A real-time exhaustive search algorithm for the weapon-target assignment problem

Naeem Feghhi[1], Amir R Kosari[2*], MA Amiri Atashgah[3]

[1,2,3] *Faculty of New Sciences and Technologies, Department of Aerospace Engineering, University of Tehran, Tehran, Iran*

**** Corresponding author: kosari_a@ut.ac.ir***

Abstract: Weapon-Target Assignment (WTA) as an important part of the aerial defense cycle has long been studied. Challenges are usually finding fast-computing methods to search optimal or near-optimal solution in cases of a large number of weapons and targets. This viewpoint is more mathematically considerable but practically has limited usage in the mentioned context. In this paper, a real-time search algorithm is proposed which decomposes the WTA problem and by decreasing the size of solution space and deleting impossible solutions, provides a real-time exhaustive search algorithm. Implementation of the algorithm for three typical scenarios shows excellent real-time performance and the possibility of finding exact solutions for large-scale problems.

Keywords: Weapon Target Assignment, Exhaustive Search, Multi-objective, real-time, optimization

1- Introduction

In aerial defense cycle systems, Weapon-Target Assignment (WTA) as an important subsystem has been considered for decades. In a rough classification, there are large-size and small-size problem categories. Using exhaustive search, small-size static WTA problem is solved fast and the exact solution is gained. In this case, all possible pairs of weapon-target are checked and the cost function is calculated and the optimum solution is found. Nature of this problem is such that if the size increases, to find an exact solution, the required computational burden significantly increases (Johansson et al. 1). The literature on the WTA problem is classified into two study categories; mostly, some researchers have tried to use or propose new algorithms to solve the problem more effectively. Exhaustive search (Johansson et al. 1), Genetic algorithm (GA) (Johansson et al. 1; Ling *et al.* 2; Yan, Y., Zha, Y., Qin, L., et al. 3, Wen, Y. et al. 4), Ant Colony together with GA (Jiuyong et al., 5), particle swarm optimization (Zhu et al. 6), Large Scale Neighborhood (Mei-Zi 7; Zhou, T., Zhang, J., Shi, J., et al. 8), Fuzzy (Sahin et al. 9), Game theory (Gürdal et al. 10; Lechevin et al. 11; Seung et al.12) and Markov decision process (Plamondon et al. 13) are some examples of the research approaches to this problem. Subsequently, a number of studies compare the efficiency of different algorithms such as Ant colony, GA, PSO, and Maximum Marginal Return (Johansson et al. 14), and similarly GA, Tabu Search, Simulated Annealing and Variable Neighborhood Search (Tokgöz et al. 15). In those research interests, the problem is solved for the target-based WTA problem or asset-based problems. Furthermore, two types of static and dynamic WTA are studied in the literature. The problem is often considered as a single objective case while in some studies multi-objective methods are focused (such as Lotter et al. 16, Li et al. 17, Lotter et al. 18, Zhou et al. 19, and Lotter 20). For more, the proposed algorithms are usually implemented for a static target-based form of WTA problem.

Exact algorithms are proposed to solve the WTA problem for some special cases, such as: (1) when all the weapons are identical, or (2) when the targets can receive at most one weapon (Ahuja et al. 21). WTA problems are mostly NP-complete, discrete, stochastic, nonlinear and usually large-size; this is why obtaining the optimal solution of WTA problem is usually impossible (Chi et al. 22). In large-size problems, if real-time solutions are required, heuristic algorithms are usually used. Using Heuristic methods, near-optimal solutions may possibly be found in a short time. Weapon assignment problem is also a case of general multi-robot task allocation problem, wherein the objective is to optimally assign a set of robots to a set of tasks in such a way that the overall system performance, subject to a set of constraints, is optimized (Khamis et al. 23). Scheduling problem is another similar field of problems which contains two sub-problems: resource allocation and scheduling. In the resource allocation sub-problem, the computational burden is a function of the number of resources allocated (Ziaee, M. 24). Another branch of allocation problems is the Flexible Job Shop problem which includes routing (finding the best route to accomplish a job) and scheduling (Imanipour, N. . In the present study, a simple fast computing method is presented which tries to find the exact solution of a large-size problem at a higher level (multi weapon systems level) by simplifying the WTA problem. Concisely, our proposed algorithm tries to be simultaneously real-time and exact; two properties that generally are not collected in the previously proposed algorithms. The proposed algorithm decomposes the problem, distributes the targets to suitable defensive resources, executes favorite constraints and finally generates the reduced size solution space; in comparison with common exhaustive search method which assumes all weapons engaging all targets.

2. WTA general considerations

In the following, some considerations about viewpoints in solving WTA problem are presented:

2.1. The problem viewpoints

Traditional WTA Problem Equation: The traditional WTA problem, in the static target-based form, is a function of the form:

$$\min_{x_{i,k} \in \{0,1\}} F = \sum_{i=1}^{|T|} V_i \prod_{k=1}^{W} (1 - P_{i,k})^{x_{i,k}}, \qquad (1)$$

$$\text{subject to } \sum_{i=1}^{|T|} x_{i,k} = 1, \quad k = 1, \ldots, |W|. \qquad (2)$$

In which the parameters are as follows:

T: a set of targets. $T = \{T_1, \ldots, T_N\}$

W: a set of weapons. $W = \{W_1, \ldots, W_M\}$

V_i : the value of target i.

$P_{i,k}$: kill probability for the pair (T_i , W_k)

$x_{i,k}$ **:** decision variable. Its value is 1 if W_k is assigned to T_k and otherwise equals 0.
(Johansson et al. 1).

It is desired to find an optimal solution which minimizes the probability of targets survival, in which, the solution is a set of $\{T_i , W_k\}$ for all targets. Using Equation (1), the formula considers all T^W solutions. It is assumed that all weapons can engage all targets without weapon inventory limitation. Equation (1) can possibly be modified to consider special problems using further constraints.

Problem Sizes*:* In WTA problem, challenges are usually finding fast-computing methods to examine optimal or near-optimal solution in cases of a large number of weapons and targets. This viewpoint is more mathematically considerable but practically has limited usage in the mentioned context. Usually ignored point is that in the practical cases, a target rarely faces with all weapons. By removing never-occurring cases, the problem is simplified and consequently, the size of it significantly decreases.

Heuristic Methods*:* In solving a large-size WTA problem, the optimal solution is attainable using the exhaustive search. While the computation time becomes long, heuristic methods are often developed and improved to solve the problem faster and more analogous to the optimal solution. Heuristic methods have special search algorithm and search only part of possible solution space and the solution will be near-optimal.

By removing the impossible solutions and using criteria in the possible solutions space, based on decision-maker preferences, more unacceptable solutions are removed. As the size of the search space significantly decreases, all the remained solution space may be searchable and the exact best solution is obtained.

In (Zbigniew et al. 26) it is observed that the WTA solution space is decreased by deletion of solutions with kill probability less than minimum acceptable value. Through this filtering, the reduced problem is solved in less than a second for large size problems.

