Journal of Naval Science and Engineering
2016, Vol. 12, No.2, pp.67-78
Endüstri Mühendisliği/Industrial Engineering

GENERALIZATION OF A UAV LOCATION AND ROUTING PROBLEM BY TIME WINDOWS

Ertan YAKICI

Industrial Engineering Department, Turkish Naval Academy, Istanbul, Turkey
eyakici@dho.edu.tr

Date of Receive: 12.09.2016 **Date of Acceptance: 04.11.2016**

Abstract

In this study we extend and generalize a locating and routing problem for UAVs, with an objective of maximization of the total score collected from interest points visited. By solving the problem we determine simultaneously take-off and landing stations and visit order of interest points for each UAV. The problem is defined by an integer linear programming (ILP) formulation. An ant colony optimization approach is altered for the introduced problem. Computational experiments are performed to compare CPLEX solver and the heuristic. We observe that the heuristic performed well on the experienced instances.

ÖZ

Bu çalışmada İHA'lar için kullanılan, ziyaret edilen noktalardan toplanan puanları ençoklamayı amaçlayan bir yerleştirme ve rotalama problemi geliştirilerek daha genel bir problem haline getirilmiştir. Bu problemin çözümü ile her bir İHA için kalkış ve iniş istasyonları ile noktaların ziyaret sıraları eşzamanlı olarak belirlenmektedir. Problem tamsayılı doğrusal programlama modeli olarak formüle edilmiştir. Bir karınca kolonisi optimizasyon yaklaşımı problem için modifiye edilmiştir. Sayısal denemelerde CPLEX çözücüsü ile sezgisel yaklaşım karşılaştırılmış, sezgisel yaklaşımın tecrübe edilen problem örnekleri üzerinde iyi performans gösterdiği tespit edilmiştir.

Keywords: *Location and Routing Problem; Ant Colony Optimization;UAV.*
Anahtar Kelimeler: *Yerleştirme ve Rotalama Problemi; Karınca Kolonisi Optimizasyonu; İHA.*

1. INTRODUCTION

In recent years we have witnessed that UAVs can increase the capability of military power by achieving difficult tasks that are unsafe for pilots. More specifically, small UAVs are employed by navies and used as surveillance drones by launching from small platforms.

For a navy, having mobile platforms and changing interest points, most survelliance tasks require a predetermined plan for stationing and routing the UAVs. A problem for optimal planning such an operation is defined by Yakıcı [1], and named as prize collecting location and routing problem (PCLRP). In PCLRP, it is assumed that identical UAVs are allocated to bases. Each UAV takes off from its base follow a route and land on its base where each UAV is limited by a maximum flight time. Optimal solution to this problem maximizes the collected scores (considered as importance factors) from visited interest points. Since the Integer Linear Programming (ILP) solvers provide poor solutions or no solution in reasonable period of times, an Ant Colony Optimization method is suggested by the author.

In this study, this basic problem is generalized to allow UAVs to take off and land at different bases, and to include time windows for assigned tasks to interest points. To the best of our knowledge, this problem is not introduced before. We give a formulation of this new PCLRP generalized with time windows. We also propose some modifications to the solution method suggested by Yakıcı [1] to employ it in solving new PCLRP which we call PCLRPTW from now on.

Since PCLRPTW and solution method proposed in this study are similar to PCLRP and its solution method, we do not give a detailed literature review here. For this purpose we refer to the literature review given by Yakıcı [1]. However, here we should at least specify the most relevant paper which is introduced by Ahn et al. [2]. It is defined in the context of planet exploration missions. The details of the solution method are presented by Ahn, DeWeck, Geng, and Klabjan in another paper [3]. Their problem is a rich version of PCLRP. However, it does not consider time window for each site visit as we do in PCLRPTW.

The readers are referred to recent survey papers by Drexl and Schneider [4] and by Prodhon and Prins [5] for a general review of the LRP literature.

In the following sections, we introduce the problem, explain the suggested metaheuristic method and present the result of our computational experience. Finally, in the last sections we provide concluding remarks.

