Sādhanā Vol. 41, No. 2, February 2016, pp. 183–191

A new intelligent approach for air traffic control using gravitational search algorithm

KAZEM DASTGERDI[1], NASSER MEHRSHAD[2,*] and MOHSEN FARSHAD[2]

[1]Department of Electronics, Operational Aviation Branch, Birjand International Airport, South Khorassan Airports General Administration, Birjand, Iran
[2]Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran
e-mail: k.dastgerdi@airport.ir; nmehrshad@birjand.ac.ir; mfarshad@birjand.ac.ir

MS received 17 January 2014; revised 4 April 2014; accepted 24 April 2015

Abstract. Aircraft landing planning (ALP) is one of the most important challenging problems in the domain of air traffic control (ATC). Solving this NP-hard problem is a valuable aid in organizing air traffic in terminal control area (TCA), which itself leads to a decrease in aircraft fuel consumption, costs of airlines, and workload undertaken by air traffic controllers. In the present paper, the ALP problem is dealt with by applying effective rich knowledge to the optimization process (to remove obvious non-optimal solutions), and the first use of Gravitational Search Algorithm (GSA) in resolving such a case. In this regard, while the specific regulations for safe separation have been observed, the optimal landing time, the optimal runway, and the order of consecutive landings have been determined so that the main goal (minimizing total flight delays) would be best met. Results of simulations show that this approach, compared to previous ones, which are based on Genetic and Bionomic algorithms, GLS, and Scatter search method, considerably decreases total flight delays. Attaining zero in the total flight delays in three scenarios with real data shows that the suggested intelligent approach is more decisive than others in finding an optimal solution.

Keywords. GSA; minimizing total flight delays; ALP; minimum separation time; TCA.

1. Introduction

Considering many advantages that air travels nowadays have-including convenience, cost, high speed and safety-air traffic has achieved significant growth and it is expected that this growth promotes even more in the following years [1]. The air traffic congestion appears to be due to the imbalance between air transportation demand, the capacity of routes, sectors, terminal control area (TCA), and airports. Therefore, poor management of this congestion may lead to a lot of flight delays, increase of operational errors by air traffic control personnel (controllers), and passengers' dissatisfaction [2].

The growing trend of traffic congestion and many problems emerging from it made aviation industry specialists look for appropriate and practical solutions for optimal control of air traffic in various phases, one of the most important of which is aircraft landing problems. Obviously, in the case of having few aircrafts under the area of radar coverage, or long intervals between planned landing times (PLT), controllers are easily able to simply and appropriately plan for the aircrafts landing. However; if the traffic congestion under the controlled area exceeds a specific limit, with regard to the difficulty of the aircraft landing planning (ALP) problem (i.e. it is being NP-hard, non-linear and non-convex), controllers encounter many problems in managing air traffic [3, 4]. The controllers' major problems in these conditions are enormous workload and calculative complexity to find an optimal solution, which can cause improper decisions in due course while intending to control traffic congestion of the TCA [5].

In this regard, solutions like constructing new runways and airports used to be considered, but even these solutions are too expensive, time-consuming and-in some situations-probably impossible. Thus, such problems led to applying of computer planning by those in charge of aviation industry. Usual computer planning methods and some other methods like "First Come First Served" (FCFS) did not produce optimal results. These made researchers use the available capacities in airports optimally to minimize total flight delays and, simultaneously, maximize the admitted number of aircrafts into airports. They used intelligent methods based on random search algorithms for planning.

Cheng *et al* [6] presented four Genetic search methods for the air traffic control (ATC). Beasley *et al* [7] presented general description of the model along with its objectives and mathematical formulation for the ALP in both one and several runways. In addition, regarding ALP,

*For correspondence

many studies have been made with respect to planning and determining the order of consecutive landings in order to minimize deviation of the allocated landing time (ALT) from the PLT [8–11], and decreasing the duration of scheduling [12, 13]. Hansen [3], Hu & Paolo [14], Tang *et al* [15], Salehipour *et al* [16], and Yu *et al* [17] presented four Genetic search methods, one effective Genetic algorithm using uniform crossover, MONSDE, variable neighborhood descent (VND) algorithm, and cellular automata optimization (CAO), respectively. On hybrid methods of different optimization algorithms, Jia *et al* [18] introduced the algorithm of CSA-RHC. In this hybrid algorithm, which is based on Clonal Selection Algorithm and Horizon Control Receding, techniques of excellent gene segment spread (EGSS) and infeasibility degree (IFD) have been, respectively, used to accelerate the algorithm converging trend and guiding the optimization process. Another hybrid intelligent method for the ATC was based on genetic algorithms and Ant Colony Optimization (ACO) that was presented by [19].

The present paper, by presenting a new approach; i.e. adding rich knowledge to the optimization process, maintains that GSA can be used as a new optimization means to attain intelligent flight planning with minimum flight delays. In the second section of this paper, base notions and mathematical formulations of the problem are presented. In the third section, a summary of the GSA is given. In the fourth section, the suggested intelligent approach introduced in the paper is elaborated on, as a means of the optimal control of air traffic congestion. In the fifth section, simulations are introduced in four different scenarios (using objective data). Then, the results of the suggested approach are compared to those of the last methods. Finally, in the sixth section, our conclusion is arrived at.