2.2. Simplified WTA

Usually, in order to find the WTA (near-optimal) solution in real-time, a decrease in solution accuracy is acceptable. This is because of the large computation load of the problem. It is claimed that in a real-world scenario, a very large-scale problem seldom occurs and the exact solution possibly can be attainable in real-time by decreasing the size of WTA problem. For more, a realistic air defense scenario for a naval task group practicably consists of less than ten targets and ten weapon systems (Johansson et al. 1). The WTA problem can be decomposed and solved in multiple stages. This causes omission of some impossible solutions, before starting the search process for finding the optimal solution (Frini, *A.* Guitouni, A. and Benaskeur, A. 27). By this solution viewpoint, it is possible to find the exact solution of a WTA problem under single/multiple objectives, using exhaustive search, even in large size WTA cases. Accordingly, an algorithm is designed to be practical, fast, and with some assumptions results in an exact solution.

The proposed algorithm (PA) can be compared with a rule-based heuristic search (RHS) algorithm (Rathinam et al. 28) as follows:

1- RHS uses some rules – which are extracted using methods such as data mining in solution space – to opt a path to approach the proper solution among all possible cases, however PA firstly removes absolute impossible solutions from the search space and reduces the problem and search space size and required rules are predefined.

2- The starting point of the search process in RHS is estimated based on proper primary solution(s) with some computational burden but in PA the starting point of search is selected based the priorities of the tasks and requires no estimation.

3- In RHS solution space is searched by a predefined resolution which causes neglecting some of possible solutions – a common way in heuristic methods. While in PA, no possible solution is neglected in the search process.

3. The proposed WTA algorithm

The WTA is a part of the command and control process in an air defense process. After receiving targets' kinematic data and extracting their features, priorities are assigned. In the following, the WTA process is discussed more in-depth. Figure 1, presents the Weapon Assignment Algorithm, and subsequently Figure 2 depicts Weapon Assignment Decision Maker (WADM), inputs, geometric features, priorities, outputs pairs, and order of engagement. According to Figure 2, the processes in WADM are as follows:

Engageable Targets: Based on the cross-parameter (Figure 3), for each weapon-target pair (WTP), it is calculated whether the target predicted path is within WS engagement radius or not.

Crossing Weapons: Subsequently, for each target, Engageable targets matrix (ETM) is generated (Figure 4, ①). ETM shows that each target sequentially could be engaged by which weapon systems. Crossing weapon systems are sorted in ETM based on chronological order.

Engagement combinations: by having ETM, all possible engagement pairs are generated. if the maximum of two shots against each target is allowable, and a target predicted path crosses *m* weapon systems, all combination containing one or two shots is calculated as follows:

$$\textit{number of combinations} = \binom{m}{1} + \binom{m}{2} + m = \frac{m(m+3)}{2} \qquad (3)$$

one shot

two shots
different

two shots
same WS

For example, if a target's predicted path is crossing two weapon systems; using equation (3), there are 9 possible combinations for engaging the target, containing one-shot and two-shot combinations.

Solution Criteria: There are decision factors which affect WA problem solution. Some of them are listed in (Lotter et al. 16 and Lotter 20). Before generating possible solutions, some criteria are defined. In this work the probability of kill, the number of weapons used against each target, and engagement cost are used to find best solution (Figure 4: ③) Furthermore, it is necessary to notice that among the solutions satisfying minimum acceptable kill probability condition, the solution that temporally occurs first is chosen.

Solution Space: All possible solutions are generated for all WTPs according to the criteria. For each WTP a vector is defined (Figure 4: ④) which contains information of the solution cost, number of engagements, probability of kill, and solution number (Equation (4)).

$$\textit{Solution Vector} = [\text{cost}, \text{eng}, \text{Pk}, \#] \qquad (4)$$

Engagement
cost

Number of

Probability of

Solution

For example, vector [8, 2, 0.9, 22] indicates a solution which costs 8 unit, engagement is done with two shots against the target, total probability of kill with two shots is 0.9 and this vector belongs to the 23th generated solution for the target (counter starts from 0).

Solution Filters: Having all possible solutions, the searching process for best solutions starts. Here, the Decision Maker can filter the solutions based on decision criteria. For example, it can select solutions; with kill probability greater than a margin, with a maximum limit on cost, or the number of engagements with each target.

Engagement Strategy: In addition to criteria and filters, Engagement Strategy (ES) affects the solution space, as well. Each ES defines rules for engagement and constraints for solution space and resembles optimization strategies. The optimal solution based on each ES differs from the others; engaging the target as soon as possible, engaging with the highest possible probability of kill, and engaging with minimum cost are some examples of ESs. It is worth pointing out that each ES can be a combination of two or more criteria.

Solution Maker: Solution Maker searches the solution space and applies the filter. Solution maker tires to find a set of the best weapon candidates to engage targets and starts weapon assignment from the target with the highest priority (Figure 4: ⑤). If the weapon inventory allows, the first best solution is selected. Filters may cause deletion of all solutions for a specific target. For example, if there is no weapon or combination of weapons (in case of multiple engagements with targets) to meet minimum acceptable P_k, another solution for the target will be chosen based on other criteria or solution with smaller P_k.

In the following, the problem is solved with the proposed algorithm through a limited or unlimited inventory condition. If the inventory is limited, the solution may not be optimal, however, in the other case, the solution is optimal.

4. Case Studies and Simulation Results

Performance of the proposed algorithm was tested in the context of a surface-based air defense system.

4.1 Engagement model

***Test scenario*:** There are six long-range (L1 – L6) and six medium-range (M1 - M6) weapons with limited inventory; positioned around a defended asset to protect it against six aerial targets approaching toward the asset. Weapons range and cost data are considered as shown in Table 1. Six targets approach the defended asset. The future path, according to the current position and velocity vector, is predicted. Table 2 displays the threat priorities based on the degree of their threat to the defended asset. In the following, weapon systems inventory is shown in Table 3. If a target crosses a Weapon System (WS), it can be engaged at a predictable distance and cross. Cross-parameter is a minimum distance of target predicted path with the center of a weapon system, as illustrated in Figure 3. Subsequently, crossing weapons for each target is shown in Table 4.

The probability of kill (**P_k**): It is assumed that kill probabilities for a specific resource are functions of weapon-target distance and cross. For a particular weapon–target pair, an effect depends on the range between them because of the following reasons: (1) The range of a weapon is limited, and (2) The accuracy of a weapon depends on a range to the target (Bogdanowicz et al. 29). While P_k increases, the target approaches the WS, and reaches a maximum value and then decreases. Similar behavior was seen in (Benaskeur et al. 30 , Lotter et al. 18). Likewise, for a specific distance, as cross increases, P_k decreases (Figure 5). According to this behavior, a matrix for P_k is defined; whenever a target is processed, based on the distance and cross wrt WS, possible kill probabilities are predicted.

Engagement Strategies: The WTA problem is solved for two different ESs; (1) Engaging all targets with maximum P_k and one shot against each target, and (2) Engaging all targets with a minimum allowable P_k, wherein maximum two shots are allowable against each target and minimum cost is desired.

