Chapter 5

A Two-Step Optimisation Method for Dynamic Weapon Target Assignment Problem

Cédric Leboucher, Hyo-Sang Shin, Patrick Siarry, Rachid Chelouah, Stéphane Le Ménec and Antonios Tsourdos

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/53606

1. Introduction

The weapon target assignment (WTA) problem has been designed to match the Command & Control (C2) requirement in military context, of which the goal is to find an allocation plan enabling to treat a specific scenario in assigning available weapons to oncoming targets. The WTA always get into situation weapons defending an area or assets from an enemy aiming to destroy it. Because of the uniqueness of each situation, this problem must be solved in real-time and evolve accordingly to the aerial/ground situation. By the past, the WTA was solved by an operator taking all the decisions, but because of the complexity of the modern warfare, the resolution of the WTA in using the power of computation is inevitable to make possible the resolution in real time of very complex scenarii involving different type of targets. Nowadays, in most of the C2 this process is designed in order to be as a support for a human operator and in helping him in the decision making process. The operator will give its final green light to proceed the intervention.

The WTA arouses a great interest among the researcher community and many methods have been proposed to cope with this problem. Besides, the WTA has been proved to be NP-complete [1]. There are two families of WTA: the Static WTA (SWTA) and the Dynamic WTA (DWTA). In both of these problems, the optimality of one solution is based either on the minimisation of the target survival after the engagement or the maximisation of the survivability of the defended assets. The main feature of the SWTA stands in its single stage approach. It is considered that all the information about the situations are provided and the problem can be considered as a constrained resource assignment problem. In contrast, the DWTA is a multi-stage problem in which the result of each stage is assessed, then use to update the aerial situation for the upcoming stages. The DWTA can also be expressed as a succession of SWTA, but the optimality of the final solution cannot be guaranteed since it comes to the same as in a greedy optimisation process. One other difference stands in the temporal dimension of the DWTA which does not exist in the SWTA. The weapons can intervene within a certain defined time because of physical, technical and operational constraints. In addition, any DWTA problem has to be solved in using real-time oriented method. By real-time it is assumed that the proposed method has to be fast enough to provide an engagement solution before the oncoming targets reached their goals. Most of the previous work on the WTA was focused on the resolution of the SWTA. Hosein and Athans was among the first to defined a cost function based on the assets [2]. This model was reused in [3] and [4]. Later, a second modelling has been proposed by Karasakal in [5], aiming to maximise the probability to suppress all the oncoming targets. One other variant of the WTA is to take into account a *threatening* value to each target according to its features and the importance of the protected assets. The research of Johansson and Falkman in [6] proposed a good overview of all the possible modelling, taking into account both of the developed models and enabling to take into consideration the value of the defended assets and the threatening index of the incoming target. Kwon *et al.* explored further this principle in assigning a value to the weapon in [7]. The main researches on the SWTA started around the 1950's. Most of the proposals to solve this problem was based on the classic optimisation processes: branch and bound algorithm appears in the survey conducted in 2006 by Cai *et al.* [8]. With the evolution of the new technologies, some more complex methods appeared in [9] in using the neural networks. The genetic algorithms are used in [4], [10] and [11] to solve the SWTA. Cullenbine is using the Tabu Search method in [12]. A different approach angle is used in [13]. In this former approach the WTA problem is treated as a resources management problem and the reactivity of the proposed approach, based on the Tabu Search, was able to deal with real-time requirement. Nowadays, this method is used in many military systems like Rapid Anti-Ship Missile-Integrated Defense System (RAIDS) [14] [13]. Whereas the SWTA had aroused the interest of the researchers first, lately the DWTA had attracted much more attention. The first DWTA was proposed by Hosein and Athans around 1990 [15]. In the proposed approach of Hosein and Athans, a sub-optimal solution was studied in order to determine a solution which was considered as "good enough" [15]. Later they developed exact methods to solve some simplified DWTA [16] [17]. The dynamic programming enables to solve the DWTA in [18], but under the assumption that all the engaged targets are destroyed. Despite its study to decrease the computational time, the problem was still treated in exponential complexity [18]. A more complex DWTA model is designed by Wu *et al.* in [19] where the temporal dimension is included under the form of firing time windows.

The studied DWTA in this chapter slightly differs from the common defined DWTA in the literature. The proposed model has been designed to fit a specific requirement from industrial application. Whereas the classic problem is considering a multi-stage approach, the solved problem considers a continuous time where the targets are evolving in the space according to their own objectives and features. The targets trajectories are designed in using Bezier's curves defined by 4 control points which the last one is set to the centre of area that we are defending. The choice of this trajectory modelling has been done in order to add more diversity in the tested scenarii. The current situation is updated in real time, which means that the proposed algorithm must be as reactive as possible to cope with the oncoming targets. In order to solve the presented problem in the fastest and the most accurate way, a two-step optimisation method is proposed. The first step optimise the assignment of the weapons to the targets, then the optimal firing sequence is obtained in using the results obtained from the first step. The optimal assignment is determined in using the graph theory, and more especially the Hungarian method in a bipartite graph. The used of this method in the first step is motivated by the optimality and the polynomial complexity of the method. Then, the computation of the firing sequence is optimised in using a particle swarm optimisation (PSO) process combined to the evolutionary game theory (EGT). This former method has been proved as efficient in general allocation resources problem [20].

The performance index for the evaluation of the assignment is determined by three different criteria: the capacity to propose an early fire, the width of the firing time window and the minimisation of the overflying of the defended area by our own assets for security purpose. The quality of the firing sequence is obtained from the reactivity of the algorithm to treat the targets in the earliest possible way, the respect of the system constraints and the avoidance of idle time when a firing is possible.

The goal of this chapter is to develop an efficient method to solve a target based DWTA problem involving technical and operational constraints. A mission is considered as achieve only if no targets reach the defending area. The contribution of this paper includes the following aspects:

- Design of a DWTA model taking into account target trajectories and operational and technical constraints on the weapons.
- A two-step approach based on the graph theory, then a combined swarm intelligence and evolutionary game method to solve the DWTA in an optimised fashion.
- The reducing computational load in order to enable real-time applications.
- The targets are following a Bezier's curve trajectory in order to sow the confusion among the defending system.
- The success of one fire is determined by the draw of one random number in $[0, 1]$, then compared to a probability threshold of kill (PK).

The rest of this chapter is organised as follows: the second section describes the details of the studied DWTA, the third section introduces the background of Hungarian algorithm, particle swarm optimisation and evolutionary game theory. Then the fourth section details the proposed method before testing and analysing the obtained results by using a dedicated simulator designed for this DWTA problem. The chapter ends with the conclusion of this study.

2. Background of the proposed approach

2.1. The Hungarian algorithm

The assignment problem arouses the interest of the researchers community for a while. The principle consists of finding a maximum weight matching in a weighted bipartite graph. It is more commonly formulated as: there are two distinct sets, one contain agents, the other one contain tasks. Note that each agent has his own ability to realise a job properly and this capability is represented by a quantitative value. The global objective to assign all the agents to the jobs can be achieved in one optimal way. The Hungarian method published by

Kuhn in 1955 is inspired by the work of two Hungarian researchers: Dénes Kõnig and Jenõ Egervàry [21]. This method has been proved as optimal and polynomial.

