This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier’s archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/copyright
Invited Review

Shunting yard operations: Theoretical aspects and applications

Nils Boysen a, Malte Fliedner c, Florian Jaehn b, Erwin Pesch b,*

a Lehrstuhl für Allgemeine Betriebswirtschaftslehre/Operations Management, Friedrich-Schiller-Universität Jena, Carl-Zeiß-Straße 3, D-07743 Jena, Germany
b Institut für Wirtschaftsinformatik, Universität Siegen, Hölterlinstraße 3, D-57068 Siegen, Germany
c Fachgebiet Operations Research/Management Science Research, Technische Universität Darmstadt, Hochschulstraße 1, D-64289 Darmstadt, Germany

ARTICLE INFO

Article history:
Received 29 March 2011
Accepted 20 January 2012
Available online 28 January 2012

Keywords:
Shunting yards
Classification yards
Container transshipment

ABSTRACT

Almost ever since freight has been transported via rail, shunting yards (also called classification or marshaling yards) are operated in order to separate freight trains and reassemble new trains. The efficient use of shunting yards has a deep impact on the efficiency and reliability of rail freight services. Thus, much research on shunting yards has been published, starting from the 1950s. Lately, several publications mostly focusing on the sorting procedures have livened up research on shunting yards. This paper reviews the literature on the operational processes at shunting yards over the last 40 years and discusses the operational challenges of freight transshipment. The approaches are classified according to different sorting strategies which allows an easy access to the models for both, researchers and practitioners. The paper concludes with an overview on future research challenges.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Shunting yards, also called marshaling or classification yards, play an important role as nodes in rail freight transport networks. Here, inbound trains are disassembled and the railcars are then assembled such that desired outbound trains are generated. By this procedure, railcars can be sent through the network so that every origin–destination combination can be served without providing a huge number of point-to-point connections. However, these classification procedures are rather time consuming. Bontekoning and Priemus (2004) state that in Europe, shunting operations may take 10–50% of trains’ total transit time. These numbers manifest that time savings on shunting operations can lead to significant time and cost savings on the whole rail network. This is surely a reason, why plenty research papers have been written on how to use shunting yards efficiently.

The important planning and control processes to be performed at a shunting yard are mostly at the operational level. While the exact process implementation depends to some extend on the specific yard characteristics, especially the core operations of sorting and rearranging railcars via a system of tracks and switches can often be sufficiently generalized to include a wide variety of different shunting yards. Due to their critical importance for yard efficiency, we focus on these operations in the following. In particular, we discuss the applicability of theoretical insights with respect to different yard characteristics and thereby aim to illuminate the influence of practical requirements on the structure of decision problems.

To the best of our knowledge, there is no survey paper considering the entire operations of a shunting yard. On the one hand, there are a few recent surveys (e.g. Di Stefano et al., 2007; Hansmann and Zimmermann, 2008; Gatto et al., 2009) that treat special problems arising at shunting yards. These papers are part of our review and they are summarized below. On the other hand, there are two older surveys, which consider more general rail operations and cover shunting yards as a side issue only. The literature review by Assad (1980b) dedicates a section to yard and terminal models and classifies papers into three groups. Firstly, papers considering queuing models are presented. In these models, the cars have to visit several service facilities (reception and inspection, disassembling, connection, etc.) and average queuing time, average length of the queue, etc. are to be determined. Secondly, papers presenting yard simulation models (mostly from the 1960s) are surveyed. Finally, papers are reviewed, which try to estimate some kind of yard production function in terms of a yard performance in dependency on the amount of incoming traffic. The yard performance can be measured in processing time or monetary costs. Cordeau et al. (1998) present a literature review on various optimization problems in train logistics. Although they focus on tactical problems such as routing, blocking, and scheduling, they briefly address problems arising at transshipment yards such as the disassembling and reassembling of cars, and the prediction of total connection times of cars handled in the yard.

The main intention of this work is to provide a comprehensive review of core shunting processes and contrast theoretical
achievements of the field to practical requirements and applications. The remainder of this paper is organized as follows. Section 2 precisely defines the scope of this review. Afterwards, Sections 3–5 structure and review the research in this field with respect to different sorting strategies. An outlook on future research directions concludes the paper in Section 6.

2. Scope and structure of review

This paper investigates the operational processes that are executed at freight shunting yards and discusses quantitative approaches developed to improve shunting operations. In order to line out the scope we will start out with a characterization of different yard types. Afterwards, we describe the hierarchical planning process connected to shunting yards and define the (operational) tasks considered in this review.

2.1. Yard and terminal types

Shunting yards, as we consider them in this paper, are characterized by the disassembling and reassembling procedures of trains via a system of tracks and switches. In contrast to other yard or terminal types, loads itself are not moved from one train to another, but the railcar carrying the load can be assigned to a different train. We furthermore restrict ourselves to shunting yards, which are employed in freight transport.

Besides the traditional shunting yards, several types of rail yards or rail terminals exist. Boysen and Fleider (2010) classify rail yards into three generations. Shunting yards are considered as first generation yards. Furthermore, rail-road terminals (second generation) and fully automated rail-rail transshipment yards (third generation) are mentioned. Both, second and third generation yards focus on standardized loading units and thus, they cannot completely replace traditional shunting yards.

Rail-road terminals are mostly, but not exclusively, applied to implement a point-to-point rail-network. They take advantage of intermodal transport. The pre-haulage and end-haulage is performed by trucks, transporting goods to and from the terminals. The main-haulage is then provided by rail. The terminals are constructed in such a way that one or more gantry cranes span over several rail tracks, truck lanes, and sometimes storage lanes. The cranes move the standardized loading units so that the trains themselves are not decomposed. As decomposing of trains is an essential attribute to be considered in this paper, we omit considering rail-road terminals. Literature on these terminals includes several simulation approaches for layout planning (see e.g. Ferreira and Sigut, 1993; Abacoumkin and Ballis, 2004; Ballis and Golas, 2003; Kozan, 2006; Vis, 2006; Benna and Gronalt, 2008), treats several operational problems (e.g. Lai et al., 2008, consider aerodynamic effects of loading units, Brun and Knust, in press, assign load units to the railcars, Montemanni et al., 2009, sequence crane moves, Boysen and Fleider, 2010, determine fixed crane areas), and presents holistic approaches (see e.g. Froyland et al., 2008). For surveys, see Macharis and Bontekoning (2004) and Bontekoning et al. (2004).

Modern rail-rail transshipment yards also use gantry cranes for moving loading units. However, they are designed to quickly exchange loading units from one train to another so that a hub-and-spoke system can be applied even if all trains intermittently stopping at the terminal have to travel the whole distance from origin to destination within one night. In order to ensure such a quick transshipment, the terminals are usually equipped with an automated ground sorting system, which quickly moves loading units lengthwise. This allows the cranes to work more efficiently. Only few of these hub yards have yet been built (e.g. Port-Bou, see Martinez et al., 2004; González et al., 2008) or will be built (e.g. Hannover-Lehrte, see Aliche, 2002; Rotter, 2004). As trains are not decomposed at these yards, we omit a detailed description of the literature. The interested reader may for instance be referred to Meyer (1999), Wiegmans et al. (2007), Boysen et al. (2011a), Boysen et al. (in press) and Boysen et al. (2011).

Shunting yards are not necessarily only connected to freight trains. In some cases, passenger trains (Freling et al., 2005; Kroon et al., 2008) have to be shunted. Certainly, the tasks to be performed when shunting passenger trains are similar to the operations of a freight train shunting yard. However, there are differences. The number of trains and especially the number of railcars to be shunted is usually much higher if freight trains are considered. Furthermore, single units of passenger trains might be self-propelled, as it is the case in the above mentioned papers. Although it was hard to draw a clear line, we decided to only include papers, which have at least some relevance to shunting yards dedicated for freight trains. Thus, similar problems of tramway traffic (Blasum et al., 1999; Winter and Zimmermann, 2000; Eggermont et al., 2009) are not included in this survey.