2. Base notions of the problem and its mathematical formulations

In this part, first we present the underlying notions and thereafter the mathematical ALP relations. A main goal of intelligent landing scheduling is minimizing the difference between PLT and ALT. PLT is determined based on the kind and speed of any aircraft, its distance from the destination airport, weather conditions, and traffic of the specific air route. Thus, if ALT is equal to PLT, the flight delay is zero. However, this may be hard to achieve due to traffic congestion in the TCA or non-optimal landing scheduling. In aviation industry, one of the common solutions to reduce delays in landing operations is a holding maneuver that keeps the aircrafts of the TCA on standby until the ALT for each. The speed, the fuel of any aircraft, and traffic congestion in the controlled air space have significant roles while performing the holding maneuver.

Since airlines' scheduling is a consecutive orderly process, for each aircraft there is a determined flying time range and ground services in airports. Obviously, having too much delay in some flight operations may lead to other flight delays of that airline. Based on this, the main goal is minimizing total delays. If so, and considering the prediction of probable flight delays by those in charge of flight planning of airlines, hopefully no disruption will happen to the following flight plans and extra costs will significantly decrease, as well [20].

According to the formulation presented by [16], having defined the following parameters, we discuss the formulation of the ALP:

N : The number of aircrafts that are landing in a congested airport,
M : The number of operational runways,
PLT_i : The planned landing time for the aircraft $i(i = 1, 2, ..., N)$,
ALT_i : The allocated landing time for the aircraft $i(i = 1, 2, ..., N)$,
SP_{ij} : The minimum of safe separation time between the consecutive landings of the aircrafts i and $j(i \& j = 1, 2, ..., N)$.

Using Eq. (1), as shown, we can calculate the delay of the aircraft i in the runway r.

$$\Delta_{ir} = \begin{cases} ALT_{ir} - PLT_{ir} & \text{if } ALT_{ir} \rangle PLT_{ir} \\ & (i = 1, 2, ..., N) \,\&\, (r = 1, 2, ..., M)\,. \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The parameter of determining the order of aircrafts landing is calculated by means of Eq. (2). The parameter has been defined so that it would cover the features of Eqs. (3) and (4):

$$Seq_{ij} = \begin{cases} 1 & \text{if aircraft } j \text{ lands after aircraft } i \\ & (i, j = 1, 2, ..., N);\ i \neq j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$Seq_{ij} + Seq_{ji} = 1 \quad (i, j = 1, 2, ..., N);\ i \langle j \quad (3)$$

$$\sum_{i=1}^{N} \sum_{j=1,\, j \neq i}^{N} Seq_{ij} = \frac{1}{2}(N^2 - N). \quad (4)$$

The index of a common runway for the aircrafts i and j is defined by Eq. (5). Having this equation in mind, we find out that this index contains the mentioned feature in Eq. (6).

$$SL_{ij} = \begin{cases} 1 & \text{if aircraft } i \text{ and } j \text{ lands on the same runway} \\ & (i, j = 1, 2, ..., N);\ i \neq j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

$$SL_{ij} = SL_{ji} \quad (i, j = 1, 2, ..., N);\ i \langle j. \quad (6)$$

The index of the aircraft i which is to be matched with the runway r is introduced by means of Eq. (7). It contains the mentioned features in Eqs. (8) and (9), too.

$$RL_{ir} = \begin{cases} 1 & \text{if aircraft } i \text{ lands on runway } r \\ & (i = 1, 2, ..., N);\ (r = 1, 2, ..., M) \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$\sum_{r=1}^{M} RL_{ir} = 1 \quad (i = 1, 2, ..., N); \ (r = 1, 2, ..., M) \quad (8)$$

$$RL_{ir} + RL_{jr} - 1 \leq SL_{ij} \quad i\langle j; \ (i, j = 1, 2, ..., N); \ (r = 1, 2, ..., M) \quad (9)$$

Regarding the reliability of the solution of ALP, the following condition must be met:

$$PLT_i \leq ALT_i \quad ; \ (i = 1, 2, ..., N) \quad (10)$$

SP_{ij} is another important statement which should be considered to ensure the safety of aviation. This condition causes a minimum safe separation interval between consecutive flights. Satisfying this condition helps to keep aerodynamic stability of a trailing aircraft under the turbulence caused by the leading aircraft [21]. Therefore, based on the types of aircrafts and established standards and instructions, a minimum safe separation time between consecutive landings can be attained using the following matrix [3]:

$$SP_{ij} = \begin{bmatrix} 1.0 & 1.0 & 1.0 \\ 1.5 & 1.5 & 1.0 \\ 2.0 & 1.5 & 1.0 \end{bmatrix}. \quad (11)$$

In this matrix, the rows represent leading aircrafts, and the columns trailing ones. In addition, the entries in the rows and columns stand for small, large, and heavy aircrafts, respectively. For example, if the leading aircraft is a heavy Boeing 747 and the trailing one is a small Islander, at least two separation time units must be met between the two.

