Online Dial-A-Ride Problem with Unequal-length Time-windows

Fanglei Yi
Faculty of Management and Administration
Macau University of Science and Technology
Macau, China
flyi@must.edu.mo

Yu Song
Zhongnan University of Economics and Law, China;
Macau University of Science and Technology, Macau
ysong@must.edu.mo

Chunlin Xin
Beijing University of Chemical Technology
Beijing, China
xinchl@sem.tsinghua.edu.cn

Ivan, K.W.Kai Walter
Faculty of Management and Administration
Macau University of Science and Technology
Macau, China
kwlai@must.edu.mo

***Abstract*—In the Online Dial-A-Ride Problem (ODARP) with time-windows, objects are to be transported between points in a metric space. Transportation requests are released over time, specifying the objects to be transported and the corresponding source and destination. One serve travels in the metric space at constant unit speed, picking up the objects at their sources and drops them at their destinations. Each request also specifies a deadline. If a request is not be served by its deadline, it will be called off. The goal is to plan the motion of server in an online way so that the maximum number of requests is met by their deadlines. Usually it is assumed that the time-windows are equal-length. That is, each request has the same length of maximum waiting time. We study the problem in a more general case in which the lengths of maximum time that requests can wait are different. We perform competitive analysis of two online strategies for the problem in two special metric spaces, the uniform metric space and the *K*-constrained metric space, respectively. The competitive ratios of the strategies are obtained.**

Keywords-Online dial-a ride problem; competitive analysis; time-windows; competitive ratio

I. INTRODUCTION

The Dial-A-Ride Problem (DARP) consists of designing vehicle (or server) routes or schedules for users who specify pick-up and drop-off requests between sources and destinations in a metric space. The aim is to plan the motion of the server to handle the requests so that some optimality criterion is met. To meet real life needs, many new side constraints have been added to the problem. One useful extension is the Dial-A-Ride Problem with Time-Windows (DARPTW) in which each request specifies a deadline, the maximum time the request can wait for serving. If a request is not be served by its deadline, it will be called off. The goal is to plan the motion of servers so that the maximum number of requests is met by their deadlines. The problem has been studied extensively in the area of operations research, management science, and combinatorial optimization because of its usefulness to the logistics and transportation industry. A common characteristic of almost all the approaches to the study of the problem is the offline point of view. The input is known completely beforehand. However, in many routing and scheduling applications the instance only becomes known in an online fashion. In other words, the input of the problem is communicated in successive steps. This makes the problem an online optimization problem.

In the Online Dial-A-Ride Problem with Time-windows (ODARPTW) requests are presented over time, requiring the server to carry the objects from sources to destinations while the server is enroute for serving other requests. And each request also specifies a deadline. If a request is to be served, the server must reach the point where the request originated during its time-window (the time between the request's arrival and its deadline). The goal of the problem is to design strategies for the server to serve as many incoming requests as possible by their deadlines in an online way. The online strategy for ODARPTW neither has information about the release time of the last request nor has the total number of requests. It must determine the behavior of the server at a certain moment t of time as a function of all the requests released up to time t (and the current time t). In contrast, an offline strategy has information about all requests in the whole sequence already at time 0.

Usually it is assumed that the time-windows are equal-length in ODARPTW [1,2]. That is, each request has the same length of maximum waiting time. However, this assumption in many practical situations is not realistic since different customers have different requirement of service level, the maximum waiting times they can tolerate are also different.

In this paper we study ODARPTM in a more general case in which the length of maximum waiting time for each request is different. We refer to this model as the *Online Dial-A-Ride Problem with Unequal-length Time-windows (ODARPUTW).*

The online DARP and in general vehicle routing and scheduling problems have been widely studied for more than

This work was partly supported by the Macao Foundation of the Macao Special Administrative Region.