4.2. Engagement scenario 1

To solve the problem based on the first ES, according to target cross wrt each crossing weapon, best distance for engagement and respectively the maximum P_k for each crossing weapon is calculated. The best solution based on the first ES is shown in Table 5. To solve the WAP for the second ES, all possible solutions containing possible single shots and combination of all weapon system for two shots are generated. In the next step, using the minimum allowable P_k filter, the outlier solutions are ignored. Finally, the solution maker searches the remained solution space for the best one .In the following, Figure 6 demonstrates the first simulation scenario, wherein the targets T1 to T6 are shown from their starting (Detection) point and their direct predicted path along defended space. For the illustrated problem in Figure 6, a number of possible solutions and final allowable solutions for each target are shown in Table 6. Furthermore, Table 7 contains outputs of the solution maker as the best results.

In the simulation, it is assumed that all targets move in straight line with random constant speed except target 1 which is hand-controlled.

4.3. Scenario 2

In the following, Figure 7 shows another typical scenario.

The scenario contains five time-steps (shown in Figure 7 by ① to ⑤). The result of targets prioritization and weapon assignment for the five time-steps is presented in Figure 8 and Table 8, wherein target 1 is not Engageable in time-steps 1 and 3; as its predicted path is out of all weapon systems engagement zones. Priority between target pairs (T2,T3), (T4,T5) and (T3,T4) are changed in time-step transitions: ① to ②, ② to ③ and ④ to ⑤, respectively.

4.4. Engagement scenario 3 and 4

The simulation is also performed for both 12*12 (scenario was shown in Figure 9, containing target starting point and predicted path) and 12*18 (weapons * targets) problem instances and the related results are presented in Table 9-10.

Table 11 shows the computation time for three problem instances of sections 4.2 and 4.4 and indicates that the proposed algorithm solves the problem in real-time.

4.5. Engagement scenario 5

The algorithm likewise was implemented for 16 different scenarios based on four different numbers of weapon systems and targets. The results (Table 12) express that adding new targets has more impact on algorithm performance than adding new weapon systems. Figure 10 shows the computational time for an equal number of weapons and targets among the 16 scenarios. As a rough estimation, a problem of size 45*45 (weapons * Targets) can be solved in less than one second (real-time) for a Symmetric emplacement of defense system (similar to Figure 9).

Performance of the proposed algorithm highly depends on the defense system arrangement. In the worst case, i.e. when all targets face all weapons, the computational burden is similar to the exhaustive search algorithm. This case in special cases may occur. However, for lots of practical applications, the proposed algorithm can find the proper solution to a large-size problem in real-time.

4.6. Discussion

In (Johansson et al. 1), the real-time condition is assumed the solutions which are obtained in about one second. Hence, to satisfy this condition, the size of a WTA problem which is solved in the static form using exhaustive search algorithm should be smaller than *7*7* (weapons * targets). The proposed algorithm, based on removing impossible zones of the search space, could solve the static WTA problem of the size *24*24* (weapons * targets) in about 0.25 second using exhaustive search algorithm. So the real-time condition is satisfied.

For more, a problem of size (12*6) in the form of equation (1), assuming that all weapons are used, requires 6^{12} iterations. Accordingly, the use of the exhaustive search is not suitable for real-time applications.

The problem with the proposed algorithm is decomposed and solved in multiple steps. Firstly, *12*6* cases are checked to find crossing weapon systems. Next, the related criteria are applied and all possible solutions for each target are generated independently. In the presented defense arrangement, a maximum number of possible solutions for each target (with 12 weapons) is *44*, because maximum crossing weapons for a target is 8. As multiple shots are possible from each weapon system, if there be no weapon inventory limitation, the solution is optimal.

The computational burden of WTA problem is reduced by using two steps. Firstly, according to the arrangement of the defense system and linear predicted path of targets, the maximum number of crossing weapons for each target is limited. For the defensive arrangement presented in Figure 6, the maximum number of weapon systems that a target meets is 8. Secondly, decision-maker strategies for engagement (such as minimum acceptable kill probability) reduce the computations more. So if there is no solution to exclude, assuming all targets do not meet all weapons, the computational burden of the problem is reduced. In the presented case study, weapon assignment problem was solved with reducing large size time-consuming (using exhaustive search) or imprecise (using heuristic search algorithm) problem to a simple fast-computing one; based on this fact that real-world problems are much simpler than basic WTA problem formulation (Equation (1)). Finally, in Table 13, one can notice the general comparison of weapon assignment algorithms in the aspects of being real-time and optimality.

5. Conclusions

Weapon-target assignment problem, typically is studied at the level of multiple weapon systems. Accordingly, the assignment problem is solved in sequential steps. By removing the impossible/outlier solutions from solution search space in primary steps, the size of search space and required time for finding the optimal solution considerably decreases. Hence, instead of using imprecise heuristic search methods, an exact exhaustive method is used for finding real-time WTA problem solution. In brief, the pros of the proposed algorithm are the possibility of being a real-time solution, low computational burden, and providing an optimal solution (with some assumptions) for the multi-criteria WTA problem. To observe the possibility of being real-time and optimality of the solutions, a general comparison of the different weapon assignment algorithms was presented.

References

1. Johansson, F. & Falkman G. "An empirical investigation of the static weapon-target allocation problem", Proceedings of the 3rd Skövde Workshop on Information Fusion Topics: 63-67 (2009).
2. Ling, W., Hang, Y. W., Fa-xing, L. and et al. "An Anytime Algorithm Based On Modified GA For Dynamic Weapon-Target Allocation Problem", IEEE Congress on Evolutionary Computation: 2020-2025 (2008).
3. Yan, Y., Zha, Y., Qin, L., et al. August. "A research on weapon-target assignment based on combat capabilities". In Mechatronics and Automation (ICMA), 2016 IEEE International Conference on (pp. 2403-2407) (2016).
4. Wen, Y., Liu, L., Wang, Z. , et al. "Multi-UCAVs targets assignment using opposition-based genetic algorithm". In Control and Decision Conference (CCDC), 2015 27th Chinese (pp. 6026-6030). IEEE (2015)
5. Jiuyong Z., Chuanqing X., Xiaojing W. and et al. "ACGA Algorithm of Solving Weapon - Target Assignment Problem", Open Journal of Applied Science, 2(4): 74-77 (2012).