Let G be a complete bipartite graph composed, one hand by a set of $|A|$ agents and one other hand by $|T|$ tasks. Then $G = (A, T, E)$, where E denotes the set of the edges linking the set of Agents with the set of Tasks. Note that each edge from E is weighted by a positive cost $c(i, j)$, where $i \in \{1, \ldots, |A|\}$ and $j \in \{1, \ldots, |T|\}$. The function $P : (A \cup T) \longrightarrow \mathbb{R}$ represents the potential if $p(i) + p(j) \leq c(i, j)$ for each $i \in A$ and $j \in T$. The potential value is obtained in summing all the potential from the set $A \cup T$: $p = \sum_{v \in (A \cup T)} p(v)$. The Hungarian method enables to find the perfect matching and the potential equalising the cost and the value, which means that both of them are optimal.

2.2. The particle swarm optimisation

Kennedy and Eberhart [22], the founders of the PSO method, was inspired by the behaviour of animals acting in society to achieve a goal. For example, the birds, the fishes, etc. can make up a very efficient collective intelligence in exchanging very basic information about the environment in which they are evolving. From this starting point, the authors have designed the PSO method to solve many optimisation problems over the last few decades. A swarm is composed of particles (representing a solution) flying on the solution space and communicating with the neighbourhood the quality of the current position.

The first step in PSO algorithm is to define the moving rules on the solution space for the particles. Let $X_i^t = [x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t,]$, $x_{id}^t \in \{0, 1\}$ be a particle in a population of P particles and composed of D dimensions. The velocity of this particle is denoted as $V_i^t = [v_{i1}^t, v_{i2}^t, \ldots, v_{iD}^t,]$, $v_{id}^t \in \mathbb{R}$. Then, as in the PSO method described in [23], the next step is to define the best position for the particle $P_i^t = [p_{i1}^t, p_{i2}^t, \ldots, p_{iD}^t,]$, $p_{id}^t \in \mathbb{R}$, and the best position $P_g^t = [p_{g1}^t, p_{g2}^t, \ldots, p_{gD}^t,]$, $p_{gd}^t \in \mathbb{R}$ of the entire population at the iteration t. The velocity of the particle i is adjusted in respect to the direction d with:

$$v_{id}^{t+1} = \omega_1 v_{id}^t + \omega_2(x(t) - x_{ind}(t)) + \omega_3(x(t) - x_{global}(t)).$$

The parameter ω_1 denotes the weight of the particle inertia. ω_2 is the coefficient associated to the individual coefficient. Then, ω_3 denotes the social coefficient. The final step of one PSO iteration is to update the position of the particles in using the following formula:

$$X_i^{t+1} = X_i^t + V_i^t.$$

This process enables to find an optimal solution in repeating this process. In the classical version of the PSO [24], these coefficients are drawn randomly in order to maximise the exploration of the solution space by the particles. It can be a weakness when the computational time has to be the shortest possible. The studied method proposes to decrease this computational time in using the Evolutionary Game Theory (EGT) to determine the three coefficients ω_1, ω_2 and ω_3. Since the particles are "jumping" on the solution space, the creators wished to limit the jumped distance to a maximum length determined by the value of V_{max} usually determined with respect to the solution space.

http://dx.doi.org/10.5772/53606

2.3. The evolutionary game theory

The evolutionary game theory appeared initially in a biologic context. The need to model the evolution phenomena led to the use of mathematical theory of the games to explain the strategic aspect of the evolution. Over the last few decades, the EGT has aroused interest of the economists, sociologists, social scientists, as well as the philosophers. Although the evolutionary game theory found its origin in biologic science, such an expansion to different fields can be explained by three facts. First of all, the notion of evolution has to be understood as the change of beliefs and norms over time. Secondly, the modelling of strategies change provides a social aspect which matches exactly the social system interactions. Finally, it was important to model dynamically the interactions within a population, which was one of the missing elements of the classic game theory. As in this former domain, the evolutionary game theory deals with the equilibrium which is a key point in both of the theories. Here the equilibrium point is called the evolutionary stable strategy. The principle of the EGT is not only based on the strategy performance obtained by itself, but also the performance obtained in the presence of the others.

2.3.1. Evolutionary Stable Strategies

An Evolutionary Stable Strategy (ESS) is a strategy such that, if all members of a population adopt it, then no mutant strategy could invade the population under the influence of natural selection. Assume we have a mixed population consisting of mostly p^* individuals (agents playing optimal strategy p^*) with a few individuals using strategy p. That is, the strategy distribution in the population is:

$$(1-\varepsilon)p^* + \varepsilon p$$

where $\varepsilon > 0$ is the small frequency of p users in the population. Let the fitness, i.e. payoff of an individual using strategy q in this mixed population, be

$$\pi(q, (1-\varepsilon)p^* + \varepsilon p).$$

Then, an interpretation of Maynard Smith's requirement [25] for p^* to be an ESS is that, for all $p \neq p^*$,

$$\pi(p, (1-\varepsilon)p^* + \varepsilon p) > \pi(p^*, (1-\varepsilon)p^* + \varepsilon p)$$

for all $\varepsilon > 0$ "sufficiently small", for agents minimizing their fitness.

2.3.2. Replicator dynamics

A common way to describe strategy interactions is using matrix games. Matrix games are described using notations as follows. e_i is the i^{th} unit line vector for $i = 1, ..., m$.

$A_{ij} = \pi(e_i, e_j)$ is the $m \times m$ payoff matrix.

$\Delta^m \equiv \{p = (p_1, ..., p_m) \mid p_1 + ... + p_m = 1,\ 0 \leq p_i \leq 1\}$ is the set of mixed strategies (probability distributions over the pure strategies e_i).

Then, $\pi(p, q) = p \cdot Aq^T$ is the payoff of agents playing strategy p facing agents playing strategy q.

Another interpretation is $\pi(p, q)$ being the fitness of a large population of agents playing pure strategies (p describing the agent proportion in each behaviour inside a population) with respect to a large q population.

The replicator equation (RE) is an Ordinary Differential Equation expressing the difference between the fitness of a strategy and the average fitness in the population. Lower payoffs (agents are minimizers) bring faster reproduction in accordance with Darwinian natural selection process.

$$\dot{p}_i = -p_i(e_i \cdot Ap^T - p \cdot Ap^T)$$

RE for $i = 1, ..., m$ describes the evolution of strategy frequencies p_i. Moreover, for every initial strategy distribution $p(0) \in \delta^m$, there is an unique solution $p(t) \in \delta^m$ for all $t \geq 0$ that satisfies the replicator equation. The replicator equation is the most widely used evolutionary dynamics. It was introduced for matrix games by Taylor and Jonker [26].

Note that this introducing to the EGT and the PSO comes from one of our previous study in [20].

3. The formulation of the DWTA: A target-based model

A common approach to the DWTA problem based on the capabilities of the defence system to minimise the probability that a target can leak the proposed engagement plan. However, the problem dealt with in this study is slightly different from the classic DWTA. Whereas the classic DWTA is considering a multi-stage approach, the solved problem considers a continuous time where the targets are evolving in the space according to their own objectives and features. The proposed model has been designed to fit a specific requirement from industrial application, which explains this different approach.

The weapon system is defending an area from oncoming targets. This area is represented by a circle. All the weapons are disposed randomly within this range. In order to make the problem as general as possible, it is assumed that each weapon has its own velocity and own range. The targets are aiming the centre of the area to defend. The trajectories of the targets are designed by Bezier's curves in using 4 control points, all randomly drawn on the space, but the last point which is set to the centre of the area to defend. Thus, the problem presents a high diversity and can test the proposed method in the most of possible tricky cases. It is also assumed that the velocity of the targets and of the weapons are constant.

