Online Algorithms for the Vehicle Scheduling Problem with Time Objective

Maolin Hu[1,2], Yinfeng Xu[2], and Weijun Xu[2]

[1] Department of Mathematics, Guyuan Teachers college, Guyuan, 756000, China
{humaolin2000, xuweijun75}@163.com
[2] School of Management, Xi'an Jiaotong University, Xi'an, 710049, China
yfxu@mail.xjtu.edu.cn

Abstract. Falling across the problem of the unforeseen congested events in the logistic transportation, we build the online vehicle scheduling model with the recoverable congested vertices in this paper. Our optimization objective minimizes the total time of online scheduling. Considering the dynamic occurrence characteristics of the congested vertices one by one, we first introduce three online scheduling strategies, i. e., Greedy Strategy, Reposition Strategy, and Waiting Strategy, and analyze the advantages and disadvantages of competitive performances of the three strategies. Then we propose the Simple Choice Strategy. By competitive performance analysis, we think that the Simple Choice Strategy could be an optimal scheduling scheme for online vehicle transportation.

1 Introduction

In practice, the carrier often transports goods by the vehicle from the origin to the destination. The crossings and roads of traffic networks are viewed as the vertices and edges of the weighted and undirected graph $G = (V, E)$, where V is the set of all vertices with $|V| = n$, and where E is the set of all weighted edges. Let O be the origin and D the destination respectively, and O_i $(1 \leq i \leq n-2)$ be the other vertex. Let $w(x, y)$ be the time needed for the vehicle to move from the vertex x to the vertex y in graph G. In fact, the carrier always hopes that the vehicle can move along the optimal route so as to minimize the time, while the optimal route is not creditable, because some vertices may be blocked so as not to pass through. Let us first consider the following problems:

Q1. If the carrier knows the congested sequence $R_k = (r_1, r_2, \cdots, r_k)$ and the relatively recoverable time sequence $T(R_k) = (t(r_1), t(r_2), \cdots, t(r_k))$ in advance, then how the vehicle is scheduled to minimize the total time?

Q2. If the information of the congested sequence R_k and the relatively recoverable time sequence T_k is dynamically available, i.e., when the carriers can only know the past and present information, but can not know the future information, how the vehicle is scheduled to minimize the total time?

Since Q1 is an offline problem, the carriers can obtain the optimal route by using the classical optimization theory. While Q2 is an online problem, with the dynamic recurrence of the congested vertices and the recoverable time of the

X. Deng and Y. Ye (Eds.): WINE 2005, LNCS 3828, pp. 574–583, 2005.

relatively congested vertex one by one, it is difficult to deal with Q2. However, in recent years, there occurs an interesting research focusing in the field of algorithms, which is the online algorithm. The method of competitive analysis can be used to solve online problem in some sense [1–9].

2 Basic Assumptions

For the convenience of discussion, basic assumptions of online vehicle scheduling problem are stated as follows.

(1) Suppose the vehicle always meet k ($k > 0$) number vertices, while going from O to D by whatever route, and suppose $R_k = (r_1, r_2, \cdots, r_k)$ be a congested sequence, and $R_i = (r_1, r_2, \cdots, r_i)$ ($0 \leq i \leq k$) be a subsequence of R_k. For $i = 0$, R_0 means that the vehicle doesn't meet the congested vertex.

(2) Blockages happen at the crossings. Let $t(r_i)$ ($i = 1, 2, \cdots, k$) be the recoverable time of the congested vertex r_i, $T(R_k) = (t(r_1), t(r_2), \cdots, t(r_k))$ be the total recoverable time sequence of R_k, i.e., $T(R_k) = \sum_{i=1}^{k} t(r_i)$. Once the congested vertex is recoverable, it cannot be blocked again.

(3) Graph G' deriving from the removal of the congested vertices in Graph G is also connected.

(4) Let $T_{opt}(OD|R_i)$ ($0 \leq i \leq k$) be the minimum required time from O to D when knowing the occurrence of the congested sequence R_i in advance. For $i = 0$, let $T_{opt}(OD|R_i) = T_{opt}(OD)$.

(5) Suppose that the vehicle meet the congested vertex r_i at the adjacent vertex O_i ($1 \leq i \leq k$). Let $T_{opt}(O_iD|R_i)$ be the minimum time for the vehicle to move along the optimal route from O_i to D when the congested sequence R_i is known in advance.

