European Journal of Operational Research 251 (2016) 432–441

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

The lockmaster's problem☆

Ward Passchyn[a], Sofie Coene[a], Dirk Briskorn[c], Johann L. Hurink[d], Frits C. R. Spieksma[a,*], Greet Vanden Berghe[b]

[a] KU Leuven, Faculty of Economics and Business, ORSTAT, Naamsestraat 69, 3000 Leuven, Belgium
[b] KU Leuven, Department of Computer Science, CODeS & iMinds-ITEC, Gebroeders De Smetstraat 1, 9000 Gent, Belgium
[c] Chair of Production and Logistics, Bergische Universität Wuppertal, Rainer-Gruenter-Straße 21, 42119 Wuppertal, Germany
[d] University of Twente, Department of Applied Mathematics, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 29 September 2014
Accepted 1 December 2015
Available online 11 December 2015

Keywords:
Transportation
Lock scheduling
Batch scheduling
Dynamic programming
Complexity

ABSTRACT

Inland waterways form a natural network infrastructure with capacity for more traffic. Transportation by ship is widely promoted as it is a reliable, efficient and environmental friendly way of transport. Nevertheless, locks managing the water level on waterways and within harbors sometimes constitute bottlenecks for transportation over water. The lockmaster's problem concerns the optimal strategy for operating such a lock. In the lockmaster's problem we are given a lock, a set of upstream-bound ships and another set of ships traveling in the opposite direction. We are given the arrival times of the ships and a constant lockage time; the goal is to minimize total waiting time of the ships. In this paper, a dynamic programming algorithm is proposed that solves the lockmaster's problem in polynomial time. This algorithm can also be used to solve a single batching machine scheduling problem more efficiently than the current algorithms from the literature do. We extend the algorithm such that it can be applied in realistic settings, taking into account capacity, ship-dependent handling times, weights and water usage. In addition, we compare the performance of this new exact algorithm with the performance of some (straightforward) heuristics in a computational study.

1. Introduction

Transportation of goods by ship, over sea as well as over waterways, is a promising alternative for transport over land. Reasons are its reliability, its efficiency (a barge of 3000 tons can transport as much as 100 train wagons or 150 trucks), its relatively low operating cost and its environmental friendliness. Hence, the relative importance of this mode of transport is rising. Maritime transport for intercontinental trade receives increasing attention due to the high cargo volumes that can be shipped at a much lower energy cost compared to air cargo traffic. Inland waterways transport is therefore seen as a mode of transport that can make a significant contribution to sustainable mobility. For instance, in a recent report, the European Commission (2015) promotes a better use of inland waterways in order to relieve heavily congested transport corridors. Not only is the energy consumption of transport over water approximately 17 percent of that of road transport and 50 percent of rail transport, it also has a high degree of safety and its noise and gas emissions are modest. This natural network is the only existing infrastructure with excess capacity and where congestion is limited (Inland Navigation Europe, 2014). In the US, total waterborne commerce has risen from about 1500 million tons of goods in 1970 up to 2600 million tons in 2006; due to the economic crisis it has lowered since then to a level of about 2200 million tons in 2009 (U.S. Army Corps of Engineers, 2009). In a report prepared for the State of New York (Goodban Belt LLC, 2010), the potential of the New York Canal System for container-on-barge logistics is extensively described and an increased flow is expected after the Panama Canal expansion in 2015. The cargo volume transiting the Panama Canal is expected to grow on average 3 percent per year between 2005 and 2025, mainly due to an increase in container transport (Panama Canal Authority, 2006). In China, 88 million tons of freight passed the Three Gorges Dam in 2010; this is nearly 5 times the maximal annual volume reported before 2003 (ChinaDaily, 2011).

Many of these waterways (the Panama Canal, the Three Gorges Dam, inland waterways in Europe such as the Kiel canal (Luy, 2010), and many others) are accessible through sea locks and are often interrupted by river locks. Locks are needed to control the water level so that large and heavy ships can continue to access the corresponding waterways. At several waterway networks, congestion is expected to increase, yielding extra pressure on the locks. Examples are the New York State Canal System (Goodban Belt LLC, 2010) and

☆ An extended version of this paper is available as a research report, Passchyn et al. (2015)

* Corresponding author. Tel.: +3216326976; fax: +3216326732.
E-mail addresses: ward.passchyn@kuleuven.be (W. Passchyn), sofie.coene@kuleuven.be (S. Coene), briskorn@uni-wuppertal.de (D. Briskorn), j.l.hurink@utwente.nl (J.L. Hurink), frits.spieksma@kuleuven.be (F.C.R. Spieksma), greet.vandenberghe@cs.kuleuven.be (G. Vanden Berghe).

http://dx.doi.org/10.1016/j.ejor.2015.12.007
0377-2217/

the North Sea Canal region (van Haastert, 2003). Some locks are being expanded, such as locks on the Twente Canal in the Netherlands (Rijkswaterstaat, 2010). Also, new locks are being built, for example, the Deurganckdock lock at the harbor of Antwerp (Antwerp Port Authority, 2011), which will become the world's largest lock by volume.

These locks are bottlenecks for transportation over water, and hence, operating locks wisely contributes to the attractiveness of transportation over water and can help to avoid expensive infrastructural interventions. However, the algorithmic problem of how to operate a lock has not been studied broadly in the scientific literature. The purpose of this paper is to fill this gap. Our point of view here is to see the lock as an entity providing a service to the ships. Then, it makes sense to identify a strategy for the lock that optimizes some criterion related to service (as we do in this paper). Another point of view would be that the lock announces times when ships can enter the lock in order to be transferred, and that the ships simply need to respect these times. Clearly, even in the latter model, there is still a decision to be made concerning these times.

1.1. Problem definition

We now give a formal description of a very basic situation that will act as our core problem: the lockmaster's problem. We first study this simplified problem; the results obtained for this problem will serve as a basis for dealing with more realistic settings later on.

Consider a lock consisting of a single chamber. Ships that are traveling downstream arrive at the lock at given times. Other ships travelling upstream arrive at the lock, also at given times. Let $A = \{1, \ldots, n\}$ represent the set of all ships. Let $t(a)$ be the given arrival time of ship $a \in A$, and $p(a)$ the arrival position of ship $a \in A$, with $p(a) = 0$ for a downstream arrival, and $p(a) = 1$ for an upstream arrival. For convenience, we assume that a total order is imposed on A so that ships are ordered by non-decreasing arrival time, i.e. $t(i) \le t(i+1)$ for $i = 1, \ldots, n-1$. Note that multiple ships may have the same arrival time; the total order thus breaks these 'ties' arbitrarily. It is important to emphasize that we do not require ships to enter the lock in the order imposed on A, except in the cases (see Sections 5.1 and 5.3) where we explicitly state this as a requirement.

Let T denote the *lockage duration*: this is the time needed for the water level to rise from the downstream level to the upstream level (or vice versa), plus the time needed to load and unload the lock, which is assumed to be constant in the basic problem. In other words, T measures the time that elapses between opening the lock such that ships can enter, and closing the lock after ships have left. Throughout the paper, we assume $T > 0$ as the problem becomes trivial for $T = 0$. We further assume that all data are integral. Our goal is to find a feasible lock-strategy that minimizes total waiting time of all ships. The waiting time of a ship is the length of the period that elapses between the ship's arrival time and the moment in time when the ship enters the lock. Thus, we need to determine at which moments in time the water level in the lock should start to go up (meaning at which moments in time downstream ships enter the lock and are lifted), and at which moments in time the water level in the lock should start to go down. For such a strategy to be feasible, (i) going-up moments and going-down moments should alternate, and (ii) consecutive moments should be at least T time-units apart.

This particular problem (to which we refer as the lockmaster's problem) is a simplified version of reality. However, we see this problem as a basic problem underlying any practical lock scheduling problem; and we show how to solve this basic problem by dynamic programming (DP) in Section 3. Practical problems obviously feature many properties that are absent in this basic problem. In Section 5 we give an overview of many such features and investigate the complexity of these problem extensions.

The contributions of the present paper can be summarized as follows. We show that (1) there exists an $O(n^2)$ algorithm (see Section 4) for the lockmaster's problem; (2) this algorithm can be extended to solve variants with capacities, ship-dependent handling times, ship priorities, non-uniform lockage times, and settings with a limited number of lockages (Section 5).

In addition, we investigate the performance of several heuristics by running them on randomly generated instances that possess real-life characteristics (Section 6).

2. Literature

2.1. Lock scheduling

Lock scheduling has not been studied very thoroughly in the academic literature, although it has recently started to attract more attention. We mainly focus on the literature that considers scheduling a single lock. A relatively early work by Petersen and Taylor (1988) considers the Welland Canal in Canada. The authors present a dynamic programming algorithm for scheduling a single lock with unit capacity and extend this to obtain a heuristic result for the entire canal. A more recent paper (Nauss, 2008) deals with optimal sequencing in the presence of setup times and non-uniform processing times for the case where all arrival times are equal to zero. Smith, Sweeney, and Campbell (2009) simulate the impact of decision rules and infrastructure improvements on traffic congestion along the Upper Mississippi River. Ting and Schonfeld (2001) use heuristic methods to study several control alternatives in order to improve lock service quality. Verstichel and Vanden Berghe (2009) develop (meta)heuristics for a lock scheduling problem where a lock may consist of multiple parallel, capacitated chambers of different dimensions and lockage times, making this problem at least as hard as a bin packing problem. The question of filling a lock with ships, i.e. the packing problem, is discussed by Verstichel, De Causmaecker, Spieksma, and Vanden Berghe (2014a). Verstichel, De Causmaecker, Spieksma, and Vanden Berghe (2014b) propose a mixed integer programming model to solve a generalized lock scheduling problem for instances with a limited number of ships. Recent work by Hermans (2014) considers the optimization problem of scheduling a single lock with a capacity that allows a single ship. A polynomial time procedure asserts feasibility with respect to given deadlines in $O(n^4 \log n)$ time; it can further be used to minimize the maximum lateness.

In a related problem, ships need to pass a narrow canal, and only a restricted number of wider areas is available where ships can pass each other. Ships need to wait in these areas and are arranged in convoys that transit in a one-way direction, see e.g. (Griffiths, 1995; Günther, Lübbecke, & Möhring, 2011; Luy, 2010; Panama Canal Authority, 2006). For instances where all ships travel in the same direction, total waiting time is equal to zero, which is not necessarily the case in the lockmaster's problem.

2.2. Machine scheduling

Smith et al. (2011) relate traffic operations at a river lock with a variant of the job shop scheduling problem with sequence dependent setup times and a two-stage queuing process. The lockmaster's problem is more general since ships (i.e. jobs) have release dates and multiple ships can be locked together. The relation between the lock scheduling problem and classical machine scheduling is also observed in the literature. In fact, the lockmaster's problem introduced in this work is closely related to a batch scheduling problem. Batch scheduling involves a machine that can process multiple jobs simultaneously. Suppose that the basic lock scheduling problem only has downstream-bound ships (we will refer to this special case of the lockmaster's problem as the *uni-directional case*). The lock can be seen as a batching machine and the jobs are the arriving ships with release dates and equal processing times, and the flow time of a job is the waiting time of a ship. Following the notation of Baptiste (2000) this