The assignment and the firing time sequence are computed in real-time in order to validate the reactivity of the studied algorithm. Which means that a timer is set at the beginning of the simulation, and the position of the targets evolves accordingly to this time.

3.1. The engagement plan

The engagement plan represents the solution space. An engagement plan is composed of one assignment weapon/target and completed by a date to fire. For example, if the following situation involves 3 weapons and 2 targets, a possible engagement plan EP could be:

$$EP(t) = \{(W_1, T_2, t + FT_1); (W_3, T_1, t + FT_3)\}$$

Where t denotes the simulation time and the W_i, $i \in \{1,2,3\}$ and T_j, $j \in \{1,2\}$ represent the weapon i and the target j. The variable FT_i denotes the Firing Time computed for the weapon i. The engagement plan evolves accordingly to the situation and depends on the current simulation time and on the aerial situation. In this application, the engagement plan is recomputed every P seconds in order to make up a very reactive engagement plan capable of dealing with the trickier cases in which the targets are constantly changing their trajectories.

3.2. The choice of a two-step optimisation method

Since the complexity of the presented problem grows exponentially with the number of targets and weapons, to design an algorithm capable of handling the real-time computation, but taking into account very diversified performance indexes, the choice of two different steps was natural. Lloyd [1] proved that the DWTA is a NP-complete problem. Therefore, it is hard to find an exact optimisation method capable of solving the DWTA problem in an exact way within a reasonable time. The reasonable time implies a high frequency which can enable the real-time application of the optimisation method. Note that the system must be able to provide results in real-time in the DWTA problem since the engagement changes as the targets keep evolving in the aerial space during the computation. With these reasons, using a heuristic approach providing suboptimal solutions in real-time could be the best way to handle the DWTA problem. One other problem is to be able to quantify the quality of one proposed solution: the performance index of the assignment, and the firing sequence cannot be evaluated in using the same performance criterion. Whereas the assignment is evaluated from the system point of view, the firing sequence is evaluated from the weapons features. Dividing the problem into two parts could lead to the modification of the solution space and the optimum solution could be not the same as the optimal one if the entire solution space was considered. However regarding the real-time computation, and the heterogeneity of the considered criteria, dividing this problem into two steps makes sense in terms of reality and applicability of the designed model, and in terms of quality of the found solution.

3.3. The weapon-target assignment

In order to assign the targets to the available weapons, the Hungarian algorithm is used. The weapons and the targets are modelled as an asymmetric bipartite graph. The Figure 1 shows an example of possible assignment graph used. In the studied problem, it is assumed that the initial number of weapons is greater than the number of oncoming targets.

The quality of the proposed assignment is evaluated according to three different criteria: the capacity to propose an early fire, the width of the firing time window and the minimisation of the overflying of the defended area by our own assets for security purpose. These criteria respectively represent:

- the capability of the system to propose an early firing date, and then its ability to cope with a target in the earliest possible date in order to avoid any risk.
- the width of the firing time windows represents the time that we have to cope with one target, then the larger is this firing time windows, the more time we have to propose one engagement solution,

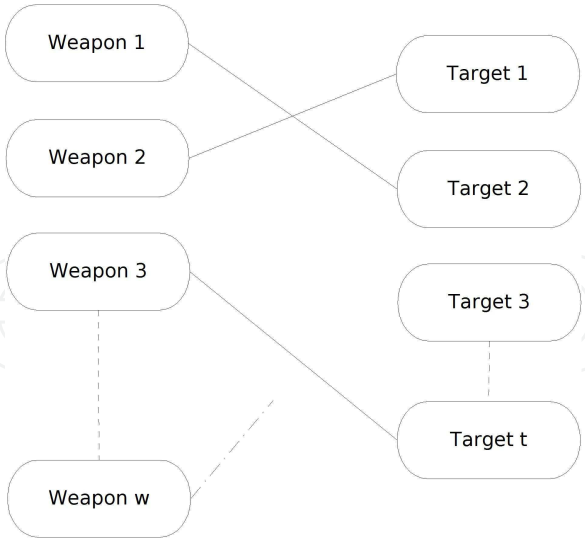

Figure 1. Example of asymmetric bipartite graph with w weapons and t targets

- limiting the overfly in our own area enables to cope with security problem in case of material failure.

3.4. The sequencing of the firing time

As soon as the weapons are assigned to the targets, the sequencing of the firing is computed with respect to the weapons properties (range, velocity) and the firing time windows as well.

In order to evaluate the quality of the proposed solution, the performance index is based on the reactivity of the algorithm, the respect of the system constraints and the avoidance of idle time when a firing is possible.

The system is subject to some technical constraints as a required time between two firing times, which depends on the system. In the designed simulator this time is fixed to 3 seconds.

3.5. Mathematical modelling

This section describes the mathematical modelling of each step followed to achieve the DWTA. The first step is the assignment of the targets to the weapons, and then the sequencing of the firing time to complete in the best possible way the destroying of all the threatening targets. The weapon-target assignment is done by using the graph theory, especially the Hungarian algorithm. The second part is done by integrating two approaches: the PSO and the EGT to make up an efficient real-time oriented algorithm to solve the firing sequence problem.

In the following section, $FTW_{w/t}$ denotes the set of the firing time windows (time windows in which a weapon w can be fired with a given probability to reach the target t). $EFF_{w/t}$ denotes the earliest feasible fire for the weapon w on the target t. The latest feasible fire for the weapon w on the target t is denoted by $LFF_{w/t}$. $E_{w/t}$ represents the edge linking the weapon w with the target t. The average speed of the weapon w is denoted by S_w. R_t and R_w denote the state of the target t (respectively the weapon w). The states are composed of the

http://dx.doi.org/10.5772/53606

(x_t, y_t) position and the speed (v_{t_x}, v_{t_y}) of the target t (respectively (x_w, y_w) position and the speed (v_{w_x}, v_{w_y}) of the weapon w) in the (x, O, y) plan. The entering point of the target t in the capture zone of the weapon w and the entering point of the defended area is computed in the same time as the $FTW_{w/t}$ and they are denoted by $P_{t_{in}}$ and $P_{t_{out}}$. The initial position of the weapon w is denoted by $P_{w_0} = (x_{w_0}, y_{w_0})$.

3.5.1. The assignment part: Hungarian algorithm

Let W be the set of the available weapons and T the set of the oncoming targets. If A represents the assignments linking the vertices W to the targets T. $G = (W, T, A)$ denotes the complete bipartite graph.

The weight of each edge is computed from the linear combination of the three criteria: earliest possible fire, width of the firing time windows and minimising the overfly of the defended area. These criteria are represented as follows:

$$f_1(E_{w/t}) = EFF_{w/t}, (w \in W), (t \in T)$$

As mentioned, $EFF_{w/t}$ denotes the earliest feasible fire for the weapon w on the target t.

$$f_2(E_{w/t}) = LFF_{w/t} - EFF_{w/t}, (w \in W), (t \in T)$$

$EFF_{w/t}$ denotes the earliest feasible fire for the weapon w on the target t. The latest feasible fire for the weapon w on the target t is denoted by $LFF_{w/t}$.

$$f_3(E_{w/t}) = d\left(P_{t_{out}}, P_{w_0}\right)$$

Here the function $d(P_1, P_2)$ represents the Euclidean distance function between the point P_1 and the point P_2. This criterion is shown in the Figure 2.

