Università degli Studi di Cagliari

DOTTORATO DI RICERCA

INGEGNERIA ELETTRONICA E INFORMATICA

Ciclo XXX

ADVANCED MODELING AND CONTROL OF INTERMODAL TERMINALS AND RAILWAY NETWORKS

Settore scientifico disciplinare di afferenza

ING-INF/04 (Automatica)

Presentata da:	**GRAZIANA CAVONE**
Coordinatore Dottorato	**Prof. FABIO ROLI**
Tutor	**Prof.ssa CARLA SEATZU**
Co-Tutor	**Prof.ssa MARIAGRAZIA DOTOLI**

Esame finale anno accademico 2016 – 2017

Tesi discussa nella sessione d'esame Febbraio-Marzo 2018

Contents

List of Figures

List of Tables

List of Publications

Journal papers

Cavone, G., Dotoli, M., Epicoco, N., Seatzu, C., "Intermodal terminal planning by Petri Nets and Data Envelopment Analysis", (2017), Control Engineering Practice, 69, pp. 9-22.

Cavone, G., Dotoli, M., Epicoco, N., Seatzu, C., "A decision making procedure for robust train rescheduling based on mixed integer linear programming and Data Envelopment Analysis", (2017), Applied Mathematical Modelling, 52, pp. 255-273.

Cavone, G., Dotoli, M., Seatzu, C., "A Survey on Petri Net Models for Freight Logistics and Transportation Systems", (2017), IEEE Transactions on Intelligent Transportation Systems, . Article in Press.

Dotoli, M., Epicoco, N., Falagario, M., Cavone, G., "A Timed Petri Nets Model for Performance Evaluation of Intermodal Freight Transport Terminals", (2016), IEEE Transactions on Automation Science and Engineering, 13 (2), art. no. 7057695, pp. 842-857.

Cavone, G., Dotoli, M., Seatzu, C., " Management of Intermodal Freight Terminals by First-Order Hybrid Petri Nets", (2016), IEEE Robotics and Automation Letters, 1 (1), art. no. 7339445, pp. 2-9.

Conference Papers

Cavone, G., Dotoli, M., Seatzu, C., "Resource planning of intermodal terminals using timed Petri nets", (2016), 2016 13th International Workshop on Discrete Event Systems, WODES 2016, art. no. 7497824, pp. 44-50.

Dotoli, M., Epicoco, N., Falagario, M., Cavone, G., "A timed Petri nets model for intermodal freight transport terminals", (2014), IFAC Proceedings Volumes (IFAC-PapersOnline), 9 (3), pp. 176-181.

Dotoli, M., Epicoco, N., Cavone, G., Turchiano, B., Falagario, M., "Simulation and performance evaluation of an Intermodal terminal using Petri Nets", (2014), Proceedings - 2014 International Conference on Control, Decision and Information Technologies, CoDIT 2014, art. no. 6996915, pp. 327-332.

Dotoli, M., Epicoco, N., Falagario, M., Turchiano, B., Cavone, G., Convertini, A., "A Decision Support System for real-time rescheduling of railways", (2014) 2014 European Control Conference, ECC 2014, art. no. 6862177, pp. 696-701.

Acknowledgement

Another great journey draws to a close and, like in all great journeys, many traveling companions left gifts and deserve acknowledgements.

First and foremost, Carla Seatzu and Mariagrazia Dotoli, who assured me their expert guidance through this journey, being my compass when I risked to lose the way and giving me support in professional and personal growth. Thanks to both, the ideas I had at the beginning of the PhD course turned from immature seeds into concrete scientific works and finally into this PhD thesis.

Bart De Schutter and Ton Van Den Boom, who supervised me during the visiting period at the Delft University of Technology. The meetings we had together were of great inspiration for me.

Cristian Mahulea and Slim Hammadi, who provided additional guidance as external reviewers of my PhD thesis.

All the colleagues and friends. Francesco Acquaviva, who introduced me to the Research, Nicola Epicoco and Raffale Carli, the soul of the D&CLab, two great professionals and valuable persons whose presence has been essential during my PhD course. Alessandro e Virginia Pilloni, Daniela Deiaco, Yin Tong, Alessandro Argiolas, Michele Nitti, Valentina Angius, Andrea Carta, Tore Martis, Maria Amenabar, Barbara Mascayano, thank you friends of Sardinia for making me discover this incredible island and myself. I thank also Gianluca Mereu and Mauro Franceschelli for their tangible support in all technical and practical issues. The multiethnic group of friends I found in Delft and conferences, Ioanna Kourounioti, Ines Uriol, Tomas Pippia, Yu Hu, Julian Murillo, Christian Portilla, Vahab Rostampour, Stefano Farì, Carlos Celemin, Shuai Yuan, Mohammd Mirz, Farid Alavi, Angelo Vermeulen, Indushree Banerjee, Amine Abdeljaouad thanks for all the moments spent together, sharing experiences and pieces of life.

All my family, especially my brother and father, who perfectly know the sense of the words "Give the ones you love wings to fly, roots to come back, and reasons to stay" (Dalai Lama).

Vincenzo Lionetti, my best travelling companion in the incredible journey of life.

Abstract

This PhD thesis presents the results of research on Information and Communication Technologies (ICTs) in the context of Smart Cities, with particular attention to the study, design, and the development of advanced models and control techniques for intermodal freight transport terminals and railway transport networks.

At the moment, the implementation of a Smart City environment is largely recognized as an effective manner to improve the quality of life in the urban context. This thesis mainly focuses on the improvement of intermodal and railway transport, which are leading alternatives to road transport for the reduction of greenhouse gas emissions. In particular, the aim is to strongly increase the benefits that such transportation systems can generate by contributing to the resolution of the respective main managerial challenges. In particular, the intrinsic discrete event dynamics of (1) intermodal freight transport terminals and (2) railway transport is here considered to derive advanced models and control techniques to be used for the resolution of the main strategic/tactical and operational decision problems characterizing such systems.

First, the viability of discrete event methods for smart transportation systems is here discussed. On one hand, it is provided a review of contributions on Petri nets for freight transportation systems; on the other hand, an overview on the discrete event MILP models for the railway rescheduling problem. On the basis of the developed reviews, contributions are here provided on (1) modeling, simulation, analysis, and control via Petri nets of Intermodal Freight Transport Terminals (IFTTs) and (2) on discrete event MILP modeling of railway traffic and the corresponding smart management when unexpected events occur in the network. Regarding topic (1), first a general modelling framework based on timed PNs is proposed which allows simulating and evaluating the performance of such key elements of the intermodal transportation chain. Then, it is shown how first-order hybrid Petri nets can be efficiently used to model and subsequently manage intermodal freight transport terminals by optimizing the terminal performance under alternative control policies Finally, it is demonstrated how timed Petri nets and the Data Envelopment Analysis (DEA) multi-objective optimization technique can be combined for the planning of intermodal terminals. The effectiveness of all of these techniques is tested on a real case study showing their practical use and ease of application. Regarding topic (2), first, a Decision Support System (DSS) for real-time management of railway networks is presented, which employs a MILP approach addressing traffic rescheduling under unexpected disturbances in a mixed- (single- and double-) tracked network. Then, it is proposed a self-learning decision making procedure for robust real-time train rescheduling in case of disturbances. The procedure is applicable to aperiodic timetables of mixed-tracked networks and it consists of three steps. The railway service provider can take advantage of this procedure to automate, optimize, and expedite the rescheduling process. Moreover, thanks to the self-learning capability of the procedure, the quality of the rescheduling is improved at each reapplication of the method. Both the presented techniques are applied to a real data set to test its effectiveness. Finally, the last contribution presents an innovative bi-level algorithm aiming at finding a feasible timetable for a mesoscopic rescheduling problem in case of disruption in a short computation time. It consists in the sequential resolution of two optimization MILP problems. It is here preliminary demonstrated, by the application of the method to a real case study (i.e., the national Dutch railway network), that the bi-level solving algorithm can be suitable for a real-time control environment thanks to its short computation time.

Chapter 1

1 Introduction

The implementation of a Smart Transportation System (STS) is one of the main challenges in the context of a Smart City environment. This is mostly due to the polyhedral meaning of smartness. In fact, it consists in creating a transportation system which can simultaneously guarantee improved safety, higher productivity and efficiency, more environmental friendliness and better life quality to citizens.

The idea of smart transport is not new and the first attempts to define this concept were made in the '90s. In particular, in 1995 the authors of [1] analyzed various virtuous cities where the transportation system was improved using advanced technologies, with the purpose of identifying available approaches, developed to evolve traditional transportation systems. Then, the American research and innovation technology administration defined an architectural structure for deployment of smart technologies in the national transportation system [2]. Moreover, the smart transportation guidebook [3] outlined different policies for making STSs for New Jersey and Pennsylvania, while [4] suggested a number of ways for improving mobility by introducing smart technologies. Finally, Debnath *et al.* in [5] discussed the smart technology initiatives developed in Singapore. All of these contributions had the great credit of creating the basis for the identification and introduction of smartness in transportation systems.

In the context of urban transport systems, various literature contributions (e.g., [6]; [7]) have recognized the fundamental role of smart technologies in reaching the features of smartness and sustainability. Some authors (e.g., [5]; [8]) have demonstrated how smart technologies can support sustainability by achieving greater economic and environmental efficiency. In particular, introducing innovative technologies in traditional transport modes can strongly improve the ability of monitoring, control, track, and improve the mobility and traffic levels, improving the quality of life, and reducing a wide set of social, economic, and environmental impacts. A smart urban transport system is frequently regarded as a system that takes advantage of smart technologies for its operational and managerial activities. Due to the inclusion of smart technologies, it has to behave in a self -operative and corrective manner, requiring little or no human intervention. Consequently, it should include three fundamental elements: sensors, command and control unit, and actuators to provide the basic capabilities of sensing, processing and decision making, acting (control), and communicating [9]. In addition, there could be some other advanced or higher-order capabilities, e.g., predictability, healing, and preventability. Predictability is the advanced level of basic sensing and processing, which refers to how accurately a system can predict a potential problem or scenario. Healing is the advanced level of control, that is, how well a system can heal potential problems to have complete recovery without any human intervention. Preventability, combination of predictability and healing, is the ultimate level of smartness, which makes a system capable of preventing potential failures by predicting and taking the appropriate preventive measures.

As already recalled, the above capabilities are fundamental for transportation systems that require to evolve the traditional operational and managerial procedures, centered on the experience of the respective operators, into a smart approach. As knowledge-based methods, the traditional techniques often lead to sub-optimal solutions when facing decision making and control problems both at strategic/tactical and operational levels. Automated procedures based on mathematical approaches combined with information and communication technologies can overcome such limit giving a great contribution to the implementation of a smart transportation system.

The research activity developed during these three years focused on ICT technologies for urban mobility, with particular attention to the modeling, simulation, analysis, and control of intermodal freight transport terminals and railway transport in the context of smart cities.

1.1 Objectives and contributions

The work in this Thesis has been developed within the PhD program in Electronic and Computer Engineering at the University of Cagliari (Italy) under the supervision of the tutor Prof. Carla Seatzu (University of Cagliari, Italy) and the co-tutor Prof. Mariagrazia Dotoli (Polytecnic of Bari, Italy). Furthermore, part of this work has been realized within the PlaceDoc program at the Technical University of Delft (the Netherlands) under the supervision of Prof. Bart De Schutter and Prof. Ton van den Boom (Technische Universiteit Delft, the Netherlands).

The thesis mainly focuses on the challenges of implementing advanced modeling and control techniques to support the management at strategic/tactical and operational levels in intermodal freight transport terminals and railway transport. With this aim, the following works have been done:

- A review of contributions on Petri Nets (PNs) for freight transportation systems, with a classification of the papers according to the addressed managerial problem, namely strategic/tactical or operational decision-making level problem, and the adopted PN formalism and an overview on the discrete event Mixed Integer Linear Programming (MILP) models for railway systems with particular attention to the railway traffic rescheduling problem.
- The implementation of a general modular modelling framework for the simulation and evaluation of Intermodal Freight Transport Terminals (IFTTs) based on timed Petri nets. This advanced model allows decision makers to identify the IFTTs' bottlenecks, as well as to test different solutions to improve the IFTT dynamics and thanks to its modularity is able to represent the different types of existing IFTTs.
- The application of the First-Order Hybrid Petri Nets (FOHPNs) formalism for the efficiently modeling and management of IFTTs. It is demonstrated the practical relevance of the formalism in enabling the terminal decision maker to choose the speeds associated with continuous transitions in order to optimize the terminal performance.
- The combination of timed Petri nets and Data Envelopment Analysis (DEA) for planning of intermodal terminals in critical conditions i.e., dimensioning number, capacity, and frequency of resources. In particular, the integration of the two approaches is used to select the most appropriate resolving scenario while taking into account conflicting requirements on the terminal performance.
- The implementation of a Decision Support System (DSS) for real-time management of railway networks based on a MILP approach for the resolution of the traffic rescheduling problem under unexpected disturbances in a mixed- (single- and double-) tracked network. The DSS simulates the network behavior with the mathematical programming model based on the railway topology and constraints, rescheduling the timetable in real time, detecting and solving conflicts in the network;
- The implementation of a three-steps self-learning decision making procedure for robust real-time train rescheduling in case of disturbances. The first two steps are executed in real-time and provide the rescheduled timetable, while the third one is executed offline and guarantees the self-learning part of the method. The railway service provider can take advantage of this procedure to automate, optimize, and expedite the rescheduling process. Moreover, thanks to the self-learning capability of the procedure, the quality of the rescheduling is improved at each reapplication of the method;
- The implementation of an innovative solving algorithm for real-time rescheduling of railway traffic in case of disruption. In the control problem the railway network is represented as a discrete event system, whose evolution is guided by the occurrence of train runs and is constrained by safety requirements. The rescheduling problem is modelled as a MILP problem

whose objective is to minimize the delays and cancellation of trains and shunting actions in the network, respecting timetable, running time, continuity, and headway constraints, and including cancelation of trains or short-turning, shunting, and ordering actions in a short computation time. The problem is solved using a two-level algorithm that consists in the sequential resolution of two optimization MILP problems. The first level optimization considers a macroscopic MILP model of the disrupted network in which all the constraints are considered but it is ideally assumed that the stations involved by the disruption have infinite capacity and no platform constraint is necessary. The second level optimization considers a mesoscopic MILP model that includes the results of the first level optimization together with capacity constraints (i.e., are included shunting-actions, short-turns with platform assignment, and ordering actions on platforms). This two-level solving algorithm can provide a feasible solution faster than full mesoscopic or microscopic optimization algorithms and can be applied in a real-time control environment.