(6) Let $TT_{ALG}(OD|R_i)$ ($1 \leq i \leq k$) be the total time that online algorithm ALG spent from O to D in the face of the congested sequence R_i one by one.

(7) Suppose that $T_{opt}(OD) + T(R_k) > T_{opt}(OD|R_k)$. Otherwise, the vehicle can reach the destination within the shortest time after each recovery of the roads when meeting each congestion.

Definition 1. *For any graph G, $T_{opt}(OD|R_k)$ and $TT_{ALG}(OD|R_k)$ is called off-line time and on-line time separately.*

Definition 2. *For any graph G, if there is only constant α, related to the number k of sequence of the congested sequence R_k, and any constant β meets this condition*

$$TT_{ALG}(OD|R_k) \leq \alpha T_{opt}(OD|R_k) + \beta,$$

then ALG is called as α-competitive for the on-line problem [1–5].

In fact, the following lemma can be directly obtained from the graph theory.

Lemma 1. *It holds that*

$$T_{opt}(OD) \leq T_{opt}(OD|R_1) \leq T_{opt}(OD|R_2) \leq \cdots \leq T_{opt}(OD|R_k).$$

3 Analysis on the Three Basic Strategies

Literature [10] has the minimum cost needed for vehicles in transportation as its optimized objective, putting forward the competitive algorithms of greedy strategy and reposition strategy. Here we will draw out these two strategies and analyze their competitive ratio and competitive performance, taking time as the optimized objective.

3.1 Greedy Strategy

Greedy Strategy: When the carrier reaches O_i and finds that the next vertex r_i is congested, he can choose the shortest path from the current vertex O_i to D after excluding the congested vertex sequence R_i, and then moves along this path.

Denote the Greedy Strategy as A, and we obtain the following theorem.

Theorem 1. *For online vehicle scheduling problem with the congested sequence R_k, the competitive ratio of A is $2^{k+1}-1$.*

Proof. The vehicle starts from O and takes the route according to Greedy Strategy when meeting congested vertex and gets to the destination D, then the route that the vehicle takes is $OO_1O_2\cdots O_kD$. The total time it takes is $TT_A(OD|R)$, which satisfies

$$TT_A(OD|R_k) \le T_{opt}(OD)+T_{opt}(O_1D|R_1)+T_{opt}(O_2D|R_2)+\cdots+T_{opt}(O_kD|R_k).$$

Note that $T_{opt}(OO_1), T_{opt}(O_1O_2|R_1), T_{opt}(O_2O_3|R_2),\cdots, T_{opt}(O_{k-1}O_k|R_{k-1})$, are parts of $T_{opt}(OD), T_{opt}(O_1D|R_1), T_{opt}(O_2D|R_2), \cdots, T_{opt}(O_{k-1}D|R_{k-1})$, respectively. Using lemma 1, it follows that

$$\begin{aligned}
T_{opt}(O_1D|R_1) &\le T_{opt}(O_1O)+T_{opt}(OD|R_1) \le 2T_{opt}(OD|R_1),\\
T_{opt}(O_2D|R_2) &\le T_{opt}(O_2O_1|R_1)+T_{opt}(O_1O)+T_{opt}(OD|R_2)\\
&\le 2^2T_{opt}(OD|R_2),\\
\cdots,&\\
T_{opt}(O_{k-1}D|R_{k-1}) &\le T_{opt}(O_{k-1}O_{k-2}|R_{k-2})\\
&\quad +\cdots+T_{opt}(O_1O)+T_{opt}(OD|R_{k-1})\\
&\le 2^{k-1}T_{opt}(OD|R_{k-1}),\\
T_{opt}(O_kD|R_k) &\le T_{opt}(O_kO_{k-1}|R_{k-1})+T_{opt}(O_{k-1}O_{k-2}|R_{k-2})\\
&\quad +\cdots+T_{opt}(O_1O)+T_{opt}(OD|R_k)\\
&\le 2^kT_{opt}(OD|R_k).
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
TT_A(OD|R_k) &\le T_{opt}(OD)+2T_{opt}(OD|R_1)+2^2T_{opt}(OD|R_2)\\
&\quad +\cdots+2^kT_{opt}(OD|R_k)\\
&\le (2^{k+1}-1)T_{opt}(OD|R_k).
\end{aligned}$$

This ends the proof.