Finally, rail terminals specialized for the transshipment of cars (see Mattfeld and Kopfer, 2003; Fischer and Gehring, 2005) and similar problems arising at seaport terminals (see e.g. Vis and de Koster, 2003; Kim et al., 2000; Kim and Park, 2004; Bierwirth and Meisel, 2010) are not considered.

All shunting yards covered in this paper can be classified into one of the following three yard types (see Petersen, 1977a):

- hump yards,
- flat yards, and
- gravity yards.

Each of these yard types may have very different characteristics (for a detailed description of different yards, see Petersen, 1977a). The layout of a typical hump yard is shown in Fig. 1.

Railcars are brought to the yard by inbound trains and hence pass through the yard from left to right where they are fetched by outbound trains. The yard compound can be generally subdivided into three main areas, which all consist of a set of parallel tracks: the receiving area where inbound railcars arrive, the classification area where railcars are rearranged and the departure area where sorted railcars wait until they are pulled out of the yard. The capacity of the respected areas is mainly determined by the number of parallel tracks available.

On the yard railcars are mainly moved by specific shunting engines and therefore bind yard resources during their stay. Hump yards seek to minimize resource demand by making use of the gravitational pull after the hump between receiving and classification area, so that railcars can be rearranged without relying on shunting engines. The classification area is often connected to the receiving area by an additional return track, which allows for a repeated classification of railcars.

Flat yards have almost the same layout as hump yards. The only difference is that there is no hump so that gravity shunting is replaced by switch engines. At gravity yards, the whole yard has a slight decline in order to ease switching. However, they require more staff for setting break shoes and, thus, they are less common in regions with comparatively high wage costs. If not indicated differently, we will consider hump yards. The typical number of classification tracks ranges between 20 and 40 tracks, whereas the largest single-sided shunting yards in North-America have up to 70 classification tracks. Note that the (hump) yards considered in the following might differ from the typical layout presented in Fig. 1, e.g. at some yards the classification tracks are only accessible from one side. If not clear from the context, we will describe the according yard layouts.
Most tasks to be performed at a yard have to follow several restrictions and several quantitative and qualitative goals are to be achieved. Depending on the specific shunting policies applied, the number and capacity of available tracks, and on the required composition of outbound trains, a large number of potential shunting problems exist. Furthermore, the decisions made may influence other decisions. Uncertainty in the input data (especially the train arrivals) complicates the finding of optimal or even good solutions. To ease decision making, some subproblems are usually solved separately. Several of these subproblems are not relevant for optimization as they have (fairly) fixed processing times (e.g., the inspections, the uncoupling and the assembly) or because they just have little relevance (e.g., the assignment of inbound trains to the receiving tracks). The problems, which are relevant for the efficiency of a yard will be in our focus.

2.2. Overview on decision problems

In the following, we describe the typical hierarchical planning process at shunting yards. On the operational level, the shunting process is described for hump yards only. This seems justified since hump yards are most widely used in practical applications and the details of the planning process can be quite readily transferred to other yard types as well.

2.2.1. Strategic tasks

Most strategic tasks to be performed in rail traffic do not or only partially concern shunting yards in itself. Decisions on service aspects, research and innovation, legislation and lobbying, and marketing are barely connected to shunting yards. Contrary if it comes to investments and modifications of the infrastructure. Shunting yards, (terminal) stations, and junctions can be seen as nodes of the rail network, which are connected by edges, i.e., by main and secondary rail lines. We distinguish between superordinate strategic decisions, which are on shunting yards, and subordinate strategic decision, which are taken at shunting yards. The former are rather connected to network design and they should give answer to the following questions:

- Where should a new shunting yard be built?
- Should an existing yard be rebuilt, renovated, enlarged (e.g., by new tracks, a new hump, etc.), reduced or even dismantled?
- Which (main) routes should be served using a certain yard?
- Should there be investments on (high-cost) equipment at a yard, e.g., a new switch control?

Certainly, there are many more superordinate decisions affecting shunting yards, but most of them can be reduced to the above questions. E.g., new standards on signaling, electrification or even the gauge of the tracks could lead to a rebuilding or a dismantling of a yard.

Subordinate strategic decisions contain decisions on the local infrastructure that only obliquely influence the rail network. For a newly built shunting yard, this implies decisions on the following attributes.

- The layout of a yard and its link(s) to the rail network.
- The number of humps.
- The number of parallel tracks and their length.
- The arrangement of switches.
- The type and implementation of the switch control.
- Service facilities (e.g., operational buildings, maintenance facilities, etc.).

Subordinate strategic decisions for an existing shunting yard include.

- single track abandonment or recovery,
- investments on medium-cost equipment such as switch engines,
- major maintenance activities.

Only few papers on shunting yards can be found in literature, which support strategic decisions. Most of them date back to the 1970s and 1980s and are beyond the scope of this review. They try to estimate the yard capacities and can therefore be used to support decisions on the number of tracks and their layout (see Petersen, 1977a; Petersen, 1977b; Turnquist and Daskin, 1982; Martland, 1982; Marinov and Viegas, 2009). A model on superordinate strategic planning in rail freight transportation was proposed by Crainic et al. (1990).

2.2.2. Tactical tasks

Once the strategic decisions regarding the infrastructure and capacity of the yard have been taken, several tactical planning tasks need to be solved on a shorter planning horizon of several months or weeks. Unlike strategic tasks, tactical decisions typically affect organizational processes and shunting policies and can therefore be more easily reversed. Similarly to the strategic tasks, we distinguish between superordinate tactics that have a direct effect on network planning and subordinate tasks, which only affect the organization of the yard itself. The most important subordinate tactical tasks comprise.

- the generation of the train schedule, defining train routes and their frequency as well as time-tables,
- the routing of loaded railcars through the network,
- railcar priority rules,
- the blocking, i.e., the grouping of railcars to blocks, which then travel together through different stages of the network,
- the train makeup, i.e., an assignment of railcars (for which a route is known) to trains,
- empty railcar distribution within the network,
- the allocation of classification work among shunting yards,
- yard dispatching policies.

Several of these problems have been studied in literature, e.g., there are many works on makeup and blocking (see e.g. Assad, 1980a; Bodin et al., 1980; Newton et al., 1998; Ahuja et al., 2007; Riezebos and van Wezel, 2009) or see the survey on empty railcar distribution by Dejax and Crainic (1987). Other papers dealing with
several tactical problems in rail traffic networks have been presented by Crainic et al. (1984) and Jovanović and Harker (1991).

Subordinate tactical tasks at shunting yards play a minor role in the literature. This may be due to the fact that many subordinate tasks are not specific to shunting yards (e.g. personnel planning, maintenance schedules) and further that their effectiveness can barely be evaluated separately from the operational implementation. We are not aware of any paper that primarily treats subordinate tasks at shunting yards. However, the results on operational tasks of the papers covered in this review can be certainly used to improve tactical decision making.

2.2.3. Operational tasks

In the following, we will describe the operational shunting processes in detail, assuming that decisions on strategic and tactical tasks have been taken. Although the scope of this review comprises all operational tasks, many tasks are not treated in the literature (mostly due to minor relevance).

An inbound train that arrives at the shunting yard is usually directly forwarded to one of the receiving tracks, where the road crew of the train is replaced by the yard crew. If no track of sufficient length is free, the train is temporarily parked in close vicinity to the receiving area and moved as soon as sufficient space is available. Depending on the shunting policy, railcars of several different inbound trains might very well be directed to the same receiving track. The set of railcars that enter one specific receiving track is called a “cut”. Especially if the number of receiving tracks is smaller than the number of inbound trains, a cut will most likely consist of a convoy of inbound trains. Although the assignment of cuts to receiving tracks is of minor relevance and can be done arbitrarily, the assembly of cuts, i.e. the possible bundling of inbound trains, has to be done carefully, as it influences the sorting of railcars.

