

Efficient Heuristic Approaches to the Weapon–Target Assignment Problem

Azad M. Madni*

Intelligent Systems Technology, Inc., Los Angeles, CA 90066

Mircea Andreucut†

University of Calgary, Canada

DOI: 10.2514/1.34254

The weapon–target assignment problem is a fundamental defense application of operations research. The problem consists of optimally assigning a given number of weapons to a set of targets, so that the post-engagement total expected survival value of the targets is minimized. Since the weapon–target assignment problem is known to be nondeterministic polynomial time-complete, there are no exact methods to solve it. This paper presents two innovative heuristic algorithms based on simulated annealing and threshold accepting methods to solve the general weapon–target assignment problem. Our computational results show that by using these algorithms, relatively large instances of the weapon–target assignment problem can be solved near-optimally in a few seconds on a standard personal computer.

I. Introduction

THE weapon–target assignment (WTA) consists of optimally assigning a finite number of weapons to a finite number of targets, so that the total expected survival value of the targets after the engagement is minimized. Research on the WTA problem dates back to the 1950s and 1960s [1–5]. A comprehensive review of the literature on the WTA problem is given in refs [4,6]. Several exact algorithms have been proposed in the literature to solve two special cases of the WTA problem: a) when all the weapons are identical [2] and b) when the targets can be attacked by at most one weapon [7,8]. A major result, obtained by Lloyd and Witsenhausen [9], is that the general WTA problem is nondeterministic polynomial time (NP)-complete. This finding implies that the computation time of any optimal algorithm for this problem will grow exponentially with the size of the problem. Thus, there are no exact methods for the general WTA problem even for small problems. Some of the heuristic algorithms proposed to solve the WTA problem are based on linear integer programming [6,7], nonlinear network flow [6,7], neural networks [10], and genetic algorithms [11–13]. However, in the absence of exact algorithms to solve the WTA problem, no estimates can be made on the quality of solutions produced by such heuristics.

In this paper, we present two heuristic algorithms to solve the WTA problem. The first is based on the simulated annealing (SA) method and the second is based on the threshold accepting (TA) method. We selected these algorithms for study because of the need for scalability and speed for WTA problems. In particular, we were interested in the performance of these algorithms when the number of targets is greater than the number of weapons and when they are not. Our numerical results show that relatively large dimensionality WTA problems can be solved near-optimally

Received 24 August 2007; accepted for publication 18 April 2009. Copyright ©2009 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 1542-9423/09 \$10.00 in correspondence with the CCC.

* President and CEO, Intelligent Systems Technology, Inc., 12122 Victoria Avenue, Los Angeles, CA 90066, Associate Fellow.

† Senior Research Associate, Institute for Biocomplexity and Informatics, University of Calgary, 2500 University Drive NW, Calgary, Alberta, Canada T2N 1N4.

by both these algorithms. In particular, the results show that SA offers slightly better solutions when the numbers of targets are strictly greater than the number of weapons. The solutions are equivalent when the number of targets are less than or equal to the number of weapons.

II. Formulation of the WTA Problem

At the outset, we briefly describe the WTA problem. This problem involves an offense that possesses a large but finite number of missiles that can be launched against the assets of the defense [14,15]. The defense employs a finite number of interceptors (i.e., weapons) to destroy the incoming missiles (threats) and thereby protect its assets. In other words, the incoming missiles are the targets that the interceptors seek to destroy [16]. These incoming missiles are assigned values by the defense. The interceptors (i.e., weapons) can engage at most one attacking missile launched by the offense. The engagement of a target by a weapon is modeled as a stochastic event, with a probability of kill assigned to each weapon–target pair (this is probability that the interceptor weapon will destroy the missile target if assigned to it). The engagement of a weapon–target pair is independent of all other weapons and targets. It is also assumed that a particular target may be engaged by more than one weapon. The objective is to assign weapons to targets so as to minimize the total expected value of the surviving targets. This is an integer optimization problem in that fractional weapon assignments are not allowed. Also, it is a large-scale problem because, in general, the number of weapons and targets is large, which makes enumeration techniques impractical. The parameters and variables of the WTA problem are:

- N , the number of targets (offense).
- M , the number of weapons (defense).
- v_n , the value of targets, $n = 0, \dots, N - 1$.
- p_{nm} , the probability of kill (the probability that weapon m destroys target n if it is assigned to it) $n = 0, \dots, N - 1, m = 0, \dots, M - 1$.
- The decision variable is the $N \times M$ matrix $x = [x_{nm}]$, where

$$x_{nm} = \begin{cases} 1 & \text{if weapon } n \text{ is assigned to target } m \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The probability that target n is not destroyed by weapon m is $(1 - p_{nm})^{x_{nm}}$. Since the engagement of a weapon–target pair is independent of all other weapon–target pairs, the probability that target m survives after all weapons have been fired is given by:

$$f_n(x) = \prod_{m=0}^{M-1} (1 - p_{nm})^{x_{nm}} \quad (2)$$

Thus, the total expected value of the surviving targets is given by:

$$F(x) = \sum_{n=0}^{M-1} v_n f_n(x) \quad (3)$$

Therefore, the WTA problem can be stated as following:

$$\min_x F(x) \text{ subject to } \sum_{n=0}^{N-1} x_{nm} = 1, m = 0, \dots, M - 1 \quad (4)$$

The constraints are due to the fact that each weapon must be assigned to exactly one target. Also, it can easily be shown that the objective function $F(x) : \{0, 1\}^{NM} \rightarrow R$ is convex [17].

III. Maximum Marginal Return for the Related WTA Problem

It is interesting to note that the WTA problem can be solved exactly by assuming a single class of weapons [18]. For this case, the kill probabilities are independent of the weapons, i.e. $p_{nm} = p_n$. This assumption is valid if the

defense has only one type of weapons and all the weapons are located in the same area, such that the geometry and time of intercept are the same for all weapon–target pairs. For this particular case, the maximum marginal return (MMR) is a polynomial time algorithm, guaranteed to find the optimal solution [2].

The MMR algorithm can also be applied to a relaxed version of the WTA problem to find a lower bound for the general WTA problem [19]. Replacing p_{nm} with $q_n = \max_m \{p_{nm}\}$ we can formulate the relaxed WTA (R-WTA) problem as follows:

$$\min_x \sum_{n=0}^{N-1} v_n (1 - q_n)^{x_n} \quad \text{subject to} \quad \sum_{n=0}^{N-1} x_n = M \quad (5)$$

where x_n is the number of weapons assigned to target n . Note that now the objective function is convex and separable, which make the problem tractable for exact MMR algorithm. MMR is a greedy algorithm, in which the weapons are assigned sequentially to the target for which the reduction in the objective cost is maximum. The algorithm terminates when all weapons are assigned. The pseudo-code of the MMR algorithm applied to the R-WTA problem is given in Fig. 1. The algorithm begins by initializing $q_n = \max_m \{p_{nm}\}$, $\sigma_n = v_n q_n$, and $x_n = 0$, for all $n = 0, \dots, N - 1$. The marginal return of adding an additional weapon to the target k is given by

$$\sigma_k = v_k [(1 - q_k)^{x_k} - (1 - q_k)^{x_k+1}] = v_k q_k (1 - q_k)^{x_k} \quad (6)$$

Since initially $x_n = 0$ for all $n = 0, \dots, N - 1$, the initial marginal return is

$$\sigma_n = v_n q_n \quad (7)$$

The MMR is a greedy algorithm in that the weapons are always assigned to the target k with the maximal marginal return.

$$k = \arg \max_x (\sigma_n) \quad (8)$$

This algorithm is repeated until all weapons are assigned to targets.

```

initialize  $p, v$  ;
for  $n = 0 : N - 1$  do
    {
         $q_n \leftarrow \max_m \{p_{nm}\}$  ;
         $\sigma_n \leftarrow v_n q_n$  ;
         $x_n \leftarrow 0$  ;
    }
for  $m = 0 : M - 1$  do
    {
         $k = \arg \max_n (\sigma_n)$  ;
         $x_k \leftarrow x_k + 1$  ;
         $\sigma_k \leftarrow v_k q_k (1 - q_k)^{x_k}$  ;
    }
return  $x$  ;

```

Fig. 1 The MMR algorithm formulated for the R-WTA problem.

IV. SA Method

In this section, we present a sub-optimal solution to the general WTA problem that employs SA, a heuristic method developed in statistical physics. In general, this method has produced extremely good solutions for hard optimization problem instances encountered in statistical physics (e.g., traveling salesmen problem and spin glasses).