All of the techniques have been tested and evaluated on real case studies.

1.2 Thesis Structure

In Chapter 2, focusing on discrete event methods for smart transportation systems, a review on Petri nets for freight transportation systems is presented together with an overview on the discrete event MILP models for railway systems. Chapter 3 presents innovative techniques based on Petri nets and data envelopment analysis for modeling, analysis, control and management of intermodal freight transportation terminals. In Chapter 4 advanced techniques for real-time railway traffic management are presented. In particular, a DSS and a decision-making procedure based on MILP modelling and DEA for the rescheduling in case of disturbances are proposed and a novel bi-level optimization algorithm is described for the resolution of the rescheduling problem in case of disruptions. Finally, conclusions and future works are drawn in Chapter 5.

Chapter 2

2 Discrete Event Systems Approaches for Smart Transportation Systems

This chapter focuses on the development of proper mathematical models that can be used for simulation, analysis, optimization, and control of transportation systems. More in detail, considering the intrinsic discrete event dynamics of these systems and the state of the art in the related context, two main classes of suitable discrete event systems approaches have been identified and developed: Petri Net and Mixed Integer Linear Programming models. On the one hand PNs, thanks to a series of generally appreciated features, allow to model, simulate, analyze, and control such systems. On the other hand MILP models permit to easily implement and solve optimization problems typical of real-time control.

Section 2.1 presents a review on contributions about Petri nets for freight logistics and transportation systems. The review has been developed during these years of research activity and was published in [10]. It aims at offering to the scientific community a global overview on the state of the art. Papers are classified according to the addressed problem, namely strategic/tactical or operational decision-making level problem, and the adopted PN formalism. Comments are also provided on the approaches' viability, discussing contributions and limitations, and identifying future research directions to enhance the successful application of PNs in freight logistics and transportation systems.

Section 2.2 provides a discussion on the application of the MILP model technique for railway traffic management. The section aims at highlighting the practical relevance of the MILP modeling approach, which allows the intuitive description of the railway network and traffic as a discrete event system and the optimization of the corresponding performance under strict constraints. In particular, the focus is on one of the main problems in railway traffic management, i.e. the real-time rescheduling problem in case of disturbance or disruptions.

2.1 A Survey on Petri nets for freight transportation systems

Freight logistics is fundamental in production chains and regards the management of materials or internal logistics and the physical distribution, also called distributive or external logistics. However, the physical distribution is one of the major waste sources in production systems and companies, as well as customers, are strongly interested in reducing such problem, which lowers profits and generates dissatisfaction. Smart Transportation Systems, considered as the integration of Information and Communication Technologies (ICTs) with transportation systems, can be used at this aim. In particular STSs enable stakeholders to be better informed, and make a safer, more coordinated, and smarter use of transport services.

As already introduced, Petri nets are a well-known formalism that can contribute to the development of efficient STSs, thanks to their recognized effectiveness in solving transport decision problems like, e.g., resources management, planning and optimization of routes, monitoring and control of transportation activities, application of safety rules, reduction of energy consumption as well as pollution, and so on [11] [12]. This relies on the intrinsic discrete event dynamic of freight physical distribution that allows its representation as a Discrete Event System (DES) and to take advantage of a series of largely valued properties of PNs to model, simulate, analyze, and control such systems. In particular, PNs are particularly valuable for their graphical and compact representation, modularity, possibility of modeling concurrent and parallel events, and definition of state (marking vector) that allows to efficiently solve various problems via linear integer programming, without requiring exhaustive enumeration.

In the literature two ways are available to offer specific and efficient methods to improve physical distribution activities in the DESs framework, i.e., DES simulation and performance analysis, and DESs analytical models [13]. This research work focuses on the PN formalism, i.e., a DES analytical model, because with respect to DES simulation models it allows not only simulation analysis, but also optimization and control of the modeled systems to solve problems related to strategic/tactical and operational decision-making levels of freight transportation systems.

It has to be noticed that while the related literature abounds with studies and reviews on PNs for urban transport and particularly passengers' transfer [14],[15], only few works discuss the use of PNs for freight logistics and transportation systems. In order to encourage and support researchers in applying the PN formalism in such a context, for the above reported advantages, in this research work it is provided an overview of the contributions using PNs for freight transportation chains and the terminals used for freight pick up, delivery, and transshipment. In particular, since any freight logistics and transportation system involves one or more transportation modes - water (sea/river) ways, railways, roads, airways and their combination- the research contributions are summarized regarding each transportation mode and respective terminals, as well as their integration into multimodal and intermodal systems. For each mode, it is considered the decision problems that each company is required to solve. Hence, the contributions are classified based on the tackled problem at strategic/tactical (i.e., over a middle term horizon) and at operational level (i.e., over a short term horizon). The classification of the related works using PNs is developed in accordance to the specific PN formalism (logical, with time, or high-level PNs) and addressed decision problem, and it is provided a debate on the viability of the approaches, discussing contributions and limitations. Finally, it is presented a discussion of the main literature lines of contribution and identify future research directions to enhance the application of PNs to freight logistics and transportation systems.

2.1.1 Petri nets

A Place/Transition (P/T) net is a bipartite graph, whose vertices can be distinguished into places, represented by circles, and transitions, represented by bars. Directed arcs connect places and transitions, while tokens, represented by black dots, describe the state of the net. Frequently, parallel arcs connecting a place (transition) to a transition (place) are represented by a directed arc labeled with its multiplicity, or weight (for more details on the mathematical formulation, see [16]). The dynamical behavior of the net is governed by the following rules.

Enabling Rule: a transition t is said to be enabled if each input place p of t contains at least a number of tokens equal to the weight of the directed arc connecting p to t.

Firing Rule: the firing of an enabled transition t removes from each input place p a number of tokens equal to the weight of the directed arc connecting p to t. It also deposits in each output place p a number of tokens equal to the weight of the directed arc connecting t to p. Consequently, the firing of a transition produces a new marking of the net, i.e., a state evolution, which can be computed by means of a *state equation*.

Several PN formalisms have been proposed to model DESs extending the basic P/T formalism, each being best suited for the desired specific purpose or degree of detail. As an example, when the issue is analyzing the system performance (e.g., determining the execution time of an activity, identifying bottlenecks, and optimizing the use of resources), the P/T net model is not appropriate, since it cannot model the duration of activities. Significant contributions have been provided extending PNs with time under the assumption that delays associated with transitions may either be deterministic -as in Timed PNs (TPNs)- or stochastic -as in Stochastic TPNs (STPNs)- these types of PNs are called PNs with time as difference to logic PN models in P/T nets. Formally a TPN is defined as a bipartite directed graph, which can be mathematically symbolized with a five-tuple $TPN=(P, T, \boldsymbol{Pre}, \boldsymbol{Post}, \boldsymbol{F})$. In particular, P represents the set of places with cardinality m, and T denotes the set of transitions, having cardinality n [16]. Set T includes two different subsets: T_I, representing the subset of immediate transitions (symbolized by bars), and T_D is the subset of deterministic timed transitions (described by black boxes). In case of STPNs the only difference is in set T that contains an additional subset T_S, i.e., the subset of stochastic timed transitions (depicted by empty boxes). Stochastic transitions can be associated with any type of probability distribution, depending on the stochastic event they represent. In the five-tuple TPN, $\boldsymbol{Pre}: P\times T\rightarrow\mathbb{N}^{m\times n}$ and $\boldsymbol{Post}: P\times T\rightarrow\mathbb{N}^{m\times n}$ respectively represent the *pre-incidence matrix* and the *post-incidence matrix* (being $\mathbb{N}$ the set of non-negative integer numbers). Such matrices specify the type of connections linking places in P with transitions in T. For each $p\in P$ and $t\in T$, $\boldsymbol{Pre}(p,t)$ and $\boldsymbol{Post}(p,t)$ are two natural numbers that represent the weight of the arc going respectively from p to t and from t to p. Only when this value is higher than one, the corresponding arc is labeled with its weight. In addition, the fifth element of a TPN is $\boldsymbol{F}$: $T\rightarrow\mathbb{R}_0^+$, namely the function defining the time delays of transitions in T (being $\mathbb{R}_0^+$ the set of non-negative real numbers). More precisely, for each deterministic timed transition $t_j\in T_D$, $\boldsymbol{F}(t_j)=\delta_j$ specifies its (constant) firing delay δ_j; for each exponentially distributed timed transition $t_j\in T_S$, $\boldsymbol{F}(t_j)=1/\lambda_j$ indicates the average firing delay, where λ_j is the characteristic parameter of the corresponding exponential distribution; finally, for each immediate transition $t_j\in T_I$, $\boldsymbol{F}(t_j)=0$ designates its corresponding zero firing delay. The state of the net is defined through its marking, that is a mapping $\boldsymbol{M}$: $P\rightarrow\mathbb{N}^m$ assigning to each place $p_i\in P$ a nonnegative number of tokens, and $\boldsymbol{M}(p_i)$ denotes the number of tokens in p_i. A TPN system $\langle TPN, \boldsymbol{M_0}\rangle$ is a TPN with an initial marking $\boldsymbol{M_0}$. Discrete events correspond to the firing of transitions. A transition $t\in T$ is enabled at a marking $\boldsymbol{M}$ if $\boldsymbol{M}(\cdot)\geq\boldsymbol{Pre}(\cdot,t)$ and it may fire provided that it remains enabled for a time interval equal to its current firing delay.

Continuous and hybrid PNs differently from TPNs and STPNs can be used to deal with the well-known state explosion problem and the consequent increase in computational costs, which typically affects PN models of large and complex systems (just as logistics and transportation systems). The continuity feature is provided by fluidization of the underlying discrete model, in which the integer content of places becomes a real nonnegative number and the time delay of transitions becomes a speed. Although not preserving all behavioral properties of their discrete counterpart, continuous PNs allow the application of numerous techniques developed for the discrete framework, e.g., supervisory control techniques like Generalized Mutual Exclusion Constraints (GMECs) that can impose limitations on the weighted sum of markings in a subset of places. When the fluidization of the PN model is partial, a hybrid model, called Hybrid PNs (HPNs), is obtained. This allows dealing with different fluid approximations that depend on the discrete state of the system. In this thesis, First-Order hybrid Petri nets are used to model, simulate, analyze, and control the behavior and performance of a complex transportation system, i.e., an intermodal freight transport terminal. Formally, the structure $N=\left(P,T,\mathbf{Pre},\mathbf{Post},\mathcal{D},\mathcal{C}\right)$ represents a FOHPN [17], where $P=P_d\cup P_c$ is the union of a set of continuous places P_c (represented as double circles) and a set of discrete places P_d (represented as circles). Moreover, $T=T_d\cup T_c$ is the union of a set of discrete transitions T_d and a set of continuous transitions T_c (represented as double boxes). The set $T_d=T_I\cup T_D\cup T_E$ is subdivided into a set of immediate transitions T_I (represented as bars), a set of deterministic timed transitions T_D (represented as black boxes), and a set of exponentially distributed timed transitions T_E (represented as white boxes). The cardinalities of T, T_d, and T_c are denoted n, n_d, and n_c. The pre- and post-incidence functions that specify the arcs from places to transitions and from transitions to places are, respectively:

$$\mathbf{Pre}=\begin{cases} P_d\times T\rightarrow\mathbb{N} \\ P_c\times T\rightarrow\mathbb{R}_0^+ \end{cases} and\,\mathbf{Post}=\begin{cases} P_d\times T\rightarrow\mathbb{N} \\ P_c\times T\rightarrow\mathbb{R}_0^+ \end{cases}$$