Remark 1. Note that, regarding to Greedy Strategy, the online time that the vehicle takes to move goods from O to D can be $2^{k+1} - 1$ times as much as it takes for the off-line problem in the worst case. So this strategy is not feasible in such case. But if the time taken to go along the optimized route every time when meeting with the congested vertex is "relatively shorter", it's quite economic to adopt this strategy. In normal situation, as for the congested vertex, the time it takes to go along the roads can be either short (i.e., "good road") or long (i.e., "bad road"). Going along the "good road" is of course more economic and going along the "bad road" takes longer time. So it comes to the situation that it's a pity to give it up but risky to adopt. The competitive performance of this strategy is positive, but risky.

3.2 Reposition Strategy

Reposition Strategy: When the carrier reaches O_i and finds that the next vertex r_i is congested, he moves back to the origin O along the current route and then chooses the optimal path in G exclusive of the set of the known congested vertices.

Denote Reposition Strategy as B, and we have the following theorem.

Theorem 2. *For online vehicle scheduling problem with the congested sequence R_k, the competitive ratio of B is $2k + 1$.*

Proof. The vehicle departs from the origin O. According to Reposition Strategy, the total online time taken to go from O to D for the vehicle when meeting the congested vertex is $TT_B(OD|R)$. It satisfies

$$\begin{aligned} TT_B(OD|R_k) &\leq 2T_{opt}(OD) + 2T_{opt}(OD|R_1) + 2T_{opt}(OD|R_2) \\ &\quad + \cdots + 2T_{opt}(OD|R_{k-1}) + T_{opt}(OD|R_k) \\ &\leq 2T_{opt}(OD|R_k) + 2T_{opt}(OD|R_k) + 2T_{opt}(OD|R_k) \\ &\leq\ + \cdots + 2T_{opt}(OD|R_k) + T_{opt}(OD|R_k) \\ &= (2k+1)T_{opt}(OD|R_k). \end{aligned}$$

This completes the proof.

Remark 2. Note that, regarding to Reposition Strategy, the time that the vehicle takes to move goods from the origin to the destination can be $2k + 1$ times as much as it takes for the off-line problem in the worst case. Compared with Greedy Strategy, Reposition Strategy is avoiding the risk from taking the "bad road" but missing the opportunity of taking the "good road". Taking Reposition Strategy is not risky but conservative. So we say, the competitive performance of Reposition Strategy is safe but conservative.

3.3 Waiting Strategy

Waiting Strategy: The carrier stays waiting every time when meeting a congested vertex until it is recoverable, and then he continues moving along the original route.

Denote Waiting Strategy as C. We have that the online time is $TT_C(OD|R_k)=$ $T_{opt}(OD) + T(R_k)$, and the offline time is $T_{opt}(OD|R_k)$. Thus, it follows that

$$\frac{TT_C(R_k)}{T_{opt}(OD|R_k)} = \frac{T_{opt}(OD) + T(R_k)}{T_{opt}(OD|R_k)}.$$

Remark 3. The competitive ratio of Waiting Strategy is not only connected with the number of the congested vertex but the waiting time $T(R_k)$. When $T(R_k)$ is small, the competitive ratio is close to 1; but when $T(R_k)$ is large, the competitive ratio is large, too. Therefore, the competitive ratio of Waiting Strategy is increasing with $T(R_k)$, to an endless amount. So the competitive performance of Waiting Strategy is both economic and risky.

Here, a very practical question is lying before the carriers: which strategy are they to choose to arrange their vehicles? Let's first look at a simple example.

3.4 A Simple Example

Suppose in Figure 1, the vehicle, going from O to D, will in turn meet three congested vertices $R = \{r_1, r_2, r_3\}$ at the the adjacent vertices O_i $(i = 1, 2, \cdots, 18)$ while travelling by whatever route. And we have $t(r_1) = 2$, $t(r_2) = 1$, $t(r_3) = 0.5$. To distinguish two routes between O_6 and D, we mark a node O_0. Suppose that no information about three congested vertices or three adjacent vertices be known before hand, nor be the information about the removing time known.