3. The gravitational search algorithm

The gravitational search algorithm (GSA), as one of the new random optimization algorithms, was presented by [22]. In designing this algorithm, Newton's laws of gravitation and motion have been used for a search space of split time and the searching factors are a collection of mass [23]. Each piece of mass (agent) has the important characteristics of position, speed, inertial mass, active and inactive gravitational mass. In this algorithm, gravity force is used as a means of transferring information between masses [24, 25].

For the GSA, first, the artificial system space, i.e. the very problem-defining range, is distinguished. Then, the mass of agents is calculated based on the target function. After this, considering the fitness of agents, the inertial and gravitational masses (active and inactive) are distinguished. Every solution of the optimization problem is definable in the form of a position or point in the search space, whose similarity to other solutions is specified as a space. Each mass, through the received force from other masses, can nearly understand the surrounding space. The algorithm must be guided so that the position of masses would improve as time is going by. The strategy is based on the mass regulation of the agents and since all the agents interact with each other regarding their mass, their interactions with the neighboring agents are more. The masses having better fitness functions have greater gravitational mass and interaction range with other masses. Therefore, as time is going by the masses are guided towards better positions. The agents in better positions are to take shorter steps. To this end, more inertial mass is attributed to more suitable masses which causes each agent to explore its surrounding space more precisely [26, 27].

Based on this, for a system with n masses and m dimensions, the position of the mass i from the dimension j is definable in the form of X_i^j, applying the following equation:

$$X_i^j = (X_1^1, \ldots, X_n^m) \qquad \text{that } (i = 1, 2, \ldots, n); \ (j = 1, 2, \ldots, m). \quad (12)$$

In the very beginning, all the masses randomly get located and are then driven into each other. At the time t and towards the dimension d, a force as much as $F_{ij}^d(t)$ from the mass j is applied to the i mass, the amount of which can be calculated using the following equation:

$$F_{ij}^d = G(t)\frac{M_{pas.i}(t).M_{act.j}(t)}{R_{ij}(t) + \varepsilon}(X_j^d(t) - X_i^d) \quad (13)$$

$$R_{ij}(t) = \left\| X_i(t), X_j(t) \right\|_2. \quad (14)$$

Here, $M_{pas.i}(t)$, $M_{act.j}(t)$, ε, $G(t)$, and $R_{ij}(t)$ are the passive gravitational mass of the agent i, the active gravitational mass of the agent j, a very small number, the gravitational constant at time t and the space between the two agents i and j, respectively. Therefore, the applied force into the mass i towards the dimension$d$ and at the time $t(F_i^d(t))$ is equal to the total applied forces from the other masses of the system on this mass that are definable by the following equation:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{n} rand_j . F_{ij}^d(t), \quad (15)$$

where $rand_j$ is a uniformly distributed random number in the range $[0, 1]$, to keep the random feature of the algorithm. Now, according to Newton's law of motion, the acceleration of the agent i towards the dimension of d and at the time t is definable on the basis of the following equation.

$$a_i^d(t) = \frac{F_i^a(t)}{M_{int.i}(t)} \quad (16)$$

where $M_{int.i}(t)$ is the inertial mass of the agent i. On the other hand, the speed and the new position of the agent i in the dimension d are expressed by Eqs. (17) and (18), respectively.

$$v_i^d(t + 1) = rand_i . v_i^d(t) + a_i^d(t) \quad (17)$$

$$x_i^d(t + 1) = x_i^d(t) + v_i^d(t + 1). \quad (18)$$

Here, $v_i^d(t)$ is the velocity in dimension d of the agent i at the time t. Moreover, $rand_i$ is a uniformly distributed random number in the range $[0, 1]$ to keep the random feature of the algorithm.

The gravitational constant $(G(t))$ is an appropriate parameter to control the exploration and productivity capacities in the gravitational optimization (GO) algorithm. The small and the large amount of this parameter improve the productivity and the search capacity of the algorithm, respectively. Because in the initial stages it is expected from the algorithm to search new positions, and in the final stages to improve the specified solutions using productivity capacity. An appropriate option for the gravitational fixed parameter is that it must start with a large initial amount, but decrease as time is going by. Regarding it various experiments done, the use of the following exponential equation is effective to decrease the gravitational constant in many problems [22]:

$$G(t) = G_0 \, e^{-\alpha \frac{t}{T}}, \tag{19}$$

where α and G_0 are fixed positive coefficients and T is the total algorithm repetitions.

