A Rig-Based Formulation and a League Championship Algorithm for Helicopter Routing in Offshore Transportation

Ali Husseinzadeh Kashan, Amin Abbasi-Pooya and Sommayeh Karimiyan

Abstract Crew must be transported to offshore platforms to regulate the production of oil and gas. Helicopter is the preferred means of transportation, which incurs a high cost on oil and gas companies. So, flight schedule should be planned in a way that minimizes the cost (or equivalently the makespan) while considering all conditions related to helicopter's maximum flight time, passenger and weight capacity, and arrival time of crew members to airport. Therefore, a mathematical formulation is proposed in this paper that has the objective of minimizing the finish time of the final tour of the helicopter (i.e., makespan) while taking the above-mentioned conditions into account. The model is solved for problems that have at most 11 rigs, which results in optimal solution in a reasonable amount of time. For large-size problems for which the computational effort for finding exact solution is not possible in an efficient way, a metaheuristic, namely League Championship Algorithm (LCA), is proposed. Computational experiments demonstrate that LCA can find good solutions efficiently, so it may be employed for large-size helicopter routing problems in an efficient manner.

Keywords Helicopter routing · Mathematical formulation
League championship algorithm

A. Husseinzadeh Kashan (✉) · A. Abbasi-Pooya
Faculty of Industrial and Systems Engineering, Tarbiat Modares University,
Tehran, Iran
e-mail: a.kashan@modares.ac.ir

A. Abbasi-Pooya
e-mail: a.abbasipooya@modares.ac.ir

S. Karimiyan
Young Researchers and Elite Club, Islamic Azad University,
Islamshahr Branch, Islamshahr, Iran
e-mail: S_karimiyan@iiau.ac.ir

A. J. Kulkarni et al. (eds.), *Proceedings of the 2nd International Conference on Data Engineering and Communication Technology*, Advances in Intelligent Systems and Computing 828, https://doi.org/10.1007/978-981-13-1610-4_3

1 Introduction

Crew transportation between offshore rigs and onshore airport is one of the major issues to address in operations management. To maintain oil and gas (O & G) production process, crew must be transported daily to the offshore rigs with helicopter. Outsourcing the helicopter service is very costly, and the cost is calculated based on flight hours. This has made the helicopter routing problem extremely important to researchers and practitioners. Therefore, proposing a model and solution approaches that can give satisfactory solutions is very interesting.

There are some researches on the Helicopter Routing Problem (HRP) for offshore O & G rigs. Galvão and Guimarães [1] designed an interactive routing procedure for HRP in a Brazilian oil company. Fiala Timlin and Pulleyblank [2] proposed heuristics for the incapacitated and capacitated problem in a Nigerian company. Sierksma and Tijssen [3] formulated the problem in a Dutch company as a Split Delivery Vehicle Routing Problem (SDVRP) and proposed column-generation and improvement heuristics. The HRP with one helicopter was modeled by Hernádvölgyi [4] as a Sequential Ordering Problem (SOP). Moreno et al. [5] proposed a column-generation-based algorithm for the mixed integer formulation of HRP, and a post-optimization procedure to remove extra passengers in generated schedules. Romero et al. [6] studied the case of a Mexican company through modeling the problem as the Pickup and Delivery Problem (PDP) and solving by Genetic Algorithm. Velasco et al. [7] presented a Memetic Algorithm (MA) with construction and improvement heuristic for the PDP without time windows. Menezes et al. [8] tested their column-generation-based algorithm on 365-day data from a Brazilian company and compared the result with manual flight plans, demonstrating the cost and time savings of the flight plans generated by the algorithm. De Alvarenga Rosa et al. [9] modeled the problem based on Dial-A-Ride Problem (DARP) and designed a clustering metaheuristic. Abbasi-Pooya and Husseinzadeh Kashan [10] proposed mathematical formulation and a Grouping Evolution Strategy (GES) algorithm for the HRP in South Pars gas field. In some other studies, safety was the objective of the problem. Qian et al. [11] formulated the problem where the minimizing the expected number of fatalities is the objective. A tabu search and comparison of different routing methods were given in Qian et al. [12]. Qian et al. [13] considered two non-split and split scenarios and drew the relation between HRP and parallel machines scheduling problem and bin packing problem. Gribkovskaia et al. [14] recommended a quadratic programming model and dynamic programming approaches and analyzed computational complexity of the problem.

This paper offers two contributions. First, it proposes a mathematical formulation for the problem of helicopter routing that takes account of operational rules while minimizing the finish time of the final tour. Second, a metaheuristic solution approach is presented for large-size problems that may not be solved efficiently with the mathematical model. Specifically, a solution approach is proposed that based on the League Championship Algorithm (LCA), which is a population-based algorithm proposed for constrained and unconstrained problems [15–18]. The problem of a case study is solved with both the mathematical model and the proposed algorithm to compare the results.

The remainder of this paper is organized as follows: The problem is described and formulated in Sect. 2. A League Championship Algorithm is proposed for the problem in Sect. 3. Section 4 presents computational experiments and results of solving the problem with the proposed formulation and LCA. The paper is concluded in Sect. 5.

2 Problem Definition and Formulation

The problem that is confronted in offshore platforms is the transportation of crew to and from platforms. The flight schedule of the helicopter must be planned in a way that each crew member is picked up from their respective origin delivered to their respective destination. The main features in a flight schedule are the routes and the passengers in each route. Additionally, the following considerations must be taken into account:

1. One helicopter is available for the daily transportation.
2. The helicopter has maximum weight capacity, passenger capacity, and maximum flight time limitations.
3. The start and finish points of each tour is the onshore airport.
4. The time to land, to pick up, to deliver and to take off of the helicopter should be considered.
5. The helicopter is not working between flights due to refueling, inspection, pilot rest, etc.
6. The crew members arrive at the airport at different times in the day.

Some special characteristics of the problem are as follows: (1) Having one helicopter requires multiple use of the helicopter, which is similar to the VRP with multiple use of vehicles or multi-trip VRP, which was first addressed by Fleischmann [19] for the problems where the fleet or length of the route is not large [20]. (2) Delivery passengers are available at different times at the start of the day. (3) Each passenger cannot be delivered to every delivery node, i.e., the destination of each linehaul passenger is specific.

A small instance of the problem is shown in Fig. 1. As shown in Fig. 1, there are some passengers (d1–d4) at the airport waiting to be delivered to their respective rigs (R1–R4). There are also some passengers that are waiting to be picked up (p1, p2, and p3) from their respective rigs (R1, R2, and R3, respectively) to be transported to the airport (AP). Figure 2 shows the problem as a graph with rigs representing nodes.

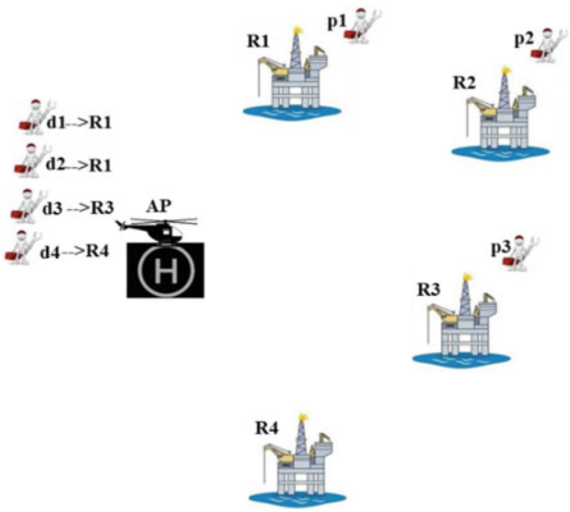

Fig. 1 An instance of the helicopter routing problem

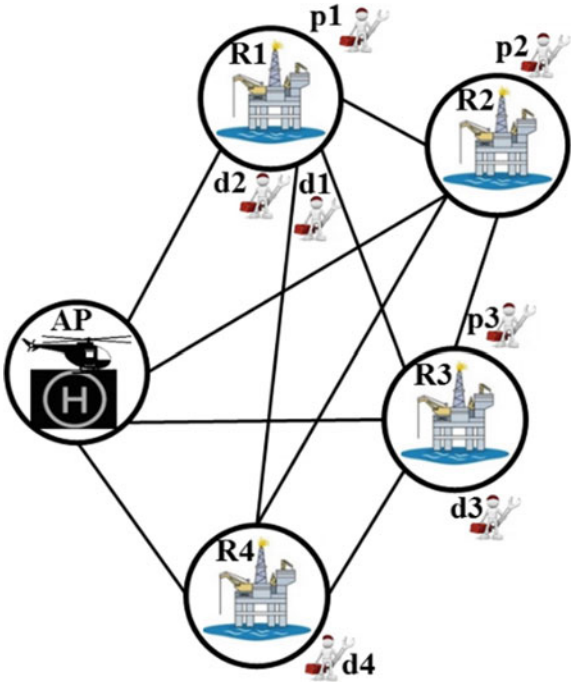

Fig. 2 Graph representing the problem

2.1 Notations

Items used in the model are as shown in Table 1.

Table 1 Definition of the items used in RBF

Item	Definition
Indices	
t	Tour counter
i, j	Rig counter ($i = 0$ or $j = 0$ represents the airport)
m	Backhaul passenger
m'	Linehaul passenger
Sets	
$T = \{1, \ldots, \text{NT}\}$	The set of the tours in a day
$R = \{1, \ldots, \text{NR}\}$ U $\{0\}$	The set of rigs, 0 represents the airport
$B = \{p_1, \ldots, p_m\}$	The set of backhaul passengers
$L = \{d_{1,} \ldots, d_{m'}\}$	The set of linehaul passengers
$C =$ B U L U $\{0\}$	The set of all passengers, 0 represents the airport
Parameters	
Q	The maximum number of passengers that the helicopter can take
FL	The maximum flight time of the helicopter in each tour
WL	The maximum weight that the helicopter can take
ST	The time required for landing, passenger dropoff, and takeoff
RT	The time between tours that is required for refueling, pilot rest, etc.
NT	Number of tours required in a day
NR	Number of rigs
$M1, M2, M3$	Adequately large numbers
t_{ij}	The flight time between the nodes i and j ($i, j \in$ R)
$\text{LB}_{m'}$	The arrival time of the passenger m' to the airport ($m' \in$ L)
W_m	The weight of backhaul passenger m ($m \in$ B)
$\text{Wp}_{m'}$	The weight of linehaul passenger m' ($m' \in$ L)
pd_{im}	=1 if m should be picked up from rig i, and = 0 otherwise ($i \in$ R and $m \in$ B)
$\text{dd}_{im'}$	=1 if m′ should be delivered to rig i, and = 0 otherwise ($i \in$ R and m′ $\in$ L)

Using the above notations, the mathematical model is built and presented in the following sections.

2.2 The Rig-Based Formulation (RBF)

Considering all of the aforementioned limitations, in order to minimize the finish time of the final tour, i.e., the makespan, a mixed integer linear formulation is presented. The variables of the model are as follows:

X_{ijt} =1 if helicopter travels from rig i to j in tour t, and = 0 otherwise
Y_{it} =1 if helicopter visits rig i in tour t, and =0 otherwise
P_{mt} =1 if passenger m is picked up in tour t, and =0 otherwise
$D_{m't}$ =1 if passenger m' is delivered in tour t, and =0 otherwise
S_t Start time of the tour t
F_t Finish time of the tour t
U_{it} Number of passengers in the helicopter after taking off from rig i in tour t,
HW_{it} Helicopter's passenger weight after taking off from rig i in tour t
A_{it} Variable for elimination of subtour
FF Tour makespan.

The proposed formulation of the problem as a mixed integer linear programming is as follows:

$$\text{RBF:Min}\, z = FF \tag{1}$$

s.t.

$$FF \geq F(t) \quad \forall t = NT \tag{2}$$

$$\sum_{j \in R-\{0\}} X_{0jt} = 1 \quad \forall t \in T \tag{3}$$

$$\sum_{i \in R-\{0\}} X_{i0t} = 1 \quad \forall t \in T \tag{4}$$

$$\sum_{j \in R} X_{ijt} = Y_{it} \quad \forall i \in R, t \in T \tag{5}$$

$$\sum_{j \in R} X_{ijt} = \sum_{j \in R} X_{jit} \quad \begin{array}{l} \forall i \in R - \{0\}, \\ t \in T \end{array} \tag{6}$$

$$X_{ijt} + X_{jit} \leq 1 \quad \begin{array}{l} \forall i, j \in R - \{0\}, \\ i \neq j, t \in T \end{array} \tag{7}$$

$$\sum_{t \in T} X_{jjt} = 0 \quad \forall j \in R \tag{8}$$

$$A_{it} + 1 \leq A_{jt} + (1 - X_{ijt})M1 \quad \begin{array}{l}\forall i \in R, j \in R - \{0\}, \\ i \neq j, t \in T\end{array} \tag{9}$$

$$P_{mt} \leq \sum_{i \in R} pd_{im} Y_{it} \quad \forall m \in B, t \in T \tag{10}$$

$$\sum_{t \in T} P_{mt} = 1 \quad \forall m \in B \tag{11}$$

$$D_{m\prime t} \leq \sum_{i \in R} dd_{im\prime} Y_{it} \quad \forall m\prime \in L, t \in T \tag{12}$$

$$\sum_{t \in T} D_{m\prime t} = 1 \quad \forall m\prime \in L \tag{13}$$

$$U_{0t} = \sum_{m\prime \in L} D_{m\prime t} \quad \forall t \in T \tag{14}$$

$$U_{it} + \left(\sum_{m \in B} pd_{jm} P_{mt} - \sum_{m\prime \in L} dd_{jm\prime} D_{m\prime t} \right) \leq U_{jt} + (1 - X_{ijt})M2$$
$$\forall i \in R, j \in R - \{0\}, i \neq j, t \in T \tag{15}$$

$$U_{it} \leq Q \quad \forall i \in R, t \in T \tag{16}$$

$$HW_{0t} = \sum_{m' \in L} D_{m\prime t} W p_{m\prime} \quad \forall t \in T \tag{17}$$

$$HW_{it} + \left(\sum_{m \in B} pd_{jm} P_{mt} W_m - \sum_{m\prime \in L} dd_{jm\prime} D_{m\prime t} W p_{m'} \right) \leq HW_{jt} + (1 - X_{ijt})M3$$
$$\forall i \in R, j \in R - \{0\}, i \neq j, t \in T \tag{18}$$

$$HW_{it} \leq WL \quad \forall i \in R, t \in T \tag{19}$$

$$S_t \geq LB_{m\prime} D_{m\prime t} \; \forall t \in T, m\prime \in L \tag{20}$$

$$S_t + \sum_{i \in R} \sum_{j \in R} (ST + t_{ij}) X_{ijt} = F_t \quad \forall t \in T \tag{21}$$

$$S_t \geq F_{t-1} + RT \quad \forall t \in T, t \neq 1 \tag{22}$$

$$F_t - S_t \leq FL \quad \forall t \in T \tag{23}$$

$$FF, S_t, F_t, U_{it}, HW_{it}, A_{it} \geq 0 X_{ijt}, Y_{it}, P_{mt}, D_{m't} \in \{0, 1\} \tag{24}$$

The objective function (1) is the minimization of the finish time of the last tour. Constraint (2) calculates the value of the objective function based on the last tour's finish time. Constraints (3) and (4) ensure that the helicopter, respectively, begins and ends each tour at the airport. Constraint (5) checks whether each rig is visited in a tour or not. Constraint (6) are flow balance constraints. Backward flights in each tour and loops in rigs are avoided by constraints (7) and (8), respectively. Subtours are eliminated by constraints (9). Constraints (10) state if a rig is visited by a helicopter, the passenger of that rig can be picked up, while Constraints (11) guarantee that all passengers picked up. Similarly, these are ensured for deliveries with constraints (12) and (13). Constraints (14) and (15) calculate the occupied capacity of the helicopter after leaving the airport and rigs, respectively, in each tour. Constraint (16) ensures that capacity requirement is met. Constraints (17)–(19) impose similar conditions to (14)–(16) for helicopter's weight. Constraint (20) calculates the start time of the flight of helicopter in each tour based on the availability time of passengers, while constraint (21) calculates the finish time of each tour. Constraint (22) considers the time between consecutive tours for refueling, etc. Constraint (23) ensures that the maximum flight time of each tour is met. Finally, all variables are declared in (24).

3 League Championship Algorithm for HRP

League Championship Algorithm (LCA) is a population-based metaheuristic proposed for constrained and unconstrained optimization. LCA was first proposed by Husseinzadeh Kashan [15] using the metaphor of sports championship.

The flowchart of LCA is depicted in Fig. 3. The algorithm is initialized by generating a constant-size random population (league) of individuals (teams) and determining the fitness function values (playing strengths). The algorithm then generates a league schedule using single round-robin schedule (see Fig. 4) to plan matches. The winner and the loser of each match are identified stochastically considering the fact that the probability of one team defeating the other is inversely proportional to the difference between that team's strength and the ideal strength. Following winner/loser determination, in each iteration (week), the algorithm moves to the next set of potential solutions (team formations) by using a SWOT matrix (see Fig. 5) that is derived from the artificial match analysis. This analysis is analogous to what a coach typically carries out after a match to determine the team's formation for the next match. Based on the strategy adopted using the SWOT matrix, the equation to move to a new solution (formation) is determined. The procedure is performed iteratively until a termination condition, such as the number of seasons, has reached.

3.1 Idealized Rules of LCA

In the implementation of LCA, there are some idealized rules implied. These rules, which actually idealize some features of normal championships, are:

Rule 1. A team that plays better is more likely to win the game.
Rule 2. The result of each game is unpredictable.
Rule 3. The probability of team i winning team j is the same from both teams' viewpoints.
Rule 4. The result of a match is win or loss (no draw).
Rule 5. If team i beats team j, any strength of team i is a weakness in team j.

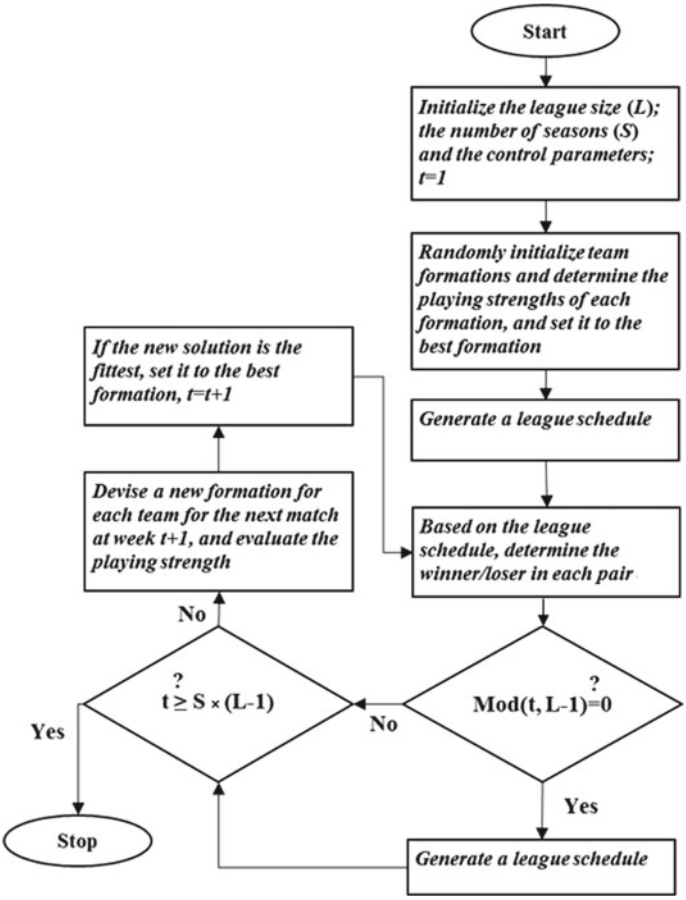

Fig. 3 Flowchart of LCA

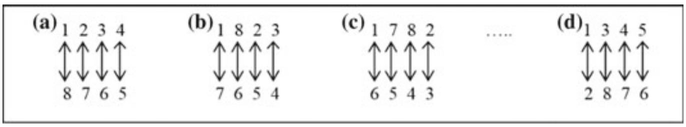

Fig. 4 An example of single round-robin algorithm

Rule 6. Teams only concentrate on their next match without taking account of any of future matches.

	Adopt S/T strategy *i won, l won.* *Focus on ...*	*Adopt S/O strategy* *i won, l lost.* *Focus on ...*	*Adopt W/T strategy* *i lost, l won.* *Focus on ...*	*Adopt W/O strategy* *i lost, l lost.* *Focus on ...*
S	own strengths (or weaknesses of *j*)	own strengths (or weaknesses of *j*)	–	–
W	–	–	own weaknesses (or strengths of *j*)	own weaknesses (or strengths of *j*)
O	–	weaknesses of *l* (or strengths of *k*)	–	weaknesses of *l* (or strengths of *k*)
T	strengths of *l* (or weaknesses of *k*)	–	strengths of *l* (or weaknesses of *k*)	–

Fig. 5 SWOT matrix for building new formation

3.2 Generating a League Schedule

A schedule must be generated in each season allowing teams (solutions) to compete with each other. To this aim, s single round-robin schedule may be used. An example of this league scheduling algorithm for a sports league of eight teams is depicted in Fig. 4. For the first week (Fig. 4a), 1 plays with 8, 2 with 7, etc. For the second week (Fig. 4b), one team (team 1) is fixed and others are rotated clockwise. This continues until generating a complete schedule.

3.3 Determining Winner/Loser

Given team i having formation X_i^t playing team j with formation X_j^t in week t, let the probability of team i beating team j be p_i^t, and p_j^t be the probability of team j beating team i. If $\hat{f}$ denotes the optimal value, according to the Rule 3, we may write

$$p_i^t + p_j^t = 1 \tag{25}$$

According to the Rule 1, we may also write

$$\frac{f(X_i^t) - \widehat{f}}{f(X_j^t) - \widehat{f}} = \frac{p_j^t}{p_i^t} \tag{26}$$

From (25) and (26), p_i^t can be obtained. In order to decide about the result of the match, a random number is generated; if it is less than or equal to p_i^t, team i wins and j loses; otherwise j wins and i loses.

3.4 Building a New Team Formation

In order to move to a new set of solutions (population), we have to change the configuration of the solutions (team formations). First, let us define the following indices:

- l the team that will play with team i ($i = 1,\ldots, L$) at week $t + 1$ according to the league schedule
- j the team that has played with team i ($i = 1,\ldots, L$) at week t according to the league schedule
- k the team that has played with team l at week t according to the league schedule

Using the SWOT matrix in Fig. 5, for determining team i's formation to play with l, if i won the previous game and l won, too, then the S/T strategy for team i is to focus on its own strength (or j's weaknesses) and strengths of l (or k's weaknesses). Other cases can be defined in a similar fashion. Based on the strategy, an equation is used to build the new team's formation for the next week. The interested reader is referred to [15] for detailed information.

3.5 Solution Representation

The solution is represented by a permutation of (NP + NT − 1) numbers in which the numbers less than or equal to NP show the passenger number and the numbers greater than NP work as delimiters for tours. A sample solution representation for NP = 7 and NT = 2 is depicted in Fig. 6, where passengers 3, 5, 7, and 2 are picked up or delivered in tour 1 and passengers 4, 1, and 6 are in tour 2 (8 is the delimiter).

3.6 Objective Function

In order to handle the constraints, penalty function is utilized in the objective function as defined in (27).

$$F(x) = \begin{cases} f(x) + h(l)H(x) & \text{in case of infeasible x} \\ f(x) & \text{otherwise} \end{cases} \tag{27}$$

3	5	7	2	8	4	1	6

Fig. 6 Solution representation for the problem of helicopter routing

where $f(x)$ is the value of the makespan relevant to solution x, $h(l)$ is the penalty value that is dynamically adjusted in the search, l is the iteration counter, and $H(x)$ is a penalty function that is defined in (28).

$$H(x) = \sum_{v} \theta(q_v(x))\, q_v(x)^{\gamma(q_v(x))} \tag{28}$$

where $q_v(x)$ is the amount of constraint v's violation, $\theta(q_v(x))$ and $\gamma(q_v(x))$ are functions of the constraint violation. Three cases of constraint violation (or a mixture of them) can happen: (1) violation of maximum flight time, (2) violation of helicopter's capacity limitation, and (3) violation of helicopter's passenger weight limitation, where the amount of violation is equal to $\sum_t \max(0, F_t - S_t - FL)$, $\sum_i \sum_t \max(0, U_{it} - Q)$, and $\sum_i \sum_t \max(0, HW_{it} - WL)$, respectively, (notations are as defined previously).

3.7 Heuristics

Three intra-route heuristics are also incorporated in the body of the algorithm to improve the solution. They are applied to a percentage of generated solutions. The description of these heuristics is as follows.

The swap: Two passengers of a tour are randomly selected and swapped, as demonstrated in Fig. 7.

The inversion: The visiting order of all passengers between two randomly selected passengers is reversed. It is depicted in Fig. 8.

The insertion: This operator moves one portion of a tour to somewhere else in the tour, as shown in Fig. 9.

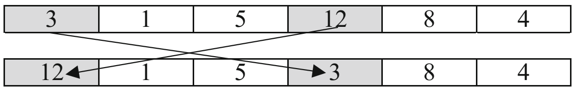

Fig. 7 Swap

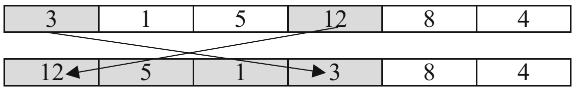

Fig. 8 Inversion

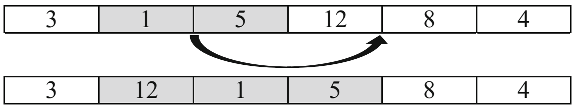

Fig. 9 Insertion

4 Computational Experiments and Results

To test the performance of the proposed formulation and algorithm, the problems of case study were solved with RBF and LCA and the results were compared as summarized in Table 2. The first section of Table 2 reports the case parameters, namely number of rigs (NR), number of pickups and deliveries, and the number of tours. The second and third sections report the results for solving RBF and LCA, respectively. A comparison of objective value and time between RBF solutions with the ones obtained by LCA is presented in the fourth section. Two metrics are computed to evaluate the mean performance of LCA: $\mathrm{Gap_{obj}}$ and $\mathrm{Gap_{time}}$. The relative difference between the average objective value obtained by LCA and the optimum solution is computed based on Eq. (29) and reported under the column $\mathrm{Gap_{obj}}$ as a percentage. In Eq. (29), $\mathrm{Obj_{LCA}}$ is the objective value obtained by LCA and $\mathrm{Obj_{opt}}$ is the objective value obtained from RBF solution. Lower values of $\mathrm{Gap_{obj}}$ show better performance of LCA.

$$\mathrm{Gap_{obj}} = (\mathrm{Obj_{LCA}} - \mathrm{Obj_{Opt}})/\mathrm{Obj_{Opt}} \tag{29}$$

The second metric, $\mathrm{Gap_{time}}$, shows the relative difference between the CPU time of RBF and LCA and is computed by Eq. (30). T_{Opt} and T_{LCA} are, respectively, the run times of RBF and LCA.

$$\mathrm{Gap_{time}} = (T_{\mathrm{Opt}} - T_{\mathrm{LCA}})/T_{\mathrm{LCA}} \tag{30}$$

As Table 2 demonstrates, the case study instances were solved to optimality when the number of rigs is between 4 and 11, and the number of passengers is between 43 and 46. The CPU time for solving the problems with RBF is approximately between 7 and 15 min, while it is about between 4 and 6 min for LCA, which proves the efficiency of both approaches. The column $\mathrm{Gap_{obj}}$ shows that LCA finds solutions with objective function gaps between 3.539 and 5.757%, which is an acceptable value. The $\mathrm{Gap_{time}}$ has positive values, which shows the efficiency of the algorithm compared to the RBF. This demonstrates the efficiency of LCA in providing suboptimal solutions. Therefore, it can be used for problems with large size, where optimal solution cannot be found within an acceptable amount of time.

5 Conclusions and Future Research

This study presented a mathematical model of the problem of routing helicopter to transport crew to and from offshore platforms. Constraints such as passenger and weight capacity of the helicopter, maximum flight time, and delivery passenger's arrival time to the airport were taken into account with the objective of minimizing

Table 2 LCA compared with RBF

Problem parameters				RBF		LCA		Comparison	
NR	No. of pickups	No. of deliveries	NT	Objective value	CPU time (s)	Objective value	CPU time (s)	$Gap_{obj}(\%)$	$Gap_{time}(\%)$
4	18	27	4	509.187442	0:07:02.781	535.58424	0:03:53.028	5.184	81.12
5	16	30	4	529.515663	0:08:57.437	549.29674	0:03:50.492	3.736	133.48
6	17	26	3	491.601450	0:08:45.907	509.94723	0:03:21.740	3.732	161.19
7	20	25	3	450.870342	0:09:37.052	476.82592	0:04:12.947	5.757	128.97
11	19	27	4	537.152514	0:14:58.505	556.16476	0:05:52.815	3.539	155.11

makespan. The proposed mathematical model is tested using case study instances where it shows efficiency and effectiveness in solving the problems.

Furthermore, an algorithm based on the League Championship Algorithm (LCA) was proposed to solve the problem. The LCA proves its efficiency in producing near-optimal solutions with an acceptable amount of computational effort. Therefore, LCA is fit for solving large problem instances where finding optimal solution is not affordable.

In this research, the objective was to minimize the makespan. Other objectives such as passenger's waiting time may also be considered which will make a multi-objective optimization problem. Furthermore, VRP heuristics may be incorporated with LCA to improve its performance in terms of solution quality. The performance of other metaheuristic algorithms, such as Optics Inspired Optimization (OIO) algorithm [21, 22] is also worth investigating in solving the problem of helicopter routing.

References

1. Galvão RD, Guimarães J (1990) The control of helicopter operations in the Brazilian oil industry: issues n the design and implementation of a computerized system. Eur J Oper Res 49:266–270
2. Fiala Timlin MT, Pulleyblank WR (1992) Precedence constrained routing and helicopter scheduling: heuristic design. Interfaces 22(3):100–111
3. Sierksma G, Tijssen GA (1998) Routing helicopters for crew exchanges on off-shore locations. Ann Oper Res 76:261–286
4. Hernádvölgyi IT (2004) Automatically generated lower bounds for search. Doctoral dissertation, University of Ottawa
5. Moreno L, de Aragao MP, Uchoa E (2006) Column generation based heuristic for a helicopter routing problem. In: Experimental algorithms. Springer, Berlin, pp 219–230
6. Romero M, Sheremetov L, Soriano A (2007) A genetic algorithm for the pickup and delivery problem: an application to the helicopter offshore transportation. In: Theoretical advances and applications of fuzzy logic and soft computing. Springer, Berlin, pp 435–444
7. Velasco N, Castagliola P, Dejax P, Guéret C, Prins C (2009) A memetic algorithm for a pick-up and delivery problem by helicopter. In: Bio-inspired algorithms for the vehicle routing problem. Springer, Berlin, pp 173–190
8. Menezes F, Porto O, Reis, ML, Moreno L, de Aragão MP, Uchoa E, Abeledo H, Nascimento NCD (2010) Optimizing helicopter transport of oil rig crews at Petrobras. Interfaces 408–416
9. de Alvarenga R, Machado AM, Ribeiro GM, Mauri GR (2016) A mathematical model and a clustering search metaheuristic for planning the helicopter transportation of employees to the production platforms of oil and gas. Comput Ind Eng 101:303–312
10. Abbasi-Pooya A, Husseinzadeh Kashan A (2017) New mathematical models and a hybrid grouping evolution strategy algorithm for optimal helicopter routing and crew pickup and delivery. Comput Ind Eng 112:35–56
11. Qian F, Gribkovskaia I, Halskau Ø Sr (2011) Helicopter routing in the Norwegian oil industry: including safety concerns for passenger transport. Int J Phys Distrib Logistics Manage 41(4):401–415
12. Qian F, Gribkovskaia I, Laporte G, Halskau Sr Ø (2012) Passenger and pilot risk minimization in offshore helicopter transportation. Omega 40(5):584–593
13. Qian F, Strusevich V, Gribkovskaia I, Halskau Ø (2015) Minimization of passenger takeoff and landing risk in offshore helicopter transportation: models, approaches and analysis. Omega 51:93–106

14. Gribkovskaia I, Halskau O, Kovalyov MY (2015) Minimizing takeoff and landing risk in helicopter pickup and delivery operations. Omega 55:73–80
15. Husseinzadeh Kashan A (2009) League championship algorithm: a new algorithm for numerical function optimization. In: International conference of soft computing and pattern recognition
16. Husseinzadeh Kashan A (2014) League championship algorithm (LCA): an algorithm for global optimization inspired by sport championships. Appl Soft Comput 16:171–200
17. Husseinzadeh Kashan A, Karimi B (2010) A new algorithm for constrained optimization inspired by the sport league championships. In: Evolutionary computation (CEC), 2010 IEEE congress, pp 1–8
18. Husseinzadeh Kashan A (2011) An efficient algorithm for constrained global optimization and application to mechanical engineering design: league championship algorithm (LCA). Comput.-Aided Des 43:1769–1792
19. Fleischmann B (1990) The vehicle routing problem with multiple use of vehicles. In: Working paper, Fachbereich Wirtschaftswissenschaften, Universitat Hamburg
20. Olivera A, Viera O (2007) Adaptive memory programming for the vehicle routing problem with multiple trips. Comput Oper Res 34(1):28–47
21. Husseinzadeh Kashan A (2015) A new metaheuristic for optimization: optics inspired optimization (OIO). Comput Oper Res 55:99–125
22. Husseinzadeh Kashan A (2015) An effective algorithm for constrained optimization based on optics inspired optimization (OIO). Comput.-Aided Des 63:52–71