The railcars of a cut are first inspected by the yard staff for mechanical defects and are then prepared for decoupling. At this stage the switch engine of the train is typically removed, along with some special railcars (e.g. for refrigeration) which are separately handled. Next, the hump sequence of the cuts has to be determined and afterwards, the cuts are successively humped. Once a cut is ready for shunting, a hump engine pulls the railcars off the hump, where they are finally decoupled so that they can roll freely downhill. A railcar is redirected towards its designated classification track via a system of switches where it is slowed down by automatic retarders or by controlled collisions with cars already on track until it comes to a halt. One of the most important and most studied operational tasks at a shunting yard is the assignment of railcars to classification tracks, i.e. the core of a sorting procedure.

If the desired sequence of outbound railcars could not be achieved after all cuts were humped, the humping process is repeated for all or a subset of railcars. For this purpose some classification tracks typically have access to one or more special return tracks, which lead back to the receiving tracks. At some yards, railcars can be arbitrarily pulled from any classification track back to the receiving tracks for rehumping (a so called “track pull”). The pulled railcars are reassigned to new cuts on the receiving tracks and then humped again as described. This procedure is repeated until the desired order of railcars on the classification tracks is obtained. From the classification tracks, railcars are pulled to the departure tracks by switch engines. In this context, the schedule of switch engines within a shunting yard is an important operational task. On the departure tracks, the railcars are finally assembled to outbound trains and a road locomotive is attached to each outbound train. Each railcar is inspected a second time (e.g. concerning break function), the road crew replaces the yard crew and finally, each outbound train is dispatched.

Since the most important part of the operational planning at shunting yards is the sorting and rearranging of railcars, most papers presented in this survey deal with this problem and numerous variants. Although the sorting and rearranging is the key activity at shunting yards, it need not be the bottleneck, which is underlined by some other papers presented in this review.

2.3. Structure of review

We focus on quantitative planning and control approaches that aim at directly improving the operational shunting process outlined above. The vast majority of papers analyzes the performance of specific sorting schemes and is reviewed first. The second part of the review focuses on papers which provide a more comprehensive overview on shunting operations.

We start our review with papers on single-stage sorting. In this context we distinguish between strict single-stage sorting and extended single-stage sorting. In the former, a railcar only moves in forward direction through the yard, i.e. railcars must not be humped more than once. In contrast to that, extended single-stage sorting allows some track pulls for rehumping albeit only in exceptional cases.

Whenever rehumping is an integral part of the shunting policy, we refer to this strategy as multi-stage sorting. We dedicate a section on papers dealing with this sorting scheme and further distinguish three multi-stage approaches. Firstly, we review the most basic and commonly applied sorting strategies, which are sorting by train, sorting by block, triangular sorting and geometric sorting. Secondly, new and promising ideas for multi-stage sorting in deterministic scenarios are presented. Thirdly, we review robust sorting strategies that allow an easy recovery in case of certain disturbances. Among the more general papers in Section 5, we review two recent classifications of sorting problems and present their according notations. Furthermore, we discuss some papers that treat shunting yard operations as part of holistic planning approaches.

Each paragraph reviewing a new paper is highlighted by a respective underlined citation.

3. Single-stage sorting

The main challenge of single-stage sorting consists in determining the hump sequence of cuts and the assignment of blocks to classification tracks. This has to be done in such a way that all blocks of an outbound train arrive on the same track. A standard assumption in this context is that the order of blocks in each outbound train can be chosen arbitrarily, since otherwise single-stage sorting is hardly applicable (an exception is the paper by Di Stefano and Koči, 2004). Common objectives are the minimization of the throughput time (i.e. the dwell time of railcars in the yard) and the minimization of the number of classification tracks used.

Most shunting yards in America are designed for single-stage sorting. This implies that these yards usually have a high number of classification tracks which are comparably short as they do not need to hold many railcars. However, even if there are many more classification tracks than outbound blocks and block assignment is arbitrary, single stage sorting cannot guarantee to generate the desired outbound sequence in one attempt. That is why in practice rehumping occurs at least occasionally at almost all yards (extended single-stage sorting). Strict single-stage sorting certainly bears interesting theoretic challenges, which are the basis for extended single-stage sorting.

3.1. Strict single-stage sorting

The paper written by Yagar et al. (1983) is concerned with finding a hump sequence, i.e. an order in which complete inbound trains are humped. They consider a static sequencing model with
single-stage sorting. The sequence determines the time, each railcar is available for departure. If a railcar misses its originally intended outbound train (in other words: if the according connection is missed), the railcar leaves the yard with a later outbound train and its dwell time in the yard increases. The objective is to minimize “yard throughput costs”, which are mostly influenced by the time each car spends in the yard, adjusted by priority values for certain cars. For a small number of inbound trains, all possible sequences can be evaluated using dynamic programming. If this number is large, a ranking procedure based on load factors and priority values is applied in order to determine a set of trains that are suitable for being humped next. This set of trains is then sequenced according to the yard throughput costs. This approach has been tested using real life data from CN Taschereau Rail Yard in Montreal. The solutions obtained by their method have been compared to a FIFO-strategy and to the sequences, which were actually applied at the yard.

A specific single-stage sorting problem has been presented by Dahlhaus et al. (2000). Their problem called “Train marshaling problem” can be described as follows. Given a single train with \( n \) railcars and a disjoint partition \( S_1, \ldots, S_n \) \((\leq n)\) of the set of railcars. Each set of the partition corresponds to a specific outbound destination. In general, we have to assume that the railcars of a set of the partition are not located next to each other, but they are spread out within the train. The train is now humped and the railcars are sorted to classification tracks. Afterwards, the railcars are to be assembled on one departure track in such a way that for each set of the partition, all cars are located successively. The question is, how many classification tracks are needed in order to achieve such a reassembly. Of course, \( n \) is a simple upper bound, which can be achieved by assigning each railcar to a different track. The authors show that this problem corresponds to an optimization problem, which can be described as follows. Given a sequence \( 1, \ldots, n, 1, \ldots \) in which the interval \( 1, \ldots, n \) is repeated \( k \) times. Which is the minimum value for \( k \) such that there exists a permutation \( p(1), \ldots, p(t) \) of \( \{1, \ldots, t\} \) and such that the sequence contains all elements of \( S_{p(1)} \), then all elements of \( S_{p(k)} \), etc.? The equivalence of both problems can be illustrated by an example provided by Dahlhaus et al. (2000):

**Example 1.** For \( n = 11 \) railcars numbered according to their position in the inbound train, \( t = 5 \), and a partition

\[
S = \{S_1, S_2, S_3, S_4, S_5\} = \{(1, 6, 11), (2, 7), (3, 8), (4, 9), (5, 10)\}
\]

we obtain \( k = 4 \) as a minimum value of \( k \). This value can be achieved using the identity permutation. Note that the permutation delivers the sequence of the sets of the partition after assembly. Each repetition of the interval \( 1, \ldots, n \) corresponds to one track.

![Train marshaling problem example](image)

The authors show that this problem is NP-complete. They furthermore provide a tight upper bound for the case that no information about the partition is available. Thus, if only the number of railcars is known, the required number of classification tracks is \([2 + \frac{1}{2}]\).

Di Stefano and Kočić (2004) consider single-stage shunting problems, which origin from the parking of (passenger) trains in a depot at night. However, these problems closely relate to the assignment of (freight) railcars to classification tracks for a given hump sequence. If the passenger trains considered by Di Stefano and Kočić (2004) are regarded as blocks of railcars, the relation of the problems becomes obvious. For a given sequence of \( n \) inbound trains, the assignment of the trains to classification tracks needs to be determined in such a way that a given outbound sequence can be followed without extra switching. Using as few tracks as possible is the only objective. Further restrictions, e.g. track length requirements, are not considered. Four different scenarios are analyzed. Firstly, all trains can only enter each classification track from one side and they can only leave the tracks from the other side (diso = double in, single out). This problem corresponds to the chromatic number of a permutation graph, which leads to an \( O(n \log n) \) algorithm. Secondly and thirdly, trains can either enter tracks from both sides and leave from one side only (dio = double in, single out), or they can enter from one side and leave from both sides (dio = single in, double out). These two problems are equivalent and again, there is a corresponding graph coloring problem, which is to find a coloring of particular 3-uniform hypergraphs. Finding a coloring of general 3-uniform hypergraphs is known to be NP-hard, and in a later paper (Di Stefano and Zimmermann, 2005) it has been shown that the special case is NP-hard and there is a 3.42 approximation algorithm. Di Stefano and Kočić (2004), not being aware of the complexity yet, developed a heuristic, which uses at most \( \sqrt{4n-1} \) tracks and runs in \( O(n^{2.5}) \). Finally, the situation is examined in which trains can enter and leave tracks from both sides (dio = double in, double out). This corresponds to finding a coloring of particular 4-uniform hypergraphs. The problem is NP-hard in general, and Di Stefano and Zimmermann (2005) proved that the special case is also NP-hard. There is no algorithm presented for that problem.