SA is a generic, probabilistic meta-algorithm for the global optimization problem, namely locating a good approximation to the global optimum of a given function in a large search space [20–24]. Each step of the SA algorithm replaces the current solution by a random solution, chosen with a probability that depends on the difference between the corresponding function values and on a global parameter θ (called the temperature), which gradually decreases during the process. This dependency is such that the current solution changes almost randomly when θ is large, but trends increasingly downhill as θ goes to zero. The allowance for uphill moves saves the method from getting “stuck” in local minima, the bane of greedier methods. To apply the SA method to a specific problem, one must specify: i) the state space; ii) the candidate solution selection method (which enumerates the candidates for the next state); iii) the annealing schedule; and iv) the probability transition function. It is important to realize that these choices can have a significant impact on the method’s effectiveness.

In the case of the general WTA problem, the state space corresponds to $x = [x_{nm}] \in \{0, 1\}^{NM}$. We begin the computation with a feasible solution

$$x_{nm} = \delta(n, m \bmod(N)) \tag{9}$$

where

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases} \tag{10}$$

is the Dirac function, and $x \bmod(y)$ is the modulo operation. This way every weapon is assigned to exactly one target, which is the requirement of the constraint in the general WTA problem (4). We also consider an integer variable: $\xi_m = m \bmod(N)$, $m = 0, \dots, M - 1$, such that we have

$$x_{nm} = \delta(n, m \bmod(N)) = \delta(n, \xi_m) \tag{11}$$

One can see that ξ_m gives the target index, to which the weapon m is assigned.

Let us denote by x the solution corresponding to the current step of the algorithm. A candidate solution x' is found by randomly selecting a weapon m and randomly choosing another target i for this weapon. The return involved in this transition is

$$\Delta(x, x') = F(x') - F(x) \tag{12}$$

The transition $x \leftarrow x'$ is accepted with a probability given by the probability transition function discussed next.

The initial temperature must be large enough to make the uphill and downhill transition probabilities about the same. The temperature must then decrease so that it is nearly zero, at the end of the allotted time. A popular choice is the exponential schedule, where the temperature decreases by a fixed factor λ at each iteration. If θ_i and θ_f are the initial and, respectively, the final temperature then $\lambda = (\theta_f/\theta_i)^{1/T}$, where T is the final number of iterations. For each iteration $t = 0, \dots, T - 1$, one performs a fixed number of transition steps $s = 0, \dots, S - 1$, such that the equilibrium is reached at that temperature.

The probability of making the transition from the current state x to the candidate new state x' is given by the Boltzmann distribution.

$$P(\Delta(x, x'), \theta) = \begin{cases} 1 & \text{if } \Delta(x, x') \leq 0 \\ \exp(-\Delta(x, x')/\theta) & \text{if } \Delta(x, x') > 0 \end{cases} \tag{13}$$

Let us analyze the return $\Delta(x, x')$ in more detail. The initial total expected value of the surviving targets is given by

$$F(x) = \sum_{\substack{n=0 \\ n \neq i, \xi_m}}^{N-1} v_n \prod_{k=0}^{M-1} (1 - p_{nk})^{x_{nk}} + v_{\xi_m} \prod_{k=0}^{M-1} (1 - p_{\xi_m k})^{x_{\xi_m k}} + v_i \prod_{k=0}^{m-1} (1 - p_{ik})^{x_{ik}} \tag{14}$$

The final total expected value of the surviving targets is given by:

$$F(x') = \sum_{\substack{n=0 \\ n \neq i, \xi_m}}^{N-1} v_n \prod_{k=0}^{M-1} (1 - p_{nk})^{x_{nk}} + v_{\xi_m} \prod_{k=0}^{M-1} (1 - p_{\xi_m k})^{x_{\xi_m k}} (1 - p_{\xi_m m})^{x_{im} - x_{\xi_m m}} + v_i \prod_{k=0}^{m-1} (1 - p_{ik})^{x_{ik}} (1 - p_{im})^{x_{\xi_m m} - x_{im}} \quad (15)$$