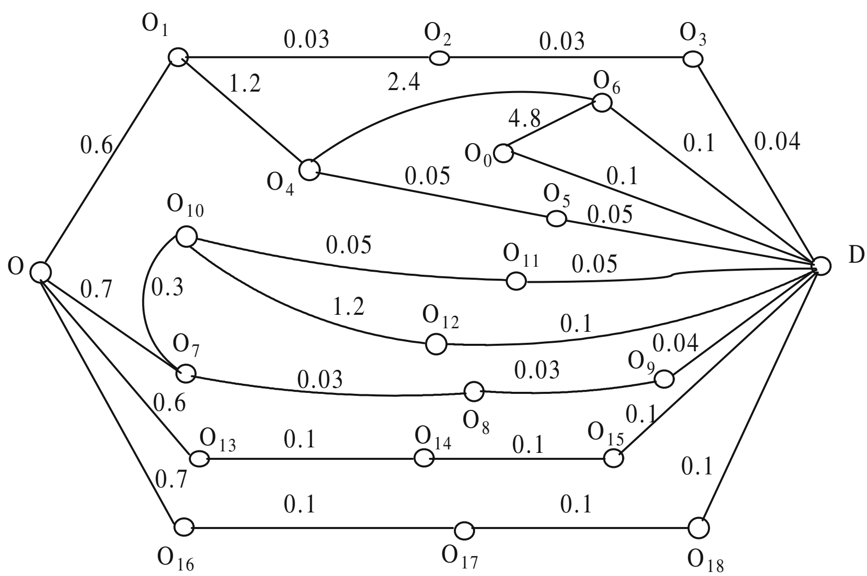

Fig. 1. The Diagram of the Competitive Analysis

The weight of edges shows the time for the vehicle to go through the distance. What is to be considered now is which road to take so that the time is relatively less.

(1) If the vehicle takes the route according to Greedy Strategy, going along the optimized route $OO_1O_2O_3D$, it meets the congested vertex r_1 when coming to O_1; if r_1 is known, the optimized route $O_1O_4O_5D$ from O_1 to D can be figured out. Then the vehicle goes along this route. When it reaches vertex O_4 and meets the congested vertex r_2, the optimized route O_4O_6D from O_4 to D can be figured out if r_1, r_2 is known. Then the vehicle goes on along this route. When it comes to vertex O_6 and meets the congested vertex r_3, if r_1, r_2, r_3 is already known, the optimized route O_6O_0D from O_6 to D can be figured out., then the vehicle goes on along this route to the destination D. Therefore, the route that the vehicle takes is $OO_1O_4O_6O_0D$. It's easy to figure out the time how long it takes the vehicle to go through this route, which is 9.1.

(2) If the vehicle takes the route according to Reposition Strategy, it goes along the optimized route $OO_1O_2O_3D$ and reaches the vertex O_1. Then when it meets the congested vertex r_1, it returns along O_1O. If r_1 is already given, going along the optimized route $OO_7O_8O_9D$, it will return along O_7O at the vertex O_7 when meeting congested vertex r_2; if r_1, r_2 is given, going along the optimized route $OO_{13}O_{14}O_{15}D$, it will return along $O_{13}O$ at the vertex O_{13} when meeting congested vertex r_3; if r_1, r_2, r_3 is given, going along the optimized route $OO_{16}O_{17}O_{18}D$, it will reach the destination D. So the route that the vehicle takes is $OO_1OO_7OO_{13}OO_{16}O_{17}O_{18}D$. It's easy to figure out the time the vehicle takes is 4.8.

(3) If the vehicle takes the route according to Waiting Strategy, it will stop to wait at the vertex O_1 when meeting congested vertex r_1 while going along the optimized route $OO_1O_2O_3D$. In a time period of two after the congested vertex r_1 is recoverable, the vehicle continues its trip along the original route. Then it reaches vertex O_2 and meets congested vertex r_2. After waiting for a time-period of one, the congested vertex r_2 is recoverable and the vehicle continues its trip along the original route and reaches the vertex O_3. At the vertex 3 it meets congested vertex r_3 then wait for a time period of 0.5. After congested vertex r_3 is removed, this vehicle goes along the same route and reaches the destination D. It's easy to figure out the time that the vehicle takes in the whole trip is 4.2.