978-1-4244-4639-1/09/$25.00 ©2009 IEEE

three decades (see [3] for a survey on the subject). Most previous researches on online routing problems focused on the objectives of minimizing the makespan [4,5,6], the weighted sum of completion times [3,7], and the maximum/average flow time [8,3]. In the paper [9,10,11], results on the online k-taxi scheduling problem have been presented, in which a request consists of two points (a source and a destination) on a graph or in a metric space. Subsequently, a similar problem, online *k*-truck scheduling problem has been studied in [12]. However, in the problem of online *k*-taxi/truck scheduling it is assumed that *k* servers (taxies or trucks) are all free when a new service request occurs. That is, new requests can not be released while the severs are busy in serving other requests. The goal is to minimize the total distance travelled by servers. In [2] the authors studied firstly the online DARP with single server in which new requests come in while the server is traveling for other requests. They considered two different cases depending on the capacity of the server. In [13] the online DARP under a restricted information model was studied, in which the online server only knows the source information at the release time of the request. The destination of a ride is released when the online server reach the source. The objective function is to minimize the time by which the server has executed all the rides and is back in the origin. All of these previous work assumed that the requests could wait for any length of time until the server completed them. That is, they did not consider the constraints of the time-window. Most previous works on the DARP with time-window constraints are the offline point of view. The input is known completely beforehand. For some related works, please refer to [14,15]. In [2], we presented the first results on the online DARP with time-windows, in which each request has a limited maximum waiting time and the aim is to complete as many requests as possible by they deadlines. In [1] we studied the same problem under a restricted information model, where two special metric spaces, the *uniform metric space* and the *K-constrained metric space,* are considered, as we will do here. We study this problem again in this paper under the *unequal-length time-windows model*.

We evaluate the quality of an online strategy by competitive analysis [16,17], which has become a standard yardstick to measure the performance. In competitive analysis, the performance of an online strategy is compared to the performance of the optimal offline strategy, which knows about all future jobs. A strategy ***A*** for ODARPTW is called α-competitive if for any instance ***R*** the number of request completed by ***A***, denoted by $B_A(R)$, is at least $1/\alpha$ times the number of request completed by an optimal offline strategy *OPT*, denoted by $B_{OPT}(R)$. That is,

$$B_A(R) \geq \frac{1}{\alpha} B_{OPT}(R) \qquad (1)$$

where α is called the competitive ratio of strategy ***A*** for ODAR.

We perform competitive analysis of deterministic strategy for ODARPUTW in two types of metric space, respectively. In the first metric space, uniform metric space, we present an online GREEDY strategy. We prove that GREEDY is 3-competitive. And in the second metric space, *K*-constrained metric space, another online strategy, Maximal Goods First strategy (*MGF*), is shown which has a $\frac{2K\lambda^2+\lambda-1}{\lambda-\rho-1}$ competitive ratio, where $\lambda=\sqrt{(\rho+1)^2+\frac{\rho}{K}}+\rho+1$, K is the diameter of the metric space, and $\rho>0$ is the coefficient of penalty.

The rest of this paper is organized as follows. In section II we give some definitions and notations. In section III we present two online strategies and discuss their competitive performances in two types of metric space, respectively. Section IV concludes this paper.

II. DEFINITIONS AND NOTATIONS

Let $M = (X, d)$ be a metric space with *n* points which is induced by an undirected unweighted graph $G = (V,E)$ with $V = X$, i.e., for each pair of points from the metric space *M* we have $d(x, y)$ that equals the shortest path length in *G* between vertices *x* and *y*. We consider two types of metric space in this paper, the uniform metric space and the *K*-constrained metric space. In the uniform metric space, the distance between any two points is unit length. It can be considered as a special metric space that is induced by a complete graph with unit edge weights. And in the *K*-constrained metric space, $d_{max}/d_{min}= K$, where $d_{max} = \mathbf{max}\, d(x, y)$, $d_{min} = \mathbf{min}\, d(x, y)$, $x \neq y$, $x, y \in V$. Without loss of generality, we assume that d_{min} equals 1 and d_{max} is *K* long in the *K*-constrained metric space. We call *K* the diameter of the metric space which can be considered the maximum time required to travel between the two farthest points in the metric space. Note that in the uniform metric space, $K = 1$. An instance of the basic ODARPUTW in the metric space *M* consists of a sequence $R = (r_1, r_2, \ldots , r_m)$ of requests. Each request is a quintuple $r_i = (t_i, z_i, a_i, b_i, h_i) \in R \times N \times X \times X \times R$ with the following meaning: $t_i \in R$ is the time that request r_i is released and $z_i \in N$ is the quantity of goods that needs to be transported by the server; $a_i \in X$ and $b_i \in X$ are the source and destination, respectively, between which the goods corresponding to request r_i is to be transported; h_i denotes the time-window length of request r_i, the maximum time that request r_i can wait. The capacity of the server is finite, denoted by constant *Z*. That is, the upper bound of the goods loaded by the server is Z units. We assume the goods of the request is partible. If a given request has overmany goods in the sense of the current capacity of the server, it can be divided into several partitions. The server is allowed to load some partitions of them. And the rest of them can be considered as the goods of new requests. In this paper we assume that $z_i \leqslant Z$, $\forall i \in N$. It is also assumed that the sequence $R = (r_1, r_2, \ldots , r_m)$ of requests is given in order of non-decreasing release times, that is, $0 \leqslant t_1 \leqslant t_2 \leqslant \ldots \leqslant t_m$. A request is said to be *accepted request* if the corresponding object is picked up by the server at source, and a request is said to be *completed request* if the corresponding object is transported to the destination. In this paper, we consider the following assumptions for ODARPTWRIM as we did in [1]:1) The speed of the server is constant 1. This means that, the time it takes to travel from one point to another is exactly the distance between the two points; 2) To make sure that the problem is feasible, we assume that the windows sizes $h_i \geq d_{\min}$ in the K-constrained metric space and $h_i \geq 1$ in the uniform metric space, for $i = 1, 2, ..., m$. Note here h_i is not a constant any more, but can be an arbitrary value of time for each request.

III. ONLINE STRATEGIES AND COMPETITIVE ANALYSIS

In this section we first study the problem in a general metric space. We prove that there is no competitive deterministic online strategy for a non-constrained metric space if the time-windows $h_i > 0$ are arbitrary. Then we study the problem in two special types of metric space, the uniform metric space and the K-constrained metric space, respectively. We propose a GREEDY strategy for the problem in uniform metric space and a MGF strategy in the K-constrained metric space. The performance guarantees of the two strategies for the problem are shown in this section.

Theorem 1. ***In a general metric space, no online deterministic strategy can obtain a constant competitive ratio for ODARPUTM.***

Proof: Assume the online and offline servers stay at origin point o at time t=0, and two requests, $r_0 = (0, 1, a_0, b_0, h_0)$ and $r_0' = (0, 1, a_0', b_0', h_0')$,are presented, where r_0 requires the server to pick up 1 unit of goods at source a_0 and deliver the goods to destination b_0, r_0' requires the server to deliver 1 unit of goods from source a_0' to destination b_0' ,and the time-windows of the requests are h_0 and h_0' ,respectively. If the distance from the origin point to the sources a_0 and a_0' are $d(o, a_0) = h_0$ and $d(o, a_0') = h_0'$, respectively, there are two possible choices for an online strategy:

1) The online server remains at the origin point, it does not move for any one of the two requests.

2) The online server leaves the origin point immediately to one of the sources.

If the online strategy chooses 1), then the request sequence stops. No new request will be presented. So the online server can not serve any one of the two requests while the offline server can move immediately to one of the source points and serve one of the requests. So $\frac{B_{OPT}(R)}{B_A(R)} = \frac{1}{0} = \infty$

If the online strategy chooses 2), we assume without loss of generality that at time h_0' the online server reaches position a_0' for serving request r_0', then offline server moves to a_0 immediately for request r_0 at time $t = 0$ and delivers the goods to the destination b_0 at time $t_0 = h_0 + d(a_0, b_0)$. Then, from time $t_0 + t_i$ $(i = 1, 2, ..., M)$ onwards a request $r_i = (t_i, 1, b_{i-1}, b_i)$ is presented, where $t_i = t_{i-1} + d(b_{i-1}, b_i)$ and M is an arbitrary large number. So, as long as $d(a_0', a_0) + h_0' > h_0$ and $d(a_0', b_i) + h_0' > t_i$ $(i = 1, 2, ..., M)$, the online server can not complete any one of the requests in sequence r_i $(i = 1, 2, ..., M)$ during their time-windows whereas the offline server can serve all the requests except for request r_0'. Thus, the online strategy can serve no more than 1 request while the offline strategy can complete at least $1 + M$ requests. That is, $\frac{B_{OPT}(R)}{B_A(R)} \leq \frac{1+M}{1}$. Since M is an arbitrary large number, the competitive ratio can be infinite. Hence we complete the proof of theorem 1.

In the rest of this section we pay our attention to two special types of metric space, the uniform metric space and K-constrained metric space.

A. Uniform Metric Space

In the case of uniform metric space we do not allow the server to drop an accepted request at any other place than its destination. This means, once a request is accepted, it will not be called off. It is also assumed that

(a). each request comes in with 1 unit of goods. That is, $z_i = 1$ for $i = 1, 2, ...$.

(b). the capacity of the server is 1. So the server can not accept new request until it completes the request on hand.

(c). the time-windows for all requests are at least 2 units of time. That is, $2 \leq h_i \leq \infty$ for $i = 1, 2, ...$.

Theorem 2. ***In the uniform metric space, for ODARPU TM under the conditions of (a), (b) and $h_i \geq 1$, $i \in (1, 2, ...)$, no online deterministic strategy can obtain a competitive ratio less than 2 and no online randomized strategy can obtain a competitive ratio less than 4/3.***

Proof: Consider two sequences of requests R and R', each with two requests r_1 and r_2. In sequence R, $r_1 = (0, 1, a_1, b_1, 3.5)$, $r_2 = (0.5, 1, b_1, b_2, 1)$. In sequence R', r_1 is same as before, $r_2 = (1, 1, b_1, b_2, 1)$. Assume that a_1, b_1, b_2 do not locate at the origin o, and $a_1 \neq b_1 \neq b_2$. If an online strategy ***A*** leave to serve request r_1 from origin at time t=0 with probability 0≤p≤1, then $B_A(R) = p \cdot 1 + (1-p) \cdot 2 = 2 - p$, and $B_A(R') = p \cdot 2 + (1-p) \cdot 1 = 1 + p$, where $B_A(R)$ and $B_A(R')$ denote the expect values of requests numbers completed by strategy ***A*** when serving the request sequences R and R', respectively. The offline strategy *OPT*, for sequence R, can leave origin point to serve request r_2 after 0.5 units of time, and then completes request r_1. For sequence R', the *OPT* can moves immediately to serve r_1 at time 0, then completes request r_2. So for both R and R', *OPT* can complete 2 requests. Thus, the competitive ratio for strategy ***A*** is at least

$$\max\left(\frac{2}{2-p},\frac{2}{1+p}\right)=\frac{2}{\min(2-p,1+p)}.$$

If ***A*** is an online deterministic strategy, $p=1$ or $p=0$, so $\min(2-p,1+p)=1$. Thus, the competitive ratio is at least 2. If ***A*** is an randomized strategy, $\min(2-p,\ 1+p)=3/2$, for $p=1/2$. Thus, the competitive ratio is $4/3$. This completes the proof.

We now give an online strategy, GREEDY, for ODARPU TM in which $2\le h_i\le\infty \ \ (i=1,2,...)$.

GREEDY: At time t, let $S(t)$ be the set of all waiting requests that have been presented but neither been served nor been called off. That is, for each request $r_i\in S(t)$, $t_i\le t\le t_i+h_i$ is hold. At any time t if the server is free and $S(t)\ne\phi$, the online server begins to serve the request which remains the smallest waiting time and can be accepted by its deadline.

Theorem 3: ***In the uniform metric space, for ODARPU TM under the conditions of (a), (b) and (c), GREEDY is a 3-competitive strategy.***

The proof is omitted for the page limitation.

B. K-constrained Metric Space

In the case of *K*-constrained metric space we allow the server to give up any accepted request at any point before it is completed. Once a request is abandoned by the server, it will not be served any more. However, some amount of penalty will take place if the server gives up an accepted request. The amount depends on the quantities of goods that are abandoned. Specifically, for a abandoned request with z_i units of goods, the amount of penalty $P_i=\rho z_i$, where $\rho>0$ is called the coefficient of penalty. The aim of the server is to deliver the maximum quantity of goods with as little penalties as possible, which can be measured by *Total income =Total goods delivered + Total penalty.*

Theorem 4. ***In the K-constrained metric space, no online deterministic strategy can obtain a constant competitive ratio for ODARPUTM if the time- windows $h_i<K$ for $i=1,2,\ldots$.***

The proof of theorem 4 is similar to that of theorem 1. The detail is omitted here.

The following discussion of this section is based on the assumptions as:

(d). all the time-windows of the requests are great than or equal to *K*.

(e). the goods quantity of requests may be different from each other, but not be great than the capacity of the server.

(f). the server can serve only one request for one time. In other words, once the server accepts a request, it can not accept another new request until the accepted request is completed or abandoned.

Now we give out an online *MGF* (Maximal Goods First) strategy.

MGF strategy: At time t, also let $S(t)$ denotes the set of all available requests. At any time t if the server is free and $S(t)\ne\phi$, the online server selects and serves the request with the maximal goods which can be accepted by its deadline. During the process of serving the request r_i with z_i units of goods, if a new request r_j with $z_j\ge\lambda z_i$ units of goods is released and can be accepted by its deadline, where $\lambda>1$, then the server gives up the accepted request and begins to serve r_j.

Theorem 5: ***In the K-constrained metric space, for ODARPUTM under the conditions of (d), (e) and (f), MGF is a c-competitive strategy, where***

$$c=\frac{2K\lambda^2+\lambda-1}{\lambda-\rho-1},\quad \lambda=\sqrt{(\rho+1)^2+\frac{\rho}{K}}+\rho+1$$ ***and $\rho>0$ is the coefficient of penalty.***

Proof: Consider a sequence R^{MGF} produced by *MGF*, it can always be divided into such *w* sub-sequences $R^{MGF}=(R_1,R_2,...,R_w)$ that in each sub-sequence R_i = $(r_{i,1},r_{i,2},...,r_{i,m})$ ($1\le i\le w$), just the last request $r_{i,m}$ is completed by the server of *MGF*. And the rest of requests $r_{i,j}$ in R_i for $j=1,...,m-1$ are abandoned by the server, implying that $z_j\ge\lambda z_{j-1}$ for $j=1,2,...,m$.

We will first consider an arbitrary sub-sequence R_i and then extent the result to the whole sequence R^{MGF}. Let $B_{MGF}(R_i)$ be the total income of *MGF* for serving sub-sequence R_i, and $B_{OPT}(R_i)$ denotes the total income that the offline strategy *OPT* gains during the time when *MGF* serves sub-sequence R_i. In fact, $B_{OPT}(R_i)$ is exactly the total quantity of goods that *OPT* delivered because *OPT*, as an offline optimal strategy, will not give up any request.

For online *MGF* strategy, the total quantity of goods that are completed is z_m. And the total penalty is $\sum_{j=1}^{m-1}\rho z_j$ caused by $m-1$ requests $r_{i,j}$ ($j=1,...,m-1$) which are abandoned, each with ρz_j penalties. So the total income of *MGF* is

$$B_{MGF}(R_i)=z_m-\sum_{j=1}^{m-1}\rho z_j\ge(1-\frac{\rho}{\lambda-1})z_m \quad (2)$$

For offline *OPT* strategy, it can design a sequence $R^* = (r_1^*, r_2^*, ..., r_{2K}^*)$ and serve it while the *MGF* strategy server is working for a certain request $r_{i,j}$ ($1 \le j \le m$). The longest time needed to complete $r_{i,j}$ for *MGF* strategy is $2K$ since it takes at most *K* units of time to travel between any two points in the *K*-constrained metric space. So there are at most 2*K* units of time for *OPT* to serve the 2*K* requests in R^* and complete all of them as long as:

1. the goods quantity of request r_l^* ($l = 1, 2, ..., 2K$) in R^*, denoted by z_l^*, is not great than λz_j ($1 \le j \le m$), where z_j is the goods quantity of $r_{i,j}$.

2. $a_1^* = a_r$, $a_l = b_{l-1}^* (l \ge 2)$ and $d(a_l^*, b_l^*) = 1$ $(l = 1, 2, ...2K)$, where a_l^*, b_l^* are source point and destination point of request $r_l^* \in R^*$, respectively. And a_r is the position of *OPT*'s server when the *MGF*'s server starts to serve $r_{i,j}$. In addition, b_1^* can be arbitrary position as long as $d(a_1^*, b_1^*) = 1$.

The first condition makes sure that the online *MGF* strategy does not give up $r_{i,j}$ when r_l^* is presented. And the second one makes it is possible for *OPT* to complete 2*K* requests in the sequence R^* within 2*K* units of time.

If the time-window length of the last request r_m is long enough, *OPT* can also complete this request after it completes all the requests in R^* (at most 2*K*). So the total quantity of goods completed by *OPT* when the *MGF*'s server is serving sub-sequence R_i is at most

$$2K\sum_{l=1}^{m-1} z_l^* + z_m < \left[2K\lambda^2/(\lambda-1)+1\right] z_m \quad (3)$$

So we have,

$$\frac{B_{OPT}(R_i)}{B_{MGF}(R_i)} \le \frac{\left(\frac{2K\lambda^2}{\lambda-1}+1\right)z_m}{\left(1-\frac{\rho}{\lambda-1}\right)z_m} = \frac{2K\lambda^2+\lambda-1}{\lambda-\rho-1} \quad (4)$$

The optimal choice in (4) for λ is $\sqrt{(\rho+1)^2+\frac{\rho}{K}}+\rho+1$, which produces a minimum value.

After *MGF* strategy finishes the whole sequence R^{MGF}, the total income of strategy is the sum of total incomes in each sub-sequence. So the competitive ratio of *MGF* strategy is $\frac{2K\lambda^2+\lambda-1}{\lambda-\rho-1}$, where $\lambda = \sqrt{(\rho+1)^2+\frac{\rho}{K}}+\rho+1$. Hence we complete the proof.

IV. CONCOLUSIONS

In this paper we discuss the ODRPUTM which is a more general case of ODRPTM. We consider the problem in two special metric spaces since no competitive deterministic online strategy exists in a non-constrained metric space. We perform the competitive analysis of GREEDY strategy in the uniform metric space and of *MGF* (Maximal-Goods-First) strategy in the *K*-constrained metric space. It will be interesting to design randomized strategies for this problem to obtain a better result. Another interesting direction is to study the problem with multiple servers, which would be a stimulating extension to our framework, both from a theoretical and practical point of view.

REFERENCES

[1] F. L. Yi, Y. F. Xu, C. L. Xin, "On the Online Dial-A-Ride Problem with Time-Windows", Information, 2007,10(2) pp.161-169

[2] F.L. Yi, Y.F. Xu, C.L. Xin. "Online Dial-a-ride Problem with Time-windows under a Restricted Information Model",Lecture Notes in Computer Science ,4041, pp. 22-41, Hong Kong, PRC, June 2006.

[3] Krumke S.O., Laura L., Lipmann M., and Marchetti-Spaccamela et al. Non-abusiveness helps: An o(1)-competitive algorithm for minimizing the maximum flow time in the online traveling salesman problem. Lecture Notes in Computer Science, 2002,pp. 200–214.

[4] Feuerstein E. and Stougie L. On-line single server dial-a-ride problems. Theoretical Computer Science, 2001, 268(1), pp.91–105.

[5] Ascheuer N., Krumke S.O., and Rambau J. Online dial-a-ride problems: Minimizing the completion time. Lecture Notes in Computer Science, 2000, pp.639–650.

[6] Ausiello G., Feuerstein E., Leonardi S., Stougie L., and Talamo M. Algorithms for the on-line traveling salesman. Algorithmica, 2001, 29(4), pp.560–581.

[7] Krumke S.O., de Paepe W.E., Poensgen D., and Stougie L. News from the online traveling repairman. Theoretical Computer Science, 2003, 295,pp.279–294.

[8] Hauptmeier D., Krumke S.O., and Rambau J. The online dial-a-ride problem under reasonable load. Lecture Notes in Computer Science, 2000,pp.125–136.

[9] Xu Y.F. and Wang K.L. Scheduling for on-line taxi problem and competitive algorithms. Journal of Xi' an Jiao Tong University, 1997, 31(1),pp.56–61.

[10] Xu Y.F., Wang K.L., and Zhu B. On the k-taxi problem. Journal of Information, 1999, vol.2, pp. 429–434.

[11] Xu Y.F., Wang K.L., and Ding J.H. On-line k-taxi scheduling on a constrained graph and its competitive algorithm. Journal of System Engineering(P.R. China), 1999.vol. 4,pp.79-84.

[12] Ma W.M., XU Y.F., and Wang K.L. On-line k-truck problem and its competitive algorithm. Journal of Global Optimization, 2001, 21(1), pp.15–25.

[13] Lipmann M., Lu X., de Paepe W.E., and Sitters R.A. On-line dial-a-ride problems under a restricted information model. Lecture Notes in Computer Science, 2002,vol. 2461, pp.674–685.

[14] Psaraftis H.N. An exact algorithm for the single vehicle many-to-many dial-a-ride problem with time windows. Transportation Science, 1983, vol.17, pp.351–357.

[15] Diana M. and Dessouky M.M. A new regret insertion heuristic for solving largescale dial-a-ride problems with time windows. Trasportation Reserch Part B, 2004,vol. 38 pp. 539–557.

[16] Manasse M.S., McGeoch L.A., and Sleator D.D.: Competitive algorithms for server problems.Journal of Algorithms , 1990,vol.11(2) pp.208-230.

[17] David S.B., Borodin A.: A new measure for the study of the on-line algorithm. Algorithmica, 1994, 11(1) ,pp. 73-91.