Then, the global weight of the assignment $E_{w/t}$ is the linear combination of the three functions described above: $H(E_{w/t}) = \alpha_1 f_1(E_{w/t}) + \alpha_2 f_2(E_{w/t}) + \alpha_3 f_3(E_{w/t})$, where $H(E_{w/t})$ denotes the weighting function of the assignment $E_{w/t}$ and $(\alpha_1, \alpha_2, \alpha_3) \in [0,1]^3$, with $\alpha_1 + \alpha_2 + \alpha_3 = 1$.

The cost matrix used for the Hungarian algorithm has the following form:

$$H = \begin{pmatrix} E_{1/1} & E_{2/1} & E_{3/1} & \cdots & E_{|W|/1} \\ E_{1/2} & E_{2/2} & E_{3/2} & \cdots & E_{|W|/2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ E_{1/|T|} & E_{2/|T|} & E_{3/|T|} & \cdots & E_{|W|/|T|} \end{pmatrix}$$

$|T|$ and $|W|$ represent the cardinal of the sets T and W.

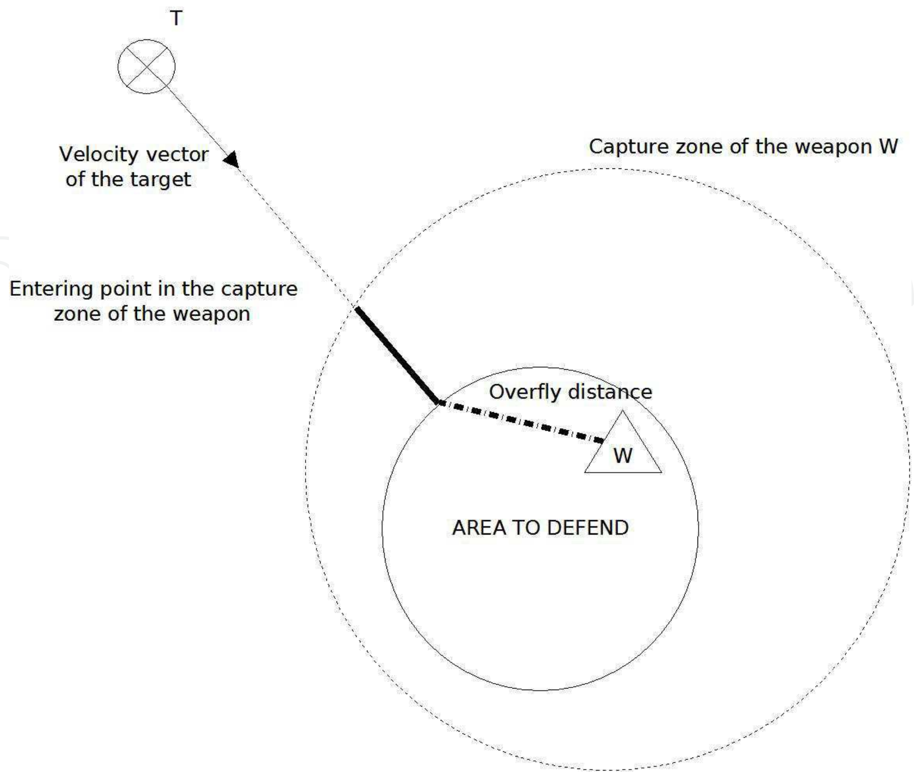

Figure 2. Representation of the overflying criterion. The used value is the Euclidean distance between the entering point of the target in the area to defend and the initial position of the weapon.

3.5.2. The firing time sequencing: EGPSO

As described in the section 2.2, the EGPSO process is based on the combination of the PSO algorithm combined to the EGT in order to increase the convergence speed [27]. In this section, $FS = [FT_i]$, $i = \{1, \ldots, w\}$ denotes a firing sequence for the w selected weapons from the previous assignment and FT_i represents the firing time of the weapon i ($i \leq |W|$). In the proposed model, FS represents one particle composed by the set of the firing times for each weapon. Since the solution space is composed by the firing time windows, it can be very heterogeneous in terms of length along each dimension. In order to avoid an unequal exploration of the solution space, the normalisation over the solution space is operated. Thus, the solution space is reduced to a $[0, 1]^{|W|}$ hypercube and enables a homogeneous exploring by the particles.

In order to evaluate the performance of a proposed solution, the global performance index is based on the reactivity of the algorithm, the respect of the system constraints and the avoidance of idle time when a firing is possible. The global cost function is obtained in multiplying each criterion. The multiplication is selected to consider evenly all the criteria. Thus, if one criterion is not respected by the proposed engagement plan, the cost function will decrease accordingly to the unsatisfied criterion.

The first performance index based on the time delay enables to quantify the reactivity of the system in summing the firing times. The function f_1 enables to express this criterion.

$$f_4(FS) = \sum_{t=1}^{T} FT_t$$

Where, FT denotes the firing time of the weapon assigned to the target t.

The second criterion evaluates the feasibility of the proposed solution to respect the short time delay due to the system constraints. This criterion is based on the presence of constraint violations. When any of the constraints is violated, the proposed solution takes the maximum value in order to avoid infeasible solution.

$$f_5(FS) = \sum_{w=1}^{W} \text{Conflict}(w)$$

The vector Conflict $= [c_i]$, $i = \{1, \ldots, |W|\}$ with $c_i = 1$ if there is a constraint violation by the weapon i, otherwise $c_i = 0$.

The third and last criterion is based on the idle time of the system. This criterion enables to avoid the inactivity of the system if there are possible fires by the current time. In the best case, this value should be reduced to the time constraint multiplied by the number of available weapons.

$$f_6(FS) = \sum_{w=1}^{W-1} (FS_{w+1} - FS_w)$$

Note that the FS vector is sorted before computing this performance index function to the current particle.

When all the criteria are computed, the global performance of the proposed firing sequence is obtained as:

$$F(FS) = \begin{cases} (f_4(FS)+1).f_6(FS) & \text{if } f_5(FS) = 0 \\ +\infty & \text{if } f_5(FS) \neq 0 \end{cases}$$

4. The proposed method

The proposed method is based on the consecutive use of the Hungarian algorithm to solve the assignment problem before determining the fire sequencing using the PSO combined with the EGT.

4.1. A two step-method

As described on the Flowchart 3, the two-step process computes first the optimal assignment of the targets to the weapons, then in a second time the optimal firing sequence is determined.

