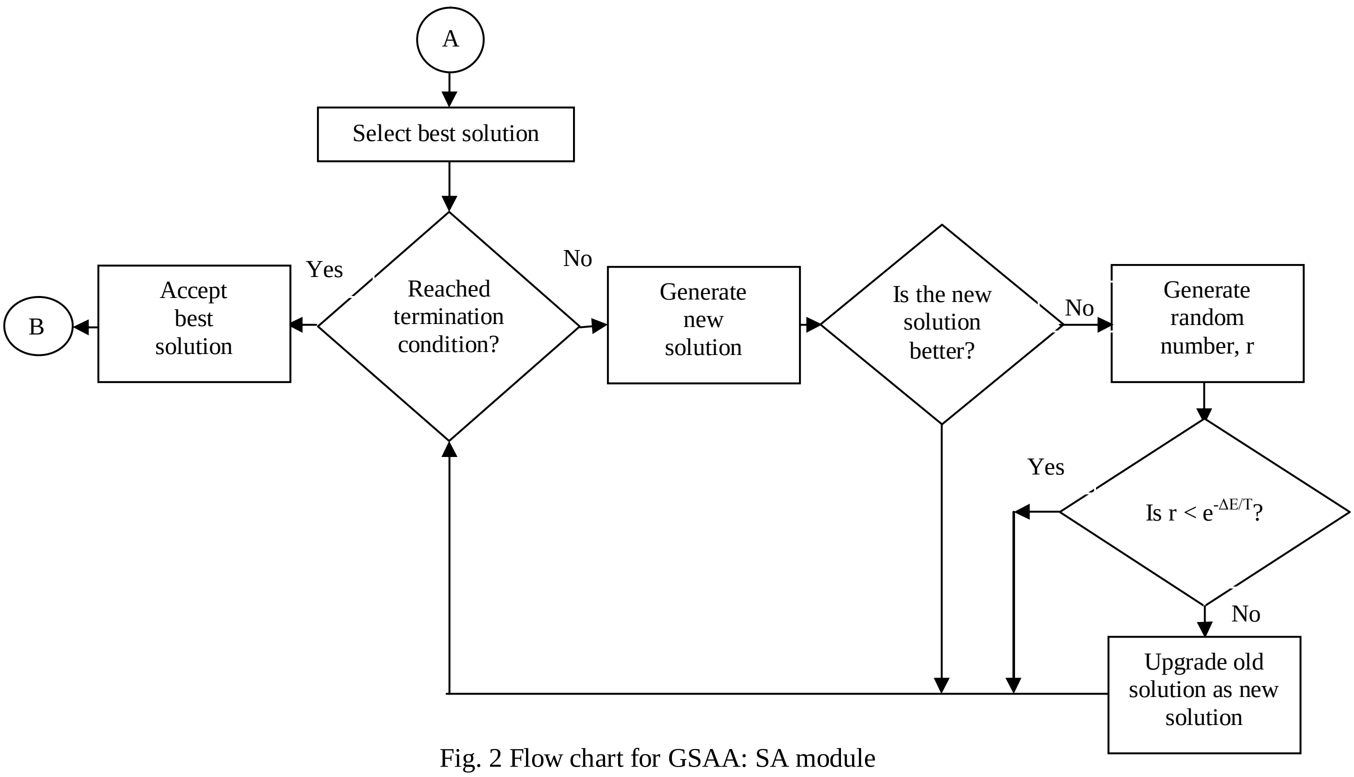

Fig. 2 Flow chart for GSAA: SA module

V. Results and discussion

A. *Results of case study from Balakrishnan and Cheng*

Performance of GSAA method for robust layout model using the data obtained from Balakrishnan and Cheng is evaluated and compared with that of simulated annealing method described in Pillai [4]. These data set consists of eight problems in each of the six situations (6 – departments 5 and 10 periods; 15 – departments 5 and 10 periods; and 30 – departments 5 and 10 periods) and thus a total of 48 problems which are solved using proposed robust model for DPLP. The results obtained by the proposed GSAA is compared with the results from SA (Robust) proposed by Pillai (2011) for different problem instances from Balakrishnan and Cheng are shown in for total material handling costs using the data set are shown in Table I.