6. Zhu, B., Zou F. & Wei J. "A Novel Approach to Solving Weapon-target Assignment Problem Based on Hybrid Particle Swarm Optimization Algorithm", International Conference on Electronic & Mechanical Engineering and Information Technology, IEEE, 12-14 August: 1385-1387 (2011).
7. Mei-Zi, L. "Constrained Weapon–Target Assignment: Enhanced Very Large Scale Neighborhood Search Algorithm", IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, 40(1): 198-204 (2010).
8. Zhou, T., Zhang, J., Shi, J., et al. "Multidepot UAV Routing Problem with Weapon Configuration and Time Window". Journal of Advanced Transportation (2018).
9. Sahin, M.A. & Kemal Leblebicioglu, K. "Approximating the optimal mapping for weapon-target assignment by fuzzy reasoning", Information Sciences 255: 30–44 (2014)
10. Gürdal A., Marden J.R. and Shamma J.S. "Autonomous Vehicle-Target Assignment: A Game-Theoretical Formulation", Transactions of the ASME, 129: 584-596 (2007).
11. Lechevin, N., Rabbath, C.A. & Lauzon M. "A Distributed Network Enabled Weapon-Target Assignment for Combat Formations", Optimization & Cooperative Ctrl. Strategies, Springer LNCIS, 381: 47–67 (2009).
12. Seung, H.R., Hwa-Sung, K.& Seung-Won, S. "The Effect of Decentralized Resource Allocation in Network-Centric Warfare", IEEE, The International Conference on Information Network: 478-481 (2012)
13. Plamondon, P., Chaibdraa, B. and Benaskeur, A.R. "A Multiagent Task Associated MDP (MTAMDP) Approach to Resource Allocation", AAAI Spring Symposium (2006).
14. Johansson, F. & Falkman, G. "A Suite of Metaheuristic Algorithms for Static Weapon-Target Allocation". GEM: 132-138 (2010).
15. Tokgöz, A. & Bulkan, S. "Weapon target assignment with combinatorial optimization techniques". International Journal of Advanced Research in Artificial Intelligence (IJARAI), 2(7):39-50 (2013).
16. Lotter, D.P., Nieuwoudt, I. and Van Vuuren J. "A multiobjective approach towards weapon assignment in a ground-based air defense environment", Orion. 29(1): 31-54 (2013).
17. Li, J., Chen, J., Xin, B. and Chen, L. "Efficient Multi-objective evolutionary algorithms for solving the multi-stage weapon-target assignment problem: A comparison study", IEEE Congress on Evolutionary Computation (CEC): 435-442 (2017).
18. Lotter, D.P. and Van Vuuren, J. "A Tri-Objective, Dynamic Weapon Assignment Model for Surface-Based Air Defence". ORiON, 32(1):1-22 (2016).
19. Zhou, D., Li, X., Pan, Q., et al. "Multiobjective weapon-target assignment problem by two-stage evolutionary multiobjective particle swarm optimization", IEEE International Conference on Information and Automation Ningbo: 921-926 (2016).
20. Lotter, D.P. "Modeling weapon assignment as a multiobjective decision problem", partial fulfillment of the requirements for the degree MComm, Stellenbosch University. (2012).

21. Ahuja, R.K. Kumar, A., Krishna, C.J. and et al. "Exact and Heuristic Algorithms for the Weapon Target Assignment Problem". Operations research, 55(6) 1136-1146. (2007).
22. Chi, H., Liu, J., Chen, Y. and et al. "Survey of the research on dynamic weapon-target assignment problem, Journal of Systems Engineering and Electronics", 17(3):559-565 (2006).
23. Khamis A., Hussein A., Elmogy A. "Multi-robot Task Allocation: A Review of the State-of-the-Art". In: Koubâa A., Martínez-de Dios J. (eds) Cooperative Robots and Sensor Networks. Studies in Computational Intelligence, Springer, 604: 31-51 (2015)
24. Ziaee, M. "Single machine scheduling problem with convex multi-resource dependent processing times and job deadlines", Scientia Iranica, 24(2), pp. 847-855 (2017).
25. Imanipour, N. "A Heuristic Approach Based on Tabu Search for Early/Tardy Flexible Job Shop Problems", Scientia Iranica, 13(1) (2006).
26. Zbigniew, R. Bogdanowicz. "Advanced Input Generating Algorithm for Effect-Based Weapon–Target Pairing Optimization". IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, 42(1), January (2012).
27. Frini, A. Guitouni, A. and Benaskeur, A. "Solving Dynamic Multi-Criteria Resource-Target Allocation Problem Under Uncertainty: A Comparison of Decomposition and Myopic Approaches". International Journal of Information Technology & Decision Making, 16(06), pp.1465-1496 (2017).
28. Rathinam, B., Govindan, K., Neelakandan, B. et al. "Rule based heuristic approach for minimizing total flow time in permutation flow shop scheduling". *Tehnički vjesnik*, *22*(1), pp.25-32 (2015).
29. Bogdanowicz, Z.R. "Advanced Input Generating Algorithm for Effect-Based Weapon–Target Pairing Optimization", IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, 42(1): 276-280 (2012).
30. Benaskeur, A.R., Kabanza, F., Beaudry and et al. "A Probabilistic Planner for the Combat Power Management Problem", ICAPS: 12-19 (2008).

list of captions

Figure 1. Weapon assignment algorithm

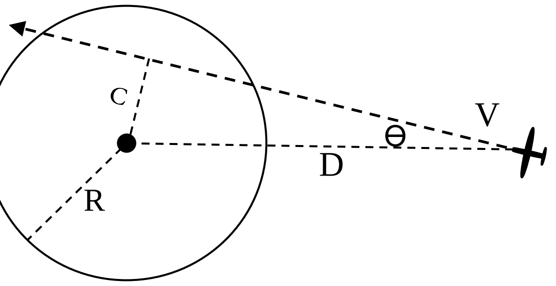


Figure 2. Weapon assignment decision maker

Figure 3. Target parameters (C: Cross, D: Distance)

Figure 4. Proposed WTA algorithm computational process

Figure 5. Pk behavior as a function of distance and cross (C1 < C2 < C3)

Figure 6. Weapon assignment test scenario (12 weapons * 6 targets)

Figure 7. A typical scenario with five time-steps

Figure 8. Targets priority and weapon assignment variations of the scenario shown in Figure 7

Figure 9   Weapon assignment test scenario (12 weapons * 12 targets)

Figure 10. Computational time for an equal number of weapons and targets

Table 1. Weapon systems range and cost data

Table 2. Targets priority

Table 3. Weapons inventory

Table 4. Crossing weapons

Table 5. The solution of WTA problem depicted in Figure 6 with the first ES.

Table 6. Possible solution combinations for WTA problem depicted in Figure 6 with the second ES.

Table 7. The solutions for WTA problem depicted in Figure 6 with the second ES.

Table 8. Prioritization and weapon assignment in five time-steps.

Table 9. Solutions for WTA problem (12 weapons * 12 targets)

Table 10. Solutions for WTA problem (12 weapons * 18 targets)

