Defence Science Journal, Vol. 70, No. 3, May 2020, pp. 240-248, DOI : 10.14429/dsj.70.15055

Using Evolutionary Algorithms for the Scheduling of Aircrew on Airborne Early Warning and Control System

Hamit Taner Ünal[*] and Fatih Başçiftçi[#]

[*]Institute of Sciences, Department of Information Technologies, Selçuk University 42071, Selçuklu, Turkey
[#]Faculty of Technology, Department of Computer Engineering, Selçuk University 42003, Selçuklu, Turkey
[*]E-mail: htanerunal@jetkite.com

ABSTRACT

Equipped with an advanced radar and other electronic systems mounted on its body, Airborne Early Warning and Control System (AWACS) enables the airspace to be monitored from medium to long distances and facilitates effective control of friendly aircraft. To operate the complex equipment and fulfill its critical functions, AWACS has a specialised flight and mission crew, all of whom are extensively trained in their respective roles. For mission accomplishment and effective use of resources, tasks should be scheduled, and individuals should be assigned to missions appropriately. In this paper, we implemented evolutionary algorithms for scheduling aircrew on AWACS and propose a novel approach using Genetic Algorithms (GA) with a special encoding strategy and modified genetic operations tailored to the problem. The objective is to assign aircrew to various AWACS tasks such as flights, simulator sessions, ground training classes and other squadron duties while aiming to maximise combat readiness and minimise operational costs. The presented approach is applied to several test instances consisting notional weekly schedules of Turkish Boeing 737 AEW&C Peace Eagle AWACS Base, generated similar to real-world examples. To test the algorithm and evaluate solution performance, experiments have been conducted on a novel scheduling software called AWACS Crew Scheduling (ACS), developed as a test bed. Computational results reveal that presented GA approach proves to be quite successful in solving the AWACS Crew Scheduling Problem and exhibits superior performance when compared to manual methods.

Keywords: AWACS; Airborne early warning; Crew scheduling; Genetic algorithms; Optimisation

1. INTRODUCTION

AWACS is an advanced airborne early warning and control system being employed by NATO and the armed forces of more than twenty countries, including USA, Australia, South Korea, France, Italy, India and Turkey[1]. It detects and identifies air and sea vehicles while facilitating guidance of friendly fighters to their targets in the face of possible threats[2]. After the second World War, many types of AWACS platforms have been developed. E-3A Sentry and Boeing 737 AEW&C (shown in Figs. 1 and 2) are most advanced systems currently being operated at various conflict zones[3]. AWACS Aircraft are defined as 'High Value Air Assets', as they significantly increase the effectiveness of Air Forces in the field of operations and are considered as strength multipliers which must be protected with the highest priority[4, 5].

Equipped with a pulse-doppler radar, UHF/VHF/Satellite Radios, Datalinks and other advanced electronic mission systems, it has critical functions such as surveillance, weapons control and electronic intelligence[6]. AWACS has a specialised flight and mission crew with a highly sophisticated training system to fulfill all these functions[7, 8] as shown in Fig. 3.

An intensive and continuous scheduling activity is carried out manually at AWACS bases[2]. Crew scheduling in this complex structure requires a very difficult and time-consuming process. To maintain the highest level of readiness in dynamic operational conditions and to assign the most suitable crew for high-cost AWACS flights, scheduling with classical methods may be insufficient, as the solution space will increase exponentially, even for a few tasks.

Figure 1. NATO E-3A AWACS *(Image Credit: NATO).*

Received : 07 October 2019, Revised : 03 March 2020
Accepted : 21 April 2020, Online published : 27 April 2020

Figure 2. Turkish Boeing 737 AEW&C *(Image Credit: Turkish Air Force).*

Figure 3. Mission Crew Cabin at Turkish B737 AEW&C *(Image Credit: Turkish Air Force).*

There are numerous studies in the literature aiming to optimise crew scheduling. While traditional approaches are based on Operations Research techniques (Evans[9], Kawakami[10], Gökcen[11], Durkan[12], Van Brabant[13], Nguyen[14], Newlon[15], Brown[16], Yavuz[17], Boyd[18], *et al.*, Vestli[19], *et al.,* Sevimli[20]) more recent research concentrates on artificial intelligence, meta-heuristics and evolutionary algorithms in particular (Aslan[21], Dyer[22], Erdemir[23], Shirley Jr.[24] and Boke[25]).

Recently significant results have been achieved by applying Genetic Algorithms and other naturally inspired methods. Most of the published work involving crew scheduling in aviation relates to crew pairing and crew assignment at commercial airlines. To the best of our knowledge, existing work does not address AWACS Crew Scheduling problem in particular and research on military aspect is very limited. Previous works mentioned above mostly utilise computer assisted decision support systems to help build a feasible schedule at military squadrons. While they effectively provide automation tools to guide schedulers, the algorithms proposed do not involve complete solutions considering all constraints in dynamic conditions.

In this paper, we implement evolutionary algorithms for scheduling aircrew on AWACS and propose a novel approach using GA with a special encoding strategy and modified genetic operations tailored to the problem. In this scope, first of all, we defined the problem and formed the chromosome structure. Then, we determined hard and soft constraints based on criteria mostly referenced in the relevant guidelines, such as mission duration, crew rest periods, continuous training requirements, equity in task distribution and task diversity. We modelled the fitness function of the algorithm mathematically and developed unique strategies for genetic algorithm operators such as selection, crossover and mutation.