In GSA, computing the masses of agents is done regarding the object function so that more mass would be attributed to the more fitted agents. The inertial masses and active or passive gravitational masses, for the agent i are named $M_{int.i}$, $M_{act.i}$ and $M_{pas.i}$, respectively. These masses, like what is in the nature, are accounted equal for each searching agent and, thus, updating the inertial and gravitational masses is done through the following equations:

$$M_{int.i} = M_{act.i} = M_{pas.i} = M_i; \quad (i = 1, 2, \ldots, n) \tag{20}$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst\ (t)} \tag{21}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{n} m_j(t)}, \tag{22}$$

where $fit_i(t)$ shows the extent of fitness in the agent i at time t, and $best(t)$ and $worst(t)$ in the minimizing problems are, respectively, computable through the Eqs. (23) and (24). It is worth mentioning that in the maximizing problems, the position of *the best* and *the worst* is exchanged in the following equations:

$$best(t) = \min_{j \in \{1,2,\ldots,\ n\}} fit_j\ (t) \tag{23}$$

$$worst(t) = \max_{j \in \{1,2,\ldots,\ n\}} fit_j(t). \tag{24}$$

Therefore, in general, we can assume the following steps for a standard GSA:

1. Determining the environment of the system and initialization.
2. The initial locating of the masses.
3. Evaluating the fitness of the masses.
4. Updating the parameters $G(t)$ '$best(t)$' $worst\ \ (t)$and $M_i(t)$ for $i = (1, 2, \ldots,\ n)$.
5. Calculating the applicable force into each mass.
6. Calculating the acceleration and speed of each mass.
7. Updating the position of masses.
8. If the stopping condition is met, show the best seen solution up to then in the output and stop the algorithm; otherwise, return to step 3.

Since the above-mentioned gravitational algorithm is statistically more complicated than other ones like PSO, to decrease the complexity in each duplication of the algorithm. Only k superior masses of this population have the potentiality of exerting force on other masses. This leads to the prevention of the less fitted factors inclusion in determining the extent of other members' motions. Thus, Eq. (15) is changed into the following one:

$$F_i^d(t) = \sum_{j \in kbest,\ j \neq i}^{n} rand_j . F_{ij}^d(t), \tag{25}$$

where $kbest$ shows the collection of k containing the superior mass of the population.

4. The introduction of the suggested intelligent approach

In the suggested intelligent approach, since the main goal is minimizing the total flight delays, the following fitness function is used:

$$Fitness = \sum_{i=1}^{N} \sum_{r=1}^{M} \Delta_{ir}, \tag{26}$$

where Δ_{ir} is the delay in landing of the aircraft i on the runway r, which is definable through Eq. (1). The prerequisite for attaining the minimum flight delays is optimal allocation of the runway, landing scheduling, and the order of consecutive flights so that the specific safe separation standards would be achieved, too.

Traffic data of the problem includes flights call signs, number and kind of aircrafts (S, L, and H), number of runways, and PLT. The suggested intelligent process to control air traffic congestion in TCA takes place in a way that the intelligent algorithm initially allots a random runway to each flight. Then, the flights allotted to each runway are arranged with respect to PLT. In the next step, considering the kinds of aircrafts and the presented matrix SP_{ij} in the 2nd section of the paper (Eq. 11), the minimum safe separation time between consecutive landings is studied. If the allocated landing times in each runway are such that the minimum safe separation time between consecutive landings is not taken into account in proportion to the entries of the matrix SP_{ij},

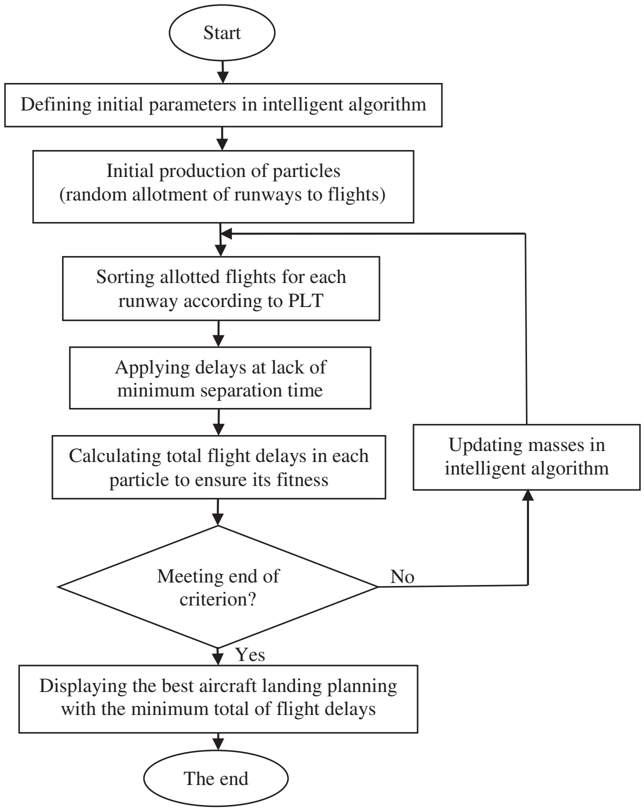

Figure 1. General flowchart for intelligent approach in solving ALP problem.

the landing of some aircrafts is delayed. Having determined the fitness of an allotment, and after a definite number of duplications, the GSA will be after finding the best solution to the problem. This suggested process leads to the improvement of the capacity of the intelligent algorithm in removing obvious non-optimal solutions. For example, if the number of the GSA masses is 30 (in a scenario with data consisting of 12 aircrafts and 3 runways), the GSA produces 30 twelve-dimension particles. Each number specific to these dimensions shows one of the three allotted runways to the landings in either of the runways 1, 2, and 3. Then, according to the general flowchart in figure 1, the next steps of the intelligent planning process are arranged.