It is assumed the *well-formed nets* hypothesis is verified, i.e., for all $t_i\in T_c$ and for all $p_i\in P_d$ it holds $\mathbf{Pre}\left(p,t\right)=\mathbf{Post}\left(p,t\right)$. The function $\mathcal{D}=T_d\setminus T_I\rightarrow\mathbb{R}^+$ insists on the timing related to timed discrete transitions. A deterministic timed transition $t_i\in T_D$ is associated with a (constant) firing delay $\delta_i=\mathcal{D}\left(t_i\right)$. An exponentially distributed timed transition $t_i\in T_E$ is associated with its average firing rate $\lambda_i=\mathcal{D}\left(t_i\right)$, i.e., the average firing delay is $1/\lambda_i$, where λ_i is the parameter of the corresponding exponential distribution. The function $\mathcal{C}=T_c\rightarrow\mathbb{R}_0^+\times\mathbb{R}_\infty^+$ specifies the firing speeds associated with continuous transitions. For any continuous transition $t_i\in T_c$, it is $\mathcal{C}\left(t_i\right)=\left(V_i^{'},V_i\right)$, with $V_i^{'}\leq V_i$. Here $V_i^{'}$ represents the minimum firing speed (mfs) and V_i represents the Maximum Firing Speed (MFS). The preset (postset) of transition t is represented as ${}^\bullet t$ ($t^\bullet$) and its restriction to discrete or continuous places as ${}^{(d)}t={}^\bullet t\cap P_d$ or ${}^{(c)}t={}^\bullet t\cap P_c$, respectively. Similar notations may be used for presets and postsets of places. The incidence matrix of the net is defined as $C=Post-Pre$. The restriction of C to P_X and T_Y ($X,Y\in\left\{c,d\right\}$) is denoted $C_{X,Y}$. Note that by the *well-formed* hypothesis, $C_{d,c}=0$. A marking

$$\boldsymbol{m}=\begin{cases} P_d\rightarrow\mathbb{N} \\ P_c\rightarrow\mathbb{R}_0^+ \end{cases}$$

is a function that assigns to each discrete place a nonnegative number of tokens, represented by black dots, and to each continuous place a fluid volume; m_p denotes the marking of place p. The value of a marking at time τ is denoted $\boldsymbol{m}(\tau)$. The restriction of $\boldsymbol{m}$ to p_c and p_d are symbolized with $\boldsymbol{m}^c$ and $\boldsymbol{m}^d$, respectively. An *FOHPN system* $\langle N, \boldsymbol{m}_0\rangle$ is an FOHPN with an initial marking $\boldsymbol{m}_0 = \boldsymbol{m}(\tau_0)$. The enabling of a discrete transition depends on the marking of all its input places, both discrete and continuous. A discrete transition t is *enabled* at $\boldsymbol{m}$ if for all $p \in {}^{\bullet}t$, $\boldsymbol{m}_p \geq Pre(p,t)$. An enabled discrete transition firing at $\boldsymbol{m}$ leads to marking $\boldsymbol{m}' = \boldsymbol{m} + C(\cdot,t)$. A continuous transition is enabled only by the marking of its input discrete places. The marking of its input continuous places, however, is used to distinguish between strongly and weakly enabling. A continuous transition t is *enabled* at $\boldsymbol{m}$ if for all $p \in {}^{(d)}t$, $m_p \geq Pre(p,t)$. An enabled transition $t \in T_c$ is *strongly enabled* at $\boldsymbol{m}$ if for all places $p \in {}^{(c)}t$, $m_p > 0$; while it is *weakly enabled* at $\boldsymbol{m}$ if for some $\overline{p} \in {}^{(c)}t$, $m_{\overline{p}} = 0$. The enabling state of a continuous transition defines its admissible *instantaneous firing speed* (IFS) v_i. The set of admissible IFS vectors can be characterized in linear algebraic terms by the following set of constraints [18]:

$$\begin{cases} V_i - v_i \geq 0 & \forall t_i \in T_\varepsilon(\boldsymbol{m}) \\ v_i - V_i^{'} \geq 0 & \forall t_i \in T_\varepsilon(\boldsymbol{m}) \\ v_i = 0 & \forall t_i \in T_N(\boldsymbol{m}) \\ \sum_{t_i \in T_\varepsilon} C(p,t_i) \cdot v_i \geq 0 & \forall p \in P_\varepsilon(\boldsymbol{m}) \end{cases}$$

where V_i is the MFS, $V_i^{'}$ is the mfs, $T_\varepsilon(\boldsymbol{m})$ ($T_N(\boldsymbol{m})$) is the subset of transitions enabled (not enabled) at marking $\boldsymbol{m}$, and $P_\varepsilon(\boldsymbol{m})$ is the set of continuous empty places at $\boldsymbol{m}$ .The FOHPN dynamics combines both time-driven and event-driven dynamics. Macro-events may be defined as the events that occur when: i) a discrete transition fires or the enabling/disabling of a continuous transition takes place; ii) a continuous place becomes empty; iii) a continuous place whose marking is increasing, reaches a flow level that enables a set of discrete transitions; iv) a continuous place whose marking is decreasing, reaches a flow level that disables a set of discrete transitions. Accordingly, the IFS vector during a macro-period (i.e., between two consecutive macro-events) keeps constant and the FOHPN dynamics can be described in the generic macro-period by a linear discrete-time state variable model where the system state collects the marking of all places and the values of all timers. Moreover, a performance index can be defined and the IFS vector can be computed solving, at each macro-period, a linear programming problem under the above speed constraints.

When the system is complex and it is necessary to model a huge amount of information, uncertain variables, logical expressions, and semantics, high-level PNs may be taken into account. The most popular are colored and fuzzy PNs. Colored PNs (CPNs) and Colored TPNs (CTPNs) allow representing different types of entities and the respective characteristics and logical constructs. The entities are represented by colored tokens that carry data values and can thus be distinguished from each other, in contrast to undistinguishable tokens of classical PNs. Each place may have an associated type or color set determining the kind of data that the place may contain.

Finally, Fuzzy PNs (FPNs) -that combine fuzzy logic with PNs- are a useful tool in dealing with uncertain and incomplete information. In FPNs fuzzy information can be enclosed using a possibility distribution in transitions, tokens, or places, and a fuzzy inference system can be implemented to model the events dynamics of the system.

2.1.2 Water transport

Water transport is historically the preferred mode to move goods over long distances. A review of the related scientific literature shows that PNs have been mainly used to solve port decision problems (Subsection 2.1.3.1), whereas few works consider PNs to solve decision problems on water transportation means (Subsection 2.1.3.2). Table 2-1 summarizes the contributions considered in this section, showing the reference number, addressed issues (i.e., simulation, analysis, and optimization/control), type of PN, and tackled applications.

A. PNs for Port Container Terminals Decision Problems

Port container terminals are characterized by highly sophisticated transportation activities, whose proper management is strictly related to the effectiveness of strategic/tactical and operational decisions. Here, it is classified and discussed the literature using PNs to solve the problems of each decision-making level [19], [20], [21], [22].

At strategic/tactical levels, PN models are used for resource planning and performance evaluation. More in detail, both logical PNs and PNs with time are used to efficiently select and dimension the resources of port container terminals (e.g., handling system, berth, and yard), eventually considering the prevention of blocking situations. In addition, deterministic or stochastic PN models are used to evaluate the performance of terminals (e.g., throughput and utilization level of resources), with a focus on the uncertainty of parameters in the latter case. A logical PN model is proposed in Liu et al. [23] to identify and reduce inefficiencies caused by an inappropriate management of berthing and container transfer operations in port container terminals. In particular, the authors use P/T nets for a structural analysis of the terminal, focusing on the occurrence of deadlocks that can lead to terminal malfunctioning. Deterministic PNs are used by Di Febbraro and Sacco [24] for optimally planning the type and number of resources, and to enhance the handling throughput of the terminal. The authors combine the deterministic TPN formalism with (max,+)-algebra, to modularly model the water transport system and evaluate the port behavior when considering different configurations. Although the authors present a customizable model of one of the fundamental cycle of the terminal (the ship to yard cycle), a significant additional contribution could consist in presenting a general model for each of the possible cycles, to build up the model of a generic port container terminal. Using stochastic PNs allows simplifying the resource planning and performance analysis of port container terminals with respect to deterministic models, as demonstrated by Li et al. in [25], Wang et al. in [26], and Zhang and Jang in [27]. More in detail, due to the stochastic nature of port services, stochastic PNs demonstrate to be effective in statistically evaluating the efficiency of container ports by allowing the measurement of their handling capacity [25], the utilization level of critical resources [26], and performing quantitative and qualitative analysis [27]. Yaxiong et al. [25] model via STPNs the entering/exiting cargoes as stochastic processes, considering the stochastic duration of the discrete events characterizing these inbound/outbound operations and then performing statistical evaluations. Wang et al. [26] use STPNs to establish a hierarchical model of the container terminal capacity and a dynamical model of subsystems. Simulations of the port container terminal by STPN models allow identifying bottlenecks by evaluating some performance indices that describe the level of utilization of the critical resources of the terminal port container. In the same direction, Zhang and Jang [27] propose a technique to avoid collisions in the operations of input and output of containers. The authors consider the employment of extended Generalized Stochastic PNs (GSPNs) to model container terminals. To cope with their complexity and implement a rule-based dynamic scheduling, the rules are integrated into GSPN models using an object-oriented approach. The extended GSPN model of the terminal allows both qualitative analyses (verification of deadlock-freeness, liveness, boundedness) and quantitative ones (determination of model dynamics, performance evaluation, dependability analysis). It would be interesting to consider advanced heuristic techniques to efficiently schedule the container transfers sequence. At the operational level, PN models are used to monitor performance and control the assignment of resources. The main activities at this level are those related to berths, the most critical resources in port terminals. These activities have been analyzed by different points of view with a correspondingly suitable type of logical PNs, PNs with time, and high-level PNs.

In particular, logical PNs are used to optimize the berth assignment in Gudelj et al. [19]. They propose P/T nets combined with genetic algorithms to model and simulate berthing and inter-terminal container transport. More recently, PNs with time are used as an alternative to the mathematical formulation of optimization problems.

TABLE 2-1 - Simulation, analysis, and control of water transport systems via PN models.

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*1	TACKLED APPLICATION*2
			PORT CONTAINER TERMINALS DECISION PROBLEMS		
[23]		Structural analysis		Place/Transition nets (**L**)	Modeling and analysis of port terminal workflow (**S/T**)
[24]	X	Sensitivity indices		Deterministic Timed PNs (**T**)	Planning the type and number of resources for seaport (**S/T**)
[23]	X	Handling capacity indices		Stochastic Timed PNs (**T**)	Handling capability of port terminal (**S/T**)
[26]	X	Capacity indices		Stochastic Timed PNs (**T**)	Management of port terminal capacity (**S/T**)
[27]	X	Structural analysis	Decision rules	Extended Generalized Stochastic PNs (**T**)	Collision avoidance in port terminal (**S/T**)
[19]	X		Genetic algorithms	Place/Transition nets (**L**)	Berth and crane assignment optimization (**O**)
[28] [29]			Precedence rules	Timed PNs (**T**)	Quay cranes operations optimization (**O**)
[30]			Mathematical programming	Colored PNs (**HL**)	Movement minimization in container stacking (**O**)
			WATER TRANSPORT MEANS DECISION PROBLEMS		
[31]			P-invariant method	Place/Transition nets (**L**)	Automatic traffic control of vessels (**O**)
[32]	X	Reachability graph	Closed loop model	Generalized Stochastic PNs (**T**)	Emergency control system for shipping (**O**)
[33]		Task time		Stochastic Timed PNs (**T**)	Optimization of ship maintenance (**O**)

Note: *1 L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS. *2 S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..

More in detail, Legato et al. [28] and Trunfio [29] tackle the problem of optimally scheduling the operations of a group of quay cranes at a maritime container terminal.

The goal is to minimize the overall vessel completion time. In this operational problem, TPN models are used as graphical optimization tool, where each sequence of tasks of the cranes is represented taking into account time and functioning constraints of the system (precedence, collision avoidance, and so on). Instead, high-level PNs are taken into account by Kefi et al. [30] for the quay management of fluvial ports and focusing on the minimization of the stacking time to quickly load ISO containers within a barge. They assimilate the container stacking optimization to the block world problem and propose a resolution that uses a method based on distributed artificial intelligence and interacting agents. In particular, the CPN formalism allows representing a negotiation procedure between agents while distinguishing the various containers.

B. PNs for Water Transport Means Decision Problems

In the context of water transport systems an emerging interest is growing towards decision problems related to safety and maintenance issues of water transportation means. Since these problems may be treated either logically or considering time, accordingly two main type of PN formalisms have been used: P/T nets for deadlock prevention and STPNs for emergency and maintenance processes management. It has to be noticed that no contribution is available for strategic/tactical decision problems. From a logical point of view, Kezic et al. [31] address the automatic traffic control of vessels in a marine canal traffic system. They propose a method for calculating the maximal permissive deadlock prevention controller, in order to regulate the traffic light system to avoid dangerous situations in case of vessels' irregular movement. The inclusion of time and stochastic data is considered by Zou et al. [32], who present an emergency control system for ships. The authors use GSPNs with finite capacity and inhibitor arcs to evaluate the performance of the system in terms of time, cost, and average utilization of resources. Moreover, the use of inhibitor arcs and synchronization structures allows solving deadlock situations. Finally, Li et al. [33] present a stochastic PN model with time focusing on the maintenance processes of ships that have an important impact on maintenance quality and schedule. For the purpose of controlling and optimizing practical maintenance processes, the paper analyzes the main characteristics of maintenance processes using STPNs, which allow evaluating the system performance under uncertainty.

2.1.3 Rail transport

Trains are one of the oldest means of transport, both for freight and passengers. They can provide an economically and ecologically sustainable service. For these reasons, railways are the most used transport mode for long-distance inland freight transport and the corresponding scientific contribution spans over a wide range of topics.

This subsection focuses on scientific contributions using PNs to minimize the waste of freight railway transport, with particular attention to the reduction of costs deriving from safety issues. More in detail, in this regard PNs have been used to define and solve decision problems concerning: collision detection, deadlock identification and avoidance, management of emergency situations, scheduling and routing of trains, traffic prediction, and evaluation of the impact of structural changes on railway traffic. To this aim, apart from some models employing P/T and timed PNs or some extension, authors mainly employ high-level PNs. Table 2-2 provides a concise description of the major contributions in this area, which are detailed in the sequel.