Therefore, we have

$$\Delta(x, x') = v_{\xi_m} [(1 - p_{\xi_m m})^{x_{im} - x_{\xi_m m}} - 1] \prod_{k=0}^{M-1} (1 - p_{\xi_m k})^{x_{\xi_m k}} + v_i [(1 - p_{im})^{x_{\xi_m m} - x_{im}} - 1] \prod_{k=0}^{m-1} (1 - p_{ik})^{x_{ik}} \quad (16)$$

```

initialize  $\theta_i, \theta_f, T, p, v$  ;
 $\theta \leftarrow \theta_i$  ;  $\lambda \leftarrow (\theta_f / \theta_i)^{1/T}$  ;
for  $n = 0 : N - 1$  do
{
  for  $m = 0 : M - 1$  do
  {
     $\xi_m = m \bmod(N)$  ;
     $x_{nm} = \mathcal{D}(n, \xi_m)$  ;
  }
}
for  $t = 0 : T - 1$  do
{
  for  $s = 0 : S - 1$  do
  {
     $m \leftarrow [rnd(M)]$  ;
     $i \leftarrow [rnd(N)]$  ;
     $\Delta \leftarrow \Delta(x, x')$  ;
     $P \leftarrow P(\Delta, \theta)$  ;
    if  $rnd(1) \leq P$  then
    {
       $\alpha \leftarrow x_{i,m}$  ;
       $x_{i,m} \leftarrow x_{\xi_m,m}$  ;
       $x_{\xi_m,m} \leftarrow \alpha$  ;
       $\xi_m \leftarrow i$  ;
    }
  }
   $\theta \leftarrow \lambda \theta$  ;
}
return  $x$  .

```

Fig. 2 The SA algorithm formulated for the WTA problem.

This expression is more efficient to calculate because it uses only the two involved targets, ξ_m and i . Thus, if the transition is successful, then the target for weapon m is updated as $\xi_m \leftarrow i$. Also, the values of the involved x elements are exchanged: $\alpha \leftarrow x_{i,m}$, $x_{i,m} \leftarrow x_{\xi_m,m}$, $x_{\xi_m,m} \leftarrow \alpha$. The pseudo-code of our SA method is given in Fig. 2.

Here, the function $rnd(x)$ returns a random number, uniform distributed in the interval $[0,x]$. Also, $[x]$ means the integer part of x .

V. TA Method

The TA method is a deterministic analog of the SA method [25,26]. It is a refined search procedure which escapes local minima by accepting solutions which are not worse by more than a given threshold. The algorithm is deterministic in the sense that the number of iterations is fixed and the neighborhood is explored with a fixed number of steps during each iteration. Analogous to the SA method, the threshold is decreased successively until it reaches the value of zero in the last iteration. The main difference between SA and TA is that for SA the threshold is modeled

```

initialize  $\theta_i, \theta_f, T, p, v$ ;
 $\theta \leftarrow \theta_i$ ;  $\lambda \leftarrow (\theta_f / \theta_i)^{1/T}$ ;
for  $n = 0 : N - 1$  do
{
  for  $m = 0 : M - 1$  do
  {
     $\xi_m = m \bmod(N)$ ;
     $x_{nm} = \delta(n, \xi_m)$ ;
  }
}
for  $t = 0 : T - 1$  do
{
  for  $s = 0 : S - 1$  do
  {
     $m \leftarrow [rnd(M)]$ ;
     $i \leftarrow [rnd(N)]$ ;
     $\Delta \leftarrow \Delta(x, x')$ ;
    if  $\Delta \leq \theta$  then
    {
       $\alpha \leftarrow x_{i,m}$ ;
       $x_{i,m} \leftarrow x_{\xi_m,m}$ ;
       $x_{\xi_m,m} \leftarrow \alpha$ ;
       $\xi_m \leftarrow i$ ;
    }
  }
   $\theta \leftarrow \lambda\theta$ ;
}
return  $x$ .

```

Fig. 3 The TA algorithm formulated for the WTA problem.

as a random variable, whereas for TA the threshold is deterministic. The TA algorithm has three advantages: 1) it is easy to parameterize; 2) it is robust to changes in problem characteristics; and 3) it works well for many hard problem instances. An extensive introduction to the TA method is given by Winker [26]. For the case of the WTA problem, we consider that the threshold corresponds to the temperature defined for the SA method. Obviously, this choice satisfies the requirements since $\theta \rightarrow \theta_f \approx 0$. The pseudo-code of our TA method is given in Fig. 3.

VI. Experimentation and Numerical Results

A. R-WTA Case

In the first experiment, we considered the R-WTA problem. The R-WTA instances were obtained by first generating v_n and q_n as uniform distributed random variables on the interval $[0, 1]$, and then setting $p_{nm} = q_n$, for all $n = 0, \dots, N - 1$ and $m = 0, \dots, M - 1$.