5. Results of simulations

In this part, the suggested intelligent approach is simulated within the framework of four different scenarios. To simulate the suggested methods, the traffic data of Dallas Fort Worth airport (DFW) in Texas, United States, and the formerly used data in Hansen Study [3], were used. The main reason for using these traffic data is to compare the results of the suggested intelligent approach with those of the previous optimizing methods.

5.1 *Scenario* 1

Nowadays, many congested airports have several runways. In some airports, these runways lie in parallel, oblique, or both forms. Considering table 1, at first, in the first scenario, we study and simulate a problem consisting of 12 aircrafts, and 3 runways. In this problem, kinds of aircrafts and their PLT for each runway are determined based on table 1.

In this scenario, the initial population with 50 masses has randomly been produced in the search space. α and G_0 have taken the amounts 1 and 10, respectively. The intelligent GSA plans landings so that, for each aircraft, while considering the PLT for each runway, that runway would be allocated which would have the minimum delay for that flight. Besides, the total flight delays of planning procedure will also be minimized. In this regard, having allocated a runway to the first flight and having planned its landing to determine the order of next flights, we must also consider the minimum safe separation time.

Notice that the entries the matrix SP_{ij} in part two, and PLT in the traffic data, are in time units. These units, depending on the airport and different conditions in the field of

Table 1. PLT values for senario 1 (12 airplanes and 3 runways).

		PLT_{ir}		
Call-sign	***Airplane-type***	***Runway 1***	***Runway 2***	***Runway 3***
DL130	H	12	11	10
AA335	S	15	17	19
UA123	H	7	9	8
DL1920	H	6	7	8
UA1133	L	10	13	15
NW2123	H	7	6	5
AA205	L	15	17	19
DL3319	H	7	8	9
SW200	S	6	7	8
DL510	H	9	12	15
UA410	H	6	5	4
SW185	L	9	7	6

Table 2. Sample of results from intelligent landing planning through GSA for scenario 1.

Call-sign	***Assigned runway***	ALT_{ir}	Δ_{ir}
SW200	Runway 1	6	0
NW2123	Runway 1	7	0
SW185	Runway 1	9	0
DL130	Runway 1	12	0
AA335	Runway 1	15	0
UA410	Runway 2	5	0
DL1920	Runway 2	7	0
DL3319	Runway 2	8	0
UA123	Runway 2	9	0
UA1133	Runway 2	13	0
AA205	Runway 2	17	0
DL510	Runway 3	15	0
	Total delays=**0**		

Table 3. Result from running suggested intelligent approach for 50 times for scenario 1.

Total delays				Computational time (s)
Worst	*Best*	*Mean*	*Std. Dev.*	
1.5	**0**	0.34	0.46	0.21

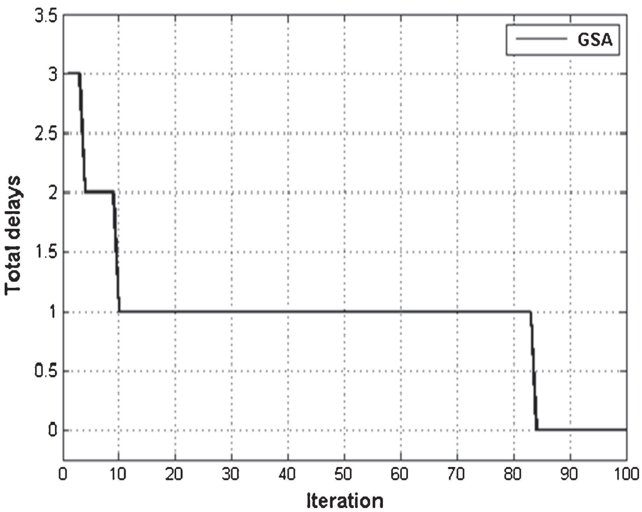

Figure 2. Sample of convergence of GSA for senario1.

aviation operation, can take 2-min, 5-min periods, etc. In table 2, a sample of the best results attained from the intelligent landing planning using the suggested approach for scenario 1 is illustrated.

The results of table 2 confirm that through meeting the required standards in planning (including the minimum safe separation), the total flight delays are rated as "zero" in this scenario; something which demonstrates reaching an optimized solution to the problem. In the following, some results, as shown in table 3, were obtained after 50 times running of the intelligent program for scenario 1. In figure 2, a sample of convergence manner by the GSA for scenario 1 is represented.

5.2 *Scenario* 2

It may be impossible to increase the number of runways in a congested airport so that the traffic congestion problem in the aircraft landing operation would be solved. Therefore, it is necessary that using optimization techniques, we control air traffic congestion in a way that while having a limited number of runways, minimum delay and maximum productivity from the same capacities would result. Thus, the impact of increasing the number of flights on the ALP problem using real data should be studied. As shown in table 4, after increasing the number of flights by 25 percent more than that of scenario 1, and not changing the number of runways, the analysis and simulation of the suggested approach in a new problem will start. In this scenario, the initial population is 40; α, G_0 are 20 and 170, respectively.

Table 4. The PLT values for the senario 2 (15 airplanes and 3 runways).

		PLT_{ir}		
Call-sign	***Airplane-type***	***Runway 1***	***Runway 2***	***Runway 3***
DL130	H	12	11	10
AA335	L	15	17	19
UA123	H	7	9	8
DL1920	H	6	7	8
UA1133	S	10	13	15
NW2123	H	7	6	5
AA205	L	15	17	19
DL3319	H	7	8	9
SW200	H	6	7	8
DL510	S	9	12	15
UA410	H	6	5	4
SW185	L	9	7	6
DL200	S	7	8	9
NW410	L	6	7	8
AA1225	H	9	8	7

Table 5. Sample of results from intelligent landing planning through the suggested approach for scenario 2.

Call-sign	***Assigned runway***	ALT_{ir}	Δ_{ir}
SW200	Runway 1	6	0
UA123	Runway 1	7	0
SW185	Runway 1	9	0
AA205	Runway 1	15	0
NW410	Runway 2	7	0
DL200	Runway 2	8.5	0.5
UA1133	Runway 2	13	0
UA410	Runway 3	4	0
NW2123	Runway 3	5	0
AA1225	Runway 3	7	0
DL1920	Runway 3	8	0
DL3319	Runway 3	9	0
DL130	Runway 3	10	0
DL510	Runway 3	15	0
AA335	Runway 3	19	0
Total delay=**0.5** (time unit)			

Table 6. Results from 50 times running of the suggested intelligent planning for scenario 2.

Total delays				Computational time (s)
Worst	*Best*	*Mean*	*Std. Dev.*	
3	**0.5**	1.45	0.71	43.25

Taking into consideration the more difficult conditions of aircraft landing in scenario 2 (an increase by 25 percent in the number of flights), the presented results in table 5, which show the total flight delays as much as 0.5 time unit, imply the high capacity of the suggested intelligent approach to lead to an optimal solution to the problem.

Following this, in table 6 the results from 50 times running of the intelligent planning for scenario 2 are presented.

Table 7. The PLT values for senario 3 (20 airplanes and 5 runways (R1–R5)).

		PLT_{ir}				
Call-sign	***Airplane-type***	***Runway 1***	***Runway 2***	***Runway 3***	***Runway 4***	***Runway 5***
DL130	H	12	11	10	10	9
AA335	L	15	17	19	18	18
UA123	H	7	9	8	7	8
DL1920	H	6	7	8	7	8
UA1133	S	10	13	15	15	14
NW2123	H	7	6	5	6	5
AA205	L	15	17	19	19	18
DL3319	H	7	8	9	8	9
SW200	H	6	7	8	7	8
DL510	S	9	12	15	15	14
UA410	H	6	5	4	6	6
SW185	L	9	7	6	9	8
DL200	S	7	8	9	7	8
NW410	L	6	7	8	7	8
AA1225	H	9	8	7	8	9
SW442	S	10	11	10	10	11
AA127	L	6	7	8	6	7
AA1410	L	9	8	7	8	7
UA555	H	7	8	9	7	9
SW250	L	9	10	11	9	10

5.3 *Scenario* 3

In this step, in the form of a new ALP problem, and having PLT for 20 aircrafts and 5 operational runways. The information on PLT, kinds of aircrafts, and flight call-signs for the new scenario are given in table 7. In this scenario, the initial population is 20; α G_0 are 20 and 150, respectively.

Obviously, the larger the dimensions of the problem, the less probable it is to quickly arrive at optimal solutions by means of usual methods. Considering the performance difference of innovative intelligent search techniques for different problems, we need an appropriate powerful technique to minimize the total flight delays.

In table 8, a sample of the results attained from applying of the suggested approach for scenario 3, in which the specific separation limitations among flights have appropriately been taken into account, is offered. The study of the results obtained from simulations shows that intelligent landing control is effectively practical. Thus, the main goals of optimization are best met. In table 9, the results from 50 times running of the intelligent planning for scenario 3 are offered.

Table 8. Sample of results from intelligent landing planning through the suggested approach for scenario 3.

Call-sign	***Assigned runway***	ALT_{ir}	Δ_{ir}
AA127	Runway 1	6	0
UA555	Runway 1	7	0
NW2123	Runway 2	6	0
SW200	Runway 2	7	0
DL3319	Runway 2	8	0
UA123	Runway 2	9	0
SW442	Runway 2	11	0
AA205	Runway 2	17	0
UA410	Runway 3	4	0
SW185	Runway 3	6	0
AA1225	Runway 3	7	0
DL200	Runway 3	9	0
SW250	Runway 3	11	0
AA335	Runway 3	19	0
NW410	Runway 4	7	0
UA1133	Runway 4	15	0
AA1410	Runway 5	7	0
DL1920	Runway 5	8	0
DL130	Runway 5	9	0
DL510	Runway 5	14	0
	Total delays =**0**		

5.4 *Scenario* 4

Since in some congested airports, due to specific limitations like short length of some runways, it may not be possible for some specific types of aircrafts to land, this problem is also simulated. To simulate the problem, the traffic data of scenario 1 is used. In this scenario, the third runway is allotted only to the landing of S and L aircrafts.

Table 9. The results from 50 times running of the suggested intelligent approach for scenario 3.

Total delays				Computational time (s)
Worst	*Std. Dev.*	*Mean*	*Best*	
2.5	**0**	1.03	0.78	31.49

The initial population in GSA is 50; and α, G_0 are 1 and 10, respectively. The results of the simulation presented in table 10 show that with growing difficulty of the problem (allotting the third runway only to the landing of S and L aircrafts), the resulted solutions are the optimum ones for this scenario. Following this, the results from 50 times running of the intelligent program are presented in table 11.

5.5 *Comparing results*

Now, to study the feasibility of the suggested intelligent approach for ALP, applying table 12, we compare the results from the intelligent ALP using GSA, to with those from previously run methods. In this table, the computational time and the mean total delays for 50 times running have been presented. Reaching the optimal solutions (total delay of zero) shows that the suggested approach is more practical than the last presented intelligent approaches. The accuracy of the results is verifiable through the traffic data and Eqs. (11) and (26).

Table 10. Sample of results from the suggested simulation approach for scenario 4 (no heavy airplane to Runway 3).

Call-sign	*Assigned runway*	ALT_{ir}	Δ_{ir}
DL1920	Runway 1	6	0
DL3319	Runway 1	7	0
AA205	Runway 1	15	0
UA410	Runway 2	5	0
NW2123	Runway 2	6	0
UA123	Runway 2	9	0
DL130	Runway 2	11	0
DL510	Runway 2	12	0
AA335	Runway 2	17	0
SW185	Runway 3	6	0
SW200	Runway 3	8	0
UA1133	Runway 3	15	0
		Total delays =	**0**

Table 11. General results from intelligent landing planning through the suggested approach for 50 times running in scenario 4.

Total delays				Computational time (s)
Worst	*Best*	*Mean*	*Std. Dev.*	
2.5	**0**	0.48	0.59	35.57

6. Conclusion

ALP is one of the most important complicated problems in the domain of air traffic control. Considering its being non-linear, non-convex, and NP-hard, usual methods are not enough for reaching optimal practical solutions in this regard. In the present paper, as seen in the flowchart presented, rich effective knowledge was applied to the optimization process. This is done to increase the speed of convergence of the algorithm, remove the obvious non-optimal solutions, and find optimal solutions in the limited searching space created. To come up with this plan, optimal runway allocation and scheduling for consecutive landings were effectively done while achieving specific standards of aviation safety aiming at minimizing total flight delays. The simulations showed that, compared to the different previous optimization methods, total flight delays significantly decreased. Applying rich effective knowledge can improve scheduling results. This minimizing can help air traffic controllers with the process of effective aircraft-landing planning and decrease their workload, leading to a decrease in traffic congestion in TCA. From among several advantages of minimizing flight delays, the decrease in fuel consumption by aircrafts, environmental pollutants, and airline minor costs can be mentioned.

Table 12. Comparing results from intelligent landing planning through the suggested approach with results from the previously run methods in this regard.

Test case [a]	GA[a]	GA[b]	SS[c]		BA[c]		GLS[d]		GSA[e]		
	TD	TD	TD	CT	TD	CT (s)	TD	CT (s)	TD		CT (s)
									Best	Mean	
H1-12-3	3.5	1	3.75	8	3.75	49	3.5	0.24	0	0.34	0.21
H2-15-3	9	5.5	12.25	10	12.25	47	12.25	1.07	0.5	1.45	43.25
H3-20-5	12	7.65	8.75	12	9.75	49	7.75	8.50	0	1.03	31.49
H4-12-3	8.5	4.30	5.50	7	4.25	47	3.25	0.32	0	0.48	35.57

TD, total delays; CT, computational time.
[a]Genetic search (Method 4) by [3].
[b]An efficient GA with uniform crossover by [14]
[c]Scatter search and bionomic algorithm by [10]
[d]A genetic local search algorithm with a threshold accepting mechanism by [28].
[e]The proposed approach that examined in this study, running on an Intel Pentium 2.4 GHz CPU personal computer with 512 MB RAM.

References

[1] Malaek S M B and Naderi E 2008 A new scheduling strategy for aircraft landings under dynamic position shifting. *IEEE Aerospace Conference* 1–8

[2] Yifei Z and Kai C 2010 Air traffic congestion assessment method based on evidence theory. *IEEE Control and Decision Conference (CCDC)* 426–429

[3] Hansen J V 2004 Genetic search methods in air traffic control. *J. Comput. Oper. Res.* 445–459

[4] Amrahov S E and Ibrahim Alsalihe T A 2011 Greedy algorithm for the scheduling aircrafts landings. *Application of Information and Communication Technologies (AICT)* 1–3

[5] Oussedik S and Delahaye D 1998 Reducing air traffic congestion by genetic algorithms. *Parallel problem Solving from Nature — PPSN V, Lecture notes in computer science* 1498: 855–864

[6] Cheng V, Crawford L and Menon P 1999 Air traffic control using genetic search techniques. *Proceedings of the IEEE International Conference on Control Applications* 249–254

[7] Beasley J E, Krishnamoorthy M, Sharaiha Y M and Abramson D 2000 Scheduling aircraft landings –The static case. *Transport. Sci.* 34: 180–197

[8] Bianco L, Dell'Olmo P and Giordani S 2006 Scheduling models for air traffic control in terminal areas. *J. Scheduling* 223–253

[9] Beasley J E, Sonander J and Havelock P 2001 Scheduling aircraft landings at London Heathrow using a population heuristic. *J. Oper. Res. Soc.* 52: 483–493

[10] Pinol H and Beasley J E 2006 Scatter search and bionomic algorithms for the aircraft landing problem. *Eur. J. Oper. Res.* 171: 439–462

[11] Tavakkoli-Moghaddam R, Yaghoubi-Panah M and Radmehr F 2012 Scheduling the sequence of aircraft landings for a single runway using a fuzzy programming approach. *J. Air Transp. Manag.* 25: 15–18

[12] Bianco L, Dell'Olmo P and Giordani S 1999 Minimizing total completion time subject to release dates and sequence-dependent processing times. *Ann. Oper. Res.* 86: 393–415

[13] Atkin J, Burke E, Greenwood J and Reeson D 2007 Hybrid metaheuristics to aid runway scheduling at London Heathrow airport. *Transport. Sci.* 41: 90–106

[14] Hu X-B and Paolo E D 2009 An efficient genetic algorithm with uniform crossover for air traffic control. *J. Comput. Oper. Res.* 245–259

[15] Tang K, Wang Z, Cao X and Zhang J 2008 A multi-objective evolutionary approach to aircraft landing scheduling problems. *IEEE World Congress on Computational Intelligence* 3650–3656

[16] Salehipour A, Moslemi Naeni L and Kazemipoor H 2009 Scheduling aircraft landings by applying a variable neighborhood descent algorithm: Runway-dependent landing time case. *J. Appl. Oper. Res.* 1: 39–49

[17] Yu S P, Cao X B and Zhang J 2011 A real-time schedule method for aircraft landing scheduling problem based on cellular automation. *J. Appl. Soft Comput.* 11: 3485–3493

[18] Jia X, Cao X, Guo Y and Qiao H 2008 Scheduling aircraft landing based on clonal selection algorithm and receding horizon control. *Proceedings of the 11th International IEEE, Conference on Intelligent Transportation Systems* 357–362

[19] Bencheikh G, Boukachour J, El Hilali Alaoui A and El Khoukhi F 2009 Hybrid method for aircraft landing scheduling based on a Job Shop formulation. *IJCSNS Int. J. Comput. Sci. Netw. Security* 9: 78–88

[20] Forbes J F 2008 The effect of air traffic delays on airline prices. *Int. J. Ind. Organiz.* 26: 1218–1232

[21] Bojanowski L, Harikiopoulo D and Neogi N 2011 Multi-runway aircraft sequencing at congested airports. *American Control Conference (ACC)* 2752–2758

[22] Rashedi E, Nezamabadi-pour H and Saryazdi S 2009 GSA: A gravitational search algorithm. *Inf. Sci.* 179: 2232–2248

[23] Shaw B, Mukherjee V and Ghoshal S P 2012 A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems. *Electr. Power Energy Syst.* 35: 21–33

[24] Duman S, Güvenç U, Sönmez Y and Yörükeren N 2012 Optimal power flow using gravitational search algorithm. *Energy Convers. Manag.* 59: 86–95

[25] Güvenç U, Sönmez Y, Duman S and Yörükeren N 2012 Combined economic and emission dispatch solution using gravitational search algorithm. *Computer Science & Engineering and Electrical Engineering* 1–9

[26] Li C, Zhou J, Xiao J and Xiao H 2012 Parameters identification of chaotic system by chaotic gravitational search algorithm. *Chaos, Solitons Fractals* 45: 539–547

[27] Khajehzadeh M, Taha M R, El-Shafie A and Eslami M 2012 A modified gravitational search algorithm for slope stability analysis. *Eng. Appl. Artif. Intell.* 25: 1589–1597

[28] Liu Y H 2011 A genetic local search algorithm with a threshold accepting mechanism for solving the runway dependent aircraft landing problem. *Optimiz. Lett.* 2: 229–245