Strategic/tactical decision problems are mainly focused on guaranteeing a safe service, minimizing collisions and dangerous situations, by properly planning the allocation of resources, possibly at the lowest cost. PNs have proven to be useful when integrated in decision support systems for the investigation and verification of railway safety operations. In particular, PNs with time and high-level PNs are used in offline studies supporting the assessment of long-term decisions, design and test of on-board control systems and interlocking structures, prediction of traffic level, scheduling of trains, and avoidance of collisions. PNs with time are used by Rama and Andrews [12] to assess long-term decisions aiming at reducing costs and ensuring a safe service. The authors propose STPNs to evaluate the effects of long-term decisions on the lifecycle costs of a large-scale railway system. The modularity of PNs is here largely appreciated, due to the possibility of representing the granularity of the railway structure in a top-down approach. Furthermore, the possibility of performing statistical evaluation on the behavior of the network leads to the computation of costs and performance predictions over a long time horizon. This reveals to be essential for the provision of well-informed asset management decisions. Differently, high-level PNs are considered by Bjørk and Hagalisletto [34] and Hagalisletto et al. [35] to address trains' collisions. Authors present methods for the rapid construction of large-scale executable railway models, obtaining nets that are safe, permit collision detection, include time, and are sensitive to their surroundings. CPNs are used to provide reliable information about the network, and to understand the behavior of the system offline. In particular, the authors show that using CPNs one can simulate, monitor, and control a railway system so as to detect and avoid collision, forecast the traffic amount in the net, and support train scheduling. In addition, Daohua and Schnieder [36] propose CPNs for evaluation and test purposes. In particular, they perform a top-down scenario based evaluation of a satellite-based train control system. This is possible thanks to the modularity and high-level descriptive power of CPNs, which permits the inclusion (by means of labels) of domain specific knowledge necessary to implement a consistent and constraint adherent safety critical control system. A significant number of contributions consider the design of interlocking (coordination of points and signals at junctions) and signalization systems. In particular, to verify the safety of the railway yard, Durmuş et al. [37], [38] use automation PNs, which are extended PNs easily interfaceable with sensors and actuators. More in detail, automation PNs include inhibitor arcs, enabling arcs, firing conditions associated with transitions, and actions that may be assigned to places. The yard model is implemented into a PLC (Programmable Logic Controller) to test scenarios. Results show that modeling with automation PNs allows verifying the railway net safety. Furthermore, to ensure that the automation PN model of the system does not reach forbidden states or undesired situations, Durmuş et al. [37] provide also a PN based supervisor. Durmus et al. [38] and Yildirim et al. [39] extend the railway yard PN considering a level crossing and show how the model allows testing several possible failure situations. In the same context, Söylemez et al. [40] evaluate the use of automation PNs for the design of safety critical software. The authors show that PNs can be automatically generated from interlocking tables, so that they can be translated into the corresponding PLC code. Moreover, Durmuş et al. [41] present a PN representation of railway interlocking systems, satisfying the recommendations of related functional safety standards. The interlocking software is tested and verified by two simulators that convert automation PN models into PLC code. Finally, Vanit-Anunchai [41] [42] show that interlocking tables may be modeled using CPNs. At operational level, safety issues are regarded under a control and monitoring point of view.

Logical PNs, PNs with time, and high-level PNs are used to prevent deadlocks, provide control systems for safe railroad intersections, ensure robustness of track allocation, and manage traffic abnormal situation. Logical PNs are useful to simply model and impose control laws preventing collisions of trains. Giua and Seatzu [43] propose a local collision avoidance approach based on P/T nets and GMECs that enforce limitations on the weighted sum of markings in a subset of places, resulting in a PN supervisor that specifies a state feedback control law preventing the net from reaching forbidden markings. This contribution highlights the possibility of using logical PNs not only for strategic/tactical purposes as discussed for water transport, but also to avoid the occurrence of deadlocks at the operational level. Stochastic PNs with time are appreciated for real-time control of rail crossing at railroad intersections, thanks to the possibility of properly modeling the system and computing a suitable timing of traffic lights. In particular, Weng et al. [44] demonstrate the effectiveness of STPNs in implementing a parallel railroad level crossing control system to avoid critical scenarios (traffic jams) in the system. High-level PNs apply to real-time deadlock prevention, safety verification of the railway signaling system, and management of abnormal situations. Fanti et al. [45] present a CPN model of a railway system controlled through colored GMECs, to deal with the real-time traffic control for deadlock prevention. The authors use CPNs to model the dynamics of the railway network, while the prevention policy is expressed by a set of linear inequality constraints, i.e., the colored GMECs that are enforced by adding appropriate monitor places.

Based on the analysis of digraphs associated to CPNs, deadlock situations are characterized and a strategy is established to define offline a set of GMECs that prevent deadlock. CPNs are also used in the field of safe communication in the railway signaling system and for bottlenecks resolution. Guo et al. [47] use CPNs to model and verify the correctness and safety characteristics of a railway signal safety protocol. The authors implement and simulate single-channel and dual-channel communication models, based on the railway signal safety protocol, via CPNs. This simulation technique reveals to be useful in the detection of switch deadlocks in a dual channel system.

Table 2-2 - Simulation, analysis, and control of rail transport systems via PN models.

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*[1]	TACKLED APPLICATION*[2]
[11]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Lifecycle cost analysis (**S/T**)
[34]	X	Maude simulation	Control rules	Colored PNs (**HL**)	Collisions of trains (**S/T**)
[35]			Control PN submodel	Colored PNs (**HL**)	Collisions of trains (**S/T**)
[36]	X	State- space analysis and model-based testing		Colored PNs (**HL**)	Satellite-based rail control systems (**S/T**)
[37] [38]	X		PN model for control systems	Automation PNs (**HL**)	Interlocking and signalization systems design for railway yard (**S/T**)
[39]	X	Functional block diagram		Automation PNs (**HL**)	Failure investigation in railway yard with level crossing (**S/T**)
[40]	X			Place/Transition nets (**L**)	Interlocking tables (**S/T**)
[41]	X			Place/Transition nets (**L**)	Interlocking system with safety standards (**S/T**)
[46] [42]	X	State-space analysis	PN model of interlocking tables	Colored PNs (**HL**)	Railway interlocking systems (**S/T**)
[43]			Supervisory control with GMECs	Place/Transition nets (**L**)	Deadlock avoidance in railway systems (**O**)
[44]	X	Reachability graph	Control conditions and events	Stochastic Timed PNs (**T**)	Railway level crossing control system (**O**)
[45]		Analysis of digraphs	Supervisory control with GMECs	Colored PNs (**HL**)	Deadlock prevention on railway systems (**O**)
[47]	X	Simulation analysis	Monitoring module	Colored PNs (**HL**)	Safe communication in railway signaling system (**O**)
[48]	X	Simulation analysis		Timed Colored PNs (**HL**)	Track allocation robustness (**O**)
[49]	X	Scenario-based simulation analysis	Decision rules	Fuzzy PNs (**HL**)	Train dispatching in case of abnormal situations (**O**)

Note: *[1] **L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS.** *[2] **S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..**

Wenzheng et al. [48] tackle the track allocation robustness, which is seldom addressed in the literature, solving bottlenecks that may occur in track allocation schemes when trains in the net are delayed. The model of the track allocation schemes uses timed CPNs. The detection of bottlenecks is performed using the model and the indicators of the total trains' departure delay. In the context of other high-level PN formalisms, Cheng et al. [49] use a FPN approach to formulate the decision rules of train dispatchers in case of abnormality in the railway system functioning. The authors transform the train dispatchers' expertise into useful knowledge rules, to be used in abnormal situations, such as: centralized traffic control system failure, automatic train protection failure, and locomotive failure.

2.1.4 Road transport

Road transport is the most used inland mode for the pick-up and delivery of goods. It is typically used for the last mile transport, the most critical and expensive activity of supply chains. However, the literature mainly focuses on passenger transport and urban vehicular traffic [50], while only few works address freight transport via road means. Nonetheless, the resolution of decision problems connected to freight transport is not negligible, especially if it is considered the corresponding impact on company costs (fuel, maintenance, employees, and so on) and environment (pollution and energy demand). In particular, the main topics of interest of PNs for freight road transportation are: the logistic chain last mile, freight transportation via highways, and hazardous freight transportation, as reported in Table 2-3. The table also shows that typically high-level PNs are used for road transport decision problems, as detailed in the sequel.

A first subset of contributions is dedicated to using PNs at a strategic/tactical level. The analyzed works address problems related to the planning of the last mile freight transport and use both logical and high-level PNs for estimating routes, assigning vehicles to cargos, etc. Logical PNs are used by Qu et al. [51]to minimize the travel time and consequently the corresponding waste. The authors develop an algorithm based on P/T nets to estimate cargo route(s) with the least total travel time, based on finding the transition firing sequences that minimize the total travel time. In the same context, Aized and Srai prove the effectiveness of high-level PNs by developing in [52] a three-layered hierarchical CPN model for planning the last mile delivery region. The implementation of the hierarchical structure via PNs allows the reorganization of the system whether any change happens, thanks to the scalability and versatility of the PN tool. In addition, Franke and Dangelmaier [53] propose a preliminary work on a multi-agent system integrating CPNs with agents to dynamically simulate a road transportation system and plan vehicle routes and assignments to clients.

A second subset of contributions is dedicated to the use PNs at operational level for hazardous material (hazmat) transport and for long-distance road transport, typically motorways or freeways. More in detail, PNs with time and high-level PNs have been applied also to road transport systems for the special cases of hazardous material transport, and accidents of heavy means. Timed PNs are considered by Yuanchun et al. [54] to evaluate the response of emergency system in case of accidents involving hazmat. The technique uses STPNs and Markov chains to statistically evaluate performance. Instead, Centrone et al. [55] consider the modeling of hazmat transport on congested motorways, via timed CPNs. The model allows estimating in real time the risk of hazmat transport and supporting rescue decisions, taking into account the type of transported hazardous material, the traffic and the density of population living close to the motorway. An extension of this work is developed by Fanti et al. [56] who propose a decision support system to monitor hazmat vehicles. Furthermore, high-level PNs are used by

Kabashkin [57] who proposes the implementation of decision support systems based on evaluation PNs for the choice of alternative routes in a large-scale transportation transit system. A heuristic approach and a simulation tool are developed that allow the practical realization of the support tool in real environment. As regards long-distance road transport, due to the physical extension and complexity of the considered systems, high-level PNs have typically been proposed to allow compacting the transportation system representation. In particular, Julvez and Boel [58] propose a macroscopic model based on continuous PNs as a tool for designing control laws that improve the road transport system dynamics. Taking advantage of the network fluidization that leads to a concise continuous PN model, a model predictive control strategy is proposed that allows approximating the fundamental traffic diagram and considering various traffic conditions. Rather than using continuous PNs, Dotoli et al. [59], Dotoli [60] and Fanti et al. [61] consider only a partial fluidization of the system and employ first-order HPNs to model and control freeways. The model is modular and takes advantage of HPNs to represent traffic flows as continuous fluids and control signals and interruptions as discrete dynamics. Moreover, in [59] and [61] such a HPN formalism allows to mimic the fundamental traffic diagram and simulate and optimize the road transportation system, either by using an on-line optimal control coordination of speed limits [48] or by a ramp-metering control policy, to maximize the traffic flow [59]. Dotoli [61] uses a similar HPN model to control in real time freeways by route guidance, i.e., recommendation of alternative routes, to maximize the traffic flow. Demongodin [62] proposes the use of another HPN formalism, called generalized batched PNs, for modelling freeways and controlling their speed with a variable speed limit control policy that makes use of the defined PN formalism. In particular, generalized batched PNs extend the HPN class by defining the concept of batch, i.e., of a group of entities moving through a transfer zone at a certain speed, and the corresponding notion of batch mode. Moreover, batches may be controlled according to a switching dynamics between two behaviors: the so-called free behavior and the accumulator behavior. These concepts are straightforwardly applied to model and control freight logistics and transportation systems.

2.1.5 Air transport

Air transport is employed only for very valuable goods, due to high costs. Hence, the corresponding contributions are more limited than those on other modes.

Table 2-4 summarizes the literature and clearly shows that, due to the complexity of the air transport system, typically PNs with time are used, eventually with stochastic or colored features.

Stochastic PNs with time and high-level PNs are mainly used to solve air terminal strategic decision problems concerning airplanes delays propagation, cargo processing time, air traffic capability, handling capability, air traffic management, and taxiing operations. PNs with time are used both for the analysis of air traffic flows and for the management of air terminal resources. More in detail, airport strategic decision problems include the efficient management of the aircraft flows, taking into account costs induced by the aircraft arrival delays. In this context, Ding et al. [63] use TPNs, to investigate the evolution and the propagation of airplane delays in the airport. Thanks to the temporal features of TPNs, two different models allow the investigation of the propagation of airplane delays. Two types of airplanes groups are then studied: priority and non-priority airplanes. For both of them, authors calculate the average and total delay propagation. Consistent results are obtained in test simulations of the technique on some airport hubs. As regards the management of terminal resources, stochastic PNs with time find application in the work by Yang et al. [64], who analyze the handling part of the air cargo export, examining how strongly it can influence the cargo transport speed. They present a STPN model of the air cargo handling system and build the associated homogeneous Markov chains.

TABLE 2-3 - Simulation, analysis, and control of road transport systems via PN models.

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*1	TACKLED APPLICATION*2
[51]	X	Simulation analysis	Mathematical programming	Place/Transition nets (**L**)	Least cost travel routes estimation (**S/T**)
[52]	X	Simulation analysis		Colored PNs (**HL**)	Planning of the last mile transport (**S/T**)
[53]	X		PNs control model	Colored PNs (**HL**)	Planning vehicle routes and assignment to clients (**S/T**)
[54]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Hazmat road transportation accidents (**O**)
[55]	X	Simulation analysis		Timed Colored PNs (**HL**)	Hazmat road transport on congested motorways (**O**)
[56]	X	Simulation analysis		Colored PNs (**HL**)	Hazmat vehicles monitoring (**O**)
[57]	X		Heuristics	Evaluation PNs (**HL**)	Alternative routes in large-scale transportation systems (**O**)
[58]	X	Simulation analysis	Model predictive control	Continuous PNs (**HL**)	Model based feedback control of road networks (**O**)
[59]	X	Simulation analysis	Closed loop optimal control	First-Order Hybrid PNs (**HL**)	Freeway ramp metering control (**O**)
[60]	X	Simulation analysis	Closed loop optimal control	First-Order Hybrid PNs (**HL**)	Freeway route guidance (**O**)
[61]	X	Simulation analysis	Mathematical programming	First-Order Hybrid PNs (**HL**)	Freeway speed limit control (**O**)
[62]	X	Performance analysis	Decision algorithms	Generalized Batched PNs (**HL**)	Freeway speed limit control (**O**)

Note: *1 **L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS.** *2 **S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..**

The model allows analyzing the system handling capability based on the probability distribution of the Markov chain stable state, revealing the bottlenecks and providing useful insight regarding the involved resources dimensioning and use. It is possible to see a strong similarity between this work and that by Wang et al. [26] discussed in sub-section 2.1.3.1, both for aim and formalism (STPNs). Furthermore, in [65] a real airport is modelled and simulated by means of STPN and its traffic capability is evaluated. The work has a twofold aim: on the one hand to test the simulator ability in representing a large airport, and on the other hand to evaluate the airport traffic capability, in case it should receive traffic from two adjacent airports closed for severe weather conditions. On the other hand, high-level PNs give the opportunity to analyze in detail the terminal operations, by distinguishing the flows of different type of freights. In this respect, Lee et al. [66] develop a simulation model based on high-level PNs to analyze the terminal operations, focusing on the retrieval part of terminal operations. Due to the complex cargo processes, stochastic customer requests, and processing times, the formulation of an analytical model is impractical. Hence, the authors employ timed CPNs to model and simulate the terminal operations. Colors and time stamps are associated with tokens, with the purpose of distinguishing cargos that require distinct retrieval processes. The simulation model allows investigating the results of airline assignment and the automated storage and retrieval system policy proposed by the authors to minimize the cargo processing time index and several performance indices. High-level PNs also permit to plan air traffic allowing cooperation between airborne and ground side. In [67] a CPN model is implemented to simulate a potential future arrival planning process in air traffic control. It establishes a favorable sequence in which aircraft can be led to the runway. CPNs are used to generate and evaluate the potential solutions to the sequence planning problem. In the same direction, Werther et al. [68] use CPNs as a formal approach for the description of the whole human machine system in remote tower operation human machine interface. The authors use CPNs to conduct a formal analysis of the system, identifying critical states and inconsistencies.