(4) Now let's consider another strategy. For the convenience of discussion, we assume that the minimum time be the optimal route, because $w(\cdot)$ shows the time of the optimal route, i.e., the minimum time. The vehicle starts at O along the optimized route $OO_1O_2O_3D$ and meets congested vertex r_1 at the vertex O_1. If r_1 is given, the optimized route from O_1 to D can be figured out, which is $w(O_1O_4O_5D) = 1.3$; and the optimized route from O to D, which is $w(OO_7O_8O_9D) = 0.8$. Meanwhile, both the waiting time at vertex O_1, which is $t(r_1) = 2$, and the sum of it with the distance $w(O_1O_2O_3D) = 0.1$ to go after the congested vertex r_1 is removed, can be figured out, which is 2.1. It can be seen that $w(OO_7O_8O_9D) = 0.8$ is the shortest among the three alternatives. Then the decision is to go back from O_1 to O, and go on along

$OO_7O_8O_9D$. It meets congested vertex r_2 when coming to the vertex O_7. If r_1, r_2 is given, the following items can all be figured out: the optimized route from O_7 to D is $w(O_7O_{10}O_{11}D) = 0.4$ and the optimized route from O to D is $w(OO_{13}O_{14}O_{15}D) = 0.9$; the waiting time at the vertex O_7 is $t(r_2) = 1$, plus the distance $w(O_7O_8O_9D) = 0.1$ to go after r_2 is recoverable, the sum is 1.1. The smallest among the three is $w(O_7O_{10}O_{11}D) = 0.4$, so the vehicle chooses to go along this route. It comes to the vertex O_{10} and meets congested vertex r_3. Knowing r_1, r_2, r_3, the optimized route $w(O_{10}O_{12}D) = 1.3$ from O_{10} to D; and the optimized route $w(OO_{13}O_{14}O_{15}D) = 0.9$ from O to D can be figured out. And the waiting time at the vertex O_{10} is $t(r_3) = 0.5$; the sum of it with the distance $w(O_{10}O_{11}D) = 0.1$ to go after the congestion at the vertex r_3 is recoverable, is 0.6. And the smallest of the three is $t(r_3) = 0.6$, so the decision is to wait at the vertex O_{10} for a time period of 0.5 until the congestion at the vertex r_3 is recoverable. The vehicle goes on along the route $O_{10}O_{11}D$ and reaches the destination D. It's easy to figure out the time for the vehicle to finish the whole trip is 2.8.

Remark 4. It can be seen from this example that, just as what has already been said, the performance of Greedy Strategy is active but risky; Reposition Strategy, conservative but safe; and the performance of Waiting Strategy is directly related to $T(R_k)$. In Case (4), Reposition Strategy is adopted to avoid the risk of going along the "bad road". It also choose Greedy Strategy to grasp the chance of going the "good road". Meanwhile, it chooses to wait for some time to go on the optimized route. As a result, it is better than other strategies in practical use. Being illuminated by Case (4), we hope to find a good strategy in way of combining the three strategies together, so that the best choice may be made among the three strategies: Greedy Strategy, Reposition Strategy and Waiting Strategy, every time when the vehicle meets congestion. Thus, we will present the following Simple Choice Strategy that is more economic and feasible.

4 Competitive Analysis of Simple Choice Strategy

Simple Choice Strategy: When the vehicle meets a congested vertex r_i with the number i $(i = 1, 2, \cdots, k)$, the carrier first figures out

$$T_i = \min\{T_{opt}(O_iD|R_i), T_{opt}(OD|R_i), t(r_i) + T_{opt}(O_iD|R_{i-1})\},$$

and then he chooses which strategy to go the later route according to the following condition.

(1) If $T_i = T_{opt}(O_iD|R_i)$, the vehicle will choose the route according to Greedy Strategy.

(2) If $T_i = T_{opt}(OD|R_i)$, the vehicle will go according to Reposition Strategy.

(3) If $T_i = t(r_i) + T_{opt}(O_iD|R_{i-1})$, the vehicle will go according to Waiting Strategy .

Note that, to figure out T_i, knowing $T_{opt}(O_iD|R_i), T_{opt}(OD|R_i)$ and $t(r_i) + T_{opt}(O_iD|R_{i-1})$. If three of them are equal, or at least two of them are equal, the carrier takes $T_i = t(r_i) + T_{opt}(O_iD|R_{i-1})$ first, then $T_i = T_{opt}(O_iD|R_i)$.