To test the algorithm and evaluate the performance of genetic parameters, while visually presenting the obtained results, we developed a novel software called AWACS Crew Scheduling (ACS) as a test bed. We applied presented approach to a notional weekly schedule of Turkish B737 AEW&C AWACS squadron, generated similar to real-world examples. We conducted several experiments on the notional weekly schedule consisting diverse tasks, including flights, simulator sessions, ground training classes and other squadron duties. Finally, we presented and analysed computational results.

2. MATERIAL AND METHODS

2.1 Genetic Algorithms

Genetic algorithms are state-of-the art methods that aim to optimise functions by modeling the nature[26]. It was first introduced by John Holland[27] and continues to be developed by many researchers today[28-40]. Genetic Algorithm parameters represent the genes of chromosomes, while all parameters form chromosomes. Each genetic algorithm consists of populations represented as chromosomes (individuals). The fitness of the population (fitness) is aimed to be maximised or minimised in the direction of certain constraints. Each new generation goes through processes such as selection, crossover and mutation and proceeds towards the next generation[28].

2.2 Problem Statement

Most of the conventional AWACS platforms have the ability to function on their own by means of integrated mission consoles and controllers in various crew positions[41,42]. The problem in this work is described for Turkish AWACS Base with its AWACS squadron having four Boeing 737 AEW&C aircraft and employing notional 77 combat-ready aircrew in 8 different crew positions. A sample weekly schedule is generated, consisting of 20 tasks and 131 scheduling slots. The tasks have been selected among various categories, such as flights, simulator sessions, ground trainings and squadron duties. Figure 4 shows notional Turkish AWACS Base created for the test scenario.

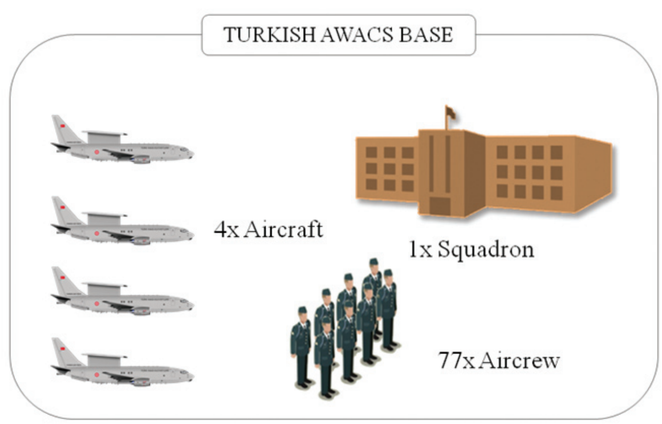


Figure 4. Notional Turkish AWACS base created for problem scenario.

2.3 Scheduling Criteria

There are several criteria assigning aircrew to tasks. The primary factors to be considered when building the schedule are explained below.

Flight Duration and Crew Rest: Similar to a combat squadron, the AWACS has the maximum flight and compulsory rest periods that aircrew must observe. The crew rest period includes the amount of time specified in the relevant guidelines, the time at which no other tasks can be given to individuals before and after the flight. These periods are taken into account when scheduling.

Training Requirements: AWACS aircrew is intensively trained from the day they join the squadron. Continuation Training (COT) starts once initial training is complete and individual is entitled to combat-ready status. Through flight career, she/he must repeat certain training events within the specified period, called 'currency'. These repetitions can be in the form of flight, simulator, theoretical classes or computer-based training. The COT requirements for each crew position are specified in detail on the relevant tables in the guidelines and executions of each training event are meticulously recorded. The failure to maintain currency in the relevant period results for an individual to lose combat-ready status temporarily.

Fairness and Equity in Distribution of Tasks: Fairness is also an important factor in the scheduling. Among aircrew, flying hours, number of tasks and amount of duties executed during the year should be as close as possible and equal distribution between individuals should be taken as basis. It is necessary to provide equal distribution of tasks not only in terms of fairness but also in terms of sharing equal experience.

2.4 Model Description

The most important step in the solution of the AWACS Crew Scheduling Problem is forming the proper chromosome structure for the application of genetic processes. Each chromosome of the population needs to have knowledge of all tasks and slots belonging to a weekly schedule. For each task, seat assignments of crew positions can be represented as genes that make up the chromosome, where we can call each gene a '*scheduling slot*'. Genetic representation and chromosome structure for the AWACS Crew Scheduling Problem are given in Fig. 5.

2.5 Fitness Function and Mathematical Model

In genetic algorithms, the quality of the solution can be determined by fitness function. The factors that determine the quality of the AWACS Crew Scheduling problem are *hard* and *soft* constraints, as in other scheduling problems. It is desirable that a candidate solution fulfills all hard constraints and meets soft constraints to maximum extent. The mathematical model is expressed by the principle of minimisation or maximisation of the fitness function. In the modelling of the AWACS Crew Scheduling Problem, functions of hard and soft constraints are aimed to be minimised by the 'cost' principle. Each gene that does not meet the desired criteria adds some cost to the function in varying amounts. The total cost of whole chromosome is obtained by sum of all these costs.

Hard Constraints: Forming the candidate pools is the first step eliminating unsuitable individuals from selection. Profession, availability and other administrative options are considered to assign aircrew to tasks, in particular to relevant slots. The most important issue emerges as *time conflict*. An individual cannot be assigned to two tasks at the same time. On the other hand, crew rest and crew duty period limitations cannot be violated either.

To overcome time conflicts, task timings are distributed to individuals' time space as timeslots. Once a timeslot receives more than one task, its value is incremented by one. Total value of timeslots, which have a value of two, or more than two add some cost to chromosome. This is called '*conflict cost*'. Since, it is a hard constraint, the total conflict cost of a chromosome must equal to zero. Chromosomes which have no conflict cost are called '*valid solution*'.