In order to reduce the dwell time of railcars in a yard, Bektaş et al. (2009) make use of the fact that two empty railcars of the same type, assigned to different outbound trains, can be reassigned to the respective other train (switch). This might positively affect the general throughput time, as a delayed empty railcar can lead to a 24 hour delay of its outbound train. The procedure can be described as follows. The hump sequence of a single-stage sorting is assumed to be given. Whenever a cut is humped, only switches are analyzed, in which one of the exchanged railcars appears in the cut to be humped next. All beneficial switches (i.e. switches reducing the total dwell time) of empty railcars from this cut with railcars to be humped later are determined. As there might be several beneficial switches involving the same car, a maximum weighted matching problem results, which can be solved exactly using the Hungarian Method. The according switches are then performed. This method has been applied to test data from a one-year period of MacMillan railyard. An average time saving of 4.98% has been reported.

3.2. Extended single-stage sorting

Kraft (2000) presents a deterministic and static hump sequencing model. In this approach almost all railcars are humped only once, but three track pulls are performed within 24 hours. He calls each pair of an inbound train and an outbound train a “connection”, if the inbound train has railcars dedicated for the outbound train. In a preprocessing step, all connections are determined. Rules are presented, deciding on which connections have to be made and which ones can be dropped. Dropping a connection implies that railcars miss their scheduled outbound train and they are assigned to a later train. The rules for dropping connections depend on the number of wagons and the (scheduled) time span for realizing the connection. All connections, which are to be made are considered as hard constraints and the objective is to minimize the exponential lateness score of all outbound trains. A mixed integer problem formulation is given, which uses arrival times, target departure times, and hump processing times as input parameters.
A sequence of the inbound trains then leads to the actual departure times of the outbound trains, which determine the objective. The problem is solved using a truncated branch-and-bound algorithm and the model was field tested at Union Pacific Railroad’s Hinkle, Oregon. Some adaptations for the real-life application of the model are suggested. Among others, it is mentioned that multistage sorting can be integrated if the set of cars to be rehumped is considered as a further inbound train. Furthermore, it is mentioned that after reviewing a solution of the model, the user might want to drop further connections and re-run the algorithm in order to prevent huge lateness.

Extensions to the approach by Kraft (2000) can be found by Kraft (2002a). He mainly refers to the following problem, called “cherry picking”. It might happen that a railcar, which has high priority to make a connection, is located on a classification track but is surrounded in its block by low priority cars. If the according outbound train has not enough capacity to take all railcars, the high priority car has to be extracted causing high extra switching effort. Thus, a “feasible connection plan” should be derived, ensuring connections and simultaneously respecting train capacity limits, while allowing a limited effort for cherry picking. In order to solve this problem, the author recommends to split inbound trains into blocks, each of which corresponds to an outbound train instead of a destination (which might be served by different trains). This strategy is presented in more detail by Kraft (2002b), to be described below. Using this strategy, it is possible to adjust trip plans of railcars before they arrive at the hump, e.g. to reassign low priority cars to later trains so that no outbound train exceeds its capacity. Although such a different assignment of blocks requires more classification tracks or more rehumps, the author states that this procedure is “far less expensive than the cost of missed connections [...] or running extra trains otherwise required to move all the traffic”. The development of a feasible connection plan as presented by Kraft (2000) is enhanced by describing how railcars should be rescheduled to different trains, even allowing railcars to be assigned to an earlier outbound train. Finally, some rules are presented in order to decide which cars should be left behind in case that an outbound train exceeds its capacity.

Kraft (2002b) presents a rule based approach for assigning blocks to classification tracks. The assignment is dynamic in terms of the possibility to reassign blocks to different tracks after each inbound train has been processed over the hump. He assumes that a hump sequencing order and a feasible connection plan has already been determined, as described by Kraft (2000) and Kraft (2002a). Thus, he assumes that for each block, the number of cars, the length, the departure time, and an accumulation profile is known. Furthermore, it is assumed that three classification tracks are reclassified once per day so that a rehumping occurs every 8 hours. Whenever a train is humped, a (possibly incomplete) assignment of blocks to classification tracks has to be determined. Each railcar is sent to the track of its block or, if its block is not assigned, to one of the rehumping tracks. The latter is only allowed if this does not lead to a missed connection. The question is to find an assignment of blocks to tracks such that all connections are made and track capacities are met. Rules are presented, which help deciding on which cars should be rehumped, how to greedily assign blocks to tracks such that track space is not exceeded, and how to split blocks if they are too large to fit onto a single track.

4. Multi-stage sorting

Whenever rehumping occurs frequently as part of the shunting strategy, we refer to this as multi-stage sorting. Additionally to the tasks performed in single-stage sorting, it has to be decided which tracks are pulled and in which order this should be done. Note that each track can be pulled several times. Again, after shunting, all blocks of an outbound train should be located on the same track and now we may assume that a specific sequence of blocks has to be achieved. Common objectives are the minimization of the number of track pulls, which correlates to the dwell time of railcars in the yard, the minimization of tracks used for shunting, and robustness.

Shunting yards all over the world but especially in Europe allow the usage of multi-stage procedures. We start with a survey on customary sorting strategies, then present recent sorting and encoding schemes, and finally discuss robustness aspects.

4.1. Basic strategies

Daganzo et al. (1983) discusses three commonly used multi-stage sorting strategies: sorting-by-train, sorting-by-block, and triangular sorting. A static setting is assumed in which a set of inbound trains is to be classified and reassigned to predetermined outbound tracks, requiring a certain order of its blocks. It is assumed that the number of outbound blocks exceeds the number of available tracks, which -- generally speaking -- requires multi-stage sorting strategies.

Using the sorting-by-train strategy means that in the first stage, blocks are sorted onto classification tracks according to departing trains. Afterwards, the blocks of each classification track are re-sorted according to the desired order of the departing train.

In the sorting-by-block strategy, the n-th block of a departing train (counted from the front) is assigned to classification track n. In the second stage, the trains are reasssembled using the blocks from classification track one, track two, and so on.

Triangular sorting is more sophisticated and based on a (non-injective) numbering of all blocks. The blocks of the d-th longest departing train receive numbers \( \frac{dd-1}{2} + 1, \frac{dd-1}{2} + 2, \ldots \), see Table 1. The trains leave to the “right” and the sequence of blocks is given, which is indicated by parentheses.

The lowest block number assigned to a classification track k is \( \frac{kk-1}{2} + 1 \) and then the j-th lowest block number assigned to track k is \( \frac{kk-1}{2} + jk + 1 + \frac{d}{2} - \frac{j(j+1)}{2} (j = 2, 3, \ldots) \). This leads to an assignment of block numbers to tracks as shown in Table 2. Note that the sequence of blocks on one track is random and thus, need not be increasing with the block number.

In the second stage, firstly track 1 is pulled and its railcars are assigned to successive tracks (block number 1 remains on track 1, blocks numbered 3 to track 2, ...), see Table 3. Only the sequence of blocks within a set of parentheses is not known.