Consequently, the represented formal work process model is a support for the communication between domain experts and system developers. At ground side, the management of the movement of airplanes, also called taxiing, is a critical and complex activity. In [69] Podgórski and Skorupski present a hierarchical CPN model to simulate the actual and projected air traffic for determining alternative taxi routes in case of congestion in aerodromes traffic, i.e. the so-called conflict points. The model allows finding the best alternative route and represents a promising base for the identification of the optimal global solution for the management of the ground traffic in the whole aerodrome. This section is concluded by recalling the work by Jamal and Zafar [70], who present mobile PNs, i.e., logical PN frameworks integrated with agent based modelling, in order to model and verify the main operations of an aircraft: takeoff, enroute, and landing.

TABLE 2-4 - Simulation, analysis, and control of air transport systems via PN models.

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*[1]	TACKLED APPLICATION*[2]
[63]	X	Statistical analysis		Timed PNs (**T**)	Delay propagation in airports (**S/T**)
[64]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Handling capability of export airport terminals (**S/T**)
[65]	X	Simulation analysis		Stochastic Timed PNs (**T**)	Air traffic capacity of airports (**S/T**)
[66]	X	Simulation analysis		Colored Timed PNs (**HL**)	Analysis of import airport terminal operations (**S/T**)
[67]	X	State space analysis		Colored PNs (**HL**)	Air traffic capacity of airports (**S/T**)
[68]	X	State space analysis		Colored PNs (**HL**)	Air traffic capacity of airports (**S/T**)
[69]	X	Simulation analysis		Colored PNs (**HL**)	Aerodrome taxiing management (**S/T**)
[70]	X			Agent Based Mobile PNs (**HL**)	Aircraft operations modeling and verification (**S/T**)

Note: *[1] L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS. *[2] S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..

2.1.6 Pipeline transport

Pipeline transport plays an important role in the oil and gas transportation system for its advantages in energy consumption, remote centralized management, and profit. Table 2-5 reports the related contributions and shows that logical PNs, PNs with time, and high-level PNs are used to tackle some of the related decision problems.

At strategic/tactical levels, logical PNs are considered to plan operating procedures, while PNs with time are used to evaluate the system performance, and high-level PNs find application in the risk evaluation context. More in detail, logic PNs are used by Chou and Chang [71] to propose a systematic strategy that allows the creation of detailed operating procedures necessary to clean any given pipeline network. A P/T net model is proposed to select the cleaning routes of all material-transfer paths. The operation steps are identified using the results of the system simulations of the PN model. The technique demonstrates to be effective in real systems. Differently, Lai et al. [72] use logic PNs to compute the optimal sequence of operating procedures necessary to transfer material in a pipeline network. The technique is flexible, allowing the definition of different objective functions that can be customized and permit the reduction of different types of waste. PNs with time are used by Xiong et al. [73] to analyze the performance of emergency response for oil and gas pipeline accidents by predetermined emergency plans. To this aim, STPNs are proposed as a modeling framework and the results obtained by the authors show that the average execution time of the STPN model can be used to evaluate the effectiveness of emergency responses. Furthermore, Gursesli and Desrochers [74] use PNs with time to analyze the interdependencies of the systems that compose power plants (i.e. power distribution, oil and natural gas production, pipeline networks for these products, water, and communication systems) and evaluate the effects of power disruption on these components. STPNs are used to represent the complex system and conduct an analytical inspection of the interdependencies. This can conduct to the identification of recovery strategies whose efficiency can be analyzed by means of structural analysis. In [75] Ren et al. compare various formalisms for the performance evaluation of pipeline networks. The authors show that STPNs are a typical tool to model nonlinear asynchronous pipelines with choices, fundamental in design planning. The modeling methods reduce to linear programming, which can be very time-consuming when the design is large. Finally, high-level PNs are proposed by Guo et al. [76] for a risk evaluation in long-distance oil and gas transportation pipelines. The method is based on a FPN model that admirably describes the unknown likelihood, the relationships of most of the input events for the accident, and the dynamic changes of the system.

The decision problems at operational level are solved using high-level PNs and mainly regard the short-term scheduling of operations in refineries and the computation of the shortest path in pipeline networks. More in detail, Wu et al. [77] use high-level PNs for the first problem. PNs allow overcoming the limits of mathematical programming in finding a feasible solution. In particular, the contribution presents a hybrid CTPN for the schedulability analysis and the short-term schedule definition. The PN framework allows the modular representation of a refinery distinguishing the multiple types of crude oil circulating in the system. The short-term scheduling problem is solved hierarchically: at the upper level the schedule is determined and at the lower level it is refined. For the same problem, Wu et al. [78] propose a heuristics to test the realizability for a given target refining schedule and in [79] and [80] they include in the scheduling problem the constraints regarding the setup cost of high fusion point oil transport, the cost of tank charging and discharging, the residency time and charging tank-switch-overlap. Moreover, Wu et al. [81] use hybrid CPNs also to solve the scheduling problem for refineries with two pipelines, each one devoted to the transportation of a different type of oil. Finally, in [82] the authors, after obtaining schedulability conditions, decompose the problem by decoupling continuous and discrete variables and solve each sub-problem hierarchically. For sub-problems with continuous variables linear programming-based techniques are used, while for those with discrete variables heuristics is applied. Thus, the problem can be efficiently solved and the approach is applicable to solve real-life problems. On the other hand, Kadri and Zouari in [83] and in [84] address the problem of determining dynamic shortest path in oil pipeline networks where the reliability condition can vary with time and environment.

TABLE 2-5 - Simulation, analysis, and control of pipeline transport systems via PN models.

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*1	TACKLED APPLICATION*2
[71]	X	Reachability graph		Place/Transition PNs (**L**)	Planning of cleaning operations in pipelines (**S/T**)
[72]	X		Mathematical programming	Place/Transition PNs (**L**)	Computation of optimal route for transfer of material (**S/T**)
[73]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Evaluation of the effectiveness of emergency plans (**S/T**)
[74]	X	Invariants analysis		Stochastic Timed PNs (**T**)	Interdependency analysis in complex systems (**S/T**)
[75]	X	Simulation analysis		Stochastic Timed PNs (**T**)	Performance evaluation of asynchronous pipelines (**S/T**)
[76]	X	Fuzzy reasoning algorithm		Fuzzy PNs (**HL**)	Risk assessment of pipelines (**S/T**)
[77]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Short-term scheduling of operations in crude oil refineries (**O**)
[78]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Refining short-term scheduling of operations in crude oil refineries (**O**)
[79]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Refining short-term scheduling of operations in crude oil refineries (**O**)
[80]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Refining short-term scheduling of operations in crude oil refineries (**O**)
[81]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Refining short-term scheduling of operations in crude oil refineries (**O**)
[82]	X	Safeness analysis	Heuristics	Hybrid Colored PNs (**HL**)	Refining short-term scheduling of operations in crude oil refineries (**O**)
[83] [84]	X	Simulation analysis	Dijkstra's algorithm	Colored PNs (**HL**)	Shortest path search in dynamic reliability space (**O**)

Note: *1 **L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS.** *2 **S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..**

The contribution determines the value of the parameters necessary to calculate the dynamic reliability and then, applying the Dijkstra's algorithm, determines the most reliable path based on a CPN model.

2.1.7 Multimodal and intermodal transport

The evolution of transport has led to the extensive use of combined transport modes, namely, multimodal and intermodal transport. This modern transport framework allows to take advantage of the best characteristics of each mode, i.e., to combine the speed, security, reliability, and sustainability of rail/sea modes for long distance transport, with the space penetration features of road transportation [85], [86], [87].

The related literature on PNs focuses mainly on problems concerning multimodal and intermodal terminals, while few works address the use of PNs for the whole transportation chain. This is due to the complexity of the transportation system and includes decision problems regarding various transportation means. On the other hand, the management of terminals is also complex and greatly influences the performance of the entire transportation system. As discussed in the following subsections, available contributions applying PNs to freight multimodal and intermodal transport typically employ PNs with time, i.e., deterministic or stochastic, and high-level PNs (see Table 2-6Table 2-6 and Table 2-7).

A. PNs for Multi/Intermodal Terminals Decision Problems

The management of multimodal or intermodal container terminals should mainly guarantee fast transshipment between modes and optimized container stacking, so as to contribute to the minimization of the freight shipping time in the whole logistic chain. For this reason, multi/intermodal terminals are usually provided with modern handling equipment, advanced transportation systems, and up-to-date ICT tools whose planning and control concur to the terminal efficiency [85]. In the literature, great attention is given to the minimization of the residence time spent by containers in the terminal and PNs contributions have analyzed, both at strategic/tactical and operational level, different aspects of the problem.

The decision problems of strategic/tactical levels include planning of resources allocation, analysis of terminal operations, and design of the yard area. The contributions propose the use of PNs with time to model and analyze the system, obtaining predictions of the system behavior and evaluating alternative policies to solve the above problems. More in detail, PNs with time are mainly considered for the representation and evaluation of the evolution of terminals. The obtained information are then used to reduce the delays in the transshipment process. Filipova et al. [88] highlight the importance of reducing the permanence of the cargo in the intermodal transport terminal. They identify and describe by TPNs the events that in a water-road terminal can interrupt the freight flow between customers, with the purpose of decreasing the delays caused in the transportation chain by these events. Nevertheless, no control policy is suggested to avoid such events. Later on, Di Febbraro et al. [89] cope with the management of an automated system called Metrocargo system designed for the handling of containers by trains in an intermodal terminal. They model the system by means of TPNs with the purpose of creating an effective tool for IFTT performance evaluation. First attempts to take advantage of stochastic PNs with time for analysis and design of intermodal freight terminals are in [90] by Fischer and Kemper. The authors suggest the use of STPNs as an intermediate language to combine modeling of logistic systems with existing analysis methods. In particular, they provide a steady state analysis of the terminal, which reveals to be useful for the appropriate dimensioning of a waiting space dedicated to the trucks involved in the intermodal chain. More recently, Maione et al. [91] and Maione et. al. [92] provide a contribution to define a complete modular model of container handling processes in container terminals, which can be used for simulation, test, monitor, and control purposes. In the first paper, a GSPN formalizes the sequence of operations, each involving the synchronization of resources, for the freight transshipment process. In the latter one, a GSPN model is used to micro-simulate key human operators' activities. In both cases the authors use GSPNs for the statistical analysis of the behavior of the system. It is evident that the framework also appears promising as a helpful tool for real-time control purposes. Moreover, Cavone et al. [93], one of the works developed on Petri nets for intermodal transportation, STPNs were used to tackle the strategic planning of number, capacity and frequency of resources in an intermodal terminal. Finally, Dotoli et al. [13] focus on modeling and performance evaluation of an intermodal freight terminal using stochastic PNs with time. The model allows estimating the terminal's performance by suitable performance indices and identify criticalities and bottlenecks. Furthermore, it allows evaluating different solutions to the recognized criticalities under alternative scenarios (e.g., when inflow traffic increases and congestion may occur).

The decision problems of the operational level mainly regard the real-time control of the activities of the terminal, while optimizing some performance indices. The literature contributions highlight that logical PNs, PNs with time, and high-level PNs are particularly suitable to this purpose thanks to their time features, the existing evaluation indices, and the control frameworks available. At first, operational level problems are discussed by mentioning a preliminary work by Kabashkin [94], presenting the structure of a decision support system for regional transit multimodal transport that allows choosing alternative multimodal routes using PN based simulation. The proposed formalism is that of evaluation PNs, an extension of Place/Transition nets. On the other hand, in the context of control of handling resources, Degano and Di Febbraro in [95], [96], [11] and Degano and Pellegrino in [97] develop in consecutive steps a PN model of a partially automated material transportation system inside an intermodal container terminal. The authors aim at optimizing the resource allocation, while synchronizing the handling operations and reducing the time spent for the transport inside of the terminal. When an unpredictable event interferes with the nominal scheduled behavior of the intermodal terminal, this can cause a delay in its activities. The authors use TPNs to predict the delay occurrence by monitoring the firing times of the transitions of the model and communicating any missed firing to a regulation module. If a missing firing is detected, various policies are proposed to minimize the propagation of the delay. Dotoli et al. [85] use stochastic PNs with time to evaluate the impact of ICT tools on the performance of intermodal systems. Using a STPN model, the authors show that the communication between strategic parts of the system can improve the transportation process in case of congestion.

TABLE 2-6 - Simulation, analysis, and control of multi/intermodal transport systems via PN models (terminals decision problems).