The goal of this experiment was to compare the MMR, SA, and TA methods. Since the MMR algorithm gives an exact solution to the R-WTA problem, we were interested in determining whether or not the results calculated with the SA and TA methods converged to the optimum assignment found by the MMR algorithm. To this end, we carried out extensive simulations for $25 \leq N, M \leq 200$ using the following parameters: $T = 100, S = 100, \theta_i = 1$, and $\theta_f = 10^{-5}$. It is important to note that the SA and TA results converged to the optimum assignment (calculated with the MMR method) for all considered R-WTA instances. In Fig. 4, we present three such examples of the results from numerical simulations.

One can see from Fig. 4 that in about 50 steps, both the SA and TA algorithms converge to the optimum assignment found by the MMR algorithm (the black base line on the figures). As a measure of convergence, we considered the time dependence of F , which is the total expected value of the surviving targets at time step t . These results show that the SA and TA methods are accurate and appropriate for moderate size instances of R-WTA problems. Obviously, the MMR method is faster for large instances $N, M > 200$, since its complexity is only $O(N + M \log N)$.

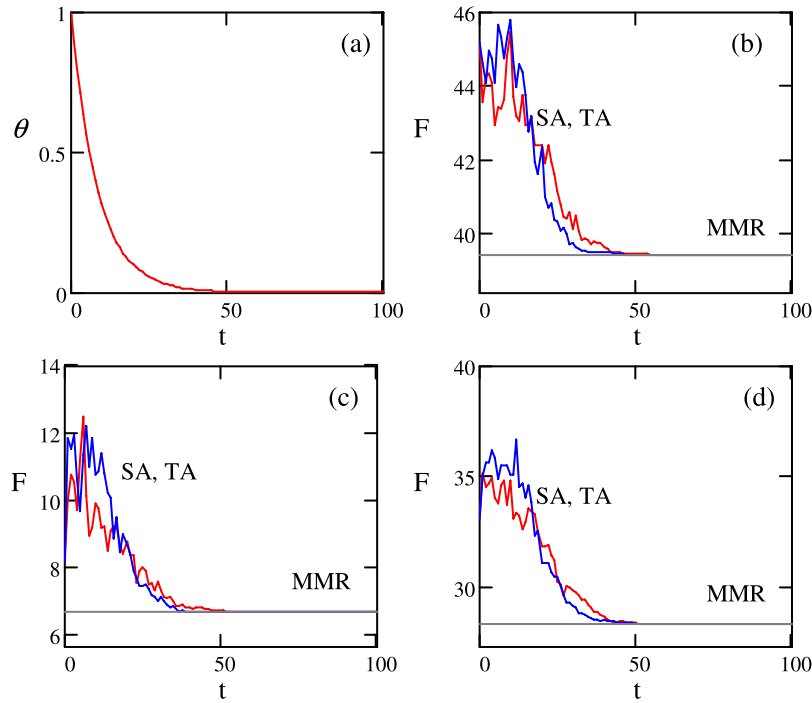


Fig. 4 Simulation results for the R-WTA problem: a) The annealing schedule used for the SA and TA algorithms; b) $N = 100, M = 50$; c) $N = 50, M = 100$; d) $N = 100, M = 100$; [SA, red; TA, blue; MMR, black (optimum assignment)].