Table 11. Approximate computation time for the proposed algorithm

Table 12. Computational burden (in milliseconds) for 16 different scenarios

Table 13. A general comparison of weapon assignment algorithms

list of Figures and Tables

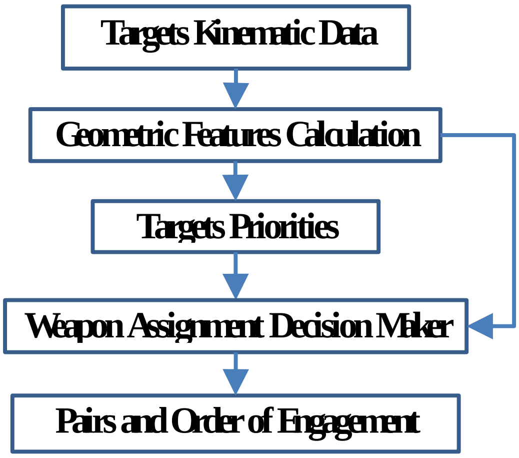


Figure 1. Weapon assignment algorithm

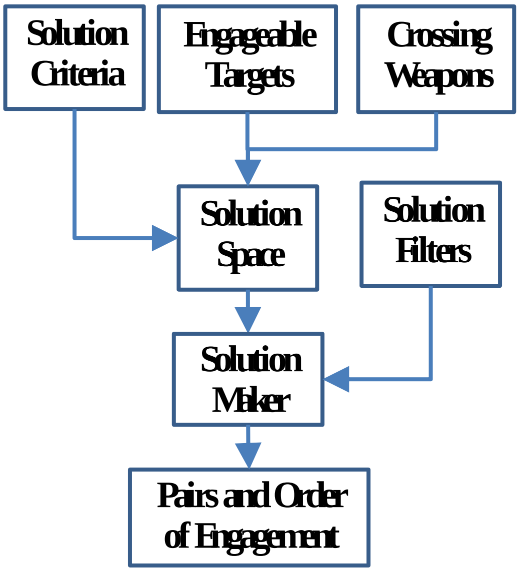


Figure 2. Weapon assignment decision maker

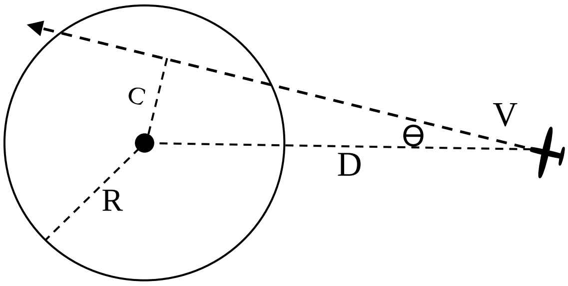


Figure 3. Target parameters (C: Cross, D: Distance)

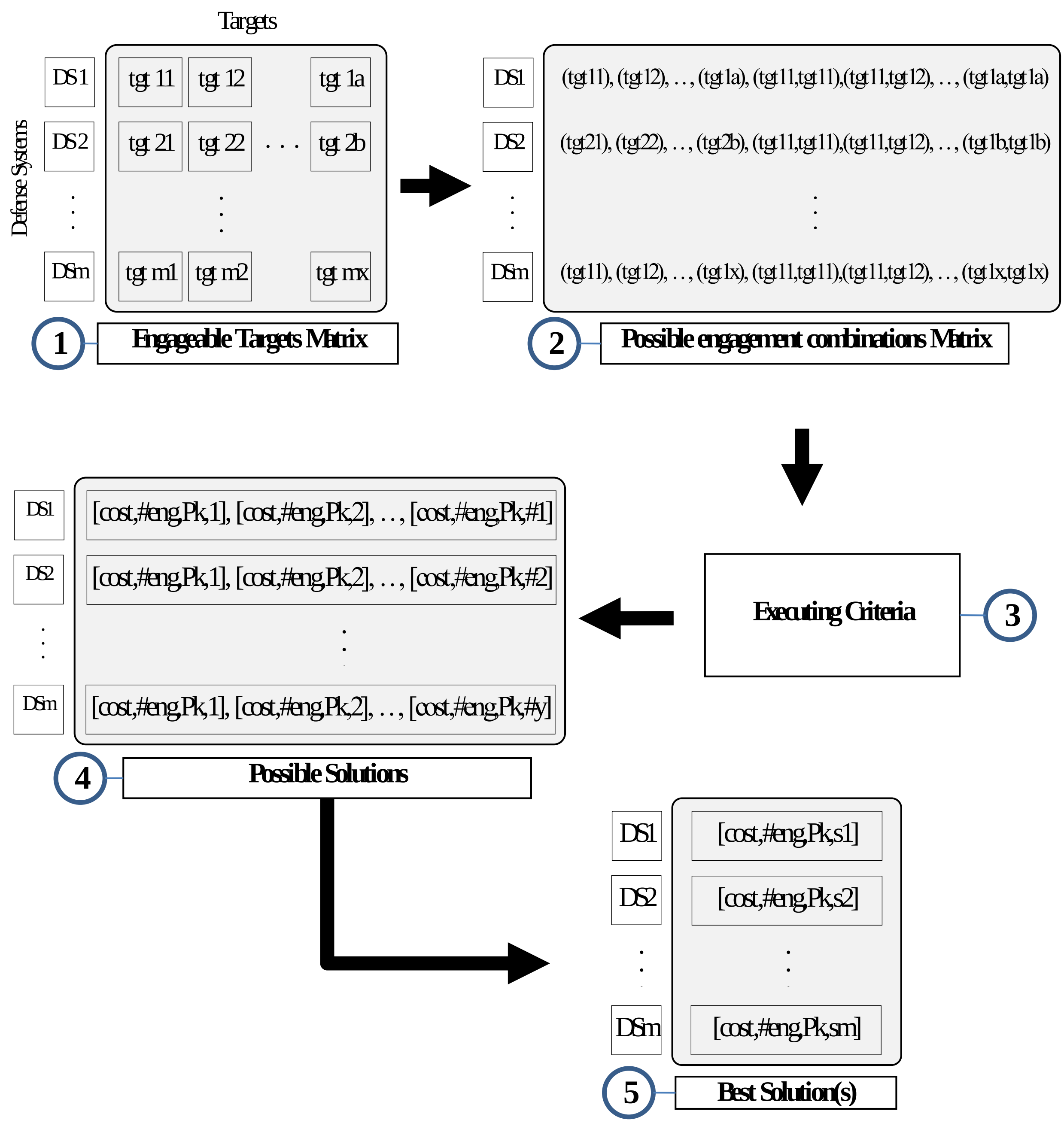
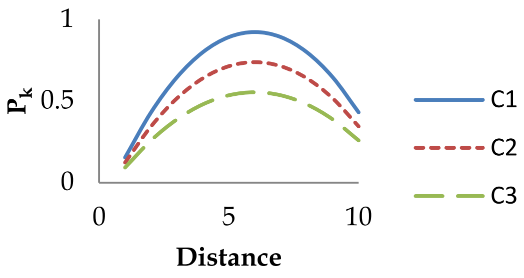


Figure 4. Proposed WTA algorithm computational process



(Pdf file of this figure is attached)

Figure 5. P_k behavior as a function of distance and cross ($C_1 < C_2 < C_3$)