2. PROBLEM DEFINITION

In our problem, we assume a fleet composed of identical UAVs. Therefore, we only specify one maximum flight time, one required time for each interest point visit and one cruising speed. Although we assume sufficient number of platforms, a limit may be introduced on the maximum number of active stations where platforms are stationed. The interest points and their time windows are assumed to remain fixed.

A solution to the problem is a number of routes, which is equal or less than the total number of UAVs, each takes off and lands in the allowed time and without violating time windows defined for interest points.

Below, we present indices, sets, parameters, variables and ILP formulation for the problems PCLRP and PCLRPTW. PCLRP and PCLRPTW are defined by the equations and inequalities (1-13) and (1-4, 7-22), respectively.

$u \in U$	set of UAVs.
$i, j \in I$	set of interest points.
$i, j \in S$	set of stations.
d_{ij}	expected elapsed time in flight between *i* and *j*.
p_i	importance of interest point *i*.
t_i	expected elapsed time on interest point *i*.
y_{max}	maximum number of active stations allowed.
t_{max}	maximum time between takeoff and landing for UAV.
b_i	beginning time for time window of interest point *i*.
e_i	ending time for time window of interest point *i*.
X_{iju}	binary variable indicating if UAV *u* has a leg from point *i* to point *j*, or not.

Y_i	binary variable indicating if station *i* is activated, or not.
F_{iju}	a continuous variable.
A_{ui}	arrival time of UAV *u* to interest point *i*.

$$\max z = \sum_{j \in I \cup S} \sum_{i \in I} \sum_{u \in U} p_i X_{jiu} \qquad (1)$$

subject to

$$\sum_{i \in S} Y_i \le y_{\max} \qquad (2)$$

$$\sum_{j \in I} \sum_{u \in U} X_{iju} \le Y_i \, |I| \qquad \forall i \in S \qquad (3)$$

$$\sum_{i \in S} \sum_{j \in I} X_{iju} \le 1 \qquad \forall u \in U \qquad (4)$$

$$\sum_{j \in I} X_{iju} = \sum_{j \in I} X_{jiu} \qquad \forall i \in S, u \in U \qquad (5)$$

$$\sum_{i \in I \cup S} \sum_{j \in I \cup S} X_{iju}(d_{ij} + t_j) \le t_{\max} \qquad \forall u \in U \qquad (6)$$

$$\sum_{i \in I \cup S} X_{iju} = \sum_{i \in I \cup S} X_{jiu} \qquad \forall j \in I, u \in U \qquad (7)$$

$$\sum_{i \in I \cup S} \sum_{u \in U} X_{jiu} \le 1 \qquad \forall j \in I \qquad (8)$$

$$\sum_{i \in I \cup S} \sum_{u \in U} F_{jiu} - \sum_{i \in I \cup S} \sum_{u \in U} F_{iju} \leq \varepsilon * \sum_{i \in I \cup S} \sum_{u \in U} X_{iju}$$

$\forall j \in I$ (9)

$F_{iju} \leq X_{iju}$ $\forall i \in I \cup S, j \in I \cup S, u \in U$ (10)

X_{iju} 0 1

$\forall i \in I \cup S, j \in I \cup S, u \in U$

(11)

$Y_i \in \{0,1\}$ $\forall i \in S$ (12)

$F_{iju} \geq 0$ $\forall i \in I \cup S, j \in I \cup S, u \in U$ (13)

$$\sum_{j \in I} \sum_{u \in U} X_{jiu} \leq Y_i \, |I|$$

$\forall i \in S$ (14)

$$\sum_{j \in I} \sum_{u \in U} X_{jiu} \leq \sum_{j \in I} \sum_{u \in U} X_{iju} \, |U|$$ $\forall i \in S$ (15)

$$\sum_{i \in S} \sum_{j \in I} X_{jiu} \leq 1$$

$\forall u \in U$ (16)

$$\sum_{i \in S} \sum_{j \in I} X_{iju} = \sum_{i \in S} \sum_{j \in I} X_{jiu}$$ $\forall u \in U$ (17)