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*1	TACKLED APPLICATION*2
[88]		State equation resolution		Timed PNs (**T**)	Delay reduction in water/road terminals (**S/T**)
[89]				Timed PNs (**T**)	Container handling with Metrocargo system in rail/road terminal (**S/T**)
[90]	X	Steady state analysis		Stochastic Timed PNs (**T**)	Dimensioning of trucks waiting area (**S/T**)
[91] [92]	X	Statistical analysis		Generalized Stochastic PNs (**T**)	Container handling with physical and human resources in sea/rail/road terminals (**S/T**)
[93]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Planning of resources in intermodal terminals (**S/T**)
[98]	X	Statistical analysis		Stochastic Timed PNs (**T**)	Efficiency evaluation of intermodal terminals (**S/T**)
[94]	X			Evaluation PNs (**L**)	Route generation for multimodal transport system (**O**)
[95] [96] [11] [97]	X		Agent-based control algorithm	Timed PNs (**T**)	Optimization and control of containers handling resources in a sea/rail/road terminal (**O**)
[85]	X	Statistical analysis	Supervisory control with GMECs	Stochastic Timed PNs (**T**)	Evaluation of ICT impact on intermodal terminals (**O**)
[99]	X	Statistical analysis	Control rules with predicates and assertions	PNs with predicates (**HL**)	Reliability analysis of intermodal terminals (**O**)
[100]	X	Simulation analysis	Mathematical programming	First-Order Hybrid PNs (**HL**)	Optimization of intermodal terminal activities (**O**)
[101]	X	Mathematical techniques	Fuzzy logic	Hybrid PNs with fuzzy logic (**HL**)	Modeling of multimodal transport system and exchange times optimization (**O**)

Note: *1 L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS. *2 S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..

The authors show that GMECs allow the straightforward modeling and simulation of such a communication subsystem in the STPN framework, providing an effective control of the system and resolution of congestions. Also high-level PNs are used for performance evaluation and control purposes, as shown in the research developed by Silva et al.[99]. They use PNs with predicates combined with Monte Carlo simulation for the performance evaluation of intermodal terminals at an operational level with particular attention to reliability analysis. PNs with predicates allow a clear graphical representation for highly complex systems, like intermodal container terminals are. In particular, the inclusion of predicates (i.e., expressions) and assumptions (i.e., equations) combined with the duplication of some fundamental places (i.e., repeated places) leads to a reduction in the number of connections and then to an increase in the readability of the model. Furthermore, predicates and assumptions allow the representation of supervisory control rules when addressing performance optimization problems. The authors apply the procedure to a real case study and prove that a low computational time is needed also for long prediction windows (e.g., 5 years for the analyzed case-study). In addition, in [100] me and my colleagues, we have shown how first-order HPNs can be efficiently used to model and manage both offline and in real-time intermodal freight transport terminals. As discussed in Section IX, the proposed formalism enables the terminal decision maker to choose the speeds associated with continuous transitions to optimize the terminal performance by two alternative control policies: the maximization of the container flows and the minimization of the stored containers. The proposed results are validated on a real terminal in southern Italy. The conclusive part of this section regards the contribution of Mahi et al. [101] who apply HPNs with fuzzy logic to model, analyze, and control a multimodal transportation system. Although the recalled work is intended mainly for passengers' transfer, it may be straightforwardly applied to freight logistics and transportation systems. The authors propose to employ HPNs to model the exchange of flows between different transportation modes and evaluate the IFTT performance (connection feasibility, waiting time, etc.) and fuzzy logic to control the system evolution while minimizing the connection time for a transfer between modes.

2.1.8 PNs for Terminal Handling Resources Decision Problems

This sub-section discusses papers that use PNs to deal with problems related to automated resources, typically Automatic Guided Vehicles (AGVs). These are often used in terminals for the efficient handling of containers especially when it is not possible to extend the terminal and the allocation of the existing storage space is thus optimized. In particular, it is summarized the literature related with operational level decision problems, such as: collision avoidance, deadlock avoidance, dynamic vehicle routing, dynamic dispatching, fault detection and identification, and activity scheduling (see Table 2-7). The main works are presented where the aim is to avoid collisions, taking advantage of the liveness property of PNs and imposing appropriate control rules. The contributions consider the application of logical PNs, PNs with time, and high-level PNs, which alone or combined with mathematical programming or control algorithms allow to easily represent complex control rules. Logical PNs features are considered for the implementation of fault tolerant and deadlock free systems. Yan & Li [102] consider the multiple faults' detection and identification in PNs that have state machine structures. In particular, the authors propose the design of the net controller leading to a fault tolerant AGV system. Finally, Gudelj et al. [103] focus on the scheduling of AGVs activities, inside a seaport container terminal. They provide a technique to find an optimal conflict and deadlock free schedule in container terminal systems, based on the combination of genetic algorithms and a PN structural analysis procedure. In particular, for the modeling of the multiclass re-entrant flowline system, PNs are used to formulate the problem in algebraic terms, leading to a substantial limitation of the problem formulation complexity. PNs with time are considered by Liu and Ioannou [104] for collision avoidance in automated container terminals using AGVs. The authors present a modular TPN model of the system. They build up small and simple PNs for each part of the automated terminal yard and prove the properties of liveness, safeness, and reversibility. Then, they integrate the subnets into a unique model, without losing the properties of liveness, safety, and reversibility. The AGVs' collision avoidance is then guaranteed by a first-come, first-pass control rule, which is modelled using inhibitor and actuator arcs embedded in the sub-models. PNs with time are also used for the optimization of dynamic dispatching and routing in bi-directional AGV systems by Nishi et al. [105] and Nishi and Tanaka [106]. They consider a PN decomposition approach and the problem consists in finding an optimal transition firing sequence for a TPN model assuring that no blocking situation occurs.

TABLE 2-7 - Simulation, analysis, and control of multi/intermodal transport systems via PN models (handling resources decision problems).

REF.	SIMULATION	ANALYSIS	OPTIMIZATION/ CONTROL	PN TYPE*1	TACKLED APPLICATION*2
[102]			PN controllers	Place/Transition nets (**L**)	Multiple AGV fault detection and identification (**O**)
[103]	X	Structural analysis	Genetic algorithms	Place/Transition nets (**L**)	Scheduling of AGVs (**O**)
[104]	X	Structural analysis	First-come first-pass control rule	Timed PNs (**T**)	AGV collision avoidance (**O**)
[105] [106]	X	Statistical analysis	Mathematical programming	Timed PNs (**T**)	AGV dynamic dispatching and routing (**O**)
[107]	X		Control rules	Colored Timed PNs (**HL**)	Deadlock and collision avoidance in zone controlled AGVs (**O**)
[108]	X		Control policy	Colored Resource-Oriented PNs (**HL**)	AGV collision avoidance (**O**)
[109]	X		Control policy	Colored Resource-Oriented PNs (**HL**)	AGV collision avoidance (**O**)
[110]	X		Supervisory control	Colored Resource-Oriented PNs (**HL**)	AGV routing with minimum travel time and collision avoidance (**O**)
[111]	X	Structural analysis	Control rules	Colored PNs with undirected arcs and directed tokens (**HL**)	Dynamic AGVs routing (**O**)
[112]		Structural analysis	Logical expressions	Colored PNs (**HL**)	Safety of open-path multi agent AGV system (**O**)

Note: *1 L=LOGIC PNS; T= PNS WITH TIME; HL=HIGH-LEVEL PNS. *2 S/T=STRATEGIC/TACTICAL LEVEL PROB.; O=OPERATIONAL LEVEL PROB..

The authors chose to decompose the TPN model to reduce the optimization problem computational complexity. The same approach is used both for single and bi-objective functions' optimizations. High-level PNs are also adopted for the resolution of the collision avoidance problem, allowing the implementation of more complex control rules. Dotoli and Fanti [107] present a control strategy to avoid deadlock and collisions in zone controlled AGVs system. The control scheme manages the assignment of new paths and the acquisition of the next zone in a bidirectional network. The AGVs system structure is modeled and analyzed via timed CPNs so as to simply implement the control strategy, which works on the basis of the knowledge of the system state. A similar approach is proposed by Wu and Zhou [108], who tackle the collision avoidance by the zone control policy and implement it by resource-oriented CPNs. Furthermore, Wu and Zhou in [109] develop colored resource-oriented PNs to model AGV systems and create a control policy to avoid deadlock and conflicts, while in [110] they use the same type of PNs to impose a supervisory control that avoids deadlock and blocking with the goal of minimizing the AGV traveling time. Roszkowska [111] proposes the modeling of AGVs using a CPN with undirected arcs and directed tokens, so as to reduce the number of the net components and simplify the insight into the model. Besides the use of zone control for collision avoidance, the author focuses on dynamic vehicle routing, taking advantage of marking liveness. In addition, Giglio [112] uses CPNs to model an open-path multi AGV system. The CPN model is able to represent the behavior of a variable number of AGVs which freely travel on the system layout, and the use of colors, guards, and arc expressions allows assuring the safety requirements of the AGV system.

2.1.9 Discussion and open issues

The literature review clearly shows that PNs have been employed in the logistics and freight transport context as regards fundamentally two research macro-areas directly related to strategic/tactical and operational decision problems: (1) **resource planning and performance evaluation** and (2) **monitoring and control of resources**, especially as regards safety/liveness assurance. Indeed, the use of PN models in both research areas is motivated by some properties that all PN formalisms have in common, namely: modularity, scalability, graphical aspect, and mathematical formalization. In the sequel conclusions are drawn regarding each freight transportation mode and outline new future research lines.

Water transport: For water transport, PNs have been taken into account in both research areas (1) and (2). Considering resource planning and performance analysis, Place/Transition nets can be effectively used to represent workflows in port container terminals and strategically evaluate the efficiency of the system according to appropriate performance indices (obviously not taking time into account). Place/Transition nets can be also useful for deadlock analysis and prevention, using structural analysis techniques. However, Place/Transition nets do not include events temporization. This may lead to overestimating the number of deadlocks. Moreover, the absence in this formalization of timing information makes the framework unsuitable for simulation purposes. Instead, TPNs and STPNs overcome this limitation and offer the possibility to perform realistic simulations of the system evolution taking into account deterministic or stochastic events. With TPNs and STPNs it is possible to define performance indices that take into account time and identify eventual critical resources that may generate bottlenecks in port container terminals. On the other hand, TPNs, STPNs, GSPNs, and CPNs are appreciated for monitoring and control purposes as an alternative to mathematical programming, since they can provide a simpler representation and resolution of complex optimization problems, that otherwise may require non trivial mathematical formulations. Concerning resource control, the main contributions regard resource allocation and container stacking optimization, collision avoidance and emergency management, and consider GSPN models for their descriptive power and simple control rules implementation. However, PNs have not yet been taken into account to manage emergency situations in real time.

Railway transport: The complexity of railways has led authors to deploy PNs' scalability and effectiveness to solve safety/liveness assurance problems, both at strategic/tactical and operational levels. Almost all works focus on high-level PNs, such as colored and fuzzy PNs. The CPN approach is particularly appreciated for its descriptive power: places, arcs, and transitions may be mathematical functions of several parameters and allow a simpler representation of the system with respect to analytical models. In the railway transport context, CPNs allow offline simulations that may be used for bottlenecks detection. Furthermore, CPNs allow solving computationally complex problems in a relatively short time. This makes them a valuable tool for real time control purposes. In particular, by associating to the system model appropriate GMECs, it is possible to control rail traffic in real time and prevent deadlocks. Nevertheless, CPNs are not suitable for knowledge based control systems: in such cases FPNs can support modelers in representing knowledge based rules to control rail traffic in case of abnormal behavior.

Road transport: In the context of freight road transport, PNs are considered both for resource planning and performance analysis, and for monitoring and control purposes in safety/liveness assurance applications. The number of related works using PNs mainly offers contribution for the last mile decision problem and for the modeling and control of freeways, with some application to hazmat road transport. The high variability of transportation routes and high quantity of unexpected events from which the transportation framework can be affected make the employment of logical PNs and discrete PNs with time not pursuable. Vice versa, the use of STPNs, CPNs, and HPNs allows the correct representation of road transport of goods. The first type of PNs is used for the evaluation of the operational performance of emergency systems in case of hazmat road transport accidents. The second instead, thanks to its scalability, allows the route planning process in the last mile region, offering the possibility to update the net structure at each change of the system, and to manage or prevent accidents and risks. It is worth noting that both PN formalisms have not yet been employed for real time applications, like rerouting in case of congested roads or delivery re-planning in case of unexpected events. Finally, HPNs allow modelling and controlling both off-line and in real-time road transportation systems, but their application has been proposed mainly for freeways only.

Air transport: Contributions for air transportation systems take advantage of the PNs simplicity and modularity to deal with resource planning and particularly safety/liveness assurance problems. Indeed, PN models are much easier to learn and use than linear programming, dynamic programming, stochastic models or other techniques like genetic algorithms. The use of PNs for safety applications assures easy comprehension of the model, easy identification, and elimination of errors, and extension of the model to improve its functionalities. TPNs and STPNs are particularly appreciated for performance evaluation or structural analysis. It is shown that these formalisms can provide sufficiently accurate results for these purposes. The use of high-level PNs like timed CPNs is an alternative to more complex formalizations and provide reliable simulations and estimations of the terminal behavior and performance. However, PNs have not yet been used in this area to solve monitoring and control problems.

Pipelines transport: Logical PNs, PNs with time, and high-level PNs are all used to solve problems of resource planning arising in the pipeline transport. Particularly, logical PN models are considered to plan operative procedures necessary to transfer materials in the network. This is possible by PNs structural analysis and combining PNs with mathematical programming techniques. On the other hand, the inclusion of time in PN frameworks is considered for performance analysis. Particularly, STPNs are used to evaluate in advance the effectiveness of emergency plans and analyze the behavior of the system in case of critical situations. The risk evaluation is conducted also with high-level PNs (FPNs) that properly capture the evolution and dynamics of these systems. For monitoring and control problems only high-level PNs have been taken into account. In more detail, CPNs and hybrid CTPNs allow the computation of shortest paths in the pipeline transport overcoming the limits of mathematical programming in finding solution for these extremely complex systems.