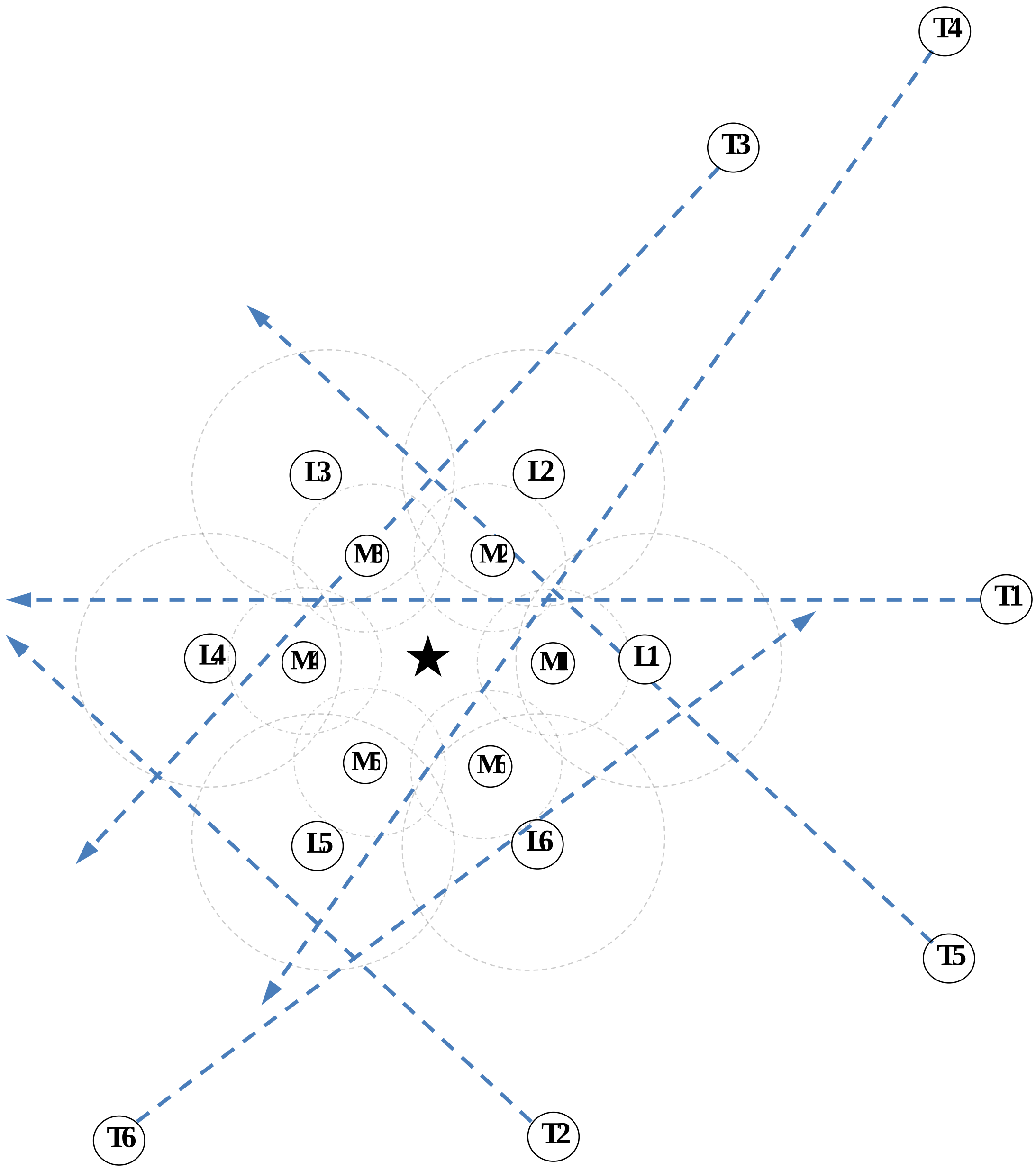


Figure 6. Weapon assignment test scenario (12 weapons * 6 targets)