TABLE I
COMPARISON OF RESULT FOR *BALAKRISHNAN AND CHENG* CASE

Description	Data Set 1		Data Set 2		Data Set 3		Data Set 4	
	SA (Robust)	GSAA	SA (Robust)	GSAA	SA (Robust)	GSAA	SA (Robust)	GSAA
6-departments 5-periods	106419	106419	105731	105731	107650	107650	108260	108260
6-departments 10-periods	220776	220776	217412	217412	219024	219024	217350	217350
15-departments 5-periods	506847	506847	500284	500284	508011	508011	503699	503699
15-departments 10-periods	1059100	1059100	1022447	1022447	1068402	1068402	1054997	1054997
30-departments 5-periods	579704	579704	576350	576350	586831	586831	584318	584318
30-departments 10-periods	1172691	1172691	1182286	1183857	1188620	1188620	1198487	1199263

Description	Data Set 5		Data Set 6		Data Set 7		Data Set 8	
	SA (Robust)	GSAA	SA (Robust)	GSAA	SA (Robust)	GSAA	SA (Robust)	GSAA
6-departments 5-periods	108188	108188	107765	107765	108114	108114	107248	107248
6-departments 10-periods	217142	217142	217397	217397	219788	219788	220144	220144
15-departments 5-periods	502622	502622	499891	499891	502919	502919	507970	507970
15-departments 10-periods	1051395	1051395	1057543	1057543	1037066	1037066	1040450	1040450
30-departments 5-periods	570492	570492	572782	573156	571703	571703	596835	597543
30-departments 10-periods	1198674	1198674	1202033	1202033	1210573	1210573	1209088	1209088

The results showed that the proposed GSAA gives lesser material handling costs as that of SA (Robust). Another important parameter for the measurement of performance of an algorithm is the time taken to obtain the results. The computational time for the proposed GSAA is much lower than that of SA (Robust) as shown in Table II. Especially when the number of departments increases, the rate of increase in the computational time is less for GSAA than SA (Robust).

TABLE II
COMPUTATIONAL TIME FOR OBTAINING SOLUTION FOR VARIOUS PROBLEMS

Problem size	No. of periods	Time taken in sec.	
		SA (Robust)	GSAA
6	5	9.937	2.48722
	10	12.935	3.6452
15	5	125.214	72.781
	10	164.531	111.56
30	5	1074.194	502.7279
	10	1344.017	829.2769

VI. CONCLUTION

In this paper a genetic - simulated annealing algorithm is used for solving robust dynamic facility layout problems. This heuristic combines the exploration features of GA as well as exploitation features of the SA algorithm. The local search characteristic of SA is introduced to the global search feature of the GA enabling the proposed GSAA an efficient algorithm for solving combinatorial problems. The performance of this algorithm was compared with previously proposed SA (Robust). It was found that the proposed GSAA gives similar results with much lesser computational time. As the problem size is increasing this advantage is getting more evident.