Secondly, the track 2 is pulled (i.e. blocks numbered 2, 6, 9, ... and 3). There are two blocks with block number 2, which are designated to the first and second train, respectively. The former is assigned to track one, the latter is assigned to track two. The next blocks are again assigned to successive tracks. Blocks of number 3, i.e. blocks belonging to trains 1 and 2, are assigned to the first two tracks, see Table 4. After resorting all tracks in the same

<table>
<thead>
<tr>
<th>Departing train</th>
<th>Block numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[... 6 5 4 3 2 1]</td>
</tr>
<tr>
<td>2</td>
<td>[... 7 6 5 4 3 2]</td>
</tr>
<tr>
<td>3</td>
<td>[... 8 7 6 5 4]</td>
</tr>
<tr>
<td>4</td>
<td>[... 12 11 10 9 8 7]</td>
</tr>
<tr>
<td>5</td>
<td>[... 16 15 14 13 12 11]</td>
</tr>
<tr>
<td>[...</td>
<td></td>
</tr>
</tbody>
</table>
fashion, the \( d \)-th train is completed on track \( d \). Each block will be moved at most three times.

For each sorting strategy, equations for the required number of tracks and for the required number of switches per group of railcars are derived. A switch is the operation of separating two adjacent railcars, or - in flat yards - the separation of two adjacent sets of railcars. Considering these values, sorting-by-train is slightly better than sorting-by-block. In general, triangular sorting requires far less tracks at the cost of a higher number of switches. If no train has more than \( B \) blocks, triangular sorting requires \( \sqrt{2B} - 1.75 + 0.5 \) tracks. The required number of switches for triangular sorting has been determined for both, humping yards and flat yards. In case of the latter, the required switches were only slightly higher than those of the other two strategies.

\textbf{Daganzo (1986)} enhances the results of the previous paper, focusing on sorting-by-train and triangular sorting. Furthermore, he briefly considers the geometric sort-by-block number strategy. Although it does not appear in the paper, let us give a brief description of geometric sorting, as this procedure is of high practical relevance but its most general version is rarely described in literature.

Just like triangular sorting, a numbering of blocks is required. However, this time, the blocks of the \( d \)-th longest departing train receive numbers \( 2(d - 1) + 1, 2(d - 1) + 2, \ldots \), see Table 5.

The \( j \)-th lowest block number assigned to classification track \( k \) is \( 2^{k-1}(2j - 1) \), see Table 6.

In the second of two stages, the tracks are successively pulled for resorting, starting with track 1. Just like triangular sorting, the \( d \)-th train is to be formed on track \( d \). Whenever the \( d \)-th track is pulled, blocks with label up to \( 2d - 1 \) are assigned to the track containing blocks labeled \( j - 1 \). The according block patterns after pulling the first and second track can be seen in Tables 7 and 8, respectively.

The main result on geometric sorting is that the required number of tracks increases much slower with an increasing number of blocks in the yard. But at the same time, the number of switches increases dramatically. Daganzo notes that all strategies can be applied using less tracks, if some trains are bundled to convoys, which in turn requires a greater length of tracks. Thus, space requirements are considered as well. If sorting-by-train and triangular sorting work near capacity concerning the number of tracks, triangular sorting requires longer tracks than sorting-by-train. In contrast to \textbf{Daganzo et al. (1983)}, the required switches for both strategies are not only calculated for the minimum number of tracks, but for any feasible number of tracks. Daganzo concludes that for a given number of tracks, sorting-by-train needs less switches per car than triangular sorting. However, if tracks are scarce, only triangular sorting might be applicable.

The two aforementioned papers by Daganzo both consider static sorting strategies, which means that the assignment of blocks to tracks does not change over time. However, it is not uncommon that some cars of a block stay at the yard (e.g. because of length restrictions of the departing train). These cars will be added to the next scheduled train serving the according block. In order to prevent that these cars occupy space on the classification tracks for a long time, they are set aside for later sorting. Such a situation in which blocks are consecutively filled by cars of incoming trains and emptied by outbound trains, and in which blocks can be assigned to different classification tracks over time, is called dynamic blocking. This strategy has been considered by \textbf{Daganzo (1987a)} and \textbf{Daganzo (1987b)}. The first of these two papers analyses the situation under homogeneous traffic, i.e. a scenario in which the outbound trains for all blocks depart equally frequently. Traffic flow changes such as rush hours are considered in neither paper. The minimum number of tracks, the required track length and the required number of switches in dependency of the number of available tracks are (approximately) determined for triangular sorting and sorting-by-train. The results coincide with the static case: triangular sorting requires less tracks at the cost of more switching operations.

In \textbf{Daganzo (1987b)}, dynamic blocking for heterogeneous traffic is analyzed. In this paper, only sorting-by-train has been considered. The paper presents two major results. Firstly, the number of required tracks is generally higher compared to the case of homogeneous traffic. Secondly, in certain situations (e.g. if most traffic aggregates in few blocks) the number of switches can be lower than in the homogeneous case.

\subsection{4.2. Deterministic approaches}

The papers presented in this section focus on a special substructure of the hump yard, containing only the track over the hump as well as the classification tracks, but not the departure tracks. The classification tracks are therefore treated as stacks, see Fig. 2 (II).

**Table 2** Block pattern after the first stage of triangular sorting (Daganzo et al., 1983).

<table>
<thead>
<tr>
<th>Classification track</th>
<th>Block Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 3 5 8 12 17</td>
</tr>
<tr>
<td>2</td>
<td>2 6 9 13 18</td>
</tr>
<tr>
<td>3</td>
<td>4 10 14 19</td>
</tr>
<tr>
<td>4</td>
<td>7 15 20</td>
</tr>
<tr>
<td>5</td>
<td>11 21</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
</tr>
</tbody>
</table>

**Table 3** Block pattern after the first step of the second stage (Daganzo et al., 1983).

<table>
<thead>
<tr>
<th>Classification track</th>
<th>Block numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(1)</td>
</tr>
<tr>
<td>2</td>
<td>(3)</td>
</tr>
<tr>
<td>3</td>
<td>(5)</td>
</tr>
<tr>
<td>4</td>
<td>(7)</td>
</tr>
<tr>
<td>5</td>
<td>(11)</td>
</tr>
<tr>
<td>6</td>
<td>(17)</td>
</tr>
</tbody>
</table>

**Table 4** Block pattern after the second step of the second stage (Daganzo et al., 1983).

<table>
<thead>
<tr>
<th>Classification track</th>
<th>Block numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(3)</td>
</tr>
<tr>
<td>2</td>
<td>(5)</td>
</tr>
<tr>
<td>3</td>
<td>(7)</td>
</tr>
<tr>
<td>4</td>
<td>(11)</td>
</tr>
<tr>
<td>5</td>
<td>(17)</td>
</tr>
</tbody>
</table>

**Table 5** Block numbering of departing trains for geometric sorting.

<table>
<thead>
<tr>
<th>Departing train</th>
<th>Block numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(3)</td>
</tr>
<tr>
<td>2</td>
<td>(5)</td>
</tr>
<tr>
<td>3</td>
<td>(7)</td>
</tr>
<tr>
<td>4</td>
<td>(11)</td>
</tr>
<tr>
<td>5</td>
<td>(17)</td>
</tr>
</tbody>
</table>
The required output order is obtained by iteratively pulling tracks. As a track pull is rather time consuming, the number of track pulls is to be minimized. One classification track is considered as a departure track that must not be pulled.

A number of very interesting results has been presented by Jacob et al. (2011). They introduce a simple though very powerful encoding for a shunting schedule. A shunting schedule has to contain the tracks pulled and the assignment of railcars being pulled to classification tracks. Instead of labeling all available tracks, the authors use “logical tracks” such that the \( i \)-th track pull is performed on logical track \( i \). Of course, a real track may be pulled more than once and thus, might refer to several logical tracks. Moving a railcar can be described using a binary encoding. If the \( i \)-th character of the binary number (counted from the right) is one, then the railcar is pulled during the \( i \)-th track pull. This implies that it has previously been assigned to the \( i \)-th (logical) track. Thus, after the \( i \)-th track has been pulled, the railcar will be assigned to (logical) track \( j \), where \( j \) is the next character being one. If there is no such \( j \), the railcar will be moved to the departure track. In order to illustrate this encoding, we use the example of Jacob et al. (2011).