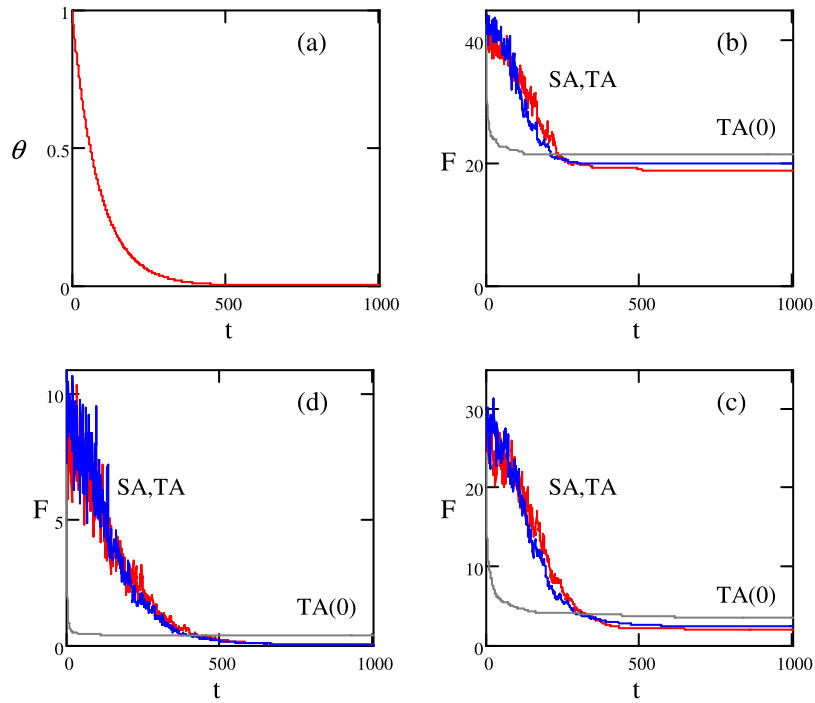


Fig. 5 Simulation results for the WTA problem: a) The annealing schedule used for the SA and TA algorithms; b) $N = 100, M = 50$; c) $N = 50, M = 50, M = 100$; d) $N = 100, M = 100$; SA, red; TA, blue; TA(0), black.

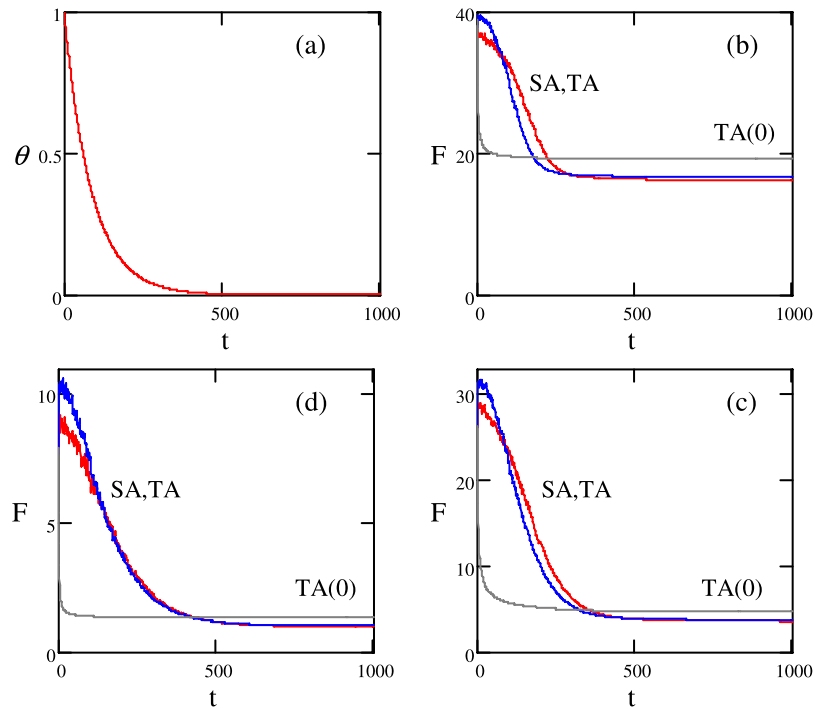


Fig. 6 Simulation results averaged over 100 general WTA instances: a) The annealing schedule used for the SA and TA algorithms; b) $N = 100, M = 50$; c) $N = 50, M = 100$; d) $N = 100, M = 100$; SA, red; TA, blue; TA(0), black.

B. WTA Case

Since the MMR algorithm cannot be applied to the more general WTA instances, we are interested only in comparing the results provided by the SA and TA algorithms. For the same reason, we also consider an extreme case of the TA algorithm, where the threshold is kept constant all the time at $\theta = 0$. We denote this algorithm as TA(0). As before, the target values v_n and the probabilities p_{nm} were generated by uniform random variables on the interval $[0,1]$. Representative examples of the simulations are presented in Fig. 5.