REFERENCES

[1] A. S. Ramkumar, S. G. Ponnambalam, and N. Jawahar, "A new integrated fast local search heuristic for solving QAP formulation in facility layout design". *Robotics and Computer-Integrated Manufacturing*, vol. 25, pp. 620-629, 2009.
[2] A. Tompkins, J. A. White, Y. A. Bozer, E. H. Frazelle, M. A. Tanchoco, and J. Trevino, *Facilities Planning*, New York: Wiley, 1996.
[3] M. J. Rosenblatt, "The dynamics of plant layout", *Management Science*, vol. 32 (1), pp. 76–86, 1986.
[4] V. M. Pillai, I. B. Hunagund, and K. K. Krishna, "Design of robust layout for dynamic plant layout problems", *Computers and Industrial Engineering*, vol. 61, pp. 813-823, 2006.
[5] M. R. Garey, D. S. Johnson, and R. Sethi, "The complexity of the flowshop and jobshop scheduling". *Math Operation Research* , vol. 1, pp. 117–129,1976.
[6] J. Balakrishnan, and C. H. Cheng, "Dynamic layout algorithms: a state-of-the art survey", *Omega The International Journal of Management Science*, vol. 26 (4), pp. 507–521, 1998.
[7] B. Ulutas and A. A. Islier, "A new approach to the dynamic layout problem", *in Proc. 35th International Conference on Computers and Industrial Engineering*, 2005, pp. 2019–2024.
[8] S. K. Konak, "Approaches to uncertainties in facility layout problems: perspectives at the beginning of the 21st Century", *Journal of Intelligent Manufacturing*, vol. 18, pp. 273–284, 2007.
[9] J. Balakrishnan, and C. H. Cheng, "Genetic search and the dynamic layout problem", *Computers and Operations Research*, vol. 27 (6), pp. 587–593, 2000.
[10] J. Balakrishnan, and C. H. Cheng, "The dynamic plant layout problem: incorporating rolling horizons and forecast uncertainty", Omega *The international Journal of Management Science*, vol. 37, pp.165–177, 2009.
[11] J. Balakrishnan, C. H. Cheng, and D. G. Conway, "An Improved pair-wise exchange heuristic for the dynamic plant layout problem", *International Journal of Production Research*, vol. 38 (13), pp. 3067–3077, 2007.
[12] J. Balakrishnan, C. H. Cheng, D. G. Conway, and C. M. Lau, "A hybrid genetic algorithm for the dynamic plant layout problem", *International Journal of Production Economics*, vol. 86, pp. 107–120, 2003.
[13] A. Baykasoglu, T. Dereli, and I. Sabuncu, "An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems", *Omega The International Journal of Management Science*, vol. 34, pp. 385–396, 2006.
[14] A. Baykasoglu, and N. Gindy, "A simulated annealing algorithm for dynamic layout problem", *Computers and Operations Research*, vol. 28 (14), pp. 1403–1426, 2001.
[15] P. Corry, and E. Kozan, "Ant colony optimisation for machine layout problems", *Computational Optimization and Applications*, vol. 28, pp. 287–310, 2004.
[16] A. R. Jr. McKendall, and J. Shang, "Hybrid ant systems for the dynamic facility layout problem," *Computers and Operations Research*, vol. 33, pp. 790–803, 2006.
[17] A. R. Jr. McKendall, J. Shang, and S. Kuppusamy, "Simulated annealing heuristics for the dynamic facility layout problem", *Computers and Operations Research*, vol. 33, pp. 2431–2444, 2006.

[18] G. Aiello, and M. Enea, "Fuzzy approach to the robust facility layout in uncertain production environments," *International Journal of Production Research*, vol. 39 (18), pp. 4089–4101, 2001.
[19] S. Benjaafar, and S. Sheikhzadeh, "Design of flexible plant layouts", *IIE Transactions*, vol. 32, pp. 309–322, 2001.
[20] P. Kouvelis, A. A. Kuawarwala, and G. J. Gutierrez, "Algorithms for robust single and multiple period layout planning for manufacturing systems", *European Journal of Operational Research*, vol. 63, pp. 287–303, 1992.
[21] V. M. Pillai, and K. Subbarao, "A robust cellular manufacturing system design for dynamic part population using a genetic algorithm", *International Journal of Production Research*, vol. 46 (18), pp. 5191–5210, 2008.
[22] W. M. Chan, C. Y. Chan, and C. K. Kwong, "Development of the MAIN algorithm for a cellular manufacturing machine layout", *International Journal of Production Research*, vol. 42 (1), pp. 51–65, 2004.
[23] D. Beasley, D. R. Bull, and R. R. Martin, "An overview of genetic algorithms, part 1, fundamentals", *University Computing*, vol. 15, pp. 58-69, 1993.
[24] C. L. Yang, S. P. Chuang, and T. S. Hsu, "A genetic algorithm for dynamic facility planning in job shop manufacturing", *International Journal of Advanced Manufacturing Technology*, vol. 52, pp. 303-309, 2011.
[25] S. Bisht, "Hybrid Genetic-simulated Annealing Algorithm for Optimal Weapon Allocation in Multilayer Defence Scenario", *Defence Science Journal*, vol. 54, pp. 395-405, 2004.