Example 2. Given one inbound train with six railcars labeled as follows (6,1,4,2,3,5). The outbound train must have increasing order of the labels. The binary encoding for each railcar is given as indicated in Fig. 2 (I).
As all characters starting from the fourth digit are zero, there are exactly three track pulls, three logical tracks, and four humping procedures. The position of all railcars before humping (II) and after each humping procedure (III–VI) is shown in Fig. 2.

From the observation that the railcars of the outbound train have nondecreasing binary numbers, a general statement for the number of required track pulls for one inbound and one outbound train can be derived. Two railcars, which appear consecutively in the outbound train, but which are not in the desired order in the inbound train, are called a “break”. Let \( c - 1 \) denote the number of breaks, i.e. the breaks split the railcars into chains with increasing order. Then the optimal shunting schedule requires \([\log_2 c]\) track pulls, if the number of tracks and the length of each track is not restricted. Thus, for any restricted case this number is a lower bound. This result can easily be extended to multiple outbound trains. In that case, the number of track pulls corresponds to the maximal number of track pulls required for solely shunting one outbound train. If multiple inbound trains without a fixed order are considered, a humping sequence of these inbound trains has to be determined such that the number of breaks is minimal. Jacob et al. (2011) prove that this problem corresponds to finding a minimum feedback arc set in special graphs. In general graphs, this problem is known to be NP-hard.

If the number of tracks within the yard is limited by \( W \) (including the departure track), the logical tracks furthermore have to be assigned to real tracks. The authors show that it is optimal to pull tracks in a “round robin” fashion, i.e. logical track \( i \) corresponds to physical track \( i \mod W \). If each track has unlimited capacity, the following result is obtained, which has independently been derived by Hansmann and Zimmermann (2008). An inbound train with \( c - 1 \) breaks can be classified with \( h \) track pulls iff \( c \in \mathcal{R}_W(h) \), with

\[
\mathcal{R}_W(h) := \left\{ \sum_{i=h}^{h+W} \mathcal{R}_W(i) \mid \begin{array}{l} h < W, \\
\forall i \in \mathbb{N} \end{array} \right\}
\]

Note that \( \mathcal{R}_W(h) \) is a generalization of the Fibonacci numbers as \( \mathcal{R}_2(h) \) corresponds to the \( h + 2 \)-th Fibonacci number. Moreover, this result states that for \( W \geq 2 \) and sufficiently large tracks, there is always a feasible shunting schedule.

Jacob et al. (2011) furthermore consider the case of restricted track capacity. Each track is assumed to hold at most \( C \) railcars. They present a proof showing that this problem is NP-hard even if only one inbound and one outbound train is considered. However, they present a polynomial time algorithm delivering a solution that needs at most twice as many track pulls as in the optimal solution. Below, when summarizing the results of Hauser and Maue (2010), this 2-approximation will be discussed in more detail. Finally, the authors consider the case that the outbound train corresponds to the inversed inbound train, i.e. the number of railcars corresponds to the number of breaks plus one. This special case can be solved in polynomial time and covers a worst case scenario for an unknown sequence of the input train.

Note that many of the results of Jacob et al. (2011) can also be found in Jacob (2007) and Jacob et al. (2007).

Based on the binary encoding of shunting schedules presented by Jacob (2007), Jacob et al. (2007), and Jacob et al. (2011), Maue and Nunkesser (2009) propose four integer programming models for deriving optimal feasible shunting schedules. They analyze the NP-hard case of restricted track capacities. One inbound train (or several inbound trains with a given humping sequence) and a variable number of outbound trains are considered. The number of track pulls is the main objective, and among all solutions with a minimum number of track pulls, the one with minimum railcar roll-ins (i.e. the total number of railcars pushed over the hump) should be chosen. The first IP model analyses this situation.

Referring to the practical application at classification yards in Switzerland, this model is enhanced by a restriction forcing all railcars to be placed on at most \( W \) tracks after each car is humped once. At that time, the remaining tracks are still used for other operations. The third model considers a restricted number \( W \) of available tracks in general. Note that in this paper, \( W \) includes the tracks on which the outbound trains are formed. Finally, the fourth model incorporates the situation appearing at Lausanne Triage classification yard, where two parallel humps are available. Thus, the outbound trains have to be partitioned into two groups. All outbound trains of one group are then formed in the part of the yard, which is connected to a certain hump. This model furthermore allows for including departure time constraints of outbound trains. Some valid inequalities for the IP are obtained.

The model has been tested on one instance of real life data from Lausanne Triage classification yard, containing 328 railcars and 23 outbound trains. An optimal solution was derived within 3 minutes. The obtained solution requires nine track pulls and occupies nine tracks. Both reduces are a saving of 1 compared to the method currently applied at the yard. The results of further, randomly generated instances are presented as well. The authors derive the conclusion that whenever a track pull can be saved, the according increase of railcar roll-ins is rather mild. E.g. there was one instance in which the number of roll-ins was minimized for a solution with 13 track pulls and another one with 7 track pulls (the optimum). In the first case, 223 railcars passed the hump, whereas in the latter 270 railcar roll-ins were required.

Gatto et al. (2009) present an introduction to shunting operations. They very generally define three types of sorting problems. Train sorting, which considers one inbound train, a fixed number of track pulls and which is to find a certain outbound sequence. The second type of problem furthermore aims at minimizing the number of track pulls. The third problem formulation is to minimize the number of tracks. Common sorting methods are reviewed, whereby triangular and geometric sorting (see above, Daganzio et al., 1983) are explained for the case of one outbound train. The authors furthermore summarize some results of Jacob et al. (2011) and Maue and Nunkesser (2009). They present the binary encoding and an IP formulation.

Márton et al. (2009) extend the IP formulation of Maue and Nunkesser (2009), using slightly different notations. Contrary to Maue and Nunkesser (2009), the number of tracks is restricted at both parts of the classification yard, each of which is connected to one hump. This restriction holds true for both, the initial roll-in and all later roll-ins. After the first roll-in, more tracks successively become available.

Again, a one day instance of Lausanne Triage classification yard has been evaluated. Although there are 1346 railcars in total, only 331 cars have been considered for the multi-stage sorting, as a sophisticated rehumping is not required for the remaining cars. From a theoretical point of view, the solution of the IP model requires one track pull and one track less than the solution applied at the yard (9 instead of 10). In order to verify the practical application of this solution, the processes of the yard have been simulated using “Villon” simulation system. It turned out that the solution of the IP model is applicable without any conflicts and the saving of a track pull and a track indeed materials. Furthermore, even the number of railcar roll-ins was slightly decreased, and the number of settings of switches was significantly decreased.

The NP-hard case of an unbounded number of tracks but limited track capacity is analyzed by Hauser and Maue (2010). They present variants of the 2-approximation developed by Jacob et al. (2011). This basic 2-approximation works in two stages. In the first stage, the capacity constraint is somewhat relaxed. Instead of requiring that the number of railcars assigned to a logical track does not exceed \( c \), it is only required that the total assignment of
railcars (i.e. the number of roll-ins) does not exceed $h_C$, with $h$ being the number of track pulls. An optimal solution for this relaxed problem (concerning $h$) can be derived in polynomial time by separately determining schedules for each outbound train. Let $h'$ be the optimal number of track pulls. If the schedule of one outbound train requires less than $h'$ track pulls, the according railcars remain untouched during the last pulls. This implies that the binary numbers of these railcars have leading zeros. Thus, the optimization problem at hand is now to find a feasible shunting schedule of the original problem such that track pulls are not be changed. The authors argue that this assumption is necessary as it takes a certain time to distribute a new shunting schedule among all people involved. After performing these $p$ sorting steps, up to $k$ (for $k \in \mathbb{N}$ given) new sorting steps might be introduced. It is shown that it is optimal to introduce these steps as soon as possible. Such a recovery schedule is called $(p,k)$-extension of the original shunting schedule. The $(p,k)$-extension is called feasible with respect to a certain scenario, if it delivers a feasible solution for the disrupted instance. The optimization problem at hand is to find a feasible solution with respect to all people involved. After performing these $p$ sorting steps, up to $k$ (for $k \in \mathbb{N}$ given) new sorting steps may be introduced. It is shown that it is optimal to introduce these steps as soon as possible. Such a recovery schedule is called $(p,k)$-extension of the original shunting schedule. The $(p,k)$-extension is called feasible with respect to a certain scenario, if it delivers a feasible solution for the disrupted instance. The optimization problem at hand is now to find a feasible shunting schedule of the original problem such that track pulls are minimal and such that for each scenario of a given set of scenarios, a feasible $(p,k)$-extension can be found. This problem is shown to be NP-complete.

A further situation is analyzed, in which the set of potential scenarios is limited. If there are at most $j$ inbound trains delayed, the above mentioned optimization problem can be solved in polynomial time. The algorithm presented can also be applied to the general case, but its runtime is exponential. The algorithm has been tested on real life instances from Lausanne Triage classification yard for different values of $p$, $k$, and $j$. The results are compared to geometric sorting and triangular sorting, which is currently applied at the yard.
5. General approaches

In this section, we present papers that do not focus on single sorting problems but take a broader look on shunting operations. Firstly, there are two papers in which sorting problems at shunting yards are classified and nomenclatures are proposed to structure the field. We briefly review these systematizations, as they provide an excellent basis for future research. Afterwards, we focus on approaches that consider several shunting yard operations simultaneously.

5.1. Classifications

Hansmann and Zimmermann (2008) present a fruitful classification of sorting problems at hump yards, considering the number of classification tracks as the only objective. For many of these problems, complexity results are derived or summarized. The authors do not restrict themselves to (freight) railcars, but any units of rolling stock (e.g. trains or trams) can be considered. For a given input sequence of units, certain requirements on the outbound sequence (see below) have to be followed. The sorting problems are classified according to seven attributes:

1. As described above, trains can enter and leave classification tracks according to the rules sido, diso, st, ub, or dido. Siso (single in, single out) is further classified to tracks, which are entered and left from the same side (stack), from different sides (queue), or which can be used arbitrarily as a stack or as a queue. Abbreviations: st (stack), qu (queue), sq (stack/queue), dis (diso), sd (sido), dd (dido).

2. A classification track might have length restrictions so that at most h units can be placed on a track. Otherwise, tracks have unbounded capacity. Abbreviations: ub (unbounded), bd (bounded by b).

3. Problems are classified to single-stage sorting (which is called “no shunting” in the original paper) and multi-stage sorting with h rehumping procedures (h hump shunting). Abbreviations: nsh (no shunting), hsh (h hump shunting).

4. If all units have to enter the tracks, before the first unit leaves, the sorting is called “sequential” and otherwise “concurrent”. If exact arrival and departure times have to be followed for each unit, the authors refer to “time windows”. Abbreviations: se (sequential), co (concurrent), tw (time windows).

5. Units of one group (grouped units) have the same characteristics concerning the outbound sequence requirements, e.g. they are to be placed in the same block) might be forced to be placed on at most s different tracks before leaving the classification tracks (s-split). Such a restriction might be reasonable in order to reduce the effort of switching engines in case of units that are not self-propelled. Otherwise, there is no such restriction, i.e. each block may be split over all tracks (split). In case of a sequential (se), single-stage (nsh) sorting problem, units may be split over an arbitrary number of tracks only if the required outbound sequence can be achieved by iteratively pulling all units from one track for departure (in contrast to interrupting the pull out operation for pulling out units from other tracks). This restriction is called chain-split. Abbreviations: s-sp (s-split), sp (split), csp (chain-split).

6. It is assumed that all positions of the outbound sequence are labeled with one of g different integers. Similarly, all units are labeled with one of g integers (for the sake of clarity, let us assume that different integers are used). If a unit labeled a is assigned to a position b, then all units labeled a have to be assigned to positions labeled b and vice versa. If there is an assignment of unit labels to position label, the outbound sequence is ordered. Otherwise, it is free. Abbreviations: fr (free), or (ordered).

7. A situation in which the labels of the outbound sequence appear in blocks (i.e. there are exactly g - 1 positions, in which the label of the next position is different), is referred to as g-blocks. The more general case is a g-pattern. Abbreviations: g-bl (g-blocks), g-pa (g-pattern).

In the style of the scheduling notation, sorting problems are classified using an abbreviation for each of the seven attributes. For example, the problem considered by Dahlhaus et al. (2000) is described by st,ub,nsh,se,cs,pr,fr,g-bl. Similarly, the four problems by Di Stefano and Koči (2004) are classified as st,ub,nsh,se,sp,or,g-bl (note that the problem also covers queues), sd,ub,nsh,se,sp,or,g-bl, ds,ub,nsh,se,sp,or,g-bl, and dd,ub,nsh,se,sp,or,g-bl. Hansmann and Zimmermann (2008) do not refer to the papers presented in Section 4.1 (Daganzo et al., 1983; Daganzo, 1986; Daganzo, 1987a; Daganzo, 1987b). These papers hardly fit into their classification, as the number of humping operations is variable and the number of tracks is not the only objective considered. Besides deriving and summarizing complexity results for a great number of further problems, mostly for single-stage shunting, Hansmann and Zimmermann (2008) present a linear time algorithm for st,ub,h-hsh,se,sp or g-bl. This algorithm has been tested with practical data from hump yard owned by BASF in Ludwigshafen, Germany. Note that Hansmann and Zimmermann (2008) present a further result on the number of different paths along which a railcar may move through a yard, if the number of tracks and the number of track pulls is given. Jacob et al. (2011) independently obtained the same results, which we described in the previous section.

At the same time as Hansmann and Zimmermann (2008), Di Stefano et al. (2007) present another, more abstract classification of shunting problems for various objectives. They mention three main groups, namely single-stage sorting, multi-stage sorting, and train matching problems. All problems are described using the theory of formal languages. That means that a train is seen as a string \( t = c_1 c_2 \cdots c_g \) of characters \( c_i \in \Sigma \) with \( \Sigma \) being a finite alphabet. Each railcar is represented by a position of this string.

Two single-stage sorting problems are presented. In both, a string \( t_{in} \) representing an inbound train and a set of strings \( T_{out} \) of which representing a possible outbound sequence, are given. The first problem is called “Single-Stage Classification Problem” (SSC). It is to find a subdivision of \( t_{in} \) (i.e. a set of subsequences of \( t_{in} \) covering all elements of \( T_{out} \), which is also a subdivision of an element of \( T_{out} \)). The corresponding subdivision corresponds to assigning railcars to classification tracks and afterwards pulling out the railcars onto a departure track. The pull out operation does not require that all railcars of a classification track are pulled out at once. If this is required, i.e. the concatenation of the elements of the subdivision of \( t_{in} \) has to be an element of \( T_{out} \), the problem is called “Complete Single-Stage Classification Problem” (CSSC). Both problems are to minimize the number of classification tracks used, corresponding to minimizing the elements of the subdivision. Three special cases of SSC and CSSC are considered, each of which requiring certain characteristics of \( T_{out} \). Firstly, \( T_{out} \) has to contain all twinning strings. In a twinning string, all characters of the same type must be located successively (corresponding to g-blocks, see Hansmann and Zimmermann (2008)). CSSC with this requirement corresponds to the problem described by Dahlhaus et al. (2000). Secondly, \( T_{out} \) has only one element (corresponding to an ordered sequence, see Hansmann and Zimmermann, 2008). Finally, \( T_{out} \) has only one element and no character appears twice, i.e. no two railcars are interchangeable.

The multi-stage classification problem (MSC) is less formally described. A classification schedule, i.e. a sequence of track pulls in combination with an assignment of railcars to tracks, has to be obtained such that the input string \( t_{in} \) turns into an element of \( T_{out} \).
The number of track pulls is to be minimized. Restrictions on the number of tracks or the track lengths are not mentioned.

The train matching problem is to minimize the number of uncoupling operations of an inbound train such that the required order of an outbound train is achieved. In other words, the inbound train is to be separated into as few blocks as possible. The costs (e.g., in terms of time) for unloading are not considered. In a more general version of this problem, a set of inbound and a set of outbound trains is given. Again, trains can be denoted by strings. Then these problems correspond to the minimum common string partition problem and the minimum common string partition problem for multisets of strings, respectively.