The conflict cost of a chromosome can be calculated as follows;

$$c_{ij} = \sum_{t=0}^{n} v_{tij} \tag{1}$$

where c is conflict cost, i is aircrew, i=0...n; n is total aircrew in the pool; j is task, j=0...n; n is task count, t is timeslot, t=0....n; total timeslots, v is conflict value. Here

$$v = \begin{cases} 0, & \textit{no time conflict} \\ > 0, & \textit{time conflict exists} \end{cases}$$

Total time *conflict cost* of a chromosome can be described as;

$$\Delta c_{ij} = \sum_{l=0}^{n} c_{ljk} \tag{2}$$

where l is gene, l=0....n; n is gene count in total.

Soft Constraints: To find the total cost of a chromosome, it is necessary to evaluate the soft constraints that affect the performance of the solution. Each soft constraint adds a certain cost to the fitness function. Costs added by each gene belonging to the candidate solutions are collected and the overall cost of the chromosome is obtained. In the AWACS Crew Schedules, the training requirements, task diversity and fairness in the task distribution constitute general soft constraints.

The most important factor in assigning AWACS aircrew to the schedule is the fulfillment of COT requirements. For this purpose, amount of days remaining for currency should be calculated. Every task has specific training events. If a task contains a training event, which a person is to execute for COT, days remaining for currency indicate how urgent a person is to fly corresponding mission. In other words, if the deadline is approaching for a specific training event, that person has higher priority to be

Day 1 Task 1 Cell 1	Day 1 Task 1 Cell 2	Day 1 Task 1 Cell 3	Day 1 Task 1 Cell 4	....	Day 5 Task 2 Cell 12	Day 5 Task 3 Cell 1	Day 6 Task 1 Cell 1	Day 7 Task 1 Cell 1

Chromosome Structure

Figure 5. Genetic representation and chromosome structure.

assigned to schedule for a task containing the dedicated training event. For example, Pilot A has 15 days left for currency on training event 'Air-to-air refueling (AAR)'. Person B has 5 days left for currency for the same training event. If task X contains AAR, Person B should be assigned to schedule with higher priority.

Training Requirements cost can be calculated as follows;

$$te_{ijk} = \min \frac{d_{ijk}}{P_{k\max}} \tag{3}$$

where $i = 1,\ldots,n,\ j = 1,\ldots,m,\ k = 1,\ldots,q,\ d_{ijk}$ is Days remaining for currency, for person i, in task j, containing training event k. $P_{k\max}$ is Maximum currency period for training event k, for all repetition columns. n is number of aircrew in the same crew position, m is task count, q is raining event count.

For the whole chromosome, all training requirements costs are summed, To obtain total training requirements cost.

$$\Delta te_{ijk} = \sum_{l=0}^{n} \min \frac{d_{ijk}}{P_{k\max}} \tag{4}$$

Duties: The above calculation is for tasks such as flight, simulator and refresher trainings. The calculation is slightly different for squadron duties. It is important to collect statistics on how often a duty is performed, and which days that specific duty have been assigned (weekday, weekend). The duty cost can be obtained as follows.

$$q_{ij} = \frac{r_{ij}}{S_{ij}} + \frac{r_{ijw}}{S_{ij}} \tag{5}$$

where q_{ij} is Duty cost of aircrew i for task j, r_{ij} is Amount of duties performed before, S_{ij} is amount of total days stayed in duty roster. r_{ijv} is Amount of duties performed in day type w (weekdays, weekends).

Total duty cost of a chromosome is;

$$\Delta q_{ij} = \sum_{l=0}^{n} \left(\frac{r_{ij}}{S_{ij}} + \frac{r_{ijw}}{S_{ij}} \right) \tag{6}$$

Task Diversity: In an AWACS Squadron, each individual should perform each type of mission in equal amounts for sharing equal experience. The diversity in task distribution provides the maximum benefit, while providing equal opportunity to aircrew. Especially in the practice of the exercises and operational flights, it is highly valuable.

On the other hand, aircrew roles in the aircraft should be equally distributed as well. For example, AWACS Surveillance Operators (SO), as a team, have different roles on each flight. An SO can either be a datalink operator, tracker or a runner. Each has specific duties and training items. Thus, it is beneficial to track the number of roles performed and distribute them equally.

The *task diversity cost* of the aircrew can be obtained by dividing the number of duty repetitions by the number of days that the individual is in her/his position of duty;

$$\mu_{ij} = \frac{b_{ij}}{R_i} \tag{7}$$

μ_{ij} *Task Diversity cost* of aircrew i, for task j; b_{ij} is Amount of repetition in task/role

R_i Amount of days in active duty

Total *task diversity cost* of a chromosome can be calculated as;

$$\Delta\mu_{ij} = \sum_{l=0}^{n} \frac{b_{ij}}{R_i} \tag{8}$$

Equal Distribution of Tasks (Fairness): While keeping high readiness, with COT requirements fulfilled properly, aircrew should be assigned to tasks in equal amount for fairness. Every person should have equal, or close to equal number of flying hours and simulator tasks executed or instructional duties performed on an annual basis. This will ensure high morale for the whole aircrew in the squadron.

Fairness cost of an individual can be obtained by;

$$h_i = \frac{u_i}{R_i} \tag{9}$$

h_i is fairness cost of aircrew i; u_i is total flying hours/tasks performed; R_i is amount of days in active duty.

Total fairness cost of a chromosome can be calculated as;

$$\Delta h_i = \sum_{l=0}^{n} \frac{u_i}{R_i} \tag{10}$$

Total Cost: As a result of the calculations above, the total cost of a chromosome can be obtained by summing all costs together.

$$T_{ijk} = \sum_{l=0}^{n} \left(\Delta te_{ijk} + \Delta c_{ij} + \Delta q_{ij} + \Delta\mu_{ij} + \Delta h_i \right) \tag{11}$$

The objective of the fitness function in our case is the minimisation of costs. Thus, Overall fitness function of GA, specific to our problem can be described as;

$$\min T_{ijk} = \min \sum_{l=0}^{n} \left(\Delta te_{ijk} + \Delta c_{ij} + \Delta q_{ij} + \Delta\mu_{ij} + \Delta h_i \right) \tag{12}$$

where

T_{ijk} is total cost of fitness function, Δte_{ijk} is the training event readiness function for flight, Simulator and ground training; Δc_{ij} is cost of time confliction, Δq_{ij} is duty equity cost, $\Delta\mu_{ij}$ is cost for equal distribution of tasks, Δh_i is general equity cost,

i =0....n; flying individual in crew

j = 0....n; tasks

k = 0....n; training events

It must be noted that, to obtain valid solutions, the conflict cost of the function must be zero.

$$\begin{aligned} \min \Delta c_{ij} &= 0 \Rightarrow valid \\ \min \Delta c_{ij} &> 0 \Rightarrow invalid \end{aligned} \tag{13}$$

To summarise; it is aimed to minimise the total cost of GA fitness function To select the best candidate for scheduling slots. A gradual decrease in total cost is expected in every iteration of GA process.

To measure the performance of solutions in this work, performance charts have been prepared for each individual at the AWACS Squadron. The solutions obtained by GA have been compared with those performance charts to measure solution quality. Although the criteria are met, based on performance charts, the algorithm will continue to seek more suitable solution alternatives. Considering the problem has many aspects, it should be aimed at obtaining the optimum value within the acceptable solution time. The pseudo code for whole GA Process is given at Fig. 6.

```
Function GA
1     Begin
2     Load aircrew list and classify based on crew positions
3     Load task records and create training requirements table
4     Generate initial population randomly
5     Create time space for all aircrew and insert timeslots for
      assigned tasks
6     Calculate costs
7     [  Calculate conflict cost
           Calculate training requirements cost
           Calculate task diversity cost
           Calculate fairness cost
           Calculate duty cost
       ]   Sum all costs to obtain total cost of fitness function
8     Sort total costs
9     Identify elite chromosomes and separate them for direct
      transfer to the next generation
10    Apply tournament selection and obtain new population
11    Apply crossover and obtain new population
12    Mutate new population and obtain final population
13    Continue loop until stopping criteria is achieved
14    End
```

Figure 6. Pseudo code for GA solution.

3. EXPERIMENTAL SETUP

We developed a Windows desktop application called ACS to conduct tests for GA solutions. The software is programmed in C# language by using Microsoft Visual Studio. The generated tasks in the notional weekly schedule, basic aircrew data and past flight records have been pre-loaded into the program. The parameters, such as number of squadrons in the base, total amount of aircrew, crew positions and any parameter belonging to any AWACS system in the world can be flexibly changed as desired. A screenshot from the test bed software is given in Figs. 7 and 8.

All experiments were performed on an x64-based desktop computer with Windows 10 operating system running on Intel i7 3770 (3.40 Ghz) processor and 8 GB of RAM. To shorten the test periods, flying hours and duty records of the individuals were kept constant and the focus was on cost minimisation for COT Requirements. The ultimate goal is to find the lowest-cost-solution as soon as possible.

Authors conducted 20 experiments with different sets of GA parameters. For each parameter set, we kept the remaining variables fixed to ensure measurable results. Each experiment was repeated at least three times and the average data was recorded. Finally, the schedule performance was evaluated based on performance charts.

Within the test scenario, the COT requirements and current status of each individual has been listed in the training module. A screenshot from the training module is shown in Fig. 8. To test the quality of the algorithm correctly, the COT status of each person has been determined differently and it is desired to measure whether the solution output can evaluate the COT priorities properly. As shown in the table in Fig. 8, for each individual, the training events for crew position's COT are listed and the number of repetitions required and maximum days to remain currency are specified. The current status is listed as 'actual' remaining days for currency.

4. RESULTS AND DISCUSSION

Experiment parameters and results of all experiments are given in Table 1. The results show that a minimum of 90% success has been achieved in all parameter groups while 100% schedule performance guaranteed on 16 experiments out of 20.

The experiments showed that the approach we put forward with this study was able to achieve a more robust schedule in a much shorter time than the solution provided by manual methods. In AWACS Squadrons, a scheduler from each crew position works intensively and collaborates with other schedulers to prepare a weekly schedule. As a result of the consecutive meetings and endless corrections, the final schedule is obtained and released. The resulting schedule is a non-ideal product, where only conflicts are resolved. However, the ACS software, which was developed with our approach, was able to prepare the optimal schedule in less than an hour with a classic desktop computer.

The scheduling module is the interface that presents the user with the best solution and other alternative suggestions as a result of running the genetic algorithm (Fig. 9). In this

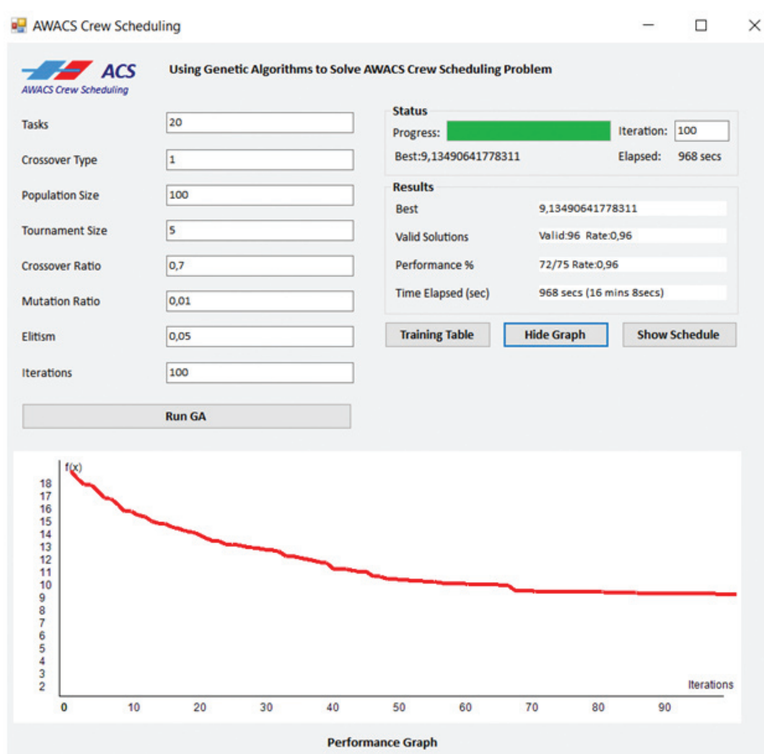
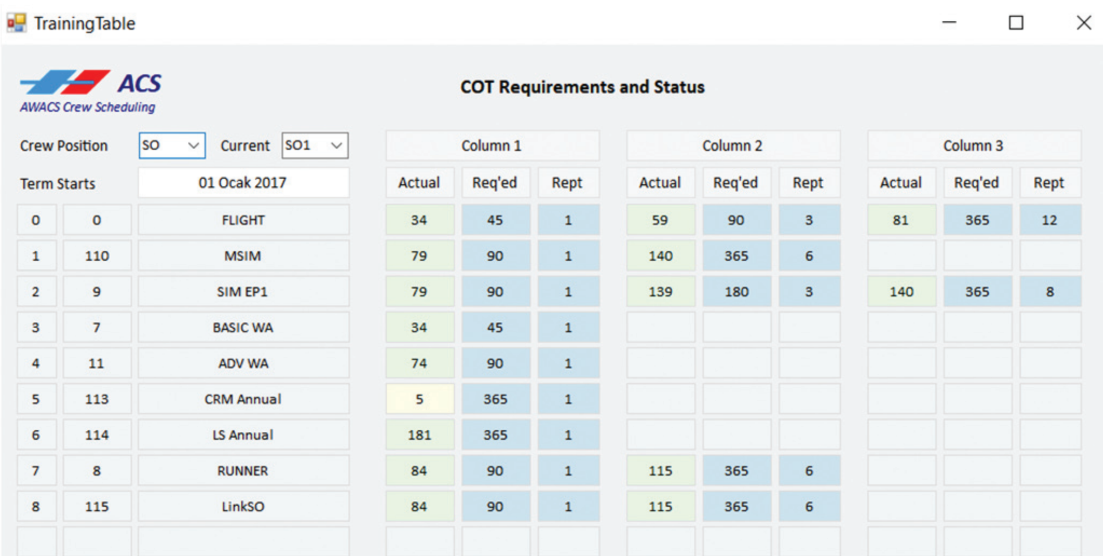


Figure 7. Test bed software (ACS)- main window.



			Column 1			Column 2			Column 3		
			Actual	Req'ed	Rept	Actual	Req'ed	Rept	Actual	Req'ed	Rept
0	0	FLIGHT	34	45	1	59	90	3	81	365	12
1	110	MSIM	79	90	1	140	365	6			
2	9	SIM EP1	79	90	1	139	180	3	140	365	8
3	7	BASIC WA	34	45	1						
4	11	ADV WA	74	90	1						
5	113	CRM Annual	5	365	1						
6	114	LS Annual	181	365	1						
7	8	RUNNER	84	90	1	115	365	6			
8	115	LinkSO	84	90	1	115	365	6			

Figure 8. Test bed software (ACS) - COT tables.

Table 1. Experimental results

Test #	Population size	Tournament size	Crossover point	Crossover rate	Mutation	Elitism	Iteration	Best Solution	Performance	Time elapsed (secs)
1	20	0.1	1	0.6	0.05	0.1	500	10.38	%93	1160
2	50	0.1	1	0.6	0.05	0.1	500	8.43	%97	2600
3	100	0.1	1	0.6	0.05	0.1	500	8.06	%100	5500
4	200	0.1	1	0.6	0.05	0.1	500	7.95	%100	10700
5	100	0.1	1	0.6	0.05	0.1	500	8.06	%100	5500
6	100	0.4	1	0.6	0.05	0.1	500	8.04	%100	5300
7	100	0.7	1	0.6	0.05	0.1	500	7.89	%100	5400
8	100	0.1	1	0.6	0.05	0.1	500	8.06	%100	5500
9	100	0.1	2	0.6	0.05	0.1	500	8.11	%100	5200
10	100	0.1	3	0.6	0.05	0.1	500	8.01	%100	5200
11	100	0.1	1	0.3	0.1	0.05	500	8.11	%100	4800
12	100	0.1	1	0.6	0.1	0.05	500	8.06	%100	5500
13	100	0.1	1	0.9	0.1	0.05	500	8.01	%100	5700
14	100	0.1	1	0.6	0.01	0.1	500	7.80	%100	5000
15	100	0.1	1	0.6	0.05	0.1	500	8.06	%100	5500
16	100	0.1	1	0.6	0.1	0.1	500	8.90	%99	5600
17	100	0.1	1	0.9	0.2	0.1	500	10.92	%91	5600
18	100	0.1	1	0.6	0.1	0.05	500	8.07	%100	5500
19	100	0.1	1	0.6	0.1	0.1	500	8.06	%100	5500
20	100	0.1	1	0.6	0.1	0.2	500	8.10	%100	5500

interface, the notional weekly schedule and the information about each task, such as task type, mission name, date, start and end times, are displayed in the relevant cells.

Based on performance charts, cells are automatically coloured as red, yellow and green. A red cell indicates a hard constraint is violated. A yellow cell shows soft constraints have not been met for designated aircrew, while fulfilling all hard constraints. A green cell demonstrates all hard and soft constraints have been met.

The final schedule obtained with an optimal run is demonstrated in Fig. 9. For each slot, the most suitable aircrew assigned to the corresponding task is indicated in green. While guaranteed for any time conflict, unsuitable candidates are shown in yellow.

5. CONCLUSIONS AND FUTURE WORK

The AWACS Crew Scheduling problem falls into the NP-Hard problem class as most of the scheduling problems, and it is impossible to solve with classical methods, because the solution space exponentially increases as the number of tasks to be scheduled and aircrew to be assigned.

In this paper, we implemented evolutionary algorithms for scheduling aircrew on AWACS and proposed a novel approach using Genetic Algorithms (GA) with a special encoding strategy and modified genetic operations tailored to the problem. The objective was to assign aircrew to various AWACS tasks such as flights, simulator sessions, ground training classes and other squadron duties while aiming to maximise combat readiness and minimise operational costs. We applied our proposed approach to several test instances consisting notional weekly schedules of Turkish Boeing 737 AEW&C Peace Eagle AWACS Base, generated similar to real-world examples. To test the algorithm and evaluate solution performance, we developed a novel scheduling software called AWACS Crew Scheduling (ACS) as a test bed.

Several experiments were planned and carried out to monitor solution performance with different GA parameters such as population size, crossover strategy, crossover rate, mutation rate and elitism rate. In this context, each parameter has been examined by keeping other parameters constant and the results were recorded.

With the solutions obtained from the experiments are examined; Genetic Algorithms have proven to be quite successful in solving the AWACS Crew Scheduling Problem. In this scope;

(a) The defined hard constraints have been 100% satisfied in all experiments and at least 80% of the entire population for each experiment was found to meet the entire hard constraints.

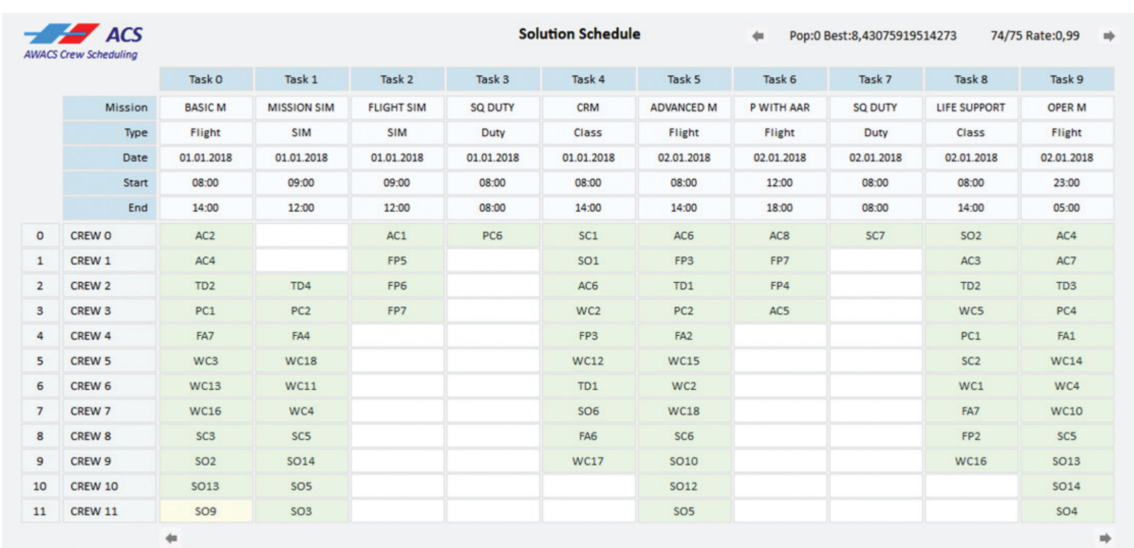
(b) Based on the performance chart to analyse soft constraints, it has been observed that in a majority of experiments, aircrew are assigned to proper tasks, with more than 90% success, defined as high priority in COT Status tables. With optimised parameters, this ratio has increased up to 100%.

(c) Especially for refresher trainings, the GA was able to assign a complete list of aircrew required to maintain currency.

(d) The solution durations have been at an acceptable level, meeting the hard constraints only in seconds and fulfilling the soft constraint requirements in minutes.

In addition to the results obtained above, some COT training events, which is still difficult to be managed by manual methods, can easily be followed by the software and proper assignments can be made with the proposed algorithm. For example, the training events named as 'LinkSO' and 'Runner' for the SO task position is not only task-based, but also seat-based roles in aircraft. In other words, the assignment of the individual to that flight is not sufficient by itself but needs to be assigned to the corresponding seat. The algorithm has successfully scheduled these priorities and has been able to meet other requirements not currently included in the scope of the directives but required for the success of the tasks.

We estimate that, in practice, the algorithm will generate very fast and robust schedules when it is considered that the generated artificial schedule contains only test-oriented and difficult-to-solve criteria that are rarely encountered under real conditions.



		Task 0	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9
	Mission	BASIC M	MISSION SIM	FLIGHT SIM	SQ DUTY	CRM	ADVANCED M	P WITH AAR	SQ DUTY	LIFE SUPPORT	OPER M
	Type	Flight	SIM	SIM	Duty	Class	Flight	Flight	Duty	Class	Flight
	Date	01.01.2018	01.01.2018	01.01.2018	01.01.2018	01.01.2018	02.01.2018	02.01.2018	02.01.2018	02.01.2018	02.01.2018
	Start	08:00	09:00	09:00	08:00	08:00	08:00	12:00	08:00	08:00	23:00
	End	14:00	12:00	12:00	08:00	14:00	14:00	18:00	08:00	14:00	05:00
0	CREW 0	AC2		AC1	PC6	SC1	AC6	AC8	SC7	SO2	AC4
1	CREW 1	AC4		FP5		SO1	FP3	FP7		AC3	AC7
2	CREW 2	TD2	TD4	FP6		AC6	TD1	FP4		TD2	TD3
3	CREW 3	PC1	PC2	FP7		WC2	PC2	AC5		WC5	PC4
4	CREW 4	FA7	FA4			FP3	FA2			PC1	FA1
5	CREW 5	WC3	WC18			WC12	WC15			SC2	WC14
6	CREW 6	WC13	WC11			TD1	WC2			WC1	WC4
7	CREW 7	WC16	WC4			SO6	WC18			FA7	WC10
8	CREW 8	SC3	SC5			FA6	SC6			FP2	SC5
9	CREW 9	SO2	SO14			WC17	SO10			WC16	SO13
10	CREW 10	SO13	SO5				SO12				SO14
11	CREW 11	SO9	SO3				SO5				SO4

Figure 9. Final schedule obtained with ACS.

In future studies it is recommended to develop optimisation techniques to shorten the solution time of the algorithm and to model the dynamic criteria AWACS aircrew need.

INTEREST OF CONFLICT

'We hereby declare that all technical and military information in this paper is '*unclassified*' and does not include any classified information belonging to any person, country or legal entity. We declare that specifications for AWACS systems mentioned in this work are retrieved from open sources which have been released to international public access'.

REFERENCES

1. Mevlutoglu, A. Commentary on assessing the Turkish Defense Industry: Structural issues and major challenges. *Defence Studies*, 2017. **17**(3), 282-294. doi: 10.1080/14702436.2017.1349534
2. Milligan, M.K. Implementing COTS open systems technology on AWACS. *Crosstalk: J. Def. Software Eng.*, 2000, pp...
3. Chen, E. The future of AWACS: Technological Advancement or Technological Relic. 2017, Air Command and Staff College, Air University Maxwell AFB United States.
4. Delaney, W.P. Air Defense of the United States: Strategic Missions and Modern Technology. *International Security*, 1990, **15**(1), 181-211. doi: 10.2307/2538986
5. Christopher, S. Active electronically-steered array surveillance radar: Indian value addition. *Def. Sci. J.*, 2010, **60**(2), 184-188. doi: 10.14429/dsj.60.338
6. Van Deventer, R. Airborne warning and control system (AWACS) and space: A framework to help understand the issues. Air Univ Maxwell AFB School of Advanced Airpower Studies, 2000.
7. Elliott, L.R.; Cardenas, R. & Schiflett, S.G. Measurement of AWACS Team Performance in Distributed Mission Scenarios. Air Force Research Lab Brooks AFB TX, 1999. doi: 10.1037/e438122006-001
8. Colegrove, C.M. & Bennett Jr, W. Competency-based training: Adapting to warfighter needs. Air Combat Command Langley AFB VA Flight Operations Division, 2006. doi: 10.21236/ada469472
9. Evans, S.E. Improving the cost efficiency and readiness of MC-130 aircrew training. The Pardee RAND Graduate School: Santa Monica, 2015, pp. 146.
10. Kawakami, T. An aid for flight squadron scheduling. Naval Postgraduate School (NPS), Monterey, California. (MSc. Thesis), 1990.
11. Gokcen, O.B. Robust aircraft squadron scheduling in the face of absenteeism. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AirForce Base, Ohio. 2008. (MSc. Thesis)
12. Durkan, M. Multi-objective decision analysis for assignment problems. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson Air Force Base, Ohio. (MSc. Thesis), 2011.
13. Van Brabant, J.D. A monthly squadron sortie scheduling model for improved combat readiness. Naval Postgraduate School (NPS), Monterey, California. 1993. (MSc Thesis)
14. Nguyen, C.T. An interactive decision support system for scheduling fighter pilot training. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 2002. (MSc Thesis)
15. Newlon, T.M. Mathematical programming model for fighter training squadron pilot scheduling. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 2007. (MSc Thesis)
16. Brown, R.P. Optimizing readiness and equity in marine corps aviation training schedules. Naval Postgraduate School: Monterey, California. 1995. (MSc. Thesis)
17. Yavuz, M. Optimizing an F-16 squadron weekly pilot schedule for the Turkish Air Force. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 2010. (MSc. Thesis)
18. Boyd, Jay A.H.; Case, A. Cunningham; Darren, P. Gray; & John, L. Parker. Network flow model for optimizing fighter squadron scheduling. Dayton: Graduate Research Project, Air Force Institute of Technology, 2006.
19. Vestli, M.; Lundsbakken, L.; Fagerholt, K. & Hvattum, L.M. Scheduling fighter squadron training missions using column generation. *Optimization Letters*, 2015, **9**(8), 1659-1674. doi: 10.1007/s11590-014-0794-y
20. Sevimli, A. Sortie generation simulation of a fighter squadron. 2016, Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. (MSc. Thesis)
21. Aslan, D., A decision support system for effective scheduling in an F-16 Pilot Training Squadron. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson Air Force Base, Ohio. (MSc. Thesis), 2003.
22. Dyer, D.E. A visual programming methodology for tactical aircrew scheduling and other applications. Defense Technical Information Center, 1991. doi: 10.21236/ada244635
23. Erdemir, U. Optimizing flight schedules by an automated decision support system. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 2014. (MSc Thesis)
24. Shirley Jr, M.D. An Improved Heuristic for Intercontinental Ballistic Missile Crew Scheduling. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 1994. (MSc Thesis)
25. Boke, C. Combining and Analyzing the Tanker and Aircrew Scheduling Heuristics. Air Force Institute of Technology (AFIT) School of Engineering and Management: Wright Patterson AFB, Ohio. 2003. (MSc Thesis)

26. Mitchell, M. An introduction to genetic algorithms. MIT Press, 1998, pp. 221.
doi: 10.1002/cplx.6130010108
27. Holland, J. Adaptation in artificial and natural systems. Ann Arbor: The University of Michigan Press, 1975, pp. 232.
doi: 10.7551/mitpress/1090.001.0001
28. Angeline, P.J. Evolution revolution: An introduction to the special track on genetic and evolutionary programming. *IEEE Expert: Intelligent Syst. Applications*, 1995, **10**(3), 6-10.
doi: 10.1109/MIS.1995.10027
29. Davis, L. Handbook of Genetic Algorithms. Van Nostrand Reinhold, 1991, pp. 385.
doi: 10.1016/s0004-3702(98)00016-2
30. Belew, R. Evolving networks: Using the genetic algorithm with connectionist learning. *In* Proceedings of the Second Artificial Life Conference, Addison-Wesley, 1991, pp. 511-547.
31. Goldberg, D.E. & Holland, J.H. Genetic algorithms and machine learning. *Machine Learning*, 1988, **3**(2), 95-99.
doi: 10.1023/a:1022602019183
32. Kearney, William T., Using genetic algorithms to evolve artificial neural networks. *Honors Theses*, 2016, Paper 818.
33. Michalewicz, Z., Evolution strategies and other methods. *In* Genetic Algorithms+Data Structures=Evolution Programs. Springer, 1996, pp. 159-177.
doi: 0.1007/978-3-662-02830-8_9
34. Montana, D.J. & Davis, L. Training feedforward neural networks using genetic algorithms. *In* International Joint Conference on Artificial Intelligence, 1989.
35. Schaffer, J.D.; Whitley, D. & Eshelman, L.J. Combinations of genetic algorithms and neural networks: A survey of the state of the art. *In* IEEE International Workshop on Combinations of Genetic Algorithms and Neural Networks, 1992, pp. 1-37.
doi: 10.1109/cogann.1992.273950
36. Sexton, R.S. & Sikander, N.A. Data mining using a genetic algorithm- trained neural network. *Intelligent Syst. Accounting, Finance Management*, 2001, **10**(4), 201-210.
doi: 10.1002/isaf.205
37. Goldberg, D.E. & Deb, K. A comparative analysis of selection schemes used in genetic algorithms. *Foundations Genetic Algorithms*, 1991, **1**, 69-93.
doi: 10.1016/b978-0-08-050684-5.50008-2
38. Kitano, H. Empirical studies on the speed of convergence of neural network training using genetic algorithms. *In* AAAI, 1990.
39. Miller, G.F.; Todd, P.M. & Hegde, S.U. Designing neural networks using genetic algorithms. *In* International Conference on Genetic Algortihms, 1989.
40. Holland, J.H. Genetic algorithms and classifier systems: Foundations and future directions. Michigan Univ., Ann Arbor (USA), 1987.
doi: 10.1016/0004-3702(89)90050-7
41. Vidulich, M.A.; Nelson, W. Todd; Robert, S. Bolia; Nicole, M. Guilliams & Annie, B. McLaughlin. An Evaluation of speech controls for AWACS Weapons Directors. 2004, Sytronics Inc Dayton Oh.
doi: 10.21236/ada423447
42. Rao, P. Computers in air defence. *Def. Sci. J.*, 1987, **37**(4), 507-513.
doi: 10.14429/dsj.37.5950

CONTRIBUTORS

Mr Hamit Taner ÜNAL received his MSc from Selcuk University, in 2018 and currently pursuing his PhD at IT Engineering at Selcuk University, Faculty of Technology, Konya, Turkey. Currently works as Head Software Developer and Data Scientist at Jetkite Inc and Jetkite. AI. His research interests include : Advances in artificial intelligence, machine learning, evolutionary algorithms, simulation systems and deep learning.
Contribution in the current study, he conducted thorough literature review, collected data and analysed existing methods with peer groups and by using his previous AWACS experience. Developed the ACS software and implemented the GA solution. Conducted experiments and documented results.

Prof. (Dr) Fatih Başçiftçi, received his MSc from Department of Education in Computer Systems, Selcuk University, and PhD at Electronics Engineering at the same institution. Presently working as Professor and Head of Computer Engineering Department at Selcuk University, Faculty of Technology, Konya, Turkey. He has been engaged extensively in research works in the fields of computer science, information systems, computer hardware and education of Computer Systems. He has published many refereed papers in the academic journals and conference proceedings. His recent research interests cover artificial intelligence, evolutionary algorithms, IoT and deep learning.
Contribution in the current study, he provided general outline and managed development and implementation of proposed solution to ACS software. Created testing plan and provided feedbacks for obtained results. Provided insights for abstract and summary.