

Research Article

A New Solution Approach for Maritime Surveillance Operation: The Case of Aegean Sea

Selçuk K. İşleyen,¹ Ukbe Uçar ,² and Figen Balo²

¹Department of Industrial Engineering, Gazi University, Ankara 06000, Turkey

²Department of Industrial Engineering, Firat University, Elazığ 23200, Turkey

Correspondence should be addressed to Ukbe Uçar; uuucar@firat.edu.tr

Received 16 February 2019; Accepted 18 June 2019; Published 16 July 2019

Academic Editor: Thomas Hanne

Copyright © 2019 Selçuk K. İşleyen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, many people illegally migrate to other countries by sea because of warfare or internal conflict in their countries, and thousands of people lose their lives for this reason. Besides, illegal trafficking of drugs or historical artifacts, illegal fishing, and various terrorist activities are conducted through the seas. Maritime surveillance operations are of great importance to prevent all these threats and maintain national security. In this study, the problem of surveillance and exploration which is performed by helicopters on the seas is discussed. A solution approach based on simulated annealing is proposed for the solution of the problem and the proposed algorithm has been tested on various scenarios. This methodology, at least to our knowledge, is used for the first time in the Moving Target Traveling Salesman Problem-Time Window. The proposed solution approach was tested on 50 different scenarios where the target number ranged from 20 to 100. As a result, in all of these scenarios, it has been determined that all of the targets have been destroyed with minimum operation time within the acceptable solution period. Thanks to this method, it is aimed at intervening quickly to illegal activities on the seas and contributing to the prevention of deaths due to refugee boats.

1. Introduction

Seas and oceans are home to many important activities, mainly trade and tourism. International passenger and freight transport, fishing, and various military shipments are carried out through these waters. Maritime surveillance operations are of great importance to ensure that all activities on the seas or oceans are carried out safely and that national and international security is maintained without any problems.

Today, one of the most important problems in the seas is undoubtedly the refugee ships. People are forced to leave their countries for a variety of reasons and migrate illegally to other countries to establish new lives. The seaway is one of the main ways used for this purpose and thousands of people die by drowning in the seas every year.

In Figure 1, the news about the illegal migrant ship trying to reach Italy from Libya in 2015 is shown. According to this news, the ship sank off the coast of Mediterranean and 400 of the 550 passengers lost their lives. Another sad part of this news is that the majority of people who died were children

[1]. Besides, the seas are one of the ways that people who are sought for various crimes often use illegally to escape to other countries. Another important problem in the seas or oceans is the sea pirates. Every year hundreds of ships are looted by sea pirates; loss of lives may occur at times. In Figure 2, the news about the attacks and the robbery of refugees who cross from Turkey to Greece by the armed pirates is shown [2]. In addition, illegal fishing and infiltration into a country through the seas are two of the main problems encountered in the seas. Elimination of stated problems and fast intervention in enemies or threats is only possible with the right operation and surveillance strategy. This study discusses the problem of fast surveillance or neutralization of moving elements in the seas. The problem is addressed in the literature under the heading of maritime surveillance problem, which is investigated in relation to the MTTSP-TW (Moving Target Traveling Salesman Problem with Time Window) due to the movement of threat elements and the remainder of the targets in the system over a certain period of time.



FIGURE 1: A refugee ship sinking in the sea [1]. Figure 1 was reproduced from the news with dated 14.04.2015 of Milliyet Newspaper which is a Turkish newspaper (2015).



FIGURE 2: Aegean Sea and a pirate attack [2]. Figure 2 is reproduced from the news with dated 14.08.2015 of Sputniknews website (2015).

The MTTSP problem was first described by Helvig et al., and the problem has many fields of application, including military operations and maritime activities [34]. In the problem, $S = \{s_1, s_2, \dots, s_n\}$ constitutes the set of targets, each s_i target having a constant v_i velocity and p_i initial position. In the system, there is a pursuer starting from the initial point and the pursuer ($v > |v_i|$), a higher velocity than all targets. In the problem, the main purpose is to define a minimum tour route in which the pursuer will start from the original point, destroy all targets, and then return to the original point. The second largest application area under MTTSP is maritime surveillance and exploration activities. Scanning and tracking of ships in the seas and oceans routing security ships or aircrafts and routing of fishing vessels depending on fish density are some of the fields of application of this problem. There are many studies in the literature for MTTSP and Maritime Surveillance Problems. Groba et al. studied the fishing activity on the river Danube in their work and tried to determine how the fish harvesters on the river will be directed and when fishing nets will be collected. They proposed a solution approach based on estimation techniques (GATP) for the solution of problem and tested the efficiency of algorithm by comparing it with two different solution approaches (GA-GSP; NN). As a result, they suggested that the proposed algorithm produces better results than the other two algorithms [35]. Mercer et al. discussed the problem of maritime surveillance operations as part of Australian national security and tried to determine the scan route by Genetic Algorithm that will minimize the total lane length with a marine surveillance aircraft having a constant v velocity, θ_i initial position, r_c turn radius, and r_v scan radius [36]. Killby et al. examined the maritime surveillance problem of

an aircraft and a number of ships where velocity-direction-initial position information of ships and aircraft is known and tried to determine by Generic Algorithm by the tour route that will minimize the number of unscanned ships and total flight time within a mission field related to the solution of problem. The study is different from others in the literature as moving and fixed targets were used, the data changed dynamically, and features such as time window, priority constraints, intervention point, and demand estimation were taken into consideration [37]. Marlov et al. discussed the maritime surveillance operation that is performed by an aircraft moving with a constant speed within a limited area of mission, and developed NN technique, “2-opt”, “2-opt stationary ships”, and “2-opt jumping ships” strategies for the solution of problem. As a result of experimental trials, they found that the heuristics “2-opt” and “2-opt stationary” produce better results than other two heuristics [38]. Fang et al. discussed the problem of protection of moving marine targets by mobile security forces with an optimal strategy. They analyzed the problem for cases with single or multiple number of targets on one-dimensional and two-dimensional planes. They developed the CASS algorithm for the solution of problem and tried to determine the appropriate protection strategy where the least number of targets will be damaged [39]. Cross et al. discussed the maritime surveillance problem in Australia’s territorial waters and tried to scan and classify the vessels in relevant field of mission. In the problem, they assumed that the field of mission is limited and there are ships moving at random speeds and directions and tried to determine a flight route where all ships are scanned and classified by a constant speed maritime surveillance aircraft within a minimum travel time [40]. Ucar and Isleyen discussed the MTTSP-TW under the destruction of moving targets by an air vehicle within minimum times and tried to solve the problem with simulated annealing algorithm and tested the proposed solution approach on theoretical data and 12 scenarios with number of targets varying between 5 and 15. As a result, they suggested that all targets were destroyed in reasonable times within specified time window [41]. In addition, some of the studies in the literature related to the subject are shown in Table 1.

In this study, Maritime Surveillance Problem which is one of the application areas of the MTTSP is discussed and the problem has been tried to be solved under the time window constraints. In this problem, the number of targets and the positions of the targets change continuously over time, and the coordinates for the pursuer and targets in each unit time need to be redefined. Under this dynamic information and without exceeding the time window constraint, it is necessary to recalculate the targets, at what time, on which coordinate axis, in which order it will be destroyed. Under these requirements, the problem is very difficult to solve and the problem has a dynamic structure because the determination of the best solution from a large number of possible solutions related to order of destruction of targets. For the solution of problem, a solution approach based on simulated annealing was developed and it has been tried to determine the tour route with minimum mission time that all threat elements are destroyed within short solution

TABLE 1: Some of the studies on MTTSP literature.

Author	Year	Application Field	Solution Method	Index
Fügenschuh ve ark	2014	Military Operations	Mathematical Modelling	[7]
Stieber ve ark	2015	Military Operations	Mathematical Modelling, Heuristic Algorithm	[8]
Stieber ve Fugenschuh	2016	Military Operations	Mathematical Modelling	[9]
Jiang ve ark.,	2005	Military Operations	Genetic Algorithm	[10]
Jindal ve ark	2011	Military Operations	Heuristic Algorithm	[11]
Englot ve ark	2013	Military Operations	Heuristic Algorithm	[12]
Jindal ve ark.,	2011	Military Operations	Heuristic Algorithm	[13]
Khosravi ve ark.,	2014	Military Operations	Heuristic Algorithm	[14]
Hammar ve Nilsson	1999	Military Operations	Theoretical Inference	[15]
Zhou ve ark	2003	Military Operations	Dynamic Inver-Over Evolutionary Algorithm	[16]
Choubey	2013	Military Operations	Genetic Algorithm	[17]
Lee ve ark.	2002	Military Operations	Immunity-based Ant Colony Optimization	[18]
Pushkarini ve Bullo	2014	Military Operations	Heuristic Algorithm	[19]
Bengt	2002	Military Operations	Theoretical Inference	[20]
Knapp ve Rothe	2012	Military Operations	Simulation	[21]
Bourjolly ve ark.	2006	Space applications	Tabu Search Alg. and Heuristic Alg.	[22]
Blough et al.	2016	Space applications	Genetic Algorithm and Sequential Meeting Point Method	[23]
Mei et al.	2015	Space applications	Heuristic Algorithm	[24]
Bimbo and Pernici	2005	Security Cameras and Surveillance Activities	Optimal Solution Methodology	[25]
Bimbo and Pernici	2005	Security Cameras and Surveillance Activities	Monte Carlo Simulation	[26]
Ilavarasi and Joseph	2014	Scheduling of the robotic systems		[27]
Asahiro et al.	2004	Scheduling of the robotic systems	Heuristic Algorithm	[28]
Asahiro et al.	2008	Scheduling of the robotic systems	2-factor approach algorithm	[29]
Chalasani et al.	1996	Scheduling of the robotic systems	Approach Algorithm	[30]
Groba et al.	2018	Dynamic routing problem	Genetic Algorithm	[31]
Papadakos et al.	2011	Risky GSP		[32]
Ucar and Isleyen	2019	Fleet Routing Problem with Moving Target		[33]

times. This solution approach, at least to our knowledge, is used for the first time in the Moving Target Traveling Salesman Problem-Time Window. In addition, this study is different from other studies in the literature due to the first implementation of metaheuristic techniques for the

refugee problem in the sea. Second section of the study defines the problem and the third section describes the proposed solution method. In the fourth section, the proposed approach is tested on various scenarios by considering refugee activities in the Aegean Sea. The fifth and final



FIGURE 3: Boundaries in the Aegean Sea [3]. Figure 3 is reproduced from the news with dated 26.10.2018 of Aydınlik.com website (2018).

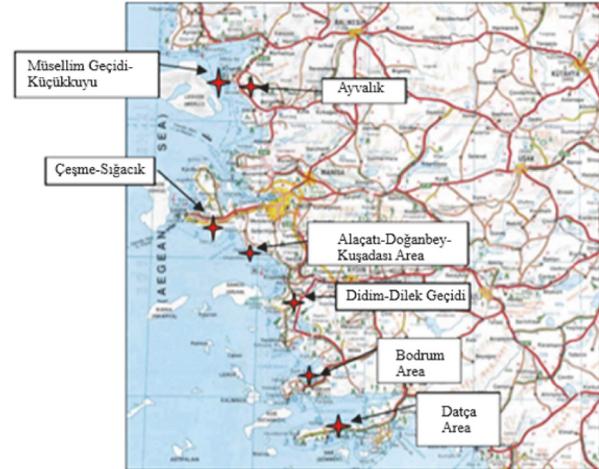


FIGURE 5: Illegal migration route along Aegean Sea Coastal Area [5]. Figure 5 is reproduced from Bodur (2010).

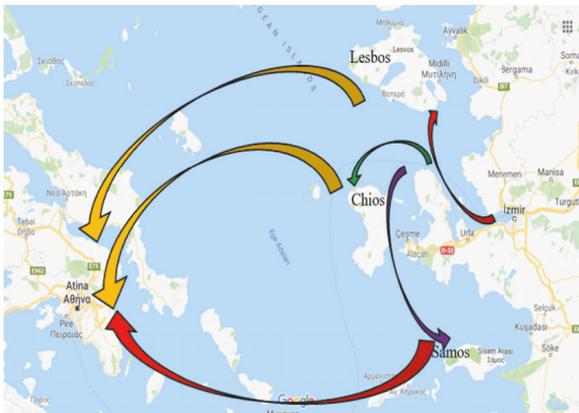


FIGURE 4: Illegal migration route in the Aegean Sea [4]. Figure 4 is reproduced from Google Maps website (2019).



FIGURE 6: Number of illegal immigrants seized in Aegean Sea in 2013 [6]. Figure 6 is reproduced from “Investigation Report on Illegal Migration in the Aegean Coast (İzmir-Aydın)” (2014).

sections cover recommendations and evaluations regarding the future studies.

2. Definition of Problem

The Aegean Sea is a sea that connects the Mediterranean Sea, with coasts surrounded by Turkey and Greece. The two countries have shares on the sea as defined in international conventions and each country performs security operations in its territorial waters. Figure 3 shows territorial water borders of the countries in the Aegean Sea. In the figure, the area marked with yellow shows the Turkish territorial waters. Since Greece is a door to Europe and the maritime control operations are harder than the land controls, thousands of people every year want to cross the Aegean Sea to the Greek islands by boats or ships, then to Europe. These transits are illegal and very risky, often resulting in drowning and death. Figures 4 and 5 show illegal migration routes related to migrations from Turkey to Greece through the Aegean Sea.

Figure 4 shows general pattern of migration from Turkey to Greece, while Figure 5 shows detailed analysis of migration from Turkish coasts to Greece. According to Figure 4, the

refugee boats first leave the İzmir coasts and then sail to the nearby Greek islands such as Lesbos, Chios, or Samos. If such transit becomes successful, they introduce themselves as refugees to the Greek authorities and are transferred to Athens after staying in these islands for about one week. These transits are not always successful and they are mostly caught by the Turkish coast guard or people get drowned due to sinking of refugee boats. Figure 5 shows a detailed analysis of migration from the Aegean coasts to Greek islands, and Figure 6 shows the number of illegal immigrants caught in these regions in 2013.

According to Figure 6, most of illegal immigrants are in Çeşme-Alaçatı, Seferihisar, and Kuşadası. Therefore, the problem of intervention in migrations from these coasts is examined in the application process.

Figure 7 shows some routes used for migrations to the Greek islands from relevant coasts. The problem of surveillance, exploration, and intervene in the refugee and trafficking boats in the seas is a type of problem that intersects



FIGURE 7: Possible migration scenarios that may occur in the coastal regions concerned.

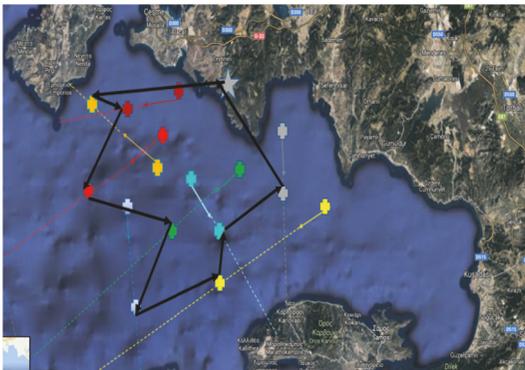


FIGURE 8: The problem of intervention to refugee, smuggling, or terror vessels.

with MTTSP in the literature. The MTTSP has moving targets in n number, with certain angles and speeds, these targets are destroyed by a constant velocity pursuer that starts from the original point, and as a result a tour route is attempted to be defined with a minimum time when all targets are destroyed and the pursuer returns the original point. Since the territorial waters have a certain boundary, the refugee boats must be intervened within the relevant boundaries, and thus the relevant problem is examined within the framework of MTTSP-TW. It is assumed that TW (Time Window) related targets will exist in the system within a certain period of time, and if they are not destroyed within such time, then the mission will fail.

Figure 8 shows a tour route related to the targets that are observed and destroyed or intervened by a maritime surveillance helicopter or unmanned air vehicle. In the route, the targets that are identified based on information obtained are neutralized by a pursuer that starts from the original point. The representations of the problem on the coordinate axis and the state of the targets in the system at any time are shown in “Figure 9”. In the figures, the red line indicates the points at which the targets can be reached, and the task is considered to fail when the targets reach the red line.

When the results in “Figure 9(a)” are examined, it is understood that there are four targets in the system at $t = 0$, when $t = 3$, a target is separated from the system,

two targets are included, and the total number of targets is reached to 5. Referring to “Figures 9(c), 9(d), and 9(e)”, it is shown that new target inputs and existing targets are out of the system. In addition, an exemplary tour route for solving the problem is described on “Figure 9(f)”. When the shapes are examined, the targets change continuously and there are “ $n!$ ” possible solutions. When the complex dynamic structure and time window constraints of the problem are taken into account, it is difficult and takes a lot of time to find the solution with minimum tour route with the approaches such as mathematical modelling among the large amount of possible solutions. For these reasons, intelligent solution strategies such as metaheuristic methods are needed to solve the problem. In this paper, problem is solved with the solution approach based on annealing simulation.

In this study, S-70-B2 Seahawk helicopters were selected as pursuer and shown in Figure 10. This helicopter is a Sikorsky type helicopter that has the capacity to carry out exploration and surveillance, search and rescue, vertical resupply, underwater and surface attack, and transport. The maximum speed of the helicopter is 270 km/h and the helicopter is equipped with torpedo, missile, and radar systems [42].

The problem has a large application area and can be addressed in a variety of ways. In this study, the following assumptions are made for the problem:

The pursuer (helicopter) must have more speed than all targets.

The pursuer (helicopter) must return to the starting point after destroying all targets.

The pursuer and targets (x, y) move in the coordinate plane.

Velocity and angles of the targets are constant. These speeds and angles do not change over time.

The pursuer velocity is fixed and does not change over time.

The targets exist in the system within specified time window and the time for each target to exit the system (time to arrive in the relevant Greek island) is different. If this target is not destroyed within the relevant time interval, then the mission is considered unsuccessful.

3. Methodology

The simulated annealing algorithm is a metaheuristic method developed by Kirkpatrick et al. in 1983. The method is a stochastic search method based on the physical annealing process of solids and iteratively proceeds through a single solution. In the algorithm, various motion mechanisms are used to improve solution quality, and the movements that cause improvement in the solution are accepted as initial solution, while the movements that do not affect solution quality are accepted with a certain probability. The Boltzmann probability ($P = e^{(-\Delta/T)}$) is used in acceptance of bad solutions [45]. It is a frequently used method as it makes

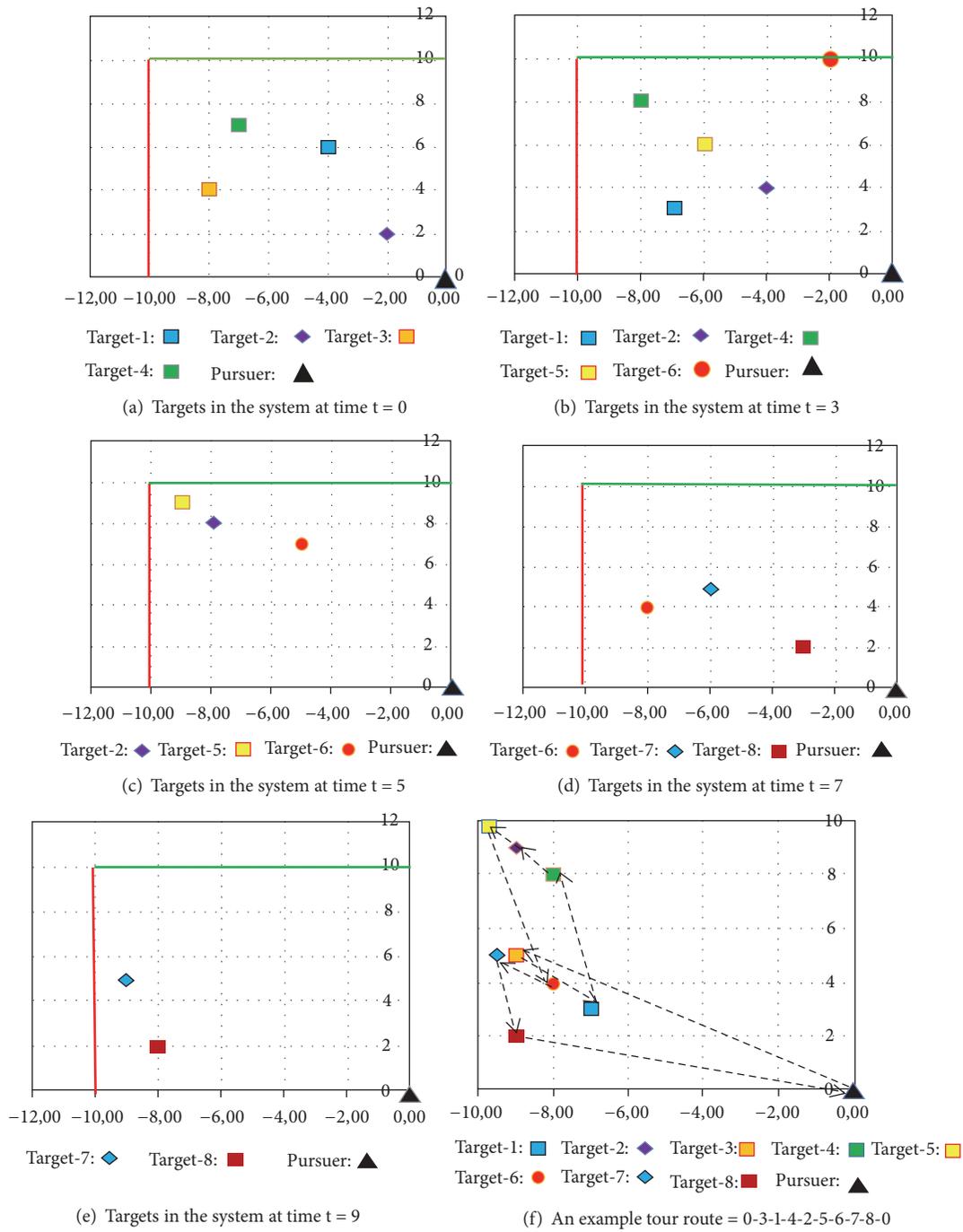


FIGURE 9: Development of the problem on the coordinate axis during the operation period.



FIGURE 10: S-70-B2 Seahawk Sea helicopter [43]. Figure 10 is reproduced from “tr.wikipedia.org” website (2019).

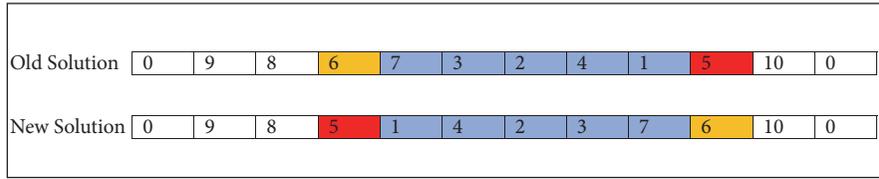


FIGURE 11: Solution perturbation mechanism in case the nodes is selected randomly.

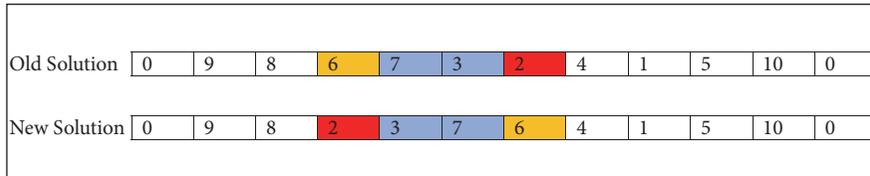


FIGURE 12: Solution perturbation mechanism in case the difference between the two nodes is 3 and the nodes are randomly selected.

good search and uses knowledge in solution of large-scale problems. In this study, the simulated annealing algorithm was used as it guarantees to find the optimum statistically, is used in continuous optimization problems and can be applied easily to different cost functions and constraints [46, 47].

There are a number of decisions that need to be made when solving problems with the simulated annealing algorithm. These decisions have a significant impact on the quality and speed of the solution. In this context, one of the first decisions to be made is about the initial temperature. The cooling schedule is another factor that affects the success of optimization algorithm. This rate is an important parameter that affects the probability of acceptance and defines temperature value in each iteration and the cooling rate can range from 0,5 to 0,99 in the studies[47]. Also, geometric cooling schedule is used to update temperature values in the proposed algorithm. In this study, suitable values for the parameters in the algorithm were determined as a result of experimental studies and these values are shown in Section 4.1.

In the algorithm, the minimum time strategy is used in determining the initial solution. In this strategy, the target that the pursuer will destroy/detect within the shortest time is selected among the targets in the system for pursuer considering relevant time window constraint and the algorithm ends when all targets are destroyed/detected. The swap method is used as neighbourhood change mechanism, and it is aimed to obtain different solutions with a solution perturbation strategy in order to prevent the solution from getting stuck in local optimal after certain iteration. The solution perturbation strategy used is shown in Figure 11.

In Figure 11, any two random points are selected in relevant solution and a new solution is created by reverse ordering of solutions remaining between these two points. In some cases, the random numbers generated (such as the cases when random numbers are the first target and the last target) cause a major change in the current solution. As a result, an efficient algorithm cannot be produced and the algorithm can shift to random search. In order to prevent the specified problem, the strategy shown in Figure 12 is used. In this study,

the difference between two points is defined as 3 and the effectiveness is provided by replacing the targets between the two points.

The solution algorithm developed for solving the related problem according to specified properties is shown in Table 2.

The developed algorithm was tested on real life application and random scenarios mentioned in Section 4; a general evaluation about the study is provided in Section 5.

4. Application

4.1. *Problem Design.* A sample display of the problem sets which are developed to test the algorithm’s efficiency is shown in Figure 13 and Table 3 of this article. The information given in Figure 13 and Table 3 is based on real data. In the Figure 13, the mission area is about 9955,58 km² and has a horizontal distance of 131 km and vertical distance of 76 km. The red lines in the figure represent the Turkish-Greek borders, which may be different in official sources. The main purpose is to define the route that will allow the S-70B Seahawk helicopter, which starts from the original point as shown in red circle, to destroy/rescue/detect the refugee boats and drug and treasury trafficking ships embarking from specified positions at various times under the assumptions given in section two within a minimum time before they reach relevant destinations, and to return to its original point.

Table 3 shows information on pursuer and targets as well as destruction times related to the result of solution.

The solution of algorithm was made by using C# on a computer with 4GB Ram and 2.5 GHz processor, the algorithm was run for 10 times based on the specified data, and the tour 0-6-12-7-1-8-11-10-2-5-9-3-4-0 with the objective of 1.99 hours was obtained in all runs. The average run time for relevant analysis is 2 minutes. Also, the results in the table show that all targets were destroyed/detected within relevant time window. There are more targets in the field of operation in real life problems and efficiency of the algorithm should be determined by considering this situation. In the article, the



FIGURE 13: Scenario examined in the application study.

TABLE 2: SA-MTH-TW (Simulated Annealing-Minimum Time Heuristic-Time Window) Algorithm.

```

 $f(S_{best}) = double.Maxvalue, S_{best}$ 
while(temperature < stopping criteria)
  Create an initial solution ( $S_0$ ) according to relevant initial solution algorithm.
  Initial temperature( $T$ )150, cooling rate ( $\alpha$ ) 0,90
   $S \leftarrow S_0, f(S) \leftarrow f(S_0)$ 
   $S_{good} \leftarrow S_0, f(S_{good}) \leftarrow f(S_0)$ 
  bestsay = 0
  while(sıcaklık  $\geq$  1)
     $f(S^*) = double.Maxvalue, S^*$ 
    Generate all neighboring solutions( $S_i^*$ ) of  $S$  according to the swap mechanism
    if  $S_i^*$  provides time window constraint
      if  $f(S^*) > f(S_i^*)$  then  $S^* \leftarrow S_i^*$  and  $f(S^*) \leftarrow f(S_i^*)$ 
      if  $f(S_{good}) > f(S_i^*)$  then  $S_{good} \leftarrow S_i^*$  and  $f(S_{good}) \leftarrow f(S_i^*)$ 
      if  $f(S_{best}) > f(S_i^*)$  then  $S_{best} \leftarrow S_i^*$  and  $f(S_{best}) \leftarrow f(S_i^*)$ 
    end
     $\Delta \leftarrow f(S^*) - f(S)$ 
    if  $\Delta \leq 0$  then
       $S \leftarrow S^*$ 
      Update location information and task time information on targets and pursuer
    else
       $u \sim u(0, 1)$ 
      if  $u < e^{(-\Delta/T)}$  then
         $S \leftarrow S^*$ 
        Update location information and task time information on targets and pursuer
      else
         $S \leftarrow S$ 
      end
    end
    if  $f(S^*) < f(S_{good})$  then bestsay ++
    if bestsay 5 then apply perturbation process 9 times to solution  $S$  by using
    inverse mechanism and use the best solution as the initial solution( $S$ ) in the next iteration
  end
   $T = T * \alpha$ 
end
end

```

TABLE 3: Information related to the problem indicated in Figure 13.

Vehicle	INPUT INFORMATION						OUTPUT
	Speed (km/h)	Initial x (km)	Initial y (km)	Angle (degree)	Entry Time	Exit Time	Destruction/ Detected Time(hour)
Helicopter	264	0	0	0	0	-----	-----
Target 1	60	-19	36	230	0,56	1,27	0,69288
Target 2	71	-21	15	145	1,06	3,69	1,14268
Target 3	55	-105	71	310	1,4	4,64	1,51284
Target 4	65	-52	43	225	1,3	2,2	1,70926
Target 5	75	-21	38	184	1,1	2,25	1,21627
Target 6	57	-61	59	268	0	0,76	0,28394
Target 7	80	-41	44	240	0,32	1,13	0,5867
Target 8	62	-61	54	233	0,5	1,48	0,8873
Target 9	66	-39	34	140	0,74	4,68	1,40631
Target 10	44	-27	35	226	0,96	2,37	1,094
Target 11	69	-58	54	243	0,86	1,81	0,96045
Target 12	42	-90	69	272	0,1	0,87	0,40171

number of targets taken into consideration in determining the algorithm efficiency is stated below.

The Aegean Sea is an international seaway where many types of ships like passenger ships, cargo vessels, tankers, and fishing boats, travel on densely. In order to determine the number of targets in the problem sets, the average daily traffic density on the Aegean Sea has to be determined first and the congestion analysis has to be made. Congestion analysis is a method that is used in real life situations for identifying the density of the system and finding out its performance. One of the main application fields of this method is the vehicle traffic in motorways. Numerous works on this topic are already present in the literature [48–52]. With this method it is aimed at controlling the traffic, correctly identifying congestions in the traffic webs, and taking precautions to relieve the traffic accordingly [53]. In calculating the level of congestion, real-time data is used and different methods can be employed [54].

The aforementioned data is also used while controlling and directing seaway traffic and the seaway density map aims to prevent accidents on the sea. In this work, the average daily number of ships in the Aegean is determined with density mapping and the problem sets which are to be used in testing the algorithm efficiency are formed. The data as taken from Marinetraffic for the traffic density in the Aegean Sea and of some of the ports on this sea is shown in Figure 14 [44].

Utilizing the density map in Figure 14, the efficiency of the algorithm is tested in 50 different scenarios where the number of targets ranges from 20 to 100. These scenarios are classified under 3 groups as small (20 and 40 targets), medium (60 targets), and large (80 and 100 targets). In the problem sets, the number of targets that enter the system in a certain amount of time is randomly generated according to the Poisson distribution and the time lapse between the targets' arrival to the system is randomly generated according to exponential distribution. Knuth's Algorithm and Inverse Transformation Function are employed for randomly generating numbers for Poisson distribution and for exponential

distribution respectively [55, 56]. In the experiments, λ parameter for Poisson distribution is taken as 5 and 10. The exponential distribution parameter β is determined according to parameter λ . In the problem sets; pursuer's speed value is randomly generated ranging from 200 km/h to 270 km/h and the target speeds are randomly generated between 20 km/h and 60 km/h. The main purpose of the algorithm is to determine the shortest time tour route in which the pursuer visits all the targets in the given time window and returns back to the starting point.

In addition to these data, experimental analysis studies were performed for determining the parameters of the algorithm. There are 5 parameters to the proposed simulated annealing: number of breaking solution (NBS), cooling rate (α), number of iterations at each temperature (N), initial (T_0), and final (T_f) temperature. The NBS parameter states in how many iterations the solution would break down in the perturbation mechanism. Among the aforementioned parameters the "N" parameter is decided deterministically (number of neighbour produced by swap mechanism + number of neighbour produced by insert mechanism). The 4 remaining parameters require a parameter tuning. The parameters used in the tuning and their levels are stated in Table 4.

In this study, parameter tuning was determined by experimental tests. The tests revealed that the ideal values are 150 for T_0 , 1 for T_f , 5 for NBS, and 0,90 for α .

4.2. Experimental Results. The analysis study for this article was executed on a personal computer with 4 GB Ram and 2.5 Ghz processor using C# software. The algorithm was run 10 times for each scenario and the obtained results are shown in Appendix (Tables 5 and 6). These tables include information about the mean, the best, the standard deviation, and the deviation of the mean from the best solution value. The results in the tables are examined in detail in the graphs below (Figures 15, 16, 17, and 18).

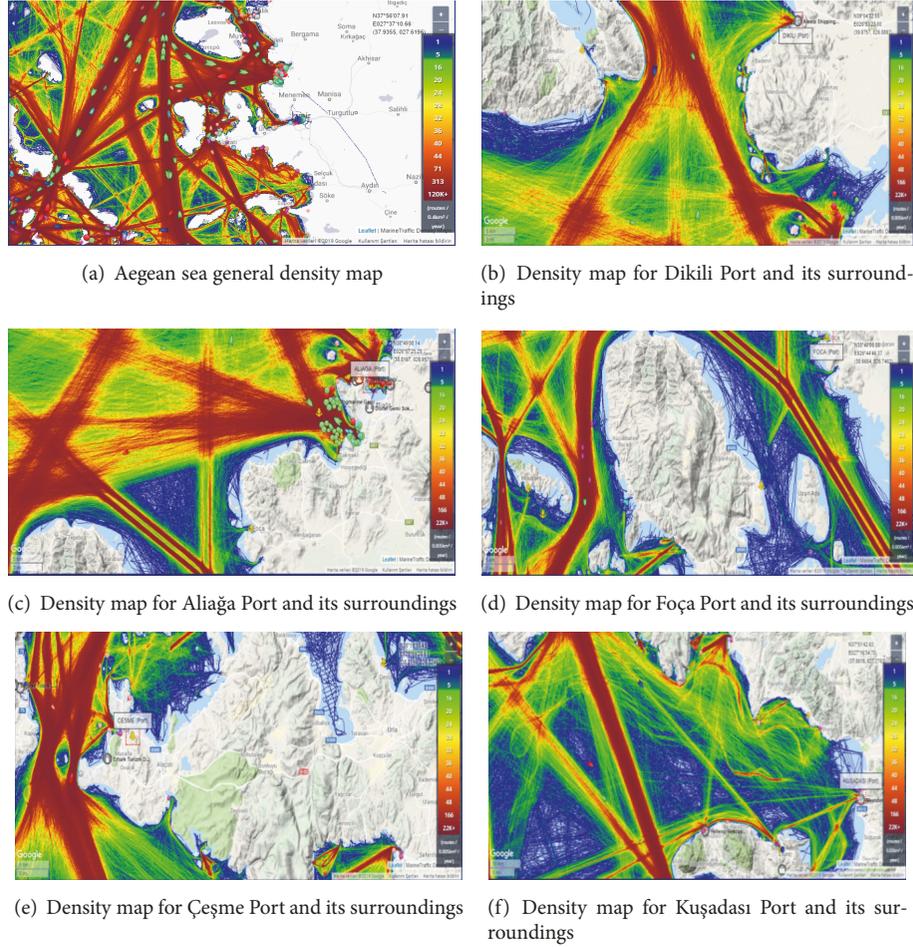


FIGURE 14: Data on Aegean Sea Density Map in 2017 according to Marinetransit [44]. Figure 14 is reproduced from www.marinetraffic.com/ website (2019).

TABLE 4: The parameters used in the tuning and their levels.

Parameter Level	A	B	C	D
	T_0	T_f	N.B.S.	α
1	50	0,005	5	0,85
2	100	0,5	15	0,90
3	150	1	30	0,95

The analysis results in terms of objective values for $\lambda=5$ are shown in Figure 15. According to these results, it was determined that the algorithm successfully destroyed/detected/rescue all targets for each scenario within minimum mission times. In addition, there was no difference between the best known values and average results in the 14 scenarios, and the difference in other scenarios was found to be quite low. When the results in Table 5 are examined, it is understood that the standard deviation values are low. In this case, it shows that the algorithm is effective for $\lambda=5$ and can be applied on different types of problems.

The response of the algorithm in terms of the size of the problem and the solution time is shown in Figure 16.

When Figure 16 is examined, it was determined that the best known solution values for small and medium size problems were reached in a very short time and that the solution time is more in large-scale problems.

The results of the analysis in terms of objective values for $\lambda = 10$ are shown in Figure 17. It is understood that the algorithm produces effective results for $\lambda = 10$ according to this figure and Table 6. When the difference between the best value and the average value was examined, it was observed that the difference in all scenarios was less than 10%. In addition, when looking at the standard deviation values in Table 6, it is determined that the algorithm produces near-average results.

TABLE 5: An overview of analysis results for $\lambda=5$.

Scenario	Number of Target	Mean	Standard deviation	Best	Deviation from the Best (%)	Average computer processing time (h / min / sec)
Scenario 1	20	0,5	0	0,5	0	3 sec
Scenario 2	20	0,51	0	0,51	0	3 sec
Scenario 3	20	0,5883	0,012	0,57	3,21	3 sec
Scenario 4	20	0,5633	0,012	0,55	2,42	3 sec
Scenario 5	20	0,81	0	0,81	0	2 sec
Scenario 6	40	3,03	0	3,03	0	25 sec
Scenario 7	40	3,47	0	3,47	0	26 sec
Scenario 8	40	4,64	0	4,64	0	26 sec
Scenario 9	40	5,83	0	5,83	0	25 sec
Scenario 10	40	5,07	0	5,07	0	25 sec
Scenario 11	60	7,7783	0,16	7,57	2,75	1 min 47 sec
Scenario 12	60	4,41	0	4,41	0	1 min 45 sec
Scenario 13	60	6,30	0,06	6,17	2,03	1 min 48 sec
Scenario 14	60	8,965	0,16	8,64	3,76	1 min 52 sec
Scenario 15	60	6,15	0	6,15	0	2 min 34 sec
Scenario 16	80	13,66	0	13,66	0	5 min 15 sec
Scenario 17	80	15,43	0,74	14,19	8,74	5 min 23 sec
Scenario 18	80	18,5583	1,05	17,52	5,93	5 min 5 sec
Scenario 19	80	12,6467	0,48	11,78	7,36	5 min 8 sec
Scenario 20	80	10,82	0	10,82	0	6 min 17 sec
Scenario 21	100	17,7433	0,78	16,16	9,80	12 min 44 sec
Scenario 22	100	22,1817	0,069	22,04	0,64	15 min 15 sec
Scenario 23	100	20,7883	0,93	19,86	4,67	12 min 55 sec
Scenario 24	100	20,6867	1,17	19,18	7,86	12 min 42 sec
Scenario 25	100	11,355	0,037	11,28	0,66	11 min 35 sec

Robustness of Algorithm
 If mean = = best then mean = yellow bar & best = blue bar
 Else mean = yellow bar & best = red bar

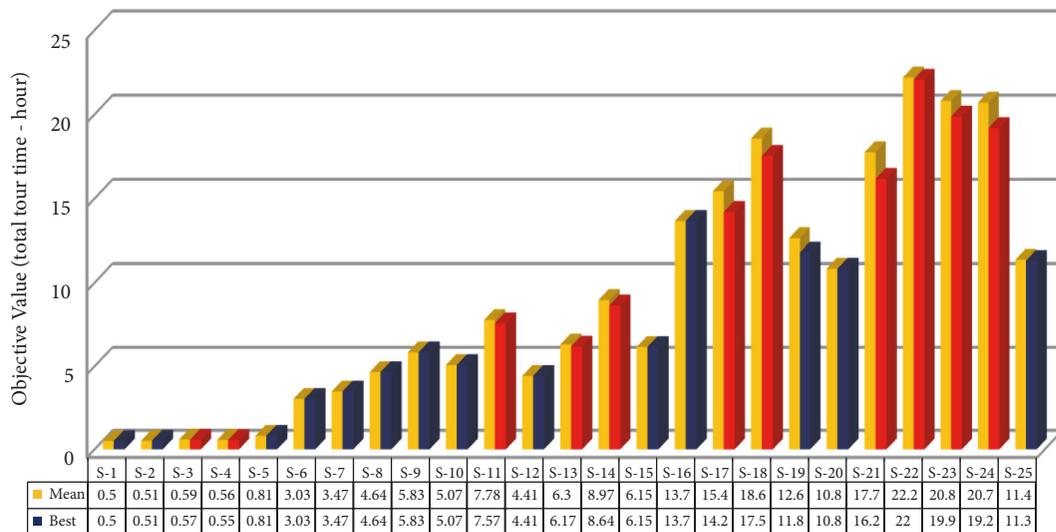


FIGURE 15: Analysis results in terms of objective values for $\lambda=5$.

TABLE 6: An overview of analysis results for $\lambda=10$.

Scenario	Number of Target	Mean	Standard deviation	Best	Deviation from the Best (%)	Average computer processing time (h / min / sec)
Scenario 1	20	0,59	0	0,59	0	2 sec
Scenario 2	20	0,68	0	0,68	0	2 sec
Scenario 3	20	0,738	0,02	0,72	2,5	2 sec
Scenario 4	20	0,722	0,026	0,68	6,176	3 sec
Scenario 5	20	0,684	0,055	0,68	0,588	2 sec
Scenario 6	40	6,922	0,104	6,75	2,548	30 sec
Scenario 7	40	3,166	0,190	2,94	7,687	29 sec
Scenario 8	40	4,686	0,224	4,49	4,365	32 sec
Scenario 9	40	6,812	0,203	6,59	3,369	30 sec
Scenario 10	40	3,852	0,044	3,78	1,905	28 sec
Scenario 11	60	10,75	0	10,75	0	2 min 1 sec
Scenario 12	60	6,748	0,183	6,42	5,109	2 min 7 sec
Scenario 13	60	12,242	0,725	11,19	9,401	2 min 8 sec
Scenario 14	60	11,206	0,210	10,86	3,186	2 min 3 sec
Scenario 15	60	9,76	0	9,76	0	1 min 59 sec
Scenario 16	80	18,956	0,255	18,5	2,465	5 min 42 sec
Scenario 17	80	8,044	0,013	8,02	0,299	5 min 45 sec
Scenario 18	80	8,938	0,273	8,45	5,775	5 min 4 sec
Scenario 19	80	21,288	1,205	19,65	8,336	5 min 36 sec
Scenario 20	80	10,414	0,148	10,15	2,60	5 min 57 sec
Scenario 21	100	14,03	0	14,03	0	12 min 56 sec
Scenario 22	100	33,678	0,631	32,55	3,465	13 min 8 sec
Scenario 23	100	6,41	0	6,41	0	12 min 53 sec
Scenario 24	100	4,322	0,097	4,17	3,645	12 min 18 sec
Scenario 25	100	19,238	0,392	18,58	3,541	13 min 6 sec

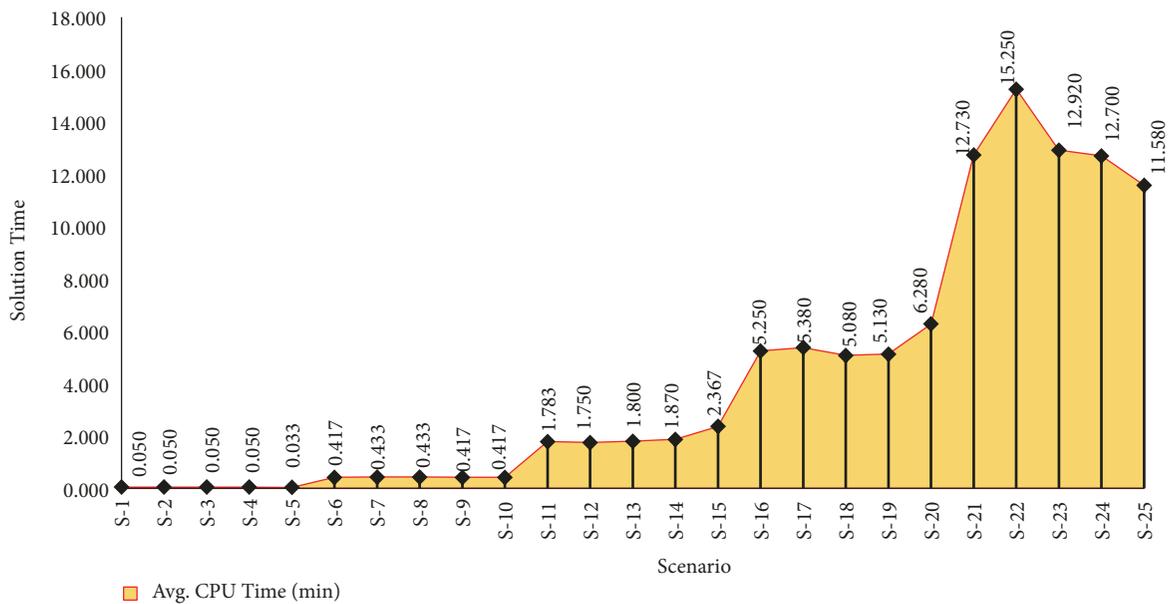


FIGURE 16: Analysis results in terms of solution time for $\lambda=5$.

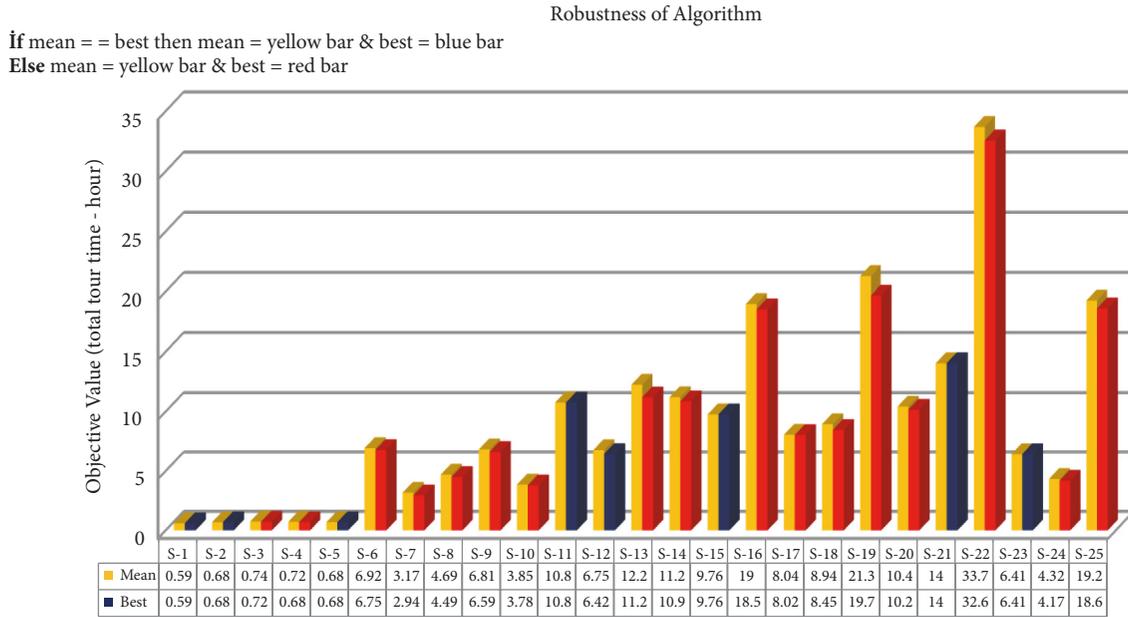


FIGURE 17: Analysis results in terms of objective values for $\lambda=10$.

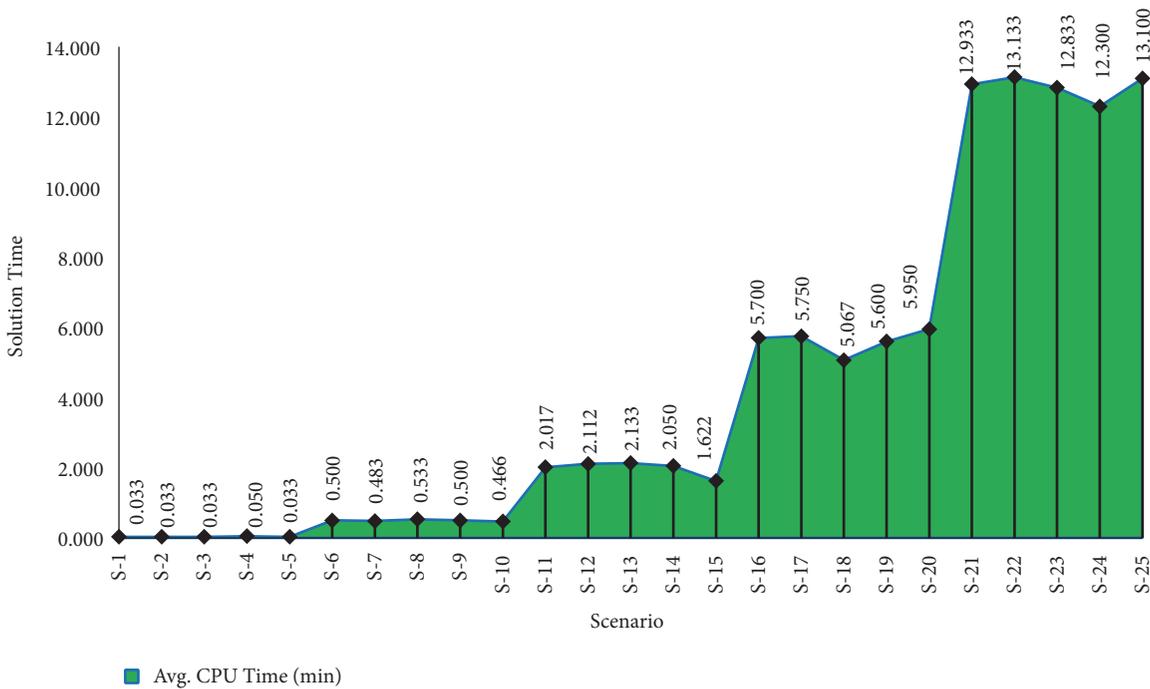


FIGURE 18: Analysis results in terms of solution time for $\lambda=10$.

Figure 18 indicates the performance of the algorithm in terms of the times of solution on the developed scenarios for $\lambda = 10$. It is understood from the results in this figure that all problems are solved within acceptable solution times. In addition, as the number of targets increases, the increase in the solution time is another result extracted from Figure 18.

In all experiments, it was observed that the solution time increased when the number of targets increased. The main

reasons for this situation are that the problem has a very complex and dynamic structure and the parameters in the problem (the position of the pursuer and targets, time of entry and exit of targets, and instantaneous change of the number of targets in the system) change momentarily. In addition, it has been determined that the algorithm responds rapidly and effectively to the changes in the number and characteristics of the targets. In particular, the algorithm produced effective

solutions in cases where the number of targets coming into the system at the unit time increases and the arrival times of the targets become frequent. Due to these reasons, it can be said that the proposed algorithm is robust and effective.

5. Conclusion

Many countries in the world have borders with seas or oceans, and these waters are home to many activities such as transportation and fishing, especially tourism. These territorial waters are also a major threat to many countries, and countries make great efforts to ensure national and international security. In today's world, many countries have internal conflict or war, and therefore many people have to migrate from their countries to others. This migration process is often prevented by coast guard forces, but sometimes it results in deaths as a result of sinking of refugee ships. Besides, trafficking and illegal fishing are carried out on the seas, and pirate activities are committed in larger water areas such as oceans. The common purpose of all these problems is that all targets are scanned, monitored, and destroyed/detected within minimum periods of time.

A solution approach based on the simulated annealing algorithm was developed to prevent the problems mentioned in this study, to assist national security forces and intervene in targets in a short time with S-70B Seahawk helicopter. The problem was associated with the MTTSP-TW problem, and the effectiveness of proposed algorithm was analyzed on various scenarios based on actual data. As a result, it was found that all targets were destroyed/detected in short times within specified time window in all scenarios and the proposed solution approach is an effective algorithm based on the analysis results.

It is thought that the proposed solution methodology can be used to routing and control maritime traffic. In this way, it is predicted that sea traffic accidents will be prevented and accident risks will decrease. It is also expected that the proposed method will be used to solve the routing problem for fish picking devices.

The problem discussed in this study is similar to many problems besides maritime surveillance operations, especially military operations, routing of unmanned air vehicles, guiding of surveillance cameras, positioning of satellites around the world, scheduling of robotic systems, and search and rescue activities. It is believed that the proposed solution approach can be easily applied for the specified problems as it has a generic structure.

It is estimated that better results can be obtained in future studies by defining different parameters and decision mechanisms for the proposed solution approach or by hybridization of the method with other algorithms. Also, it is anticipated that new areas of application may emerge with the addition of different features to given problem or discussing the problem from different perspectives.

Appendix

See Tables 5 and 6.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the Tubitak-BIDEB 2211 PhD Scholarship Program.

References

- [1] https://www.google.com.tr/search?rlz=1C1NDCM_trTR717TR717&biw=1366&bih=613&tbm=isch&sa=1&q=m%C3%BClteci+gemisi&oq=m%C3%BClteci+ge&gs_l=psy-ab.3.0.0l2j0i24k1.7690.11404.0.11985.9.9.0.0.0.154.1296.0j9.9.0...0...1.1.64.psy-ab..0.9.1293...0i67k1j0i13k1j0i13i30k1j0i8i30k1.0.hksWIdtEhsU#imgrc=JXUK_BMmHYogJM.
- [2] <https://sputniknews.com/europe/201508141025748642/>, 2015.
- [3] https://www.google.com.tr/search?rlz=1C1NDCM_trTR717TR717&biw=1366&bih=613&tbm=isch&sa=1&q=ege+denizi+t%C3%BCkr+karasular%C4%B1&oq=ege+denizi+t%C3%BCkr+karasular%C4%B1&gs_l=psy-ab.3...2632.5653.0.5906.16.15.0.0.0.204.1399.0j10j1.1.0...0...1.1.64.psy-ab..5.5.685...0j0i67k1j0i8i30k1j0i24k1j0i13k1.0.GPW6NkzHZVQ#imgrc=8HB3ME9EmPIR_M.
- [4] <https://www.google.com/maps/place/Akhisar,+Manisa/@38.4567361,25.430988,8.25z/data=!4m5!3m4!1s0x14b9d204dcf18c4d:0x4cad81cdf80f483b!8m2!3d38.917664!4d27.83662>, 2019.
- [5] M. Z. Bodur, "Ege'de Denizden Yapılan Yasa Dışı Göç ve Göçmen Profilleri, Göçmenlerin Geleceğe Yönelik Beklentileri ve Öngörüler," *Güvenlik Stratejileri Dergisi*, vol. 12, no. 12, 2010, <http://dergipark.ulakbim.gov.tr/guvenlikstrjtj/article/view/5000098898/5000092154>, Sayfa-103.
- [6] "Ege Kıyılarında (İzmir-Aydın) Gerçekleşen Yasa Dışı Göç Hakkında İnceleme Raporu (2014), Türkiye Büyük Millet Meclisi İnsan Haklarını İnceleme Komisyonu," https://www.tbmm.gov.tr/komision/insanhaklari/docs/2014/ege_kiyilarinda_yasadisigoc_hakkinda_inceleme_raporu_27052014.pdf (2014).
- [7] A. Fügenschuh, M. Knapp, and H. Rothe, *The Multiple Traveling Salesmen Problem with Moving Targets*. Helmut-Schmidt- Univ., Professur für Angewandte Mathematik, 2014.
- [8] A. Stieber, A. Fügenschuh, M. Epp, M. Knapp, and H. Rothe, "The multiple traveling salesmen problem with moving targets," *Optimization Letters*, vol. 9, no. 8, pp. 1569–1583, 2015.
- [9] A. Stieber and A. Fügenschuh, "Variants in Modeling Time Aspects for the Multiple Traveling Salesmen Problem with Moving Targets, 2016".
- [10] Q. Jiang, R. Sarker, and H. Abbass, "Tracking moving targets and the non-stationary traveling salesman problem," *Complexity International*, vol. 11, pp. 171–179, 2005.
- [11] P. Jindal and A. Kumar, "Multiple target intercepting traveling salesman problem," *International Journal of Computer Science and Technology*, vol. 2, no. 2, pp. 327–331, 2011.
- [12] B. Engiot, T. Sahai, and I. Cohen, "Efficient tracking and pursuit of moving targets by heuristic solution of the traveling

- salesman problem,” in *Proceedings of the 2013 IEEE 52nd Annual Conference on Decision and Control (CDC)*, pp. 3433–3438, IEEE, 2013.
- [13] P. Jindal, A. Kumar, and S. Kumar, “Dynamic version of traveling salesman problem,” *International Journal of Computer Applications*, vol. 19, no. 1, pp. 0975–8887, 2011.
- [14] M. Khosravi and A. G. Aghdam, “Cooperative receding horizon control for multi-target interception in uncertain environments,” in *Proceedings of the 2014 IEEE 53rd Annual Conference on Decision and Control (CDC)*, pp. 4497–4502, IEEE, 2014.
- [15] M. Hammar and B. J. Nilsson, “Approximation results for kinetic variants of TSP,” in *Proceedings of the International Colloquium on Automata, Languages, and Programming*, pp. 392–401, Springer, Berlin, Germany, 1999.
- [16] A. Zhou, L. Kang, and Z. Yan, “Solving dynamic TSP with evolutionary approach in real time,” in *Proceedings of the 2003 Congress on Evolutionary Computation, (CEC’03)*, vol. 2, pp. 951–957, IEEE, 2003.
- [17] N. S. Choubey, “Moving Target Travelling Salesman Problem Using Genetic Algorithm,” *International Journal of Computer Applications*, vol. 70, no. 2, 2013.
- [18] Z.-J. Lee, C.-Y. Lee, and S.-F. Su, “An immunity-based ant colony optimization algorithm for solving weapon-target assignment problem,” *Applied Soft Computing*, vol. 2, no. 1, pp. 39–47, 2002.
- [19] P. Agharkar and F. Bullo, “Vehicle routing algorithms to intercept escaping targets,” in *Proceedings of the 2014 American Control Conference (ACC ’14)*, pp. 952–957, IEEE, 2014.
- [20] J. Bengt, “Approximation Results for Kinetic Variants of TSP,” *Discrete & Computational Geometry*, vol. 4, no. 27, 2002.
- [21] M. Knapp and H. Rothe, “Concept for simulating engagement strategies for C-RAM systems using laser weapons,” in *Proceedings of the DMMS*, 2012.
- [22] J.-M. Bourjolly, O. Gurtuna, and A. Lyngvi, “On-orbit servicing: a time-dependent, moving-target traveling salesman problem,” *International Transactions in Operational Research*, vol. 13, no. 5, pp. 461–481, 2006.
- [23] O. P. Blough, T. K. Farrington, and J. Hudson, Trojan Asteroid Mission Design: Target Selection And Sequencing Optimization, 2016.
- [24] G. Mei, X. Ran, D. Fang, and C. Zhang, “Improved Satellite Scheduling Algorithm for Moving Target,” in *Proceedings of the fourth International Conference on Information Science and Cloud Computing (ISCC ’15)*, Guangzhou, China, 2015, <http://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=264>, id. 58.
- [25] A. Del Bimbo and F. Pernici, “Distant targets identification as an on-line dynamic vehicle routing problem using an active-zooming camera,” in *Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 97–104, IEEE, 2005.
- [26] A. D. Bimbo and F. Pernici, “Saccades planning with kinetic TSP for distant targets identification,” in *Proceedings of the IEEE International Symposium on Imaging for Crime Detection and Prevention, 2005. ICDP ’05*, pp. 145–149, IET, 2005.
- [27] K. Ilavarasi and K. S. Joseph, “Variants of travelling salesman problem: A survey,” in *Proceedings of the 2014 International Conference on Information Communication and Embedded Systems (ICICES ’14)*, pp. 1–7, IEEE, 2014.
- [28] Y. Asahiro, T. Horiyama, K. Makino, H. Ono, T. Sakuma, and M. Yamashita, “How to collect balls moving in the euclidean plane,” *Electronic Notes in Theoretical Computer Science*, vol. 91, pp. 229–245, 2004.
- [29] Y. Asahiro, E. Miyano, and S. Shimoirisa, “Grasp and delivery for moving objects on broken lines,” *Theory of Computing Systems*, vol. 42, no. 3, pp. 289–305, 2008.
- [30] P. Chalasani, R. Motwani, and A. N. I. L. Rao, “Algorithms for robot grasp and delivery,” in *Proceedings of the 2nd International Workshop on Algorithmic Foundations of Robotics*, 1996.
- [31] C. Groba, A. Sartal, and X. H. Vázquez, “Integrating forecasting in metaheuristic methods to solve dynamic routing problems: Evidence from the logistic processes of tuna vessels,” *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 55–66, 2018.
- [32] N. Papadakos, G. Tzallas-Regas, B. Rustem, and J. Thoms, “Risky traveling salesman problem,” *European Journal of Operational Research*, vol. 212, no. 1, pp. 69–73, 2011.
- [33] U. Uçar and S. Isleyen, “A New Solution Approach for UAV Routing Problem with Moving Target – Heterogeneous Fleet,” *Journal of Polytechnic*, 2019.
- [34] C. S. Helvig, G. Robins, and A. Zelikovsky, “The moving-target traveling salesman problem,” *Journal of Algorithms*, vol. 49, no. 1, pp. 153–174, 2003.
- [35] C. Groba, A. Sartal, and X. H. Vázquez, “Solving the dynamic traveling salesman problem using a genetic algorithm with trajectory prediction: An application to fish aggregating devices,” *Computers & Operations Research*, vol. 56, pp. 22–32, 2015.
- [36] G. Mercer, S. I. Barry, D. O. Marlow, and P. Kilby, “Investigating the effect of detection and classification range and aircraft dynamics on a,” *ANZIAM Journal*, vol. 49, pp. 475–492, 2008.
- [37] P. Kilby, P. Tobin, R. Luscombe, S. I. Barry, and R. Hickson, The maritime surveillance problem, 2007.
- [38] D. O. Marlow, P. Kilby, and G. N. Mercer, “The travelling salesman problem in maritime surveillance—techniques, algorithms and analysis,” in *Proceedings of the International Congress on Modelling and Simulation*, pp. 684–690, 2007.
- [39] F. Fang, A. Xin, and M. Tambe, “Protecting moving targets with multiple mobile resources,” *Journal of Artificial Intelligence Research*, vol. 48, pp. 583–634, 2013.
- [40] M. Cross, D. Marlow, and J. Looker, “Application of the non-stationary travelling salesman problem to maritime surveillance,” in *Proceedings of the MISG*, pp. 1–4, 2007.
- [41] <http://iats17.firat.edu.tr/images/Proceedings/Chapter-15.pdf>, Accessed: 08.12.2017.
- [42] https://webcache.googleusercontent.com/search?q=cache:YosbDmJ4geAJ,https://tr.wikipedia.org/wiki/S-70-B2_Seahawk+%&cd=1&hl=tr&ct=clnk&gl=tr, Accessed:16.10.2017-13:03.
- [43] https://webcache.googleusercontent.com/search?q=cache:YosbDmJ4geAJ,https://tr.wikipedia.org/wiki/S-70-B2_Seahawk+%&cd=1&hl=tr&ct=clnk&gl=tr2019, Accessed:22.06.2019-16:02.
- [44] <https://www.marinetraffic.com/en/ais/home/centerx:25.6/centery:38.4/zoom:8>, 2019.
- [45] M. M. Mafarja and S. Mirjalili, “Hybrid whale optimization algorithm with simulated annealing for feature selection,” *Neurocomputing*, vol. 260, pp. 302–312, 2017.
- [46] L. Ingber, “Simulated annealing: practice versus theory,” *Mathematical and Computer Modelling*, vol. 18, no. 11, pp. 29–57, 1993.
- [47] T. El-Ghazali, *Metaheuristics: from Design to Implementation*, John Wiley & Sons, 2009.
- [48] T. Tsubota, A. Bhaskar, E. Chung, and R. Billot, “Arterial traffic congestion analysis using Bluetooth Duration data,” 2011.
- [49] Z. Kan, L. Tang, M.-P. Kwan, C. Ren, D. Liu, and Q. Li, “Traffic congestion analysis at the turn level using Taxis’ GPS trajectory data,” *Computers, Environment and Urban Systems*, vol. 74, pp. 229–243, 2019.

- [50] B. H. Susilo and I. Imanuel, "Traffic congestion analysis using travel time ratio and degree of saturation on road sections in Palembang, Bandung, Yogyakarta, and Surakarta," in *Proceedings of the MATEC Web of Conferences*, vol. 181 of *EDP Sciences*, p. 06010, 2018.
- [51] Q. Wang, J. Wan, and Y. Yuan, "Locality constraint distance metric learning for traffic congestion detection," *Pattern Recognition*, vol. 75, pp. 272–281, 2018.
- [52] S. He, "Quantitative problem of road traffic congestion simulation and extension information analysis," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 2, 2012.
- [53] F. He, X. Yan, Y. Liu, and L. Ma, "A traffic congestion assessment method for urban road networks based on speed performance index," *Procedia Engineering*, vol. 137, pp. 425–433, 2016.
- [54] N. Petrovska and A. Stevanovic, "Traffic congestion analysis visualisation tool," in *Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 1489–1494, IEEE, 2015.
- [55] https://tr.wikipedia.org/wiki/Poisson_da%C4%9F%C4%B1%C4%B1m%C4%B1, Accessed:07.05.2019.
- [56] https://en.wikipedia.org/wiki/Inverse_transform_sampling, Accessed:07.05.2019.