It is interesting to note that the greedy TA(0) algorithm converges very rapidly to a pretty good local minimum. In contrast, both SA and TA algorithms converge more slowly; however, their solution is superior to the one calculated by TA(0). As such, TA(0) can be regarded as a fast alternative to SA and TA, when there is a high demand for speed. Nevertheless, for the investigated range, $25 \leq N, M \leq 200$, both SA and TA methods can solve a general WTA instance in only a few seconds on a standard PC. For the instance solved in Fig. 4, one can see that SA gives a slightly better solution than TA. To elucidate which method is better, we averaged the results over 100 general WTA instances. The results obtained are presented in Fig. 6. One can see that indeed SA offers a slightly better solution when $N > M$, however, SA and TA solutions become equivalent for $N \leq M$.

VII. Conclusions

The general WTA problem is known to be NP-complete and, therefore, there are no exact methods to solve it [16]. In this paper, we presented two heuristic algorithms to solve the WTA problem. These algorithms are based on the SA and the TA methods, which were originally developed for solving hard combinatorial optimization problems frequently encountered in statistical physics. Our numerical results have shown that relatively large size instances ($25 \leq N, M \leq 200$) of the general WTA problem can be solved near-optimally by these algorithms, in only a few seconds on a standard PC. Furthermore, these algorithms have the added advantage that they can be easily parallelized. When such parallelization is possible, then parallel computing is the preferred form of competition. (The specific hardware architecture used to support parallelism could be multi-core or multi-processor computers having multiple processing elements within a single machine.) In other words, the processing elements can be diverse and include resources such as a single computer with multiple processors, several networked computers, specialized hardware, or any combination of these. Assuming that there are K processors available, one can run in parallel the same WTA instance on each processor $k = 1, \dots, K$. The best solution is given by $x^* = \min\{x^{(1)}, \dots, x^{(k)}\}$, where $x^{(k)}$ is the local minimum of the results computed by each processor. Thus, the performance of these methods increases by simply adding more processors. Furthermore, the complexity of the algorithm is quite small, requiring only successive calculations of $\Delta(x, x')$ for a fixed number of time steps T . This reduced complexity, coupled with large scale parallelization, provides extremely good solutions for large instances of the WTA problem.

References

- [1] Manne, A. S., "A Target Assignment Problem," *Operations Research*, Vol. 6, 1958, pp. 346–351.
doi: [10.1287/opre.6.3.346](https://doi.org/10.1287/opre.6.3.346)
- [2] denBroeder, G. G., Jr., Ellison, R. E., and Emerling, L., "On Optimal Target Assignments," *Operations Research*, Vol. 7, 1959, pp. 322–326.
doi: [10.1287/opre.7.3.322](https://doi.org/10.1287/opre.7.3.322)
- [3] Day, R. H., "Allocating Weapons to Target Complexes by Means of Nonlinear Programming," *Operations Research*, Vol. 14, 1966, pp. 992–1013.
doi: [10.1287/opre.14.6.992](https://doi.org/10.1287/opre.14.6.992)
- [4] Matlin, S. M., "A Review of the Literature on the Missile-Allocation Problem," *Operations Research*, Vol. 18, 1970, pp. 334–373.
doi: [10.1287/opre.18.2.334](https://doi.org/10.1287/opre.18.2.334)
- [5] Cai H., Liu, J., and Wang, H., "Survey of the Research on Dynamic Weapon–Target Assignment Problem," *Journal of Systems Engineering and Electronics*, Vol. 17, No. 3, 2006, pp. 559–565.
doi: [10.1016/S1004-4132\(06\)60097-2](https://doi.org/10.1016/S1004-4132(06)60097-2)
- [6] Murphy, R. A., "Target-based Weapon–Target Assignment Problems," in *Nonlinear Assignment Problems: Algorithms and Applications*, edited by P. M. Pardalos and L. S. Pitsoulis, Vol. 7, Kluwer Academic Publishers, The Netherlands, 1999, pp. 39–53.
- [7] Rockafellar, R. T., *Network Flows on Monotropic Optimization*, Wiley, New York, NY, 1984.