Multimodal and intermodal transport: The decision-making problems of these types of transportation techniques are tackled with PNs especially in the resource planning and performance analysis research area, where PNs are widely appreciated for their scalability and modularity. The complexity of a multimodal terminal framework and the huge number of sub-activities that compose its workflows lead to the definition of sub-models to be composed and controlled to provide an efficient and effective freight transport service. The TPN and STPN formalisms, beyond offering the already discussed scalability and modularity, can be used for prediction of faulty behaviors and real time monitoring, allowing the implementation of decentralized control by means of a multilayer structure, composed by local controllers and a central supervisor. In particular, deterministic TPNs can be conveniently employed for dynamic dispatching and routing of AGVs. For example, it is possible to minimize a given objective function to find out an optimal firing sequence. In such cases, the modularity of PNs is a key factor to simplify the optimization problem and to avoid the typical state-space explosion, because it allows decomposing the main problem into sub-problems whose resolution leads to feasible sub-optimal solutions. Clearly, the use of STPNs enables to consider parameters' uncertainty in the evaluation of the transportation system performance. Conversely, the class of high-level PNs clearly encompasses a number of frameworks which may be effectively used to model, simulate, analyze and control these complex systems. However, their employment has not yet been taken into full account, except from some examples in the class of hybrid PNs, which allow a continuous and discrete modelling of terminals' activities, used in combination with mathematical programming to optimize the terminal performance off-line or in real time. Finally, CPNs are also used for paths assignment avoiding deadlock and conflicts thanks to their high descriptive power. It is of crucial importance the development of more efficient simulation software in order to avoid the execution of a custom code directly connected to the real system setting.

Some other general future research indications may be given, irrespective of the transportation mode considered in the peculiar transportation system at hand. First of all, extending the performance evaluation of the freight transportation system under uncertainty may be considered. Not only stochastic models can be adopted, which require the availability of historical collected data, but also the application of the numerous available FPN formalisms may be fully investigated, which allow modeling parameters' uncertainty by expert evaluations. Moreover, addressing the scalability of PN models of freight transportation systems needs to be better explored. Batch models, decomposable models, and continuous or hybrid (discrete and continuous) models need to be fully investigated. Thanks to fluidization, these formalisms allow a more concise representation of (some of) the system entities compared with that of microscopic discrete models. Moreover, hybrid and continuous PNs can be efficiently used for control purposes, aiming at optimizing the performance of freight transport systems. Until now, the potentials of continuous and hybrid PNs are mainly considered for traffic systems, where the traffic stream is best described by the continuous part and traffic signal control is described by the discrete part. The use of continuous and hybrid PNs therefore clearly represents an opportunity to investigate new research ways in the areas of transportation systems modeling, simulation, analysis, and control.

Finally, the integration of PNs with other tools needs to be considered for the effective modeling and control of freight logistics and transportation systems. Studying the integration of PNs formalisms with extensively employed optimization tools such as genetic algorithms need to be fully explored to guide the system optimization towards global optima. Moreover, the integration of PNs with distributed agent-based models needs to be better investigated, in order to effectively and concisely model cooperation and negotiation mechanisms in freight transportation systems. Finally, the use of PNs for fault forecasting, prevention, and identification in freight logistics and transportation systems is an emerging issue.

2.2 Discrete event MILP modeling of railway systems

The introductive section of this chapter presented the concept of smart transportation system and highlighted the importance of combining innovative technologies with traditional transportation systems to achieve a higher efficiency level, reduce environmental impacts and pollution, and provide a more efficient service. In the context of railway transportation, managing railway traffic consists in facing problems both at a strategic and at an operational level, where the former are mainly connected to offline activities [113], while the latter refer to real-time supervision and control [43]. As discussed in Section 2.1.4 Petri nets are largely used to tackle railway safety problems, however there is a lack of contributions using PNs for one of the main operational level decision problems, i.e., the real-time rescheduling. This section aims at proposing an alternative discrete event systems modeling technique which can be suitably used in the context of smart transportation systems for the resolution of the real-time rescheduling problem.

2.2.1 MILP models for railway traffic rescheduling

Basically, train rescheduling consists in retiming the offline scheduled traffic (i.e., the nominal timetable) so as to minimize undesired effects on the railway service (e.g., train delays, customer discomfort, energy consumption). Typically, unpredictable events that may occur are distinguished into disturbances and disruptions and both cause the nominal timetable to become invalid because at least one train deviates from its original schedule. Disturbances are relatively small perturbations and their effects are limited. Examples are signal malfunctions on a track section that lead to temporarily decreasing the maximum allowed train speed, as well as a no-show of staff that causes a delayed train departure. On the contrary, disruptions regard large and particularly damaging external accidents leading to the cancellation of a number of trips in the scheduled timetable, e.g., in case of trains' breakdowns and tracks' blockings [114].

Train Dispatchers (TDs), despite the problem complexity, still manage railway traffic mostly manually, so that their actions often lead to sub-optimal solutions [113], [114], [115],[116]. The two main drivers in developing a smart railway traffic control and searching for an appropriate modeling formalism are the computational complexity of the rescheduling problem and the need for short resolution times, because new schedules should not be outdated by the time they are produced. In particular, train rescheduling consists in rewriting trains timetables, actuating retiming, and reordering trains, while assuring that each train reaches its final destination without changing its nominal predefined path. Unexpected events can cause passengers' discomfort, which may be reduced by applying a rescheduling technique [117].

Three main classes of computer-based rescheduling approaches can be broadly identified [117], [118]: simulation models, heuristic procedures, and mathematical optimization models. Simulation models aim at reproducing the flow of real life, that is, a decision making process which takes place "here and now" [119], [120], and in which there is no specific objective function. Heuristic procedures take decisions that aim at decreasing some performance indicator, such as delays, conflicts, and so on (see, e.g., [117], [121], [122], [123], [124], [125], [126],[127]). Usually, on the one hand little quantitative information about the solution quality is provided, and on the other hand the computational effort in implementing these heuristics is low. Finally, mathematical optimization models have instead a well-defined objective function, which frequently refers to average or maximum delays [128], [129], total delays (i.e., considering the delay at the final destination of trains) [130] or delays at stations along the train trip. Other approaches minimize the weighted travel time for passengers [126], the deviation from the nominal schedule [131], [132], the time to recover operations and fall back to the original plan [133], [134], running cost, or spent energy [135], [136]. An exhaustive discussion on commonly used rescheduling mathematical models can be found in [137], showing that rescheduling techniques based on mathematical models can provide optimal solutions. However, their implementation and resolution is not trivial, especially when the number of variables and constraints of the considered problem is high and the rescheduling time horizon is wide. Nonetheless, effective solvers (e.g., CPLEX, GUROBI, SCIP, GLPK) may allow the resolution of such optimization problems in reasonable computation times, that is, adequate for real-time rescheduling techniques. Typically, the train rescheduling problem statement is based on integer programming, MILP, linear programming, or nonlinear programming [37]. The most common framework to reschedule railway traffic is MILP, which has been extensively tested for the resolution of optimization problems and for which a large number of effective solvers has been developed. In particular, a railway network can be modeled as a discrete event system in which a certain number of jobs have to be executed with a limited number of resources. The train runs and dwell periods are the jobs of the system while the structural features of the network as well as operators are the limited resources.

TABLE 2-8. – Summary of the Contributions on Rescheduling based on MILP Models

REFERENCE	OBJECTIVE	SOLVING APPROACH
[126] [130]	1.Minimization of the total final delay 2.Minimization of the total cost associated with delays	Heuristic approach
[138]	1.Minimization of the total delay 2. Minimization of the total cost	CPLEX, Tabu search, Simulated annealing.
[139] [140] [141] [114]	1.Minimization of the total delay 2.Minimization of the sum of the final delay of all trains	Greedy Algorithm
[142]	Minimization of the total rescheduling cost	Statistical analysis of Propagation of Incidents
[117]	Maximization of the number of transported passengers	Heuristic approach
[143]	Minimization of the deviation between nominal and rescheduled timetable	Heuristic approach
[144]	Minimization of the sum of trains' arrivals	Branch and Bound
[145] [146]	Minimization of the sum of all predicted delays and the penalty for all broken connections and switched train orders	Genetic Algorithms Permutation Based Algorithm and MILP Solvers
[147]	Minimization of the delay cost and the number of seriously impacted trains	CPLEX
[148]	Minimization of the total unexpected waiting time of all passengers within the relevant network and time period	Heuristic local rule-based dispatching strategies
[149]	Minimization of the total accumulated delays	CPLEX with branch and bound for linear programming; Evolutionary Algorithm
[134]	Minimization of the total delay and deviance	Greedy heuristic approach with CPLEX; Fixing with CPLEX
[150]	Minimization of the total costs	Decomposition Algorithm
[151]	Minimization of the total deviation	Branch and Price
[152]	Minimization of the deviation from the nominal timetable	GLPK solver
[152]	Minimization of the rescheduling cost	Right shift, Local Search, Iterative Local Search
[153]	Minimization of secondary delays	CPLEX
[154]	Minimization of the delays variance	Heuristics

The dynamics of the system is determined by synchronization and the departures and arrivals are governed by the network constraints. The rescheduling of the railway traffic consists in finding a new feasible timetable in a short computation time while respecting the limits and constraints of the system. Hence, the rescheduling problem can be modeled as a MILP problem that optimizes a certain objective. Table 2-8 summarizes the most significant contributions of the related literature, reporting in the first column the referred article, in the second column the considered objective functions and in the third one the resolution approaches. Generally, in the MILP modeling approach for rescheduling problems, the decision variables are usually binary variables and non-binary integer variables. The rescheduling tactics can be represented by binary decision variables, such as the connection maintenance, the priority of two trains, sequences of trains, assignment of resources, etc. The arrival and departure times and delays are represented as continuous decision variables. The constraints in railway network management are generally expressed by the equality or inequality constraints in the model. From the decision variables and constraints of the model, it is possible to understand which kind of rescheduling tactics are considered and the scale of the considered problem.

In the subsequent sections it will be demonstrated how the MILP modeling approach can be suitably used in the context of railway traffic rescheduling. In particular, it will be provided innovative techniques able to support the management of the railway rescheduling in case of unexpected events in a smart transportation system perspective.

Chapter 3

3 Modeling, analysis, control and management of intermodal freight transportation terminals

The development of Smart Transportation Systems (STSs) is guided by the need of creating sustainable, efficient, low impact, and effective transportation systems. At the moment, a fully sustainable transportation system has not been conceived yet, but the integration of different transport modes is sustained by the European Community as one of the most suitable ways to achieve such a purpose [155]. This chapter resumes the results of the research conducted during my PhD program and presented in [93], [98], [156], [157]. The aim is of contributing to the advance of intermodal freight transport terminals by proposing innovative techniques which can sustain the evolution of such systems towards STSs. In particular, the focus is on the modeling, analysis, control and management of intermodal freight transportation terminals. It will be demonstrated how the Petri net modeling framework can be effective for the modeling, simulation, performance evaluation and improvement, and resource planning both in an offline and real-time perspective of such systems. Furthermore, the integration Petri nets with the Data Envelopment Analysis (DEA) technique will be presented to offer an innovative decision making procedure to be used in a strategic/tactical management perspective in intermodal freight transport terminals.

3.1 Intermodal freight transport

Intermodal freight transport can be defined as the transfer of goods from an origin to a destination, involving at least two transportation modes and services, such that the transshipment between two consecutive modes is performed at an intermodal terminal. The main feature of intermodal freight transportation is that the loads are moved in one loading unit, i.e., the Intermodal Transport Unit (ITU), and are not handled in the transshipment process. Multiple transportation modes allow deploying each individual mode to its best advantage, i.e., combining the major speed, security, reliability and sustainability advantages provided by rail/sea for long distance transport, as well as their lower costs, with the increased space penetration features of road [158]. The combination of several modes into an integrated system provides a more flexible service, as well as more reliable, profitable and sustainable transport [85], [159].

An intermodal transportation system includes several actors that interact with each other, i.e., shippers that request for transportation, carriers that provide the transportation service, facility and physical infrastructure managers, institutional authorities that fix the rules for the system, and customers and citizens that ask for freights. **Shippers** are responsible for the request of transport as they usually are the senders of goods. They manage the planning of the shipment with the aim of satisfying the customers and they also contribute to the organization of the shipment process necessary for their freights. Hence, they define the logistic strategy which can include intermodal transport. **Carriers** perform the transport for the shippers. Some carriers operate dedicated services, in which an ITU contains the freights of a single customer, and others operate on the basis of consolidation, i.e., each ITU may contain different customers' freight with different origins or destinations. **Freight Logistic Providers** (FLPs), third party logistics service providers (3PLs) in particular, undertake various logics tasks within an intermodal transport system, providing various major services, such as warehousing, distribution, shipping, inventory management, co-packing, labeling, repacking, weighting, and quality control. Shipper may usually outsource logistics activities in order to focus on their core businesses and benefit from the expertise of the FLPs. On the other hand, 3PLs also interact with carriers to secure timely transportation capacity for their customers. **Facility and infrastructure managers** may be public entities or private firms with public stakeholders. They do not plan, organize, or implement freight transportation services but instead deal with the management of the physical network and infrastructure, including roads and highways, the rail infrastructure in Europe, intermodal port terminals, and so on. Thus, they play a central role by providing efficient physical networks and the necessary technology and sensors layers to control and optimize the utilization of the infrastructure and facilities. **Institutional authorities** (e.g., governments and public administrations) are the actors who tax, give incentives, set up policies, and regulate transport activities. Through the policies they set, these actors increasingly frequently aim to guide the transportation and logistics system towards "new", more beneficial to society, and resilient ways of operation (e.g., the usage of specific corridors or vehicle and motorization types, mode changes from road-based to water- and rail-based transportation, the reduction of externalities, the consideration of environmental impacts, etc.). In this class of actors, can be included also transnational institutions and national governments. Finally, customers represent the receivers of the goods. They can be the final client, retailer, distributor, or wholesaler. Customers include citizens as well, and, hence, they are mindful about emissions, safety, and viability within their local areas, and the can influence the institutional authorities through their votes.

All of these actors have their own objectives, make their decisions and are linked with each other by various interests, interconnections, interactions, and interdependencies. All of them contribute to make intermodal transportation a complex system. Moreover, these decisions and interrelations can be affected by uncertainties from many sources, often related to demand, travel times, and handling operations [160][161].