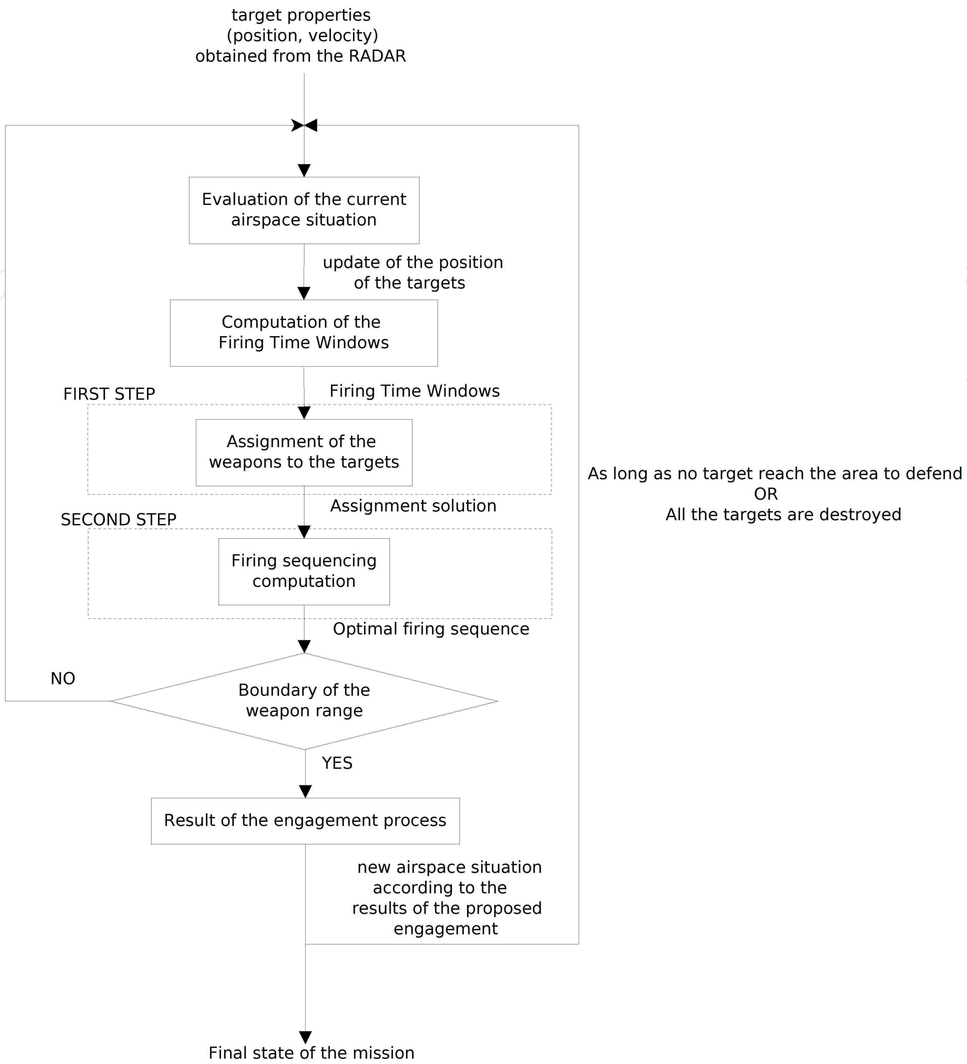

Figure 3. Representation of the two-step method to solve the DWTA.

4.2. The Hungarian algorithm

The assignment of the targets to the weapons is realised in using the Hungarian algorithm [21]. The section 3.5.1 states all the required details enabling to understand the principles of the used method. Since in real scenarii the number of targets is only rarely the same as the number of weapons, the Hungarian algorithm designed for asymmetric bipartite graphs is used. The following parameters are used to determine the best assignment: the cost matrix has a $|T| \times |W|$ form in order to assign all the targets and the coefficients of this cost matrix are determined in using the equations described in 3.5.1.

4.3. The integration of the particle swarm optimisation with the evolutionary game theory

There are two main steps in this approach, the first one is the movement of the swarm in using only, first the inertia, then only the individual component, then only the social component. From the obtained results of the movement of the three swarms, the payoff matrix is composed by the mean fitness of the particles composing each swarm. Let S be the set of the available strategies s_i, $i \in \{1,2,3\}$ which are as follows:

s_1: Use of the pure strategy *inertia*

s_2: Use of the pure strategy *individual*

s_3: Use of the pure strategy *social*

After one iteration using each strategy successively, the payoff matrix consists of the mean value of the swarm. A denotes this payoff matrix:

$$\Pi = \begin{pmatrix} \pi(s_1) & \frac{\pi(s_1)+\pi(s_2)}{2} & \frac{\pi(s_1)+\pi(s_3)}{2} \\ \frac{\pi(s_2)+\pi(s_1)}{2} & \pi(s_2) & \frac{\pi(s_2)+\pi(s_3)}{2} \\ \frac{\pi(s_3)+\pi(s_1)}{2} & \frac{\pi(s_3)+\pi(s_2)}{2} & \pi(s_3) \end{pmatrix}$$

The coefficients $\pi(s_i)$, $i \in \{1,2,3\}$ are the mean value of the swarm after using the pure strategy s_i. The evolutionary game process used to converge to the evolutionary stable strategy is the *replicator dynamic* described in [20]. As soon as the population is stabilised, the proposed algorithm stop running the replicator dynamic. This ESS gives the stable strategy rate, generally composed by a mix of the strategies s_1, s_2, and s_3. Then, the final step uses these rates as coefficients in the PSO algorithm.

The principle of the method is described on the Flowchart 4 and by the following process step by step:

1. Initialisation of the swarm in position and velocity
2. For a maximum number of iterations
   (a) Random selection of particles following the classical PSO process (exploration) and the particles following the EGPSO (increase computational speed).
   (b) Classic iteration of the PSO in using only one strategy for each swarm (inertial, individual, social)
   (c) Computation of the payoff matrix in computing the mean value of the swarm in using the strategies
   (d) Find the evolutionary stable strategy depending on the payoff matrix
   (e) Classic iteration of the PSO using the previously found coefficients
   (f) Check if the swarm is stabilised
      - If YES, restart the swarm like at the step 1
      - If NOT, keep running the algorithm
3. Obtain the optimal solution

In the presented simulation, the PSO parameters are defined as:

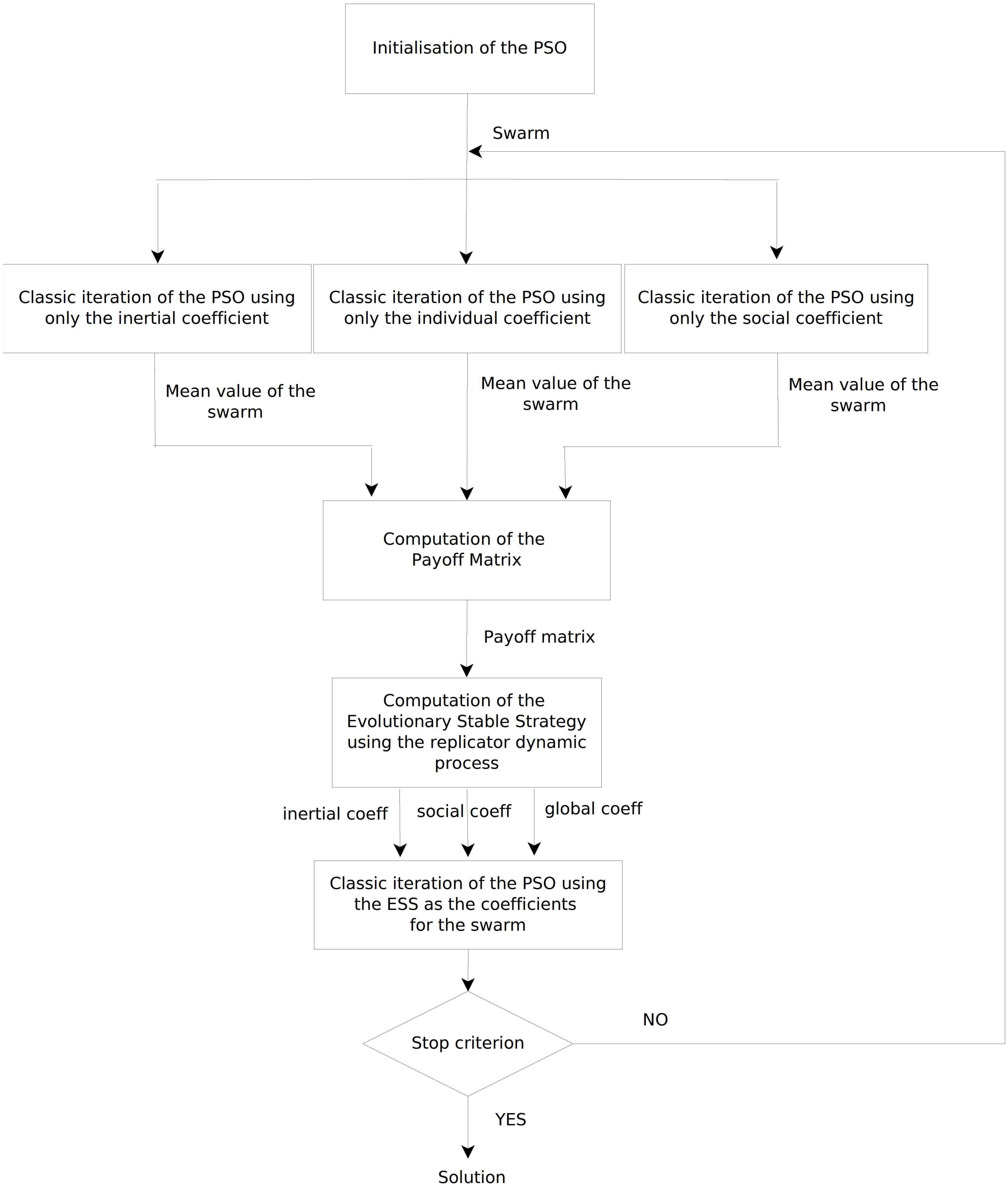

Figure 4. Details on the method designed to mix EGT and PSO.

- 50 particles are used to explore the solution space.
- The maximum distance travelled by one particle in one iteration is limited to 1/10 along each dimension. Notice that since the solution space has been normalised, the maximum velocity enables an homogeneous exploration of the solution space.
- In order to be able to be competitive in real-time, the exit criterion is a defined time of 2500 ms, after that the best found solution is considered as the optimal one.

In order to enable a quick convergence to the optimal vector rate of the PSO coefficients, the EGT process is launched in using as payoff matrix Π described in the section 4.3. The replicator equation is computed over 500 hundred generations, and then the obtained result is considered as

5. Results and comments

In this section, the efficiency of the proposed approach is analysed. After running 100 times a simulation, the number of experiences that the mission is successfully achieved is compared to the number of times it fails. Then, in a second time the evolution of the assignment is studied in analysing the target motions and the proposed engagement plan. The study ends with the analysis of the human operator point of view in order to determine if the proposed algorithm can be reliable and stable for the operator. By stable, it is assumed that the operator can have a global overview of the next engagement to execute in advance, and that this plan won't change if there are no major changes in the aerial situation (suppressed enemy or missing fire for example).

In the presented simulator, the used parameters are set up as follows:

The aerial space:
Square of 50000 m by 50000 m

Weapons
The initial position is within a radius of 3000 m around the central objective
The range of each weapon is randomly drawn between 10000 meters and 15000 meters.

Targets
The initial position is set up between 30000 m and 50000 m from the main objective located on the centre of the space.
The trajectories that the targets are following are modelled in using a Bezier curve defined by 4 control points. The last control point is automatically set as the centre of the space $(0,0)$.
The speed is randomly drawn between 50 m/s and 900 m/s.

The initial conditions:
16 Weapons vs. 12 Targets.

Condition of engagement success:
The success of an engagement one weapon on one target is determined in drawing one random number. If this number is greater than a determined value, then the shoot is considered as a success. Otherwise, it is considered that the target avoids the weapon. In this simulator this value is arbitrary fixed to 0.25, which means that the probability of operating a successful shot is 75%.

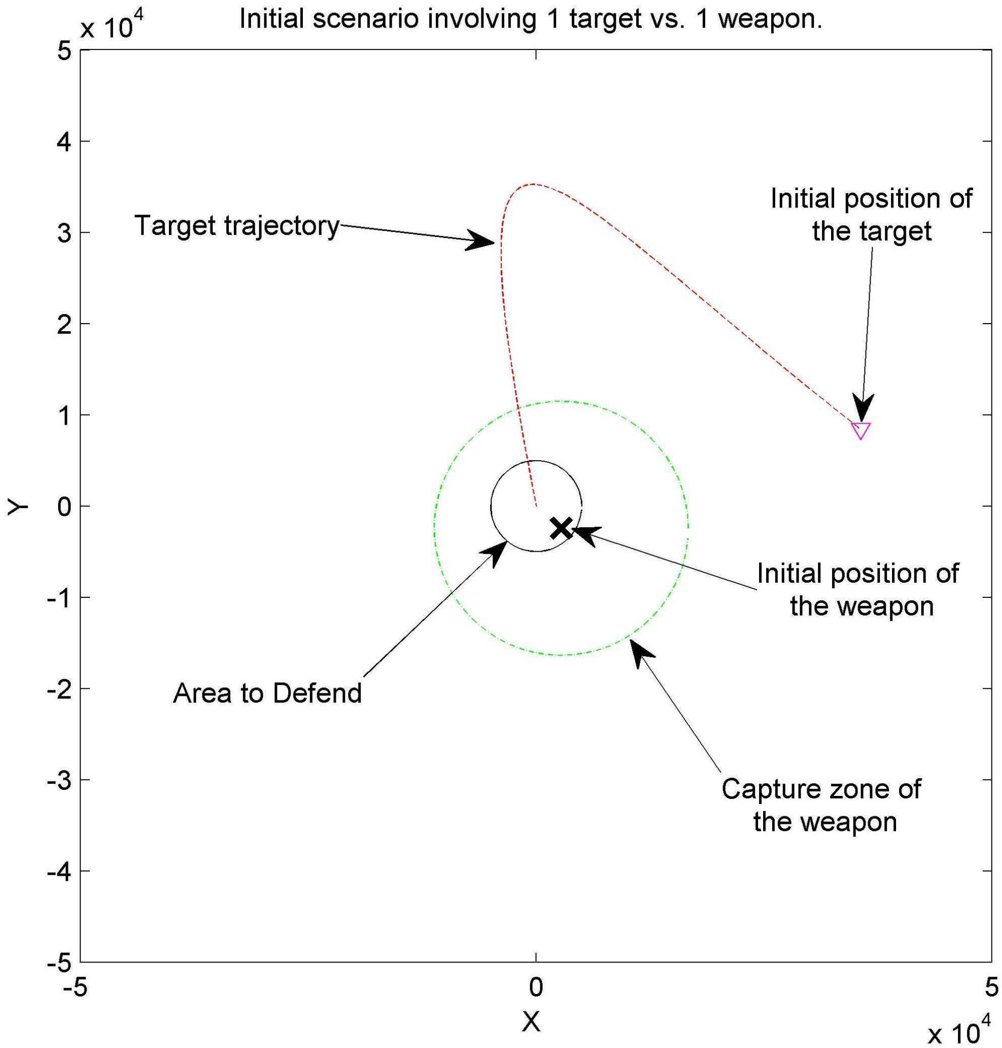

Figure 5. Representation of a possible initialisation of trajectory and weapon position. The triangle marker represents the initial position of the target. The dot line is the trajectory that the target will follow to reach its goal. The continuous line represents the area that we are defending and the cross marker surrounded by a dot line denote the defending weapon and its capture zone.