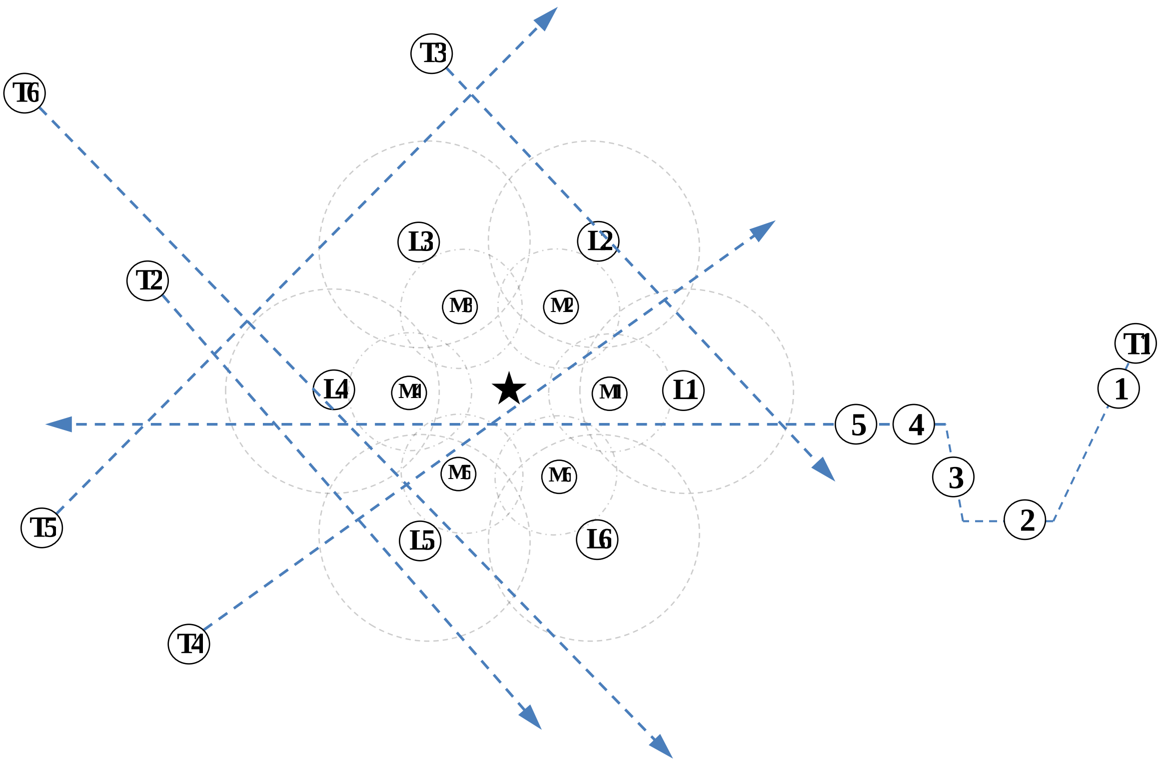


Figure 7. A typical scenario with five time-steps

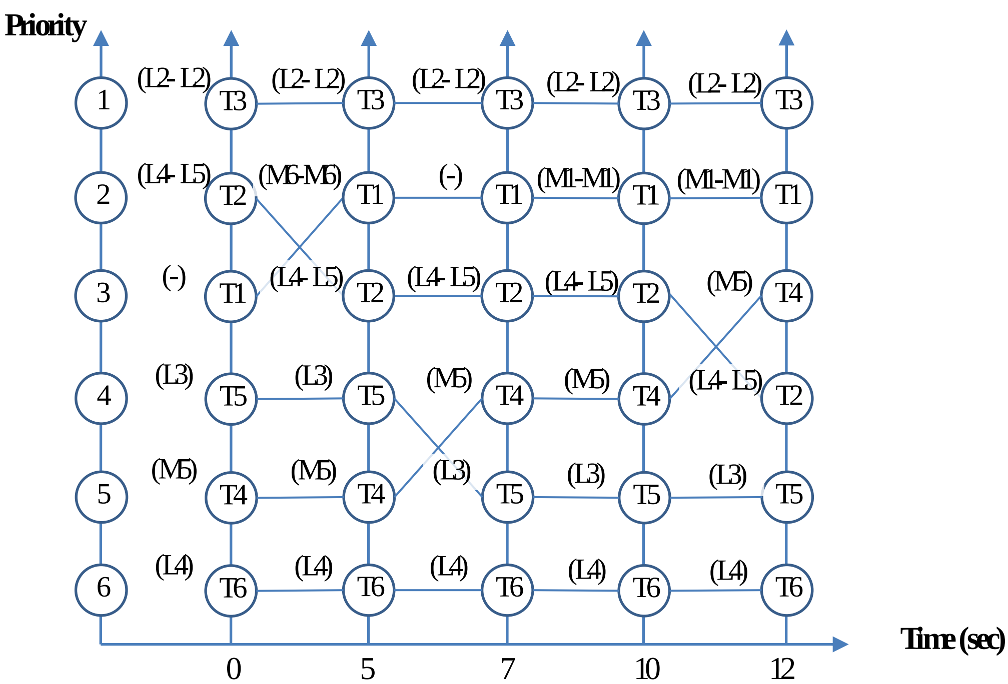


Figure 8. Targets priority and weapon assignment variations of the scenario shown in Figure 7

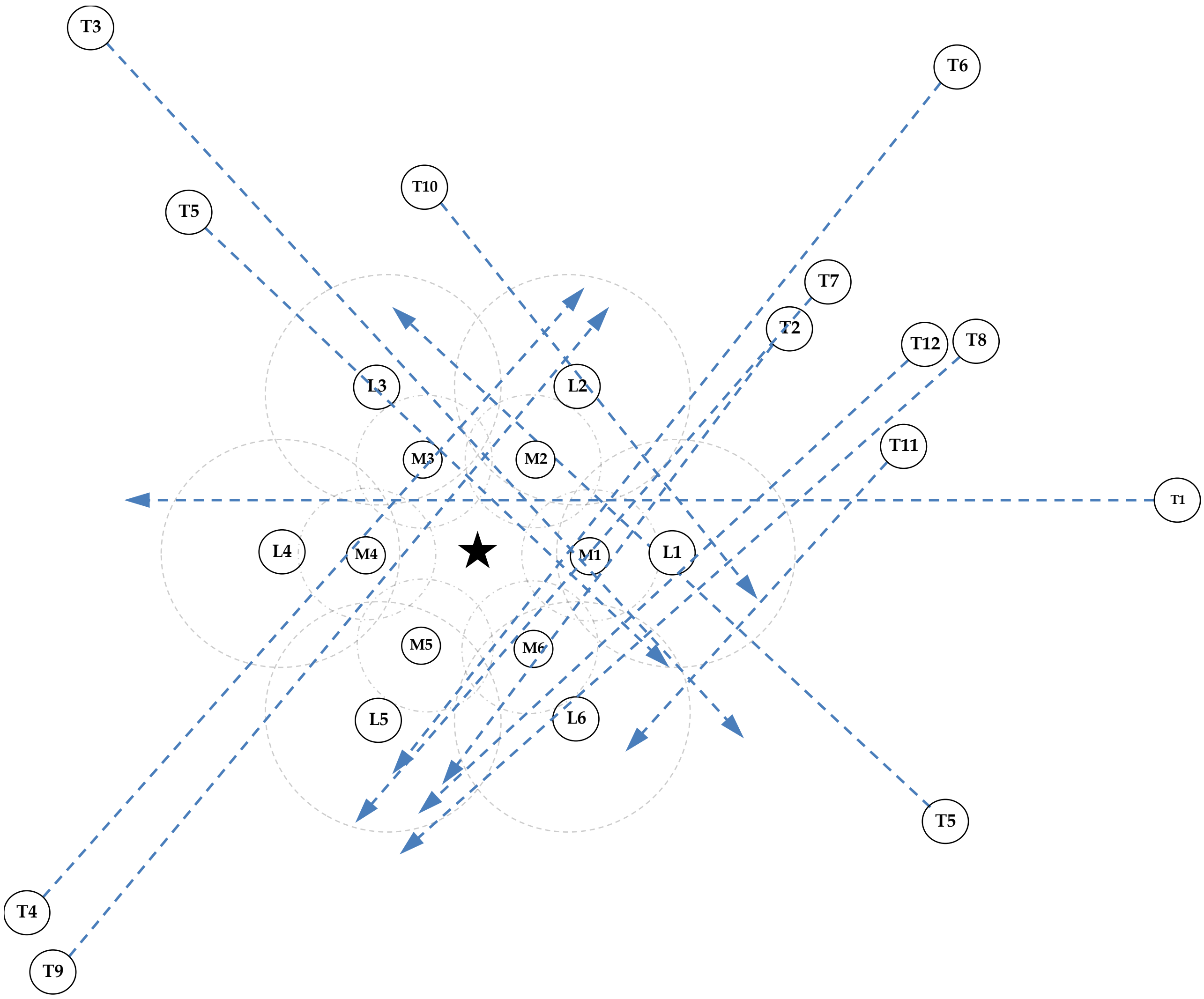


Figure 9 Weapon assignment test scenario (12 weapons * 12 targets)

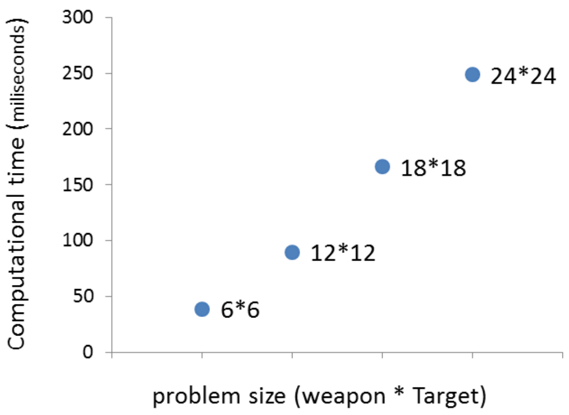


Figure 10. Computational time for an equal number of weapons and targets

Table 1. Weapon systems range and cost data

Weapon	Range	Cost
Long-Range	100	7
Medium-Range	50	4

Table 2. Targets priority

Priority	1	2	3	4	5	6
Target	1	2	3	6	5	4

Table 3. Weapons inventory

L1	L2	L3	L4	L5	L6	M1	M2	M3	M4	M5	M6
9	7	8	7	8	7	9	9	7	7	7	9

Table 4. Crossing weapons

Priority	Target	Crossing Weapon systems							
1	1	1	7	2	8	9	10	3	4
2	2	5	4						
3	3	2	3	9	10	4			
4	6	5	6	1					
5	5	1	7	8	2	3			
6	4	2	1	8	7	11	5		

Table 5. The solution of WTA problem depicted in Figure 6 with the first ES.

Priority	Target #	Weapon #	$P_k$max
1	1	L1	0.83
2	2	L5	0.76
3	3	M3	0.85
4	6	L6	0.88
5	5	L1	0.88
6	4	M5	0.88

Table 6. Possible solution combinations for WTA problem depicted in Figure 6 with the second ES.

