Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 178 (2017) 206 – 212

Procedia Engineering

www.elsevier.com/locate/procedia

16th Conference on Reliability and Statistics in Transportation and Communication, RelStat'2016, 19-22 October, 2016, Riga, Latvia

Scheduling and Routing Algorithms for Rail Freight Transportation

Wojciech Bożejko, Radosław Grymin*, Jarosław Pempera

Wrcocław University of Science and Technology, Faculty of Electronics, Department of Control Systems and Mechatronics, Janiszewskiego 11/17, 50-372 Wrocław, Poland

Abstract

The paper deals with scheduling and routing rail freight transportation. There are provided mathematical descriptions of constraints in the real rail transportation such as timetables of passenger trains, safety time buffers etc. We developed an algorithm which determines the fastest route of cargo train in a railway network. It is based on Dijkstra algorithm idea. We experimentally proved that determination of the fastest route in even large railway network, where movement of a large number of trains was planned, takes place in a time acceptable for decision makers.

Keywords: algorithm, scheduling, optimization, rail transportation

1. Introduction

Fast-growing industry requires efficient delivery of a huge amount of stuff. Transportation time is often very short and it is not easy to meet a deadline. Transportation can be accomplished by the road transportation. Unfortunately, there is a high risk of traffic jams that generates losses. However, on some roads there are weight limits imposed on vehicles so it enforces selection of longer routes or use of greater number of trucks. All these factors make road transportation uneconomic and elusive. Competitive approach employs a rail transport. Proper planning may cause lack of congestion on the railway line, it allows to carry huge amount of stuff at once, allows to

* Corresponding author.
E-mail address: radoslaw.grymin@pwr.edu.pl

doi:10.1016/j.proeng.2017.01.098

shorten delivery time and it is more safe than road transport. Operational Research among other things deals with rail transport planning. Research focuses on formulating common problems and designing efficient methods that solve them.

The freight scheduling problem is one of the most difficult problems belonging to the family of transportation problems. Due to its complexity it poses a big challenge for modern Operational Research studies thus it is in a high demand. According to Pashchenko *et al.*(2015) it consists of three sub problems: train scheduling problem (choosing proper moment for train departure along its route), locomotive assignment problem (assigning locomotives to trains) and locomotive team assignment problem (assigning teams to locomotives in an optimal way). Planning train schedules and routes imposes execution of algorithms on quite big data sets. For instance current polish rail network consists of 999 railway lines (List of railway lines managed by PKP Polskie Linie Kolejowe S.A., 2016), 588 station buildings (PKP Polskie Koleje Państwowe S.A. – Our Stations, 2016) and more than 2500 motor engines (Camp of PKP Cargo – locomotives, 2016).

However, it can be difficult to find a feasible solution since chosen train paths may mutually exclude themselves. Trains cannot move one by one close to each other. A difficulty of finding good solution increases when it comes to rail networks with a high traffic density. Furthermore, there are many expectations related to the transportation regarding safety, speed, capacity and reliability. All mentioned factors force the use of sophisticated algorithms to solve scheduling and routing problems in a reasonable time.

State of the art

According to Cadarso *et al.* (2014) such planning problems are usually solved in two phases. First phase is called the macroscoping phase. In this phase an exact model of a rail network is not known. In this phase the problem is usually formulated as a multi-commodity network flow problem considering the scheduling of the train unit. Rolling stock assignment and train sequence problems are being solved. In the second phase, called microscoping phase, all operations are considered in details. At this level all conflict situations are detected and purged, compatibility issues are taken into account and time allowances for coupling and decoupling operations are studied.

Planning rail freight transportation takes into account activities associated with movement of trains and maintenance tasks. Lidén and Joborn (2016) were dealing with dimensioning maintenance time windows. They introduced a freight traffic cost model and a passenger traffic cost model for evaluating effects of maintenance windows on regional passenger traffic. There are also researches related to real-time algorithms intended for rapid responding on unexpected situations that can disturb the normal course of daily operations. Recent study of Samà *et al.* (2016) proposed an approach based on ant colony heuristic where the real-time train selection problem is described as an integer linear programming formulation. There are rare researches devoted to integration of production scheduling and rail transportation. In the study of Hajiaghaei - Keshteli and Aminnayeri (2014) such problem was solved using the Keshtel algorithm.

2. Problem formulation

A freight routing and scheduling problem through a rail network can be defined as follows. Let us assume that a railway network consisting of n of railway junctions from the set $\{1, \ldots, n\}$ is given. Railway junction may represent a railway station, a cross dock station, an intermodal terminal, etc. A railway network can be represented as a mesh of railway tracks, which is linked in junctions. A route and a schedule of t trains from the set $T = \{1, \ldots, t\}$ are given. The goal is to determine routes and schedules for additional cargo trains.

A railway network can be modeled using directed graph $G = (V, E)$, where $V$is the set of nodes and$E$ is the set of arcs. The node $i \in V$ corresponds to railway junction i , whereas arc $e = (i, j)$, $e \in E$ denotes unidirectional railway track from junction i to junction j (bidirectional tracks are modeled as a pair of arcs (i, j) and (j, i)).

For each train $k \in T$ a sequence of railway junctions $r_k = (r_k(1), \ldots, r_k(n_k))$ defining the route of the train is known. The schedule of train is determined by the departure time $d_k(s)$ from the railway junction $r_k(s)$, $s = 1, \ldots, n_k - 1$ and arrival time $a_k(s)$, $s = 2, \ldots, n_k$, to junction $r_k(s)$.

Without loss of generality, we assume that the speed of all trains is identical. The movement time on the track $e \in E$ is $m_e > 0$. Next, the safety lag between two adjacent trains moving on the same track is constant and it is equal to L units of time. We assume that the capacity of each junction is unlimited i.e. in every junction there is a railway siding with sufficiently high number of tracks.

We assign a timetable τ_e to the track $e = (i, j)$ of a railway network. The timetable τ_e contains departure times sorted in chronological order of all trains moved from junction i to junction j, precisely

$$\tau_e = \{d_k(s) : r_k(s-1) = i, r_k(s) = j, s = \{2,\ldots,n_k\}, k \in T\}. \tag{1}$$

The timetable $\tau_e$determine time windows $W_e = (w_e(1), \ldots, w_e(n_e))$, $n_e = |W_e|$ in which it is possible to start movement of an additional train on the track $e = (i, j)$. In the time window $w_e(s) = (e_e(s), l_e(s))$, the train earliest start is $e_e(s)$, whereas latest start is $l_e(s)$. We consider some element $\tau_e(s)$ of timetable τ_e. For safety reasons, additional train can start the movement only L after the moment $\tau_e(s)$, for the same reasons the start of the movement cannot start later than L units of time before the departure of the next train from timetable i.e. $\tau_e(s + 1)$. The pair $(\tau_e(s), \tau_e(s + 1))$ generates time window $(\tau_e(s) + L, \tau_e(s + 1) - L)$ (obviously if $\tau_e(s + 1) - L> \tau_e(s + 1) + L$).

3. Determining the fastest route for single train

In this section we describe an exact algorithm for determining the fastest route for single additional train. The problem is similar to the problem of finding shortest path from fixed start node to all other nodes in weighted graph, which is solved in polynomial time by Dijkstra's algorithm. However, due to significant differences between problems, we present a modification of Dijkstra algorithm adapted to a problem of finding fastest route.

Let us assume that a railway network described by directed graph $G = (V, E)$ is given. The arc $e \in E$ has weight equal to movement time m_e and time windows W_e are known. The train is ready to travel in release time R. The goal is to find the route (and/or schedule) from junction a to junction b which minimizes arrival time to junction b. Let $\varepsilon(i)$ be an estimation of minimal arrival time to junction$i$, $i \in V$, Q be a set of unvisited nodes from V, and $Adj(i)$ be a subset of nodes reachable from the node i. The pseudocode of algorithm is presented on the Figure 1. It is easy to see that the main steps of algorithm not differ to original Dijkstra algorithm. The fundamental difference occurs in updating of estimation $\varepsilon(v)$.

The updating of estimation is realized in two steps. In the first step, for the track $e = (u,v)$ the time window $(w_e(s^*) = (e_e(s^*), l_e(s^*)$ such as:

$$l_e(s^*) = \min\{l_e(s) : l_e(s) \geq \varepsilon(u), w_e \in W_e\} \tag{2}$$

In the second step $\varepsilon(v)$ is finally updated in following way

$$\varepsilon(v) = \min\{e(v), \max\{\varepsilon(u), e_e(s^*)\} + m_e\} \tag{3}$$

1. Set$Q = V \backslash \{a\}$, $\varepsilon(a) = R$, $\varepsilon(i) = \infty$, $i \neq a$, $i = 1, \ldots, n$
2. While Q in not empty do 2-4
3. $u \leftarrow \min_{q \in Q} \varepsilon(q)$
4. Set $Q = Q \backslash u$
5. Foreach $v \in Adj(u)$ do update $\varepsilon(v)$

Fig. 1. Pseudocode of algorithm.

4. Case study

It is well-known that railway transport is the most environment-friendly type of land transport. An amount of energy needed to carry people and goods per kilometer is much smaller than in case when a competitive car transport is used. Moreover rail transport relies on electrical energy that can be produced from renewable sources. Low energy consumption results also in lower costs of transportation.

Unfortunately, despite mentioned advantages, rail transport is not attractive for transport companies by virtue of low efficiency that eventuates, inter alia, from character of managing access to a rail infrastructure (railway line). Market requirements force transport companies to make quick decisions which modes of transport should be used to realize an order and fulfill expectations of customers.

In the current section we will prove that the algorithm introduced in the previous section can in a timely fashion determine a route and a schedule ordered by a transporter. The object of study is a segment of the Polish rail network consisting of 7 cities: Gdańsk, Kraków, Lublin, Toruń, Warszawa, Wrocław and Poznań. Figure 2 presents this rail network. A line connecting rail nodes denotes a real rail connection that does not pass any other city. Table 1 shows the real passenger train timetable. On the other hand, Table 2 presents estimated time of travel for cargo trains on network segments.

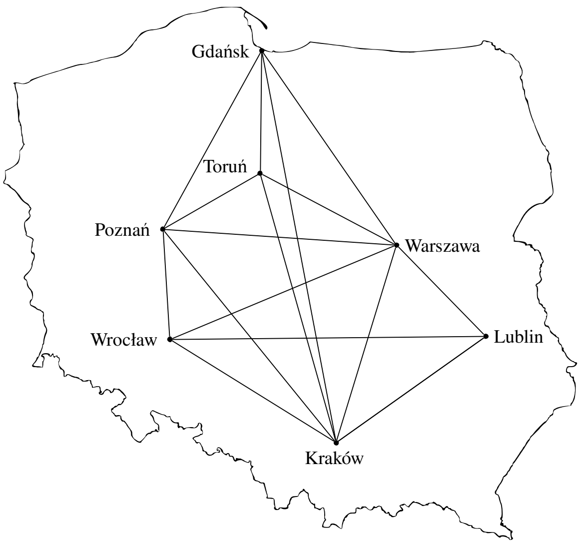

Fig. 2. Map of Poland with marked considered cities and direct connections.

A principal commissioned transit of certain number of wagons with commodity from Port of Gdańsk to Lublin. All shipping and logistic activities will be finished at 8:20 am. The aim is to determine route and schedule of transportation counting planned passenger traffic and a thirty minute safety buffer.

At the beginning we determine the shortest time of transportation from Port of Gdańsk to all cities, especially to the city pointed by the principal, that is Lublin. For this purpose we use Dijkstra algorithm. The results of the algorithm on data presented in Table 2 were collected in Table 3. In the second column we present total transportation time consists of transit times on all sections of the fastest route from fixed start node (Gdańsk) to the node mentioned in first column. In the third column we present the station which presedes the target station in fastes route. The shortest time of transportation from the Port of Gdańsk to Lublin is 11 h 43 min. The shortest route leads through rail node located in Warszawa.

Table 1. Time table.

Direction	Departure
Gdańsk-Kraków	23:32
Gdańsk-Toruń	5:33, 9:33, 13:34, 17:40, 21:50, 23:45
Gdańsk-Warszawa	7:46, 8:51, 9:50, 10:52, 11:14, 11:49, 12:52, 13:44, 13:52, 14:52, 15:50, 16:58, 17:07, 17:52, 17:59, 18:46, 18:52, 19:52
Gdańsk-Poznań	6:21, 7:32, 8:22, 12:21, 14:18, 16:23, 18:21, 21:57
Kraków-Gdańsk	21:25
Kraków-Lublin	6:00, 18:08
Kraków-Toruń	22:16
Kraków-Warszawa	7:44, 8:22, 9:46, 10:13, 11:46, 13:41, 14:14, 14:31, 15:16, 15:38, 16:44, 17:19,17:46, 19:07, 20:00, 20:07, 20:33
Kraków-Wrocław	6:18, 10:43, 11:15, 12:22, 14:54, 16:24, 18:11, 18:45, 22:25
Kraków-Poznań	7:03, 11:03, 13:13, 15:00, 21:10
Lublin-Gdańsk	5:38, 8:20, 12:22, 16:20, 20:23
Lublin-Kraków	6:39, 18:30, 00:28
Lublin- Warszawa	7:52, 9:44, 12:15, 13:52, 16:05, 18:07, 19:37, 20:23, 6:11, 09:12, 11:35, 13:13, 14:50, 17:23, 19:20,20:22
Toruń-Wrocław	12:25
Toruń-Poznań	7:32, 8:59, 11:38, 12:52, 15:28, 17:00, 19:00, 19:40
Warszawa-Gdańsk	7:16, 8:05, 08:20, 09:20, 09:30, 10:20, 11:05, 11:20, 12:20, 13:00, 13:20, 14:20,15:20, 16:20, 17:20, 18:00, 19:20, 20:20, 22:10
Warszawa-Kraków	7:50, 8:15, 9:03, 9:55, 10:30, 10:55, 11:55, 12:15, 13:00, 13:45, 14:50, 15:50, 16:17,16:55, 17:50, 18:45, 18:50, 19:50, 20:25, 21:10
Warszawa-Lublin	7:50, 9:50, 12:10, 13:35, 15:50, 11:30, 17:50, 19:50, 20:50
Warszawa-Toruń	6:30, 08:30, 10:25, 12:25, 14:15, 16:25, 18:27, 20:30, 22:00
Warszawa-Wrocław	7:45, 08:07, 09:35, 11:45, 12:05, 13:50, 16:13, 17:30, 19:25
Warszawa-Poznań	7:25, 10:00, 11:30, 13:05, 14:00, 14:35, 14:39, 15:00, 16:01, 17:00, 18:00, 19:00,22:35, 23:23
Wrocław-Kraków	6:13, 8:11, 10:50, 13:09, 14:53, 16:00, 17:27, 18:32
Wrocław-Lublin	9:23, 12:12
Wrocław-Warszawa	6:50, 8:25, 10:30, 10:45, 13:03, 14:47, 15:00, 17:20, 18:27
Wrocław-Poznań	7:29, 8:17, 08:55, 09:35, 10:21, 10:49, 11:05, 12:24, 13:27, 14:32, 15:54, 16:23, 16:42, 17:37, 18:47, 20:05, 21:31, 23:35, 23:55
Poznań-Gdańsk	6:34, 8:47, 10:50, 12:45, 14:41, 17:30, 18:47
Poznań-Kraków	7:38, 9:43, 11:40, 15:35, 17:36
Poznań-Toruń	6:46, 9:51, 10:55, 14:46, 14:51
Poznań-Warszawa	7:44, 8:41, 9:33, 9:40, 10:40, 10:45, 12:40, 14:29, 15:40, 17:40, 18:42, 19:30, 20:40
Poznań-Wrocław	5:53, 10:33, 14:32, 22:21

Table 2. Transport times.

	Gdańsk	Kraków	Lublin	Toruń	Warszawa	Wrocław	Poznań
Gdańsk		8:52		2:42	4:03		3:43
Kraków	9:00		4:13	9:09	4:21	5:05	7:02
Lublin		4:09			2:56	7:27	
Toruń	2:59	6:53			2:53		2:23
Warszawa	5:55	5:05	7:40	2:56		5:21	3:41
Wrocław		4:58	7:27		5:30		3:30
Poznań	3:42	6:52		2:25	3:55	3:20	

Table 3. Minimal transportation time.

Target station	Total transportation time	Previous station
Kraków	08:52	Gdańsk
Lublin	11:43	Warszawa
Toruń	02:42	Gdańsk
Warszawa	04:03	Gdańsk
Wrocław	07:03	Poznań
Poznań	03:43	Gdańsk

Proposed in previous section algorithm determines a route and schedule for cargo train from the chosen initial node to all other nodes. We will demonstrate execution of algorithm for two times in which a freight train will be ready to transportation: R = 8 and R = 8:30. The results of the algorithm were collected in two tables 4 and 5. For each target station we show arrival time, station which presides in fastest route, the departure time from this station and total transportation time. The total transportation time is the difference between the time of arrival to target station and the time of cargo train ready to go. It consists with transits times and downtimes in rail nodes. It is a measure of occupancy of the railway infrastructure.

Table 4. Train route and schedule for$R$ = 8:30.

Target station	Arrival time	Total time	Previous station	Departure time
Kraków	17:12	08:52	Gdańsk	08:20
Lublin	*20:20*	*12:00*	Warszawa	*12:40*
Toruń	11:02	02:42	Gdańsk	08:20
Warszawa	12:23	04:03	Gdańsk	08:20
Wrocław	*15:55*	*07:35*	Poznań	*12:35*
Poznań	*12:35*	*04:15*	Gdańsk	*08:52*

Table 5. Train route and schedule for$R$ = 8:30.

Target station	Arrival time	Total time	Previous station	Departure time
Kraków	17:22	08:52	Gdańsk	08:30
Lublin	*21:35*	*13:05*	*Kraków*	17:22
Toruń	11:12	02:42	Gdańsk	08:30
Warszawa	*14:23*	*05:53*	Gdańsk	*10:20*
Wrocław	15:55	07:25	Poznań	12:35
Poznań	12:35	04:05	Gdańsk	08:52

It can be easily noticed that in case of cities: Kraków, Toruń and Warszawa total time of transportation (see 3rd column) is minimal. It means that transport to these cities is curried out on the shortest way without any delay.

In case of Poznań departure from Gdańsk was delayed until 8:52 am because of conflict with passenger train departing from Gdańsk toward Poznań. Delayed arrival to Wrocław is caused by delayed arrival to Poznań. Finally, train riding from Lublin must wait in Warsaw from 12:23 am until 12:40 am because of conflict with passenger train departure at 12:10 am in the same direction.

For$R$ = 8:30 algorithm determined a new route to Lublin through Kraków. The new route guarantees earlier arrival to a target station then through Warszawa, because route Gdańsk-Warszawa due to safety time buffers is not available since 8:21 am (departure 8:51) until 10:20 am (departure 9:50 am).

5. Conclusion

We developed an algorithm determining the fastest route for single additional train assuming that two trains cannot move from the same station in the same direction one by one – special safety time buffer must be assumed. In our experiments we obtained valuable solutions in a sensible time.

Usually the problem is to find several fast routes for trains. Each route is described by initial and target node. Our algorithm can be used to develop an algorithm solving such problem. In such algorithm we add pairs (initial and target node) to the priority queue. In each step we take off the first element from the queue and determine the fastest route for one additional train using our algorithm. We update the rail network and continue with taking off elements, determining the fastest route and updating the rail network till the priority queue is empty. Solution consists of such problem consists of routes found by our algorithm and can be assessed by a fitness function that determines quality of the solution. Obtained solution (and its quality) depends on order of elements in the priority queue. We can check all possible solutions and provide an exhaustive search or involve metaheuristics, see Bożejko *et al.*(2016), such as tabu search or simulated annealing, also in parallel versions.

References

Bożejko W., Uchroński M. and Wodecki M. (2016) Parallel metaheuristics for the cyclic flow shop scheduling problem. Computers & Industrial Engineering 95, 156–163.

Cadarso L., Marín Á., Espinosa-Aranda J. L. and García-Ródenas R. (2014) Train scheduling in high speed railways: Considering competitive effects. Procedia-Social and Behavioral Sciences 162, 51–60.

Camp of PKP Cargo – locomotives. URL http://inforail.pl/tabor-pkp-cargo-lokomotywy_more_48391.html.

Hajiaghaei-Keshteli M. and Aminnayeri M. (2014) Solving the integrated scheduling of production and rail transportation problem by Keshtel algorithm, Applied Soft Computing 25, 184–203.

Lidén T. and Joborn M. (2016) Dimensioning windows for railway infrastructure maintenance: Cost efficiency versus traffic impact. Journal of Rail Transport Planning & Management, 6(1), 32–47.

Pashchenko F., Kuznetsov N., Ryabykh N., Minashina I., Zakharova E. and Tsvetkova O. (2015) Implementation of train scheduling system in rail transport using assignment problem solution. Procedia Computer Science, 63, pp. 154–158.

Samà M., Pellegrini P., D'Ariano A., Rodriguez J. and Pacciarelli D. (2016) Ant colony optimization for the real-time train routing selection problem. Transportation Research Part B: Methodological, 85, 89–108.

PKP Polskie Koleje Państwowe S.A., Our stations, Web page, access 02.07.2016. URL http://pkpsa.pl/pkpsa/nasze-dworce/.

List of railway lines managed by PKP Polskie Linie Kolejowe S.A. , Web page, access 02.07.2016. URL http://www.plk-sa.pl/files/public/user_upload/pdf/Reg_przydzielania_tras/Regulamin_2013_2014/zmiana_7_02_2014/P_13-14_Z_1_20140207122552.pdf.