Denote Comparison Strategy as D, and then it comes the following theorem.

Theorem 3. *For online vehicle scheduling problem with the congested sequence R_k and the relatively recoverable time sequence $T(R_k)$, the competitive ratio of D is $2k+1$.*

Proof. (1) For all $i \in \{1, 2, \cdots, k\}$, if there always comes $T_i = T_{opt}(O_iD|R_i)$, then the vehicle can choose the route according to Greedy Strategy all the way. The total time is $TT_D(OD|R_k)$, which satisfies

$$\begin{aligned} TT_D(OD|R_k) &\leq w(OO_1)+T_{opt}(O_1D|R_1)+T_{opt}(O_2D|R_2)+\cdots+T_{opt}(O_kD|R_k) \\ &\leq T_{opt}(OD)+T_{opt}(OD|R_1)+T_{opt}(OD|R_2)+\cdots+T_{opt}(OD|R_k). \end{aligned}$$

Using lemma 1, it follows that

$$TT_D(OD|R_k) \leq (k+1)T_{opt}(OD|R_k).$$

(2) In the worst case, for all $i \in \{1, 2, \cdots, k\}$, if there comes $T_i = T_{opt}(OD|R_i)$, then the vehicle will go along the route according to Reposition Strategy all the way. According to theorem 2, the total time is $TT_D(OD|R_k)$, which satisfies

$$TT_D(OD|R_k) \leq (2k+1)T_{opt}(OD|R_k).$$

(3) For all $i \in \{1, 2, \cdots, k\}$, if there always comes $T_i = t(r_i)+T_{opt}(O_iD|R_{i-1})$, then the vehicle will go on the route according to Waiting Strategy. The total time is

$$\begin{aligned} TT_D(OD|R_k) &= T_{opt}(OD) + T(R_k) \\ &\leq T_{opt}(OD) + \sum_{i=1}^{k} T_{opt}(OD|R_i) \\ &\leq (k+1)T_{opt}(OD|R_k). \end{aligned}$$

(4) There exist j vertices $r_{i_1}, r_{i_2}, \cdots, r_{i_j}$ $(1 \leq i_1 < i_2 < \cdots < i_j \leq k, 1 \leq j < k)$, so that

$$\begin{aligned} T_{i_1} &= T_{opt}(OD|r_1, r_2, \cdots, r_{i_1}), \\ T_{i_2} &= T_{opt}(OD|r_1, r_2, \cdots, r_{i_2}), \\ &\cdots, \\ T_{i_j} &= T_{opt}(OD|r_1, r_2, \cdots, r_{i_j}). \end{aligned}$$

(i) The vehicle starts from O. Before the vehicle reaches the congested vertex r_{i_1}, it is either going along the road of Greedy Strategy or along the road of Waiting Strategy. According to (1) and (3), the time it takes w_{11} satisfies

$$w_{11} \leq TT_D(OD|R_{i_1-1}) \leq i_1 T_{opt}(OD|R_k).$$

In this case, because $T_{i_1} = T_{opt}(OD|R_{i_1})$, the vehicle needs to return to the origin O by the original road according to the Simple Choice Strategy. The vehicle takes $w_{12} \leq w_{11}$ to get back, as it doesn't need to wait at the original congested vertex.

So far, the time that the vehicle needs to go back and forth w_1 satisfies

$$w_1 = w_{11} + w_{12} \leq 2i_1 T_{opt}(OD|R_k).$$

(ii) The vehicle starts from O in the second time. Before it reaches the congested vertex r_{i_2}, it is either going along the road of Greedy Strategy or Waiting Strategy. According to (1) and (3), the time it takes w_{21} satisfies

$$w_{21} \leq TT_D(OD|r_1, r_2, \cdots, r_{i_1}, r_{i_1+1}, r_{i_1+2}, \cdots, r_{i_2-1}) \leq (i_2 - i_1)T_{opt}(OD|R_k).$$

In this case, similar to (i), it takes the vehicle w_2 of time to finish its trip back and forth, which satisfies

$$w_2 \leq 2w_{21} \leq 2(i_2 - i_1)T_{opt}(OD|R_k).$$

The rest steps may be deduced by analogy.

(iii) For the time of j, the vehicle starts from O, going either along the road of Greedy Strategy or Waiting Strategy. Before reaching the congested vertex r_{i_j}, the vehicle goes back by the original road to O, similar to (ii). This time, it takes the vehicle w_j of time to finish its trip back and forth, which satisfies

$$w_j \leq 2(i_j - i_{j-1})T_{opt}(OD|R_k).$$

(iv) For the time of $j+1$, the vehicle starts from O, going along the route either of Greedy Strategy or of Waiting Strategy, until it reaches the destination D. According to (1) and (3), the time that the vehicle takes to cover this distance is w_{j+1}, which satisfies

$$w_{j+1} \leq (k - i_j + 1)C_{opt}(OD|R).$$

Therefore, from the origin O to the destination D, the total time is

$$\begin{aligned} TT_D(OD|R_k) \leq \sum_{l=1}^{l=j+1} w_l &\leq 2i_1 T_{opt}(OD|R_k) + 2(i_2 - i_1)T_{opt}(OD|R_k) \\ &+ \cdots + 2(i_j - i_{j-1})T_{opt}(OD|R_k) + (k - i_j + 1)T_{opt}(OD|R_k) \\ &= (k + i_j + 1)T_{opt}(OD|R_k). \end{aligned}$$

Taking (1), (2), and (3) together, according to the definition of competitive ratio, we can come to the conclusion that the competitive ratio of the Simple Choice Strategy D is $2k+1$ for online vehicle scheduling problem with the congested sequence R_k. This completes the proof.

Remark 5. From the discussion above, it can be seen that Simple Choice Strategy is ultimately promoted in competitive performance, although it has the same competitive ratio that is $2k+1$ as Reposition Strategy. It fully takes the dynamic features of the congested vertices into account and the advantages of the three strategies: Greedy Strategy, Waiting Strategy and Reposition Strategy. Different strategies are adopted in different situations, so that the best choice is made among the three strategies every time when the vehicle meets congestion. Greedy Strategy is used to fully grasp the chance of going the good road; in compulsive situation, Reposition Strategy is used to avoid the risk of going the bad road or waiting for too long; choosing to wait for a short time is to go on along the optimized route and avoid the risk of going the bad road. Furthermore, it can avoid the vehicle to make unnecessary reposition, which is a conservative way. All in all, the Simple Choice Strategy is a better strategy to integrate the advantages of all these strategies.

5 Conclusion

The congested vertex problem, which is not foreseeable in choosing what route to take, is a difficulty in dispatching goods. It is worthwhile to fully discuss how to put the theoretical results into practice. This paper is giving a tentative solution for online vehicle scheduling problem in transportation, which of course needs further discussion, for example, how to make the choice of optimized route, which can save time and cost, whether the congested vertex can recover as time goes on, etc.

References

1. Bar-noy, A., Schieber, B.: The canadian traveler problem. Proceedings of the second annual ACM-SIAM symposium on Discrete algorithms. (1991) 261–270
2. David, S.B., Borodin, A.: A new measure for the study of the on-line algorithm. Algorithmica. **11(1)** (1994) 73–91.
3. Du, D.Z.: k-server problem and competitive algorithms. Mathematics in Practice and Theory (in chinese). **(4)** (1991) 36–40
4. Manasse, M.S., McGeoch, L.A., Sleator, D.D.: Competitive algorithms for server problems. Journal of Algorithms. **11(2)** (1990) 208–230
5. Koutsoupias, E., Papadimitriou, C.: On the k-server conjecture. Journal of ACM. **42(5)** (1995) 971–983
6. Alon, N., Karp, P.M., Peleg, D., et al.: A graph-theoretic game and its application to the k-server problem. SIAM Journal on Computing. **24(1)** (1995) 78–100
7. Papadimitriou, C.H., Yannakakis, M.: Shortest paths without a map. Theoretical Computer Science. **84** (1991) 127–150
8. Pinnoi, A., Tung, D.T.: Vehicle routing-scheduling for waste collection in Hanoi. European Journal of Operational Research. **125(3)** (2000) 449–468
9. Woeginger, G.J.: Open problems in the theory of scheduling. Bulletin of the EATCS. **76** (2002) 67–83
10. Zhu, Z.J., Xu Y.F., Lü C.C.: Scheduling for on-line routing problem and its competitive strategy analysis. Journal of Systems Engineering. **18(4)** (2003) 324–330