Priority	Target #	Crossing Weapons #	Possible Solutions	Allowable Solutions
1	1	8	44	21
2	2	2	5	1
3	3	5	20	15
4	6	3	9	5
5	5	5	20	14
6	4	6	27	16

Table 7. The solutions for WTA problem depicted in Figure 6 with the second ES.

Priority	Target #	Weapon(s)	Pk	Cost
1	1	M1-M1	0.94	8
2	2	L5-L5	0.95	14
3	3	M3-M3	0.98	8
4	6	L5-L6	0.94	14
5	5	M1-M2	0.93	8
6	4	M2-M2	0.94	8

Table 8. Prioritization and weapon assignment in five time-steps.

Step 1: T = 0 ①

Priority	Target #	Weapon(s)	Pk	Cost
1	3	L2- L2	0.98	14
2	2	L4-L5	0.93	14
3	1	-	-	-
4	5	L3	0.5	7
5	4	M5	0.9	4
6	6	L4	0.9	7

Step 2: T = 5 ②

Priority	Target #	Weapon(s)	Pk	Cost
1	3	L2- L2	0.98	14
2	1	M6-M6	0.94	8
3	2	L4-L5	0.93	14
4	5	L3	0.5	7
5	4	M5	0.9	4
6	6	L4	0.9	7

Step 3: T = 7 ③

Priority	Target #	Weapon(s)	Pk	Cost
1	3	L2- L2	0.98	14
2	1			
3	2	L4-L5	0.93	14
4	4	M5	0.9	4
5	5	L3	0.5	7
6	6	L4	0.9	7

Step 4: T = 10 ④

Priority	Target #	Weapon(s)	Pk	Cost
1	3	L2- L2	0.98	14
2	1	M1-M	0.97	8
3	2	L4-L5	0.93	14
4	4	M5	0.9	4
5	5	L3	0.5	7
6	6	L4	0.9	7

Step 5: T = 12 ⑤

Priority	Target #	Weapon(s)	P_k	Cost
1	3	L2- L2	0.98	14
2	1	M1-M1	0.97	8
3	4	M5	0.9	4
4	2	L4-L5	0.93	14
5	5	L3	0.5	7
6	6	L4	0.9	7

Table 9. Solutions for WTA problem (12 weapons * 12 targets**)**

Priority	Target #	Crossing Weapons #	Possible Solutions	Allowable Solutions	Weapon(s)	Pk	Cost
1	4	2	5	2	M2-M2	0.969	14
2	9	6	27	21	M3	0.90	7
3	1	6	27	18	L1-L1	0.978	8
4	3	8	44	21	L1-L1	0.938	8
5	5	2	5	4	M6	0.90	7
6	18	5	20	15	L1-L1	0.949	8
7	7	6	27	21	M2	0.90	7
8	14	4	14	9	M1	0.90	7
9	6	5	20	14	L3-L4	0.934	8
10	13	7	35	22	L1	0.90	4

11	16	7	35	20	L1	0.90	4
12	12	3	9	5	M1-M1	0.978	14

Table 10. Solutions for WTA problem **(**12 weapons * 18 targets**)**

Priority	Target #	Crossing Weapons #	Possible Solutions	Allowable Solutions	Weapon(s)	Pk	Cost
1	4	6	27	18	L3-L3	0.978	8
2	9	4	14	10	M1	0.90	7
3	1	8	44	21	L1-L1	0.938	8
4	3	2	5	4	M5	0.90	7
5	5	6	27	18	L2	0.90	4
6	18	4	14	7	M1	0.90	7
7	7	6	27	21	M6	0.90	7
8	14	2	5	4	M6	0.90	7
9	6	7	35	21	M3	0.90	7
10	13	2	5	2	M3-M3	0.969	14
11	16	3	9	5	M4-M4	0.984	14
12	12	2	5	4	M3	0.90	7
13	2	6	27	21	M2	0.90	7
14	11	7	35	21	L1	0.90	4
15	8	1	2	1	M6-M6	0.938	14
16	10	1	2	2	M2	0.375	7
17	15	0	0	0	-	-	-
18	17	0	0	0	-	-	-

Table 11. Approximate computation time for the proposed algorithm

Weapons	Targets	Time (milliseconds)
12	6	30
12	12	97
12	18	127

Table 12. Computational burden (in milliseconds) for 16 different scenarios

Weapons \ Targets	6	12	18	24
6	39	67	99	127
12	51	90	131	169
18	65	114	166	212
24	73	131	193	249

Table 13. A general comparison of weapon assignment algorithms

Weapon Assignment Algorithms	Real-time	Optimal
Ant Co	●	×
Genetic		
PSO		
VLSN		
Tabu		
Simulated Annealing		
MMR		

<table>
<tr><td>Neural Networks</td><td rowspan="2">×</td><td rowspan="2"></td></tr>
<tr><td>Markov</td></tr>
<tr><td>Game Theory</td><td>×</td><td>×</td></tr>
<tr><td>Exhaustive Search</td><td>×</td><td>●</td></tr>
<tr><td>Proposed Algorithm</td><td>●</td><td>●</td></tr>
</table>

Biographies

Naeem Feghhi received his BS and MS in Aerospace Engineering from Sharif University of technology, Tehran, Iran and is currently a PhD student in the Faculty of New Sciences and Technologies, Department of Aerospace Engineering at University of Tehran, Tehran, Iran.

Amirreza Kosari received the B.S. degree in aerospace engineering from Amirkabir University, Tehran, Iran, in 1998 and the M.S. and Ph.D. degrees in aerospace engineering from Sharif University of Technology, Tehran, Iran, in 2001 and 2008.

From 2010 to 2016, he was an Assistant Professor with the Faculty of New Sciences and Technologies, University of Tehran, Iran. Since 2017, he is an Associate Professor with the same faculty. His research interests include trajectory optimization, optimal control, cooperative flights, and spacecraft attitude control. He is the author of several papers in the areas indicated above.

MA Amiri Atashgah was born in Ardabil, Iran, in 1977. He received the B.E. degree in Mechanical/Aerospace engineering in 1999, and the M.Sc. and Ph.D. degrees in Aerospace engineering in 2001 and 2011, respectively, from Sharif University of Technology (SUT), Tehran, Iran.

In 2013, he joined the Faculty of New Sciences and Technologies, University of Tehran, as a lecturer, and later, since November 2013 he has been with the Department of Aerospace Engineering, as a full-time Faculty member. In the beginning, he was an Assistant Professor and became an Associate Professor in 2018. His current research interests include Guidance, Navigation, and Control (Modeling and Simulation, Analysis, Test, and Implementation), Software and Hardware in the Loop, Vision-Based Navigation (SLAM/Mono-SLAM), Terrain Relative Navigation (TERCOM/DSMAC), Sensor Fusion (INS/GPS/Sonar/Lidar/Radar/Magnetic/Vision/TERCOM/DSMAC), Real-time Human-in-the-Loop Simulation of Aerospace Systems, and Virtual Reality (Computer Graphics, Terrain / Weather Engine).