5.2. Holistic approaches

A very expanded yard model has been presented by He et al. (2000). Inbound cars have to be assigned to outbound trains, the classification time of trains must be fixed, outbound trains are to be assigned to classification tracks, and the assembly time for trains has to be determined. Several of these variables are regarded as fuzzy. In a multi-objective approach, throughput time of cars and delay are minimized, and “flexibility” and the length of certain outbound trains are maximized. The resulting model is solved using a genetic algorithm.

The authors present a similar model (He et al., 2003), which does not make use of fuzzy sets, but which furthermore decides on the assignment of switch engines to inbound and outbound trains for classification or assembling, respectively. The resulting MIP model is very large, so that it is decomposed and the subproblems are solved using various heuristics.

Dirnberger and Barkan (2007) present an approach called “Lean Railroading”. It is to improve railyard performance based on lean manufacturing, theory of constraints, and statistical process control. Firstly, the authors point out the relevance of terminal dwell time of railcars on the whole railroad network efficiency. They analyze data from the six largest North American Class I railroads and draw the conclusion that the average dwell time is negatively correlated to the average train speed. In a second step, ideas of how to reduce the terminal dwell time are presented. Besides increasing the terminal capacity, a focus on the terminal’s bottleneck process, i.e., the throughput limiting process, is suggested. The authors identified train assembly as the bottleneck process, indicated by a long idle time of the railcars prior to assembly. They present a detailed analysis of the assembly process at Bensenville Yard in Illinois and they show how the bottleneck’s capacity can be improved: Although adding further pull-down engines is an option, it might not be cost-effective. Other possible improvements are the usage of the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times, minimizing “cherry picking” operations by optimizing the hump engine for assembly operations during idle times. In other words, the inbound train is to be shunted as soon as all according inbound trains have arrived. It is assumed that the shunting time is
only dependent on the number of railcars to be shunted, and that the
time needed per railcar decreases if more railcars are shunted. Thus, it is to be decided, at what
time the shunting operations of outbound trains should be started, or, equivalently, the railcars of
which outbound trains should be shunted together. The problem is solved in linearheuristic time using dynamic programming.

6. Summary and future research

There is a wide range of tasks to be performed at shunting yards. Although much research has been done on accounting prob-
lems, there are still several fields requiring further theoretical and applied research. We give a brief overview on emerging research
field.

Theoretical models of sorting, especially concerning multi-stage sorting, have put little emphasis on required departure times of
railcars. Certainly, it should be analyzed how accounting restrictions would affect existing sorting models and new models should be
derived considering latency as an objective.

The classification of shunting problems by Hansmann and
Zimmermann (2008) is a great basis for future research, if the
number of tracks is the only objective. For almost all problems that
can be described using their notation, at least some practical rele-
ance is obvious. However, many of these problems (especially
multi-stage problems on yards with a non-siso layout) have not
been analyzed. But there are even more problems in this field,
which have not been tackled. The classification of Hansmann and
Zimmermann (2008) does not contain cases in which the sequence
of inbound trains is variable, in which the number of humping
operations is variable, and a situation in which the sequence of
railcars within an outbound train is given, but the sequence of
the outbound trains is arbitrary.

Lately, the biggest research advancements on shunting yards
have been made in approaches minimizing track pulls. In this con-
text, the paper of Jacob et al. (2011) is worth mentioning. Their
encoding of shunting schedules allows a thorough analysis of
new problems, integrating further practical aspects. For example,
the above mentioned departure times, other objectives than track
pulls, and the interactions arising when using more than one hump
should be analyzed in more detail. The latter is important since
Lausanne Triage is not the only yard featuring two humps (e.g. Bai-
ley Yard in Nebraska, USA).

Robustness and recovery issues certainly offer a great range of
future research. This is due to the fact that there are several ap-
proaches for formalizing robustness and that a huge variety of clas-
ses of realistic disruptions can be considered. The recoverable
robustness approach (see Liebchen et al., 2009) could be applied
to further problems not being considered by Cicerone et al.
(2009a), and according algorithms should be developed (e.g. con-
sidering shunting problems with several inbound/outbound trains). An evaluation of common sorting algorithms concerning
their price of robustness under various disruptions would certainly
enhance literature in this field. Such an evaluation should consider
different sets of allowed recovery actions, and it should include the idea of \((p,k)\)-extensions proposed by Bühsing and Maue (2010).

A general issue, which in our eyes is missing some attention, is
the movement of railcars from the classification tracks to depar-
ture tracks in a yard, which has the layout as in Fig. 1. As men-
tioned by Dirnberger and Barkan (2007), it is not uncommon that
switching engines support the sorting process when pulling rail-
cars from classification tracks to departure tracks. More generally
speaking, too little focus has been put on yard layouts, which are
different from single in, single out.

We have presented an overview on the literature on shunting
yards starting from the end of the 1970s until today. Lately, many
researchers have rediscovered the importance of an efficient usage
of shunting yards as a key for freight transport improvements.
Thus, new theoretical concepts have been proposed and the appli-
cability has often been shown using real life data. The papers,
which have been discussed in this review are listed and classified
in Table 9.

References

of conventional and innovative technologies in the intermodal transport area.
European Journal of Operational Research 152, 410–419.


OR Spectrum 24, 1–17.

16A, 21–32.

16A, 21–32.


Bektas, T., Crainic, T.G., Morency, V., 2009. Improving the performance of rail yards
through dynamic reassignments of empty cars. Transportation Research Part C

Benna, T., Gronalt, M., 2008 Generic simulation for rail-road container terminals.
In: Transportation Simulation Conference, 2556–2660.

Bierwirth, C., Meisel, F., 2010. A survey of berth allocation and quay crane
scheduling problems in container terminals. European Journal of Operational
Research 202, 615–627.

Scheduling trains in the morning. Mathematical Methods of Operations
Research 49, 137–148.

trains. Transportation Research Part B 14B, 115–120.

Bontekoning, Y., Macharis, C., Trij, J., 2004. Is a new applied transportation research
field emerging? A review of intermodal rail-haul freight transport literature.

Bontekoning, Y., Priemus, H., 2004. Breakthrough innovations in intermodal freight
transport. Transportation Planning and Technology 27, 335–345.

Boysen, N., Fleider, M., 2010. Determining crane areas in intermodal
transshipment yards: The yard partition problem. European Journal of
Operational Research 204, 336–342.


Boysen, N., Jaehn, F., Pesch, E., in press. New bounds and algorithms for the

transshipment yards. Transportation Science 45, 199–211.

Brungs, F., Knust, S., in press. Optimized load planning of trains in intermodal
transportation. OR Spectrum.

Bühsing, C., Maue, J., 2010. Robust algorithms for sorting railway cars. In: de Berg,
M., Meyer, U. (Eds.), Proceedings of the 18th annual european symposium on
algorithms (ESA-10). Lecture Notes in Computer Science, Vol. 6346. Springer,


Cicerone, S., D’Angelo, G., Di Stefano, G., Frigioni, D., Navarra, A., Schachtebeck, M.,
Schöbel, A., 2009b. Recoverable robustness in shunting and timetabling. In:
Abuja, R., Miehling, R., Zaroliagis, C. (Eds.). Robust and online large-scale
Heidelberg, pp. 28–60.


freight transportation. Transportation Science 18, 165–184.


throughput: The case of multistage triangular sorting. Transportation
Science 17A, 95–106.


throughput: The case of multistage triangular sorting. Transportation
Science 20, 189–199.

Daganzo, C.F., 1986. Static blocking at railyards: Sorting implications and track
requirements. Transportation Science 24, 1–24.


throughput: The case of multistage triangular sorting. Transportation
Science 17A, 95–106.

Discrete Applied Mathematics 103, 41–54.

models in freight transportation. Transportation Science 21, 227–247.

Di Stefano, G., Köcici, M., 2004. A graph theoretical approach to the shunting