$$A_{uj} \leq \left(1 - \sum_{i \in I \cup S} X_{iju}\right) M + E_j$$ $\forall j \in I, u \in U$ (18)

$$A_{uj} \geq \left(\sum_{i \in I \cup S} X_{iju} - 1 \right) M + B_j \qquad \forall j \in I, u \in U \qquad (19)$$

$$A_{uj} \geq A_{ui} + X_{iju} t_i + d_{ij} + (X_{iju} - 1)\, M \qquad \forall j \in I, u \in U \qquad (20)$$

$$A_{uj} + t_i + X_{jiu} d_{ij} + (X_{jiu} - 1)\, M \leq t_{\max} \qquad \forall i \in S, j \in I, u \in U \qquad (21)$$

$$A_{uj} \qquad \forall j \in I, u \in U \qquad (22)$$

The function (1) represents total importance values collected from interest points. Constraint (2) limits the number of stations that can be activated. Constraints (3, 4) force each UAV to start its route from only one station to which it is assigned, while Constraint (5) force each UAV to return back to its departure point. Constraint (6) limits flight time. Constraint (7) serves as flow conservation. Constraint (8) limits the departures from interest points to one. Constraints (9, 10) prevent infeasible tours, where ε is a small positive real number. Constraints (11-13) identify the sets for decision variables. The objective function and these constraints (2-13) collectively define PCLRP.

We extend PCLRP by implementing two new features. One of them is allowing each UAV to land on any one of the active stations and the other is adding time windows for the task can be started on interest points. Removal of constraints (5, 6), and employing the constraints (14-22) provides the extended problem PCLRPTW.

Constraint (14) ensures that if station is not active, UAV cannot land on that station. Constraint (15) restricts that landing on a station can occur if a takeoff is realized at that station. Constraint (16) forces each UAV to land on at most one station. Constraint (17) ensures that if a UAV takes off, it must land. Constraints from (18-20) provide the satisfaction of time window restrictions. Constraint (21) ensures that all UAVs should return to a station before the given time. Constraint (22) declares the domain for variable A_{uj}. The constant M, used in Constraints (18-21), represents a positive real number greater than t_{max}.

3. HEURISTIC SOLUTION APPROACH

In this section we refer to the Ant Colony Optimization (ACO) metaheuristic tailored for PCLRP by Yakıcı [1].This heuristic algorithm includes two main procedures, one is related to solution construction and the other is related to pheromone update. Robust design of ACO algorithm allowed us to utilize it for our problem with a minor modification to the probability distribution employed in the routing phase of solution construction in the study of Yakıcı [1]. Since only the construction procedure is affected by the change from PCLRP to PCLRPTW, here we do not mention the procedure related to pheromone update. However, to keep the integrity of this article, we will define the parameters used in the construction phase of the algorithm, without explaining details.

The proposed heuristic technique is similar to MMAS (MAX-MIN Ant System) [6, 7]. In this method, ants represent UAVs and the collection of routes by ants constructs one solution. The algorithm repeats iterations of solution construction and pheromone trail update to converge to a good solution.

The visibility component η_{ij} is a measure of importance of interest point j per unit time elapsed both in transition between the points i and j and in executing the task at j. Two learned knowledge components $\tau^{gl}_{i,n_i,j,k}$ and $\tau^{sl}_{i,n_i,j,k}$, reflects the contribution of solution component experienced in prior solutions. A solution component identified by the indices i, n_i, j and k relays the information that i is the station, n_i is the count of UAVs assigned to station i, j and k are the current and the next location of UAV, respectively. A solution must be formed by feasibly integrated solution components.

The superscripts, *gl* and *st*, used for identifying two different pheromone trails, represents the words “global” and “stationary”. Please refer to Yakici [1] for detailed explanation about pheromone trails.

Separate probability distributions are employed for assigning UAVs to stations and routing them between points. The probability distribution for assignment is given in Equation 23. Any UAV not assigned to a station has a probability to depart any station i' to reach any interest point j'.

$$p^{a}_{i'j'} = \frac{\left(\gamma^{gl}_{i'j'}\right)^{\alpha^{gl}} \left(\gamma^{st}_{i'j'}\right)^{\alpha^{st}} \left(\eta_{i'j'}\right)^{\beta^{a}}}{\sum_{ij} \left(\gamma^{gl}_{ij}\right)^{\alpha^{gl}} \left(\gamma^{gl}_{ij}\right)^{\alpha^{gl}} \left(\eta_{ij}\right)^{\beta^{a}}} \tag{23}$$

γ^{gl}_{ij} and γ^{st}_{ij} are cumulative pheromone trails reflecting total of pheromones on the leg from station i to interest point j with greater number of UAVs at station i (compared to current UAV count). (Please see Yakici [1] for details about γ parameters). The power parameters, to which terms are raised in the formula, affect the relative importance of these terms. The superscript a identifies the “assignment” phase.

Equation 24 defines the routing probability of a UAV stationed at i', from its current point j' to k', given exactly $n_{i'}$ UAVs stationed at station i'.

$$p_{\downarrow}(i^{\uparrow\prime} \; [\![n_{\downarrow}(i^{\uparrow\prime}) \; j]\!] \; ^{\uparrow\prime} k^{\uparrow\prime})^{\uparrow} r = ((\tau_{\downarrow}(i^{\uparrow\prime} \; [\![n_{\downarrow}(i^{\uparrow\prime}) \; j]\!] \; ^{\uparrow\prime} k^{\uparrow\prime})^{\uparrow} gl)^{\uparrow}(\alpha^{\uparrow} gl) \, (\tau_{\downarrow}(i^{\uparrow\prime} \; [\![n_{\downarrow}(i^{\uparrow\prime}) \; j]\!] \; ^{\uparrow\prime} k^{\uparrow\prime})^{\uparrow} st)^{\uparrow}(\alpha^{\uparrow} st \tag{24}$$

Note that this probability is set to zero if problem constraints are violated by correponding routing. The superscript r identifies the “routing” phase.

Differently from PCLRP, here we define the parameter $\theta_{j'k'}$, which is employed to decrease the probability of a UAV to arrive at an interest point too early before its time window. With the utilization of this function, the probability decreased proportional to the waiting time before time window. Calculation of θ_{ij} value is given in the following expression:

$$\theta_{ij} = 1 - \frac{max(0, b_i - \omega - d_{ij})}{b_i} \tag{25}$$

where ω is the current time of UAV.

Pseudocode for solution construction phase is presented in Figure 1.

4. EXPERIMENTS

Keeping all of the experiment settings same as in the experiment of PCLRP, we have experienced the algorithm on the extended problem PCLRPTW for 9 instances reported by Yakıcı [1]. To activate time window constraints, a number of interest points are randomly chosen and visits to those points are restricted with certain time windows. Table 1 provides these numbers and assigned time windows (beginning and ending times).

procedure CONSTRUCTSOLUTION($iter$)
  while any $u \in U$ is not assigned to a station $i \in S$ **do**
    for $\forall i \in S$ **do**
      if $(\sum_{i|i \in A} 1 < y_{max}) \vee (i \in A)$ **then** ▷ $A|A \subseteq S$ is set of assigned stations
        Calculate $\gamma_{i,j}^{gl}$ and $\gamma_{i,j}^{st}$ $\forall i \in S, j \in I$
      end if
    end for
    for $\forall i \in S, j \in I$ **do**
      Calculate $p_{i,j}^{assignment}$
    end for
    Choose the assignment component (i, j) randomly
  end while
  $\sum_{i,n_i,j,k} (\tau_{i,n_i,j,k}^{gl})^{\alpha^{gl}} (\tau_{i,n_i,j,k}^{st})^{\alpha^{st}} (\eta_{j,k})^{\beta^r} \leftarrow 1$ ▷ sum is set to a positive number arbitrarily
  while $\sum_{i,n_i,j,k} (\tau_{i,n_i,j,k}^{gl})^{\alpha^{gl}} (\tau_{i,n_i,j,k}^{st})^{\alpha^{st}} (\eta_{j,k})^{\beta^r} > 0$ **do**
    for $\forall i \in S, j \in \{i\} \cup I, k \in I$ **do**
      Calculate $p_{i,n_i,j,k}^{routing}$
    end for
    Choose the routing component (i, n_i, j, k) randomly
  end while
end procedure

Figure 1. Pseudocode for Solution Construction Phase [1].

Table 1. Time Window Restrictions

Instance number (as given in Yakıcı [1])	Number of points restricted with time windows *x* Assigned time window
1	16 *x* (300-600)
2	10 *x* (200-600), 12 *x* (400-600)
3	10 *x* (100-300), 12 *x* (200-300)
6	24 *x* (300-600)
7	15 *x* (200-600), 24 *x* (400-600)
8	15 *x* (100-300), 24 *x* (200-300)
11	32 *x* (300-600)
12	20 *x* (200-600), 36 *x* (400-600)
13	20 *x* (100-300), 36 *x* (200-300)

Table 2 provides the results. The columns indicate instance number, best and worst heuristic solution value, gap between best CPLEX (version 12.6.2.0) solution obtained in one hour and best heuristic solution ((*C-H)/H* where *C* and *H* are best CPLEX and heuristic solution, respectively), and average solution time for one run. The experiments have been conducted on a PC with 4 GB RAM and 1.9 GHz processor.

Table 2. Experimental Results

Instance number (as given in Yakıcı [1])	Best heuristic solution (*H*)	Worst heuristic solution	Gap between best CPLEX solution and *H*	Average run time of heuristic (in sec.)
1	110	105	-65,7 %	108
2	125	121	-66,4 %	195
3	87	82	-18,4 %	162
6	116	107	-81,9 %	190
7	140	133	-77,9 %	334
8	84	76	-71,4 %	252
11	149	132	-50,3 %	316
12	194	181	-100 %	485
13	108	102	-57,4 %	396

In all of the experienced instances, we observe a significant difference between heuristic solutions and CPLEX solutions. CPLEX performs very poor in this hard combinatorial problem, while it cannot find any positive value in one of the problem instances (instance 12). On the other hand, heuristic method provides significantly better solutions in very short periods.

5. CONCLUSION

In this study, we generalize a variant of LRP, which maximize collected importance points from visited locations, introduced by Yakıcı [1]. A fleet of identical UAVs is assumed. UAV routes are constrained by allowed sortie time and the requirement of same takeoff and landing station. We enhance the problem by removing the limitation of having same takeoff and landing station and by adding a practical characteristic, time windows for interest points.

Experiments show that altered ant colony optimization metaheuristic provides the best solutions in a few minutes.

REFERENCES

[1] Yakıcı E., (2016). *Solving location and routing problem for UAVs,*, Computers & Industrial Engineering, in press, doi: 10.1016/j.cie.2016.10.029.

[2] Ahn J., DeWeck O., and Hoffman J., (2008). *An optimization framework for global planetary surface exploration campaigns*, Journal of the British Interplanetary Society, 61(12), pp. 487-498.

[3] Ahn J., DeWeck O., Geng Y., and Klabjan D., (2012). *Column generation based heuristics for a generalized location routing problem with profits arising in space exploration*, European Journal of Operational Research, 223(1), pp. 47-59.

[4] Drexl M., and Schneider M., (2015). *A survey of variants and extensions of the location-routing problem*, European Journal of Operational Research, 241(2), pp. 283-308.

[5] Prodhon C., and Prins C., (2014). *A survey of recent research on location-routing problems*, European Journal of Operational Research, 238(1), pp. 1-17.

[6] Stützle T., and Hoos H., (1997). *Max-min ant system and local search for the traveling salesman problem*, In Evolutionary Computation, IEEE International Conference on, pp. 309-314, IEEE.

[7] Stützle T., and Hoos H., (2000). *Max-min ant system*, Future generation computer systems, 16(8), pp. 889-914.