Consequently, despite its numerous advantages, intermodal transport has some critical aspects [162], among which efficiency and performance evaluation and optimization are the most significant. Indeed, the integration of multiple transport modes, decision makers, and types of load units leads to much more complex intermodal planning problems than unimodal ones. As a result, operating inefficiencies may be experienced if the integration of the complex subsystems in the transportation network is not fully effective. In this context, one of the most important and critical elements in the freight transportation chain and the evaluation of its competitiveness is the Intermodal Freight Transport Terminal (IFTT) that provides the interface between modes and also between shippers and carriers. Therefore, the performance of terminals is crucial for the transportation chain effectiveness and needs to be closely monitored and optimised [163],[164].

3.2 Performance evaluation of Intermodal Freight Transport Terminals using Timed Petri Nets

The combination of intermodal transport with Information and Communication Technologies (ICTs) has been identified as one of the main actions to improve the effectiveness and efficiency of intermodal transport [164], [165]. This integration clearly shows its advantages when applied to the key elements of the transportation chain, namely IFTTs. In this context, the availability of a suitable computer based simulation model for testing the operational functioning and management of the IFTT allows the analysis, design, and control of the intermodal terminal. This permits to achieve better performances of the system and help the decision makers in creating correct strategies to maximize the benefits of intermodal transport while constraining its limitations [166], [167], [163].

This section presents an innovative DES modular modeling framework based on TPNs that, combined with Monte Carlo simulation, allows simulating the dynamics and evaluating the performance of a generic IFTT. The idea arises from the lack of contributions using DES formalisms to represent intermodal terminals by adopting a general modular approach. More in detail, the intermodal terminals' DES models proposed in the literature are manually tailored to specific case studies and consequently are not reusable. Differently, here it is considered the concept of modularity to implement a versatile modeling technique. In particular, the modularity of the presented technique is guaranteed by the implementation of several elementary modules, which, suitably combined by a systematic technique adapted from [168], can represent any terminal in the TPN framework. It is necessary to remark that, although Petri nets are not able to describe in detail all the complex operations of an intermodal system like other simulation tools (such as Arena, Witness, ExtendSim, etc.) -which is not, however the final goal of the contribution- they offer significant advantages over discrete event simulation tools which motivate this choice [85]. In fact, analytical DES models allow the evaluation and analysis of DESs taking advantage of discrete event simulation models. In particular, in the developed work it is proposed the combination of the TPN modeling of the system with Monte Carlo simulation to obtain statistically accurate estimates of the case studies performance indices [168].

The two main reasons for using TPNs for IFTT modeling and performance evaluation are in the fact that on the one hand PNs allow to model in a modular and systematic way high dimension DESs such as IFTTs while keeping the physical meaning of the DES subsystems, and on the other hand that TPNs allow the temporization of the activities of the system, so as to quantify their duration and evaluate the IFTT dynamics, during short and/or long time periods, without requiring high computational efforts, even for large nets. Other significant advantages of the use of TPNs are: 1) the graphical aspect, which enables an easily perceived, concise and effective way to design and verify the model; 2) the simple mathematical representation, which allows simulation of the system in software environments considering different conditions characterized by a different level of information shared between terminals and operators and consequently automatically analyze their behavior; 3) the capability to reproduce typical features of DESs, as priority, synchronization, parallelism, causal-consequence connections and shared resources; 4) the ability to define simple performance indices to evaluate the system behavior; 5) the opportunity to perform structural analyses on the developed net; 6) the possibility, by means of the so-called Generalized Mutual Exclusion Constraints (GMECs) [169], to ensure control policies, which can be implemented through simple monitor places (i.e. adding new places to the net), so as to represent the exchange of information in IFTTs allowed by modern ICT tools, thus allowing to solve some of the recalled intermodal transport criticalities.

The effectiveness of the proposed modeling framework is shown by two case studies, one from the literature [85] and one referring to a real logistics company operator located in Bari (Southern Italy). Thanks to the short computational times, the model allows evaluating different alternatives in order to improve the performance of the examined terminal.

3.2.1 Timed Petri Net Modeling of Intermodal Freight Transport Terminals

The IFTTs modelling framework employs a modular bottom-up approach. In particular, the TPN representing the terminal is made of subnets, each modelling the sequence of operations on containers in a particular subsystem. Hence, each subnet behaves as a distinct DES interacting with the others by interfacing nets. The following subsystems are considered, that constitute an intermodal terminal [88], [99]:

1) highway;
2) tollbooth;
3) railway;
4) maritime/river port or airport;
6) access road;
7) parking or yard storage area;
8) customs;
9) ITUs maintenance area.

The above nine subsystems which may be duplicated and/or combined to form a complete IFTT are complemented by the following two modules that allow the IFTT control:

10) opening/closing of an IFTT subsystem;
11) checkpoint.

In the proposed TPNs framework, places represent resources and capacities or conditions, transitions model inputs, flows and activities into the terminal, and tokens represent ITUs or the vehicles on which they are transported.

Before describing in detail each subsystem, it is worth noting to clarify that the IFTT bottom-up modeling framework based on TPNs employs three fundamental structures: the IFTT Subnet (IFTTS), the Open IFTT Subnet (OIFTTS), and the Routing Net (RN). In particular, each of the above listed IFTT subsystems is modeled by a TPN module, i.e., an IFTTS, which is to be interconnected with others by way of its transitions that model the inflow and outflow of vehicles into and out of the subsystem and are hence called communication transitions. Moreover, from each IFTTS it is defined an Open IFTT Subnet, which is a place-bordered net obtained extending the IFTTS with at least one source and/or one sink place, respectively in input and output to the IFTTS communication transitions, allowing the vehicles to be routed to other subsystems. The routing is obtained by interconnecting different OIFTTSs by a RN, so that the TPN complete model of the terminal is attained. Hence, each RN connects with at least one immediate transition the source and sink places of two or more OIFTTS modeling the subsystems among which there is a flow of vehicles. In this way, the decision maker may easily combine the TPN subsystems to represent the flow of vehicles in the larger system, eventually modifying some of their features - i.e.: changing weights of some arcs, modifying the initial marking of the subnet, deleting or adding places or transitions, duplicating nets in a single subsystem (in the case of multiple resources), changing a deterministic transition into a stochastic one or vice-versa.

More formally, the IFTTS is the basic element of the framework and is a $TPN = \left(P, T, \mathbf{Pre}, \mathbf{Post}, \mathrm{F}\right)$

modeling the functioning of a specific subsystem of the IFTT, considered disconnected from the others. Given a particular IFTTS, an OIFTTS is a place-bordered extension of the IFTTS defined as a 9-tuple $\left(P,\ T,\mathbf{Pre},\mathbf{Post},\mathrm{F},P_I,\ P_O,\mathbf{Pre}',\mathbf{Post}'\right)$, where:

1. $\left(P,T,\mathbf{Pre},\mathbf{Post},\mathrm{F}\right)$ is an *IFTTS*;
2. P_I is the set of added source places, i.e., $\forall p\in P, {}^{\bullet}p=\emptyset, P_I\cap P=\emptyset$;
3. P_O is the set of added sink places, i.e., $\forall p\in P, p^{\bullet}=\emptyset, P_I\cap P_O=\emptyset, P_O\cap P=\emptyset$;
4. $\mathbf{Pre}': P_I\times T_O\rightarrow\mathbb{N}^{m'\times n}, \mathbf{Post}': P_O\times T_I\rightarrow\mathbb{N}^{m'\times n}$, are the *Pre* and *Post Incidence* sub-matrices for source and sink places;
5. $P_I\cup P_O\neq\emptyset$

Moreover, given two OIFTTSs $TPN_1=\left(P_1,T_1,\mathbf{Pre}_1,\mathbf{Post}_1,\mathrm{F}_1\right)$ with sink place p_o and $TPN_2=\left(P_2,T_2,\mathbf{Pre}_2,\mathbf{Post}_2,\mathrm{F}_2\right)$ with a source place p_i such that $T_1\cap T_2=\emptyset$, the flow of vehicles from TPN_1 to TPN_2 may be easily modeled by a RN duplicating the border places and connecting them via an immediate transitions, i.e., by a net $TPN_3=\left(P_3,T_3,\mathbf{Pre}_3,\mathbf{Post}_3,\mathrm{F}_3\right)$ that is a place-bordered TPN with:

1. $P_3=\{p_o,p_i\}$;
2. $T_3=\{t_r\}$ with ${}^{\bullet}t_r=\{p_o\}, t_r^{\bullet}=\{p_i\}$;
3. $\mathrm{F}_3(t_r)=0$.

Figure 3-1 represents an example of two OIFTTSs *A* and *B* modeling two IFTT subsystems and connected by a RN. The OIFTTS labeled *A* has a border sink place p_o that is added to the IFTTS modeling the terminal subsystem, while the OIFTTS labeled *B* is obtained adding border place p_i to its subsystem. The two subnets are connected by the *A-B* RN, duplicating the border places p_o and p_i and including an immediate transition t_r. The tokens flowing in the modular TPN of Figure 3-1 represent ITUs moving between the two subsystems *A* and *B*.

Generalizing, a RN is a TPN containing immediate transitions each serving as routing interface between the border places of two or more OIFTTSs.

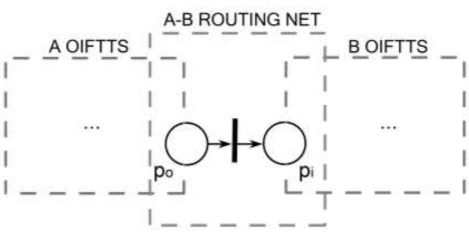


Figure 3-1. A-B Modular TPN

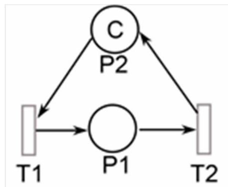


Figure 3-2. Highway portion subnet model

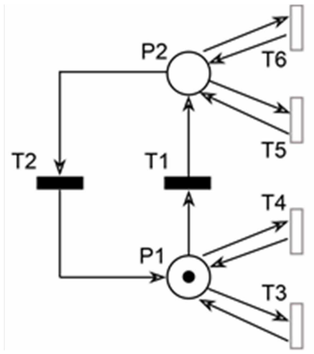


Figure 3-3. Tollbooth subnet model.

The IFTTSs modeling the terminal subsystems are detailed in the following sections. For each IFTTS, the communication transitions are depicted in grey, to which border places may be attached to connect the subsystem to others.

a. Highway

These subnets model the highways through which straight trucks or semi-trailer trucks access/leave the terminal. It is important to recall that trucks can be classified as either straight or articulated vehicles [99]. A straight truck is one in which all axles are attached to a single frame. An articulated vehicle is one that consists of two or more separate frames connected by suitable couplings. A semi-trailer truck is an articulated vehicle composed by a towing engine called tractor and one or more semi-trailers carrying freight. Figure 3-2 shows the simple model of a highway portion, where place p_1 indicates the presence of the transportation means, p_2 the highway capacity C (i.e., the maximum number of transportation means that it can accommodate), and exponential transitions t_1 and t_2 are the two communication transitions (depicted in grey) that respectively represent the incoming and outgoing flows and as such allow the combination of the subnet with others.

b. Tollbooth

These subnets model the arrival of vehicles to the IFTT from a tollbooth. Figure 3-3 represents the tollbooth subnet, with its flows differentiated on the basis of working days and holidays, and on the kind of transportation means. Place P_1 represents arrivals in working days, p_2 arrivals in holidays. Moreover, deterministic transition t_1 (t_2) represents the flow of hours at working (holiday) days. Exponential transition t_3 (t_5) models the arrivals of semi-trailer trucks during working days (holidays), while t_4 (t_6) represents the arrivals of straight trucks during working days (holidays). These four transitions are the subsystem communication transitions. Finally, note that, if different traffic conditions are present every day, it is sufficient to replicate the set given by p_1, t_1, t_3 and t_4 for each day of the week.

c. Railways

These subnets model the presence of a dedicated railway system servicing the terminal to deliver or to allow the departure of ITUs. Figure 3-4 shows a railway line arriving at the intermodal terminal, with trains delivering ITUs and/or straight trucks. Place p_1 indicates the presence of a train and p_2 its absence.

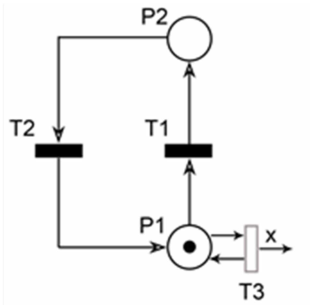


Figure 3-4. Railway subnet model: only incoming ITUs.

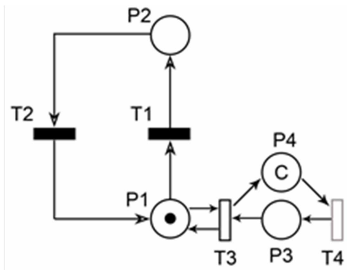


Figure 3-5. Railway subnet model: incoming and outgoing ITUs.

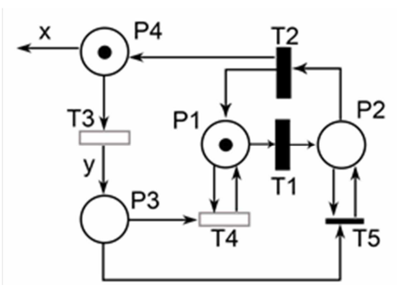


Figure 3-6. Railway subnet model: incoming and outgoing ITUs with separate load/unload.

Transition t_1 models the activity hours of the train, t_2 the hours of absence of the train, t_3 (communication transition) the average time of arrivals, and x is the number of ITUs or straight trucks at each arrival. As an alternative, Figure 3-5 illustrates the case of trains carrying ITUs and/or vehicles in both incoming and outgoing directions. Given the train load plan, it is necessary to model its maximum capacity. Hence, p_1 indicates the presence of a train, p_2 its absence, p_3 the loaded cargo, p_4 the train capacity C (the maximum number of ITUs that it can accommