



HAL
open science

On the Online Min-Wait Relocation Problem

Pascal Halffmann, Sven O. Krumke, Alain Quilliot, Annegret K Wagler,
Jan-Thierry Wegener

► **To cite this version:**

Pascal Halffmann, Sven O. Krumke, Alain Quilliot, Annegret K Wagler, Jan-Thierry Wegener. On the Online Min-Wait Relocation Problem. *Electronic Notes in Discrete Mathematics*, 2015, 50, pp.281-286. hal-02045800

HAL Id: hal-02045800

<https://hal.science/hal-02045800v1>

Submitted on 22 Feb 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On the Online Min-Wait Relocation Problem

Pascal Halffmann^a, Sven O. Krumke^a, Alain Quilliot^b,
Annegret K. Wagler^b and Jan-Thierry Wegener^{b,1}

^a *University of Kaiserslautern (Department of Mathematics)
Kaiserslautern, Germany*
{halffman,krumke}@mathematik.uni-kl.de

^b *Université Blaise Pascal (Clermont-Ferrand II, LIMOS)
BP 10125, 63173 Aubière Cedex, France*
{quilliot,wagler,wegener}@isima.fr

Abstract

In a carsharing system, a fleet of cars is distributed at stations in an urban area, customers can take and return cars at any time and station. For operating such a system in a satisfactory way, the stations have to keep a good ratio between the total number of places and cars in each station, in order to refuse as few customer requests as possible. This leads to the problem of relocating cars between stations. We consider the Online Min-Wait Relocation Problem, aiming at satisfying all customer requests with a minimal total waiting time, and show the non-existence of competitive online algorithms against several adversaries. Furthermore, we consider the max/max ratio, and show that this ratio cannot be used to theoretically evaluate online algorithms for the Online Min-Wait Relocation Problem either.

Keywords: relocation problem, competitive ratio, max/max ratio.

1 Introduction

Carsharing is a modern way of car rental, attractive to customers who make only occasional use of a car on demand. In a carsharing system, a fleet of cars is distributed at specified stations in an urban area, customers can take

¹ This work was funded by the French National Research Agency, the European Commission (Feder funds) and the Région Auvergne in the Framework of the LabEx IMobS3.

a car at any time and station and return it at any time and station, provided that there is a car available at the start station and a free place at the final station. To ensure the latter, customers do not book their requests in advance, but wait at a station until their request can be fulfilled. For operating such a system in a satisfactory way, the stations have to keep a good ratio between the total number of places and the number of cars in each station, in order to keep the waiting times for the customers as low as possible. This leads to the problem of balancing the load of the stations, called *Relocation Problem*: an operator has to monitor the load situations of the stations and to decide when and how to move cars from “overfull” stations to “underfull” ones.

Balancing problems of this type occur for any car- or bikesharing system, but the scale of the instances, possibility and time delay for prebookings and the possibility to move one or more vehicles in balancing steps differ. We consider an innovative carsharing system, where the cars are partly autonomous, which allows to build wireless convoys of cars lead by a special vehicle, such that the whole convoy is moved by only one driver (cf. [6]). This setting is similar to bikesharing, where trucks can simultaneously move several bikes during the relocation process [4,5]. Since customers of our carsharing system do not book their requests in advance, the main goal is to guarantee a balanced system during working hours (online as in [5,9]) or to set up an appropriate initial state for the morning (static situation as in [4,9]). Both, the online and the static versions are known to be \mathcal{NP} -hard [4], and different heuristics have been developed, e.g., by partitioning the problem into subproblems with clustering techniques [11], or by reducing the search space [10].

In this extended abstract, we address the online situation where cars can be transferred between stations during the working hours in order to satisfy the needs of the customers. Hereby, customers do not book their requests in advance as they do in [10] but wait at a station until their request can be fulfilled. We model the Relocation Problem in the framework of a metric task system. By [9] the studied carsharing system can be understood as a discrete event-based system, where the system components are the stations v_1, \dots, v_n , each having an individual capacity $\text{cap}(v_i)$, a system state $\mathbf{z}^t \in \mathbb{Z}^n$ specifies for each station v_i its load z_i^t at a time point t and \mathbf{z}^t changes when customers or convoy drivers take or return cars at stations. Customers demanding a car at time t from a station u , keeping the car for δ time units before bringing it back to station v , are modeled by *floating customer requests* $r = (t, u, v, \delta)$. The time t is called *release time*, u the *pickup station*, v the *drop station*, and δ the *occupation time* of r . In our considered situation, all floating customer requests must be *served*, i.e., it must be ensured that at a finite point in time

a customer can take a car from u to v . The time between the release time and the time when the floating customer request is served is called *waiting time*. Since (floating) customer requests may cause imbalances in the system, an operator has to monitor the system states and to decide when and how to move cars between the stations in order to fulfill all demands. For that, the operator monitors the evolution of system states over time, detects infeasible states ($z_v^t < 0$ or $z_v^t > \text{cap}(v)$) and creates *tasks* to move cars out of stations having an excess number of cars ($z_v^t > \text{cap}(v)$) into stations with a deficit number of cars ($z_v^t < 0$). To fulfill these tasks, tours have to be created for the convoys in order to perform the desired relocation process. For that, it is suitable to encode the urban area where the carsharing system is running as a *metric space* $M = (V, d)$ induced by a weighted graph $G = (V, E, w)$ with weight function $w : E \rightarrow \mathbb{R}_+$. The nodes correspond to stations, edges to their physical links in the urban area, and the distance d between two points $v_i, v_j \in V$ is the length of a shortest path from v_i to v_j . This yields a *metric task system*, a pair (M, \mathcal{T}) where $M = (V, d)$ is the above metric space and \mathcal{T} a set of tasks, as suitable framework to embed the tours for the convoys. A truck able to lead a convoy plays the role of a server. Each server has a capacity, corresponding to the maximum possible number of cars per convoy. A *tour* is assigned to every server, where the servers are instructed to move between stations and to add cars to (resp. remove cars from) the stations. A collection of tours, where every server has an assigned tour and that does not lead to infeasible system states is called a *transportation schedule*.

Problem 1.1 (Online Min-Wait Relocation Problem (M, R, k, L, x))

Given a metric space M , a sequence of floating customer requests R , k servers of capacity L , and x cars, find a transportation schedule with minimal total waiting time for the metric task system (M, \mathcal{T}) , where \mathcal{T} consists of tasks which are induced by R as above.

2 Competitive Ratio and Max/Max Ratio

Competitive analysis has become one standard way of measuring the quality of online algorithms, see e.g. [8]. The analysis can be viewed as a request/answer game, where an adversary generates a sequence of floating customer requests and an online algorithm tries to serve them. The performance of the online algorithm is then compared to that of an offline algorithm, which, in contrast to the online algorithm, has complete information about the whole request sequence in advance. Due to the advantage of having more information, the offline algorithm generally makes better decisions. However, the offline algo-

rithm cannot serve any floating customer request before its release time. If the ratio between the values of the solutions produced by the online $\text{ONL}(R)$ and the optimum offline algorithm $\text{OPT}(R)$ is finite for all sequences R , i.e., $\text{ONL}(R) \leq c \cdot \text{OPT}(R)$ for a $c \in \mathbb{N}$, the online algorithm is called competitive.

We show that there does not exist a competitive online algorithm for the Online Min-Wait Relocation Problem even against restricted adversaries. We derive our results for the *uniform metric space*, where all distances are equal to one. Although the uniform metric space is a strong simplification of the general situation, in transportation problems within a small central urban area, the travel distance between two stations may be determined not so much by the real distance, but by a general fixed setup time for loading and unloading.

First, we consider the *standard adversary* who knows the complete behavior of an online algorithm ONL and chooses a worst-case sequence for ONL . He is allowed to move convoys towards yet unreleased floating customer requests, but must not serve any request before its release time.

Theorem 2.1 *There does not exist a competitive online algorithm for the Online Min-Wait Relocation Problem (M, R, k, L, x) on the uniform metric space even if there are only three nodes in M , and $k = x = 1$.*

The basic idea of the proof is that the adversary releases a floating customer request with a high occupation time. The adversary waits until the online algorithm starts serving r^1 and then releases another floating customer request r^2 with occupation time 1. The adversary then serves r^2 before r^1 , while the online algorithm serves the requests in reversed order. Since we can select the occupation time of r^1 arbitrarily high, we can conclude that the ratio between the solutions produced by the online and the offline algorithm is arbitrarily high. Thus, there does not exist a competitive online algorithm for the Online Min-Wait Relocation Problem.

This proof motivates to restrict the set of floating customer requests the adversary can choose from. However, even when we restrict the adversary to select only from floating customer requests with an occupation time of 1 the situation does not change.

Theorem 2.2 *There is no competitive online algorithm for the Online Min-Wait Relocation Problem (M, R, k, L, x) on the uniform metric space against an adversary which releases only floating customer requests with an occupation time of 1, even if there are only three nodes in M , and $k = x = 1$.*

In terms of standard competitive analysis, the previous results show that one reaches the so-called *triviality barrier* [7]. With Yao's principle [2], one can

show that Theorem 2.1 and 2.2 holds for deterministic as well as randomized online algorithms. We now consider another method to analyze the efficiency of an online algorithm, the *max/max ratio*, which has been successfully applied to the k -server problem² [1,3]. A downside of competitive analysis is that some (intuitive) improvements of some algorithms do not necessarily lead to a better competitive ratio. A famous example is that online algorithms with a finite lookahead usually do not have a better competitive ratio than their counterparts with no lookahead at all. However, in [1] the authors could show, that applying the max/max ratio, there is a difference between algorithms with and without lookahead. One of the major motivations to try the max/max ratio on the Online Min-Wait Relocation Problem is that David and Borodin show in [1] that a greedy strategy (which is not a competitive online algorithm against an oblivious adversary) has a finite max/max ratio for the k -server problem. However, it turns out that there does not exist a properly defined max/max ratio for the Online Min-Wait Relocation Problem.

Let ALG be an algorithm with costs $\text{ALG}(R)$ on the input sequence R . The amortized costs $M(\text{ALG})$ are defined as $M(\text{ALG}) := \limsup_{\lambda \rightarrow \infty} M_\lambda(\text{ALG})$, where $M_\lambda(\text{ALG}) := \max_{|R|=\lambda} \text{ALG}(R)/\lambda$. The max/max ratio $w_M(\text{ONL})$ of an online algorithm ONL vs. an optimal offline algorithm OPT is then defined as

$$w_M(\text{ONL}) := \limsup_{\lambda \rightarrow \infty} \frac{M_\lambda(\text{ONL})}{M_\lambda(\text{OPT})} = \frac{M(\text{ONL})}{M(\text{OPT})}.$$

The last equation of the definition of $w_M(\text{ONL})$ is proved in [1, Lemma 4.2].

Theorem 2.3 *On the uniform metric space with at least two points, it holds: for every $\lambda \geq 2$ and for every $\mathcal{C} \in \mathbb{N}$ there exists a sequence R , with $|R| = \lambda$, so that $\text{OPT}(R) \geq \mathcal{C}$. This holds even in the case that there are only three nodes in M , and $k = x = 1$.*

The result can be proved similarly as Theorem 2.1. It implies that the value $M(\text{OPT})$ is not bounded. From $M(\text{ONL}) \geq M(\text{OPT})$, we conclude that $w_M(\text{ALG}) = \infty/\infty$ is indeterminate. However, when we restrict the set of floating customer requests as before to requests with an occupation time of 1, it turns out that several standard online algorithms for the k -server problem as “first come first serve” or “replan” have a max/max ratio of 1, since the worst-case sequences for an optimal offline algorithm are equal to the worst-case sequences for these online algorithms.

² In the k -server problem, tours for a fleet of k servers have to be computed such that every released city is visited. Hereby, the goal is to minimize the total tour length.

Overall, it seems that neither the competitive ratio nor the max/max ratio are meaningful tools for evaluating the efficiency of online algorithms for the Online Min-Wait Relocation Problem. Thus, developing a concept which allows us to theoretically evaluate online algorithms for the Online Min-Wait Relocation Problem to reflect practical observations on the different performance of online algorithms remains a goal for the future.

References

- [1] S. Ben David and A. Borodin. A new measure for the study of online algorithms. *Algorithmica*, 11(1):73–91, 1994.
- [2] A. Borodin and R. El-Yaniv. On Randomization in On-Line Computation. *Information and Computation*, 150(2):244–267, 1999.
- [3] J. Boyar, S. Irani, and K. S. Larsen. A comparison of performance measures for online algorithms. In *Algorithms and Data Structures*, volume 5664 of *LNCS*, pages 119–130. Springer, 2009.
- [4] D. Chemla, F. Meunier, and R. W. Calvo. Bike sharing system: Solving the static rebalancing problem. *Discrete Optimization*, pages 120–146, 2013.
- [5] C. Contardo, C. Morency, and L-M. Rousseau. Balancing a dynamic public bike-sharing system. Technical Report 9, CIRRELT, 2012.
- [6] M. EL-Zaher, B. Dafflon, F. Gechter, and J.-M. Contet. Vehicle platoon control with multi-configuration ability. *Proc. Computer Science*, 9(0):1503–1512, 2012.
- [7] A. Fiat and G. J. Woeginger, editors. *Online Algorithms: The State of the Art*, volume 1442 of *Lecture Notes in Computer Science*. Springer, 1998.
- [8] R. M. Karp. On-line algorithms versus off-line algorithms: How much is it worth to know the future? In *Proceedings of the IFIP 12th World Computer Congress on Algo., Softw., Arch. - IP '92*, volume 1, pages 416–429. North-Holland, 1992.
- [9] S. O. Krumke, A. Quilliot, A. K. Wagler, and J.-T. Wegener. Models and algorithms for carsharing systems and related problems. *Electronic Notes in Discrete Mathematics*, 44(0):201–206, 2013.
- [10] S. O. Krumke, A. Quilliot, A. K. Wagler, and J.-T. Wegener. Relocation in carsharing systems using flows in time-expanded networks. In *Experimental Algorithms*, volume 8504 of *LNCS*, pages 87–98. Springer, 2014.
- [11] J. Schuijbroek, R. Hampshire, and W.-J. van Hoes. Inventory rebalancing and vehicle routing in bike sharing systems. 2013. Working paper.