- [8] Papadimitriou, C. H., and Steiglitz, K., *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, New Jersey, NJ, 1982.
- [9] Lloyd, S., and Witsenhausen, H., “Weapons Allocation is NP-Complete,” *IEEE Summer Simulation Conference*, Reno, NV, 1986.
- [10] Wachholder, E., “A Neural Network-based Optimization Algorithm for the Static Weapon–Target Assignment Problem,” *ORSA Journal on Computing*, Vol. 4, 1989, pp. 232–246.
- [11] Owechko, Y., and Shams, S., “Comparison of Neural Network and Genetic Algorithms for a Resource Allocation Problem,” *IEEE World Congress on Computational Intelligence*, Vol. 7, 1994, pp. 4655–4660.
- [12] Lee, Z.-J., Su S.-F., and Lee, C.-Y., “Efficiently Solving General Weapon–Target Assignment Problem by Genetic Algorithms with Greedy Eugenics,” *IEEE Transactions on Systems Man and Cybernetics B*, Vol. 33, No. 1, 2003, pp. 113–121.
- [13] Khosla, D., “Hybrid Genetic Approach for the Dynamic Weapon–Target Allocation Problem,” *SPIE Proceeding Series: Battlespace Digitalization and Network-Centric Warfare*, Vol. 4396, 2001, pp. 244–259.
- [14] Arslan, G., Marden, J. R., and Shamma, J. S., “Autonomous Vehicle-Target Assignment: A Game Theoretical Formulation,” *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 129, No. 5, 2007, pp. 584–596.
doi: [10.1115/1.2766722](https://doi.org/10.1115/1.2766722)
- [15] Tavana, M., Bailey, M. D., and Busch, T. E., “A Multi-Criteria Vehicle–Target Allocation Assessment Model for Network-Centric Joint Air Operations,” *International Journal of Operations Research*, Vol. 3, No. 3, 2008, pp. 235–254.
doi: [10.1504/IJOR.2008.017531](https://doi.org/10.1504/IJOR.2008.017531)
- [16] Madni, A. M., Madni, C. C., and Cruz, J., “Game2Win™: Game-based Optimization of BMDS Assets,” Intelligent Systems Technology, Inc., Phase I Final Report, ISTI-FR-599-08/08, Contract # HQ0006-08-C-7837, Aug. 12, 2008.
- [17] Hosein, P., “A Class of Dynamic NonLinear Resource Allocation Problems,” PhD Thesis, Massachusetts Institute of Technology, MA, 1989.
- [18] Roux, J. N., and van Vuuren, J. H., “Threat Evaluation and Weapon Assignment Decision Support: A Review of the State of the Art,” *ORiON*, Vol. 23, No. 2, 2007, pp. 151–186.
- [19] Ahuja R. K., Kumar A., and Jha, K. C., “Exact and Heuristic Algorithms for the Weapon–Target Assignment Problem,” *Massachusetts Institute of Technology (MIT), Sloan School of Management*, Vol. 55, No. 6, 2006, pp. 1136–1146.
- [20] Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P., “Optimization by Simulated Annealing,” *Science*, Vol. 220, 1983, p. 671.
doi: [10.1126/science.220.4598.671](https://doi.org/10.1126/science.220.4598.671)
- [21] Reeves, C. R. (ed.), *Modern Heuristic Techniques for Combinatorial Problems*, McGraw-Hill, London, 1995.
- [22] Andrecut, M., and Ali, M. K., “Critical Transition in the Constrained Traveling Salesman Problem,” *Physical Review E*, Vol. 63 (4 Pt 2), 2001, 047103, 3 pages.
doi: [10.1103/PhysRevE.63.047103](https://doi.org/10.1103/PhysRevE.63.047103)
- [23] Andrecut, M., and Ali, M. K., “Finite Size Scaling and Critical Transition in Constrained Traveling Salesman Problem,” *Modern Physics Letters B*, Vol. 14, No. 24, 2000, pp. 859–870.
doi: [10.1142/S0217984900001105](https://doi.org/10.1142/S0217984900001105)
- [24] Andrecut, M., Este, R. A., and Kauffman, S. A., “Competitive Optimization of Compressed Sensing,” *Journal of Physics A: Mathematical and Theoretical*, Vol. 40, No. 16, 2007, pp. F299–F305.
doi: [10.1088/1751-8113/40/16/F02](https://doi.org/10.1088/1751-8113/40/16/F02)
- [25] Dueck, G., and Winker, P., “New Concepts and Algorithms for Portfolio Choice,” *Applied Stochastic Models and Data Analysis*, Vol. 8, No. 3, 1992, pp. 159–178.
doi: [10.1002/asm.3150080306](https://doi.org/10.1002/asm.3150080306)
- [26] Winker, P., *Optimization Heuristics in Econometrics: Applications of Threshold Accepting*, Wiley, New York, NY, 2000.

Michael Hinchey
Associate Editor