The Figure 5 shows a possible initialisation of a scenario. Note that if the trajectory is a priori known by the target, the defending side has no information at all but the final point of the target and its current position.

The analysis of the evolution of the assignment of the weapons to the oncoming targets clearly shows stability over the simulation time as long as there are no major change in the scenario. A major change in the scenario can be qualified by the suppression of one enemy which leads to the reconsideration of the entire scenario. Otherwise, the proposed method clearly shows a good stability over the simulation time which is required in the presented case. Considering the presence of a human operator having the final decision making and using this method as a help in the decision making process, it is important for the proposed engagement to be continuous over the time when the aerial situation does not vary dramatically. The upper graph of the Figure 6 displays the assignment of the target t over the number of iterations. The vertical lines identify the instants when a target has

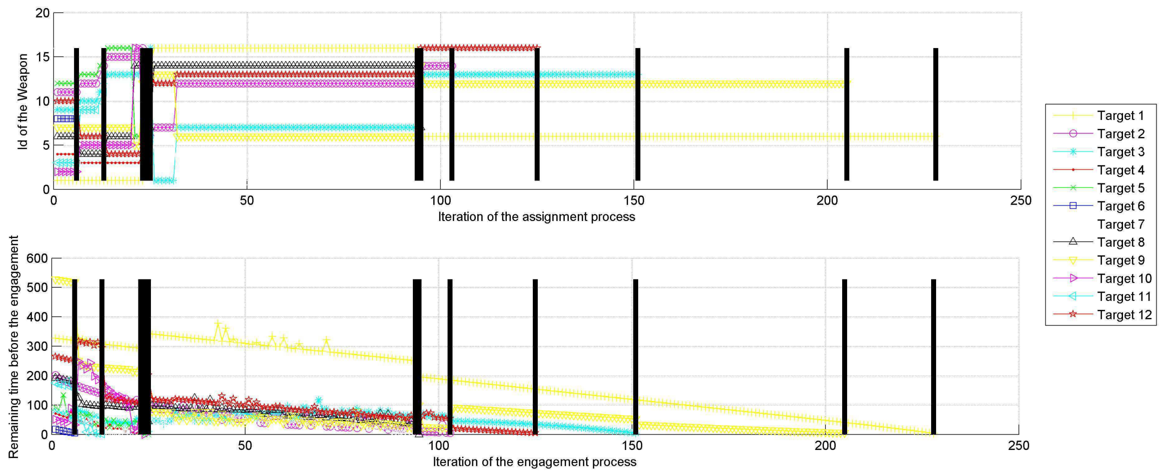

Figure 6. The upper graph illustrates the variation in the assignment process over the time. The regularity of the proposed assignment can be noticed, especially as long as the aerial situation does not change (no target are suppressed). The black vertical lines highlight these phases. The lower graph shows the evolution of the proposed firing time to engage the target over the time.

been killed, then it denotes a change in the aerial situation. During the different highlighted phases, the assignment presents some interesting features as the regularity over the time when the aerial situation keep being similar. The lower graph on the Figure 6 represents the evolution of the firing time for each target over the time. The vertical lines have the same meaning as the upper graph and denotes a change in the aerial situation like, for example, a suppressed enemy or an unsuccessful fire. This second graph highlights the continuity of the proposed firing sequence over the time. It is shown that the operator can not only approve the firing sequence in executing the firing, but the operator can follow the entire scenario and can anticipate the upcoming events. The Figure 7 focuses on the real time aspect in focusing only on the operator point of view. Indeed this Figure represents a zoom on the 25 last seconds before firing the weapons. The horizontal dash line illustrates a time limit of 5 seconds from which the operator can execute the firing.

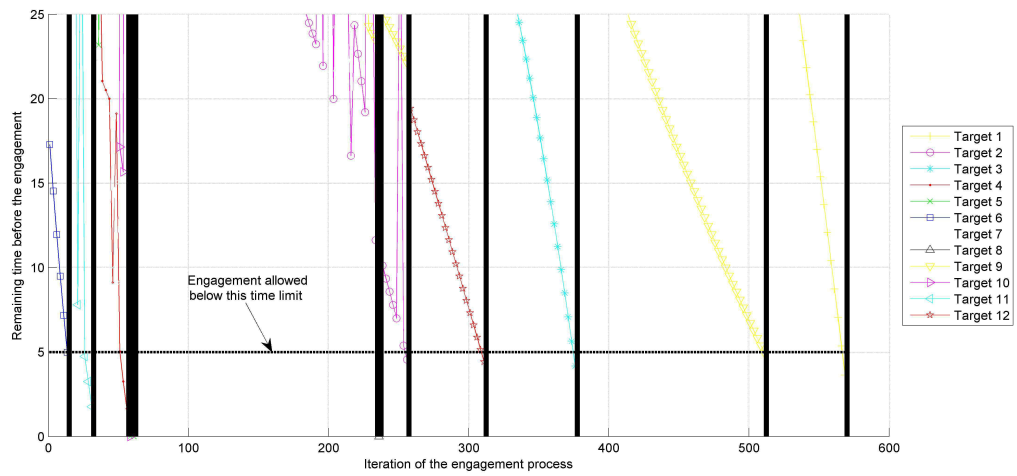

Figure 7. This graph represents a zoom on the final instruction of the operator to execute the firing of the weapon as soon as the proposed firing time is within 5 seconds of the current time. This limit is illustrated by the horizontal dash line.

In order to test the efficiency of the proposed method over different scenario, the designed experience has been launched 100 times and the final result archived. The pie diagram 8 shows the number of times that the proposed method achieved its goal versus the number of time it fails. The analysis of this result shows that the proposed algorithm successfully achieved its mission in 96% of the cases. If we look into details the causes of these failures, we can notice that 3 of the 4 failures was due to the lack of available weapons. Which means that the method does not achieved its goal because of the probability. Indeed, with PK threshold fixed to $PK = 0.90$ and 16 available weapons versus 12 targets, we have an estimate failure rate of approximatively 2 %. This last result comes from the binomial distribution, where the probability of getting exactly T success in W trials is given by:

$$P(T; W, PK) = \frac{W!}{T!(W-T)!} PK^T (1 - PK)^{W-T}$$

Thus, to solve this issue, two possible ways could be explored: first, the increasing of available weapons; second, using more accurate weapons. Although both of the proposed solutions can cope with this issue, it leads to increase the cost of the mission. Controlling this probability enables to optimise the used deployment to protect our area.

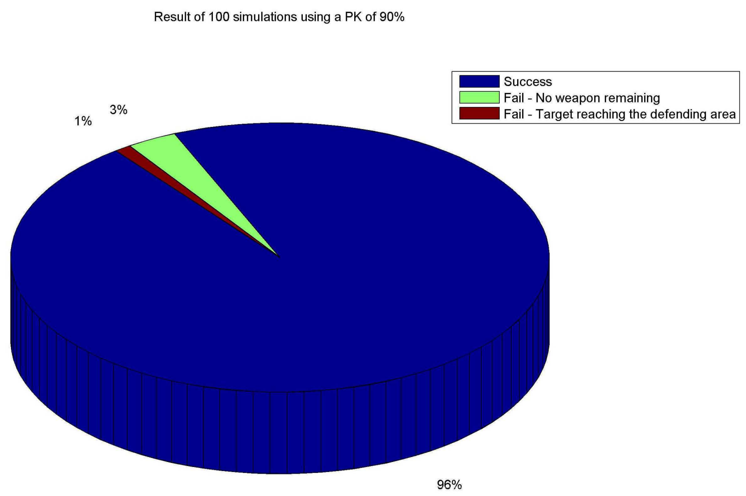

Figure 8. This bar diagram illustrates the number of time that the simulation is a success versus the number of time that it fails.

6. Conclusion

In this chapter, a two-step optimisation method for the DWTA was proposed. Based on the successive use of the Hungarian algorithm, and a PSO combined with the EGT, the proposed algorithm shows reliable results in terms of performance and real-time computation. The proposed method is verified using one simulator designed to create random scenarii and to follow the normal evolution of the battlefield in real-time. The initialised scenario was composed of 16 weapons versus 12 targets. The stability of the assignment and the continuity of the firing sequence was analysed over the launch of 100 simulations. Regarding the probability of successfully achieved the mission, a short study about the binomial distribution has been done and could be helpful in the mission planning process to determine the optimal number of available weapons before the mission. The simulation

http://dx.doi.org/10.5772/53606

results have shown the efficiency of the proposed two-step approach in various cases. The proposed algorithm achieves its objective in 96% for the given scenarii which include random simulation parameters selected for the generality of the senarii. Note that from a probability study on this application, with the chosen simulation parameters, 2% of the scenarii was expected to be failed simply because of the associated probability laws based on a Binomial distribution.

Author details

Cédric Leboucher[1,*], Hyo-Sang Shin[2], Patrick Siarry[3], Rachid Chelouah[4], Stéphane Le Ménec[1] and Antonios Tsourdos[2]

* Address all correspondence to: cedric.leboucher@mbda-systems.com

1 MBDA France, 1 av. Reaumur, Le Plessis Robinson, France
2 Cranfield University, School of Engineering, College Road, Cranfield, Bedford, UK
3 Université Paris-Est Créteil (UPEC), LISSI (EA 3956), Créteil, France
4 L@ris, EISTI, Avenue du Parc, Cergy-Pontoise, France

References

[1] S. P. Lloyd and H. S. Witsenhausen. Weapon allocation is np-complete. In *Proceeding IEEE Summer Simulation Conference*, page 1054 âĂŞ 1058, Reno (USA), 1986.

[2] P. A. Hosein and M. Athans. Preferential defense strategies. part i: The static case. Technical report, MIT Laboratory for Information and Decision Systems with partial support, Cambridge (USA), 1990.

[3] S. Bisht. Hybrid genetic-simulated annealing algorithm for optimal weapon allocation inmultilayer defence scenario. *Defence Sci. J.*, 54(3):395 – 405, 2004.

[4] A. Malhotra and R. K. Jain. Genetic algorithm for optimal weapon allocation in multilayer defence scenario. *Defence Sci. J.*, 51(3):285 – 293, 2001.

[5] O. Karasakal. Air defense missile-target allocation models for a naval task group. *Comput. Oper. Res.*, 35:1759 – 1770, 2008.

[6] F. Johansson and G. Falkman. Sward: System for weapon allocation research & development. *13th Conference on Information Fusion (FUSION)*, 1:1–7, July 2010.

[7] O. Kwon, K. Lee, D. Kang, and S. Park. A branch-and-price algorithm for a targeting problem. *Naval Res. Log.*, 54:732 – 741, 2007.

[8] H. Cai, J. Liu, Y. Chen, and H.Wang. Survey of the research on dynamic weapon-target assignment problem. *J. Syst. Eng. Electron.*, 17(3):559 – 565, 2006.

[9] E. Wacholder. A neural network-based optimization algorithm for the static weapon-target assignment problem. *ORSA J. Comput.*, 1(4):232 – 246, 1989.

[10] K. E. Grant. Optimal resource allocation using genetic algorithms. Technical report, Naval Research Laboratory, Washington (USA), 1993.

[11] H. Lu, H. Zhang, X. Zhang, and R. Han. An improved genetic algorithm for target assignment optimization of naval fleet air defense. In *6th World Cong. Intell. Contr. Autom.*, pages 3401 – 3405, Dalian (China), 2006.

[12] A. C. Cullenbine. A taboo search approach to the weapon assignment model. Master's thesis, Department of Operational Sciences, Air Force Institute of Technology, Hobson Way, WPAFB, OH, 2000.

[13] D. Blodgett, M. Gendreau, F. Guertin, and J. Y. Potvin. A tabu search heuristic for resource management in naval warfare. *J. Heur.*, 9:145 – 169, 2003.

[14] B. Xin, J. Chen, J. Zhang, L. Dou, and Z. Peng. Efficient decision makings for dynamic weapon-target assignment by virtual permutation and tabu search heuristics. *IEEE transaction on systems, man, and cybernetics - Part C: Application and reviews*, 40(6):649 – 662, 2010.

[15] P. A. Hosein and M. Athans. Preferential defense strategies. part ii: The dynamic case. Technical report, MIT Laboratory for Information and Decision Systems with partial support, Cambridge (USA), 1990.

[16] P. A. Hosein, J. T. Walton, and M. Athans. Dynamic weapon-target assignment problems with vulnerable c2 nodes. Technical report, MIT Laboratory for Information and Decision Systems with partial support, Cambridge (USA), 1988.

[17] P. A. Hosein and M. Athans. Some analytical results for the dynamic weapon-target allocation problem. Technical report, MIT Laboratory for Information and Decision Systemswith partial support, 1990.

[18] T. Sikanen. Solving weapon target assignment problem with dynamic programming. Technical report, Mat-2.4108 Independent research projects in applied mathematics, 2008.

[19] L. Wu, C. Xing, F. Lu, and P. Jia. An anytime algorithm applied to dynamic weapon-target allocation problem with decreasing weapons and targets. In *IEEE Congr. Evol. Comput.*, pages 3755 – 3759, Hong Kong (China), 2008.

[20] C. Leboucher, R. Chelouah, P. Siarry, and S. Le Ménec. A swarm intelligence method combined to evolutionary game theory applied to the resources allocation problem. *International Journal of Swarm Intelligence Research*, 3(2):20 – 38, 2012.

[21] H.W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83 – 97, 1955.

[22] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, pages 1942 – 1948, Piscataway (USA), 1995.

[23] M. Clerc. Discrete particle swarm optimization, illustred by the travelling salesman problem. *New Optimization Techniques in Engineering*, 1:219–239, 2004.

[24] J. Kennedy and R.C. Eberhart. A discrete binary version of the particle swarm algorithm. In *The 1997 IEEE International Conference on Systems, Man, and Cybernetics*, volume 5, pages 4104–4108, Orlando (USA), October 1997.

[25] J. Maynard-Smith. *Evolution and the theory of games*. Cambridge University Press, 1982.

[26] P. Taylor and Jonker L. Evolutionary stable strategies and game dynamics. *Mathematical Bioscience*, 40:145–156, 1978.

[27] C. Leboucher, R. Chelouah, P. Siarry, and S. Le Ménec. A swarm intelligence method combined to evolutionary game theory applied to ressources allocation problem. In *International Conference on Swarm Intelligence*, Cergy (France), June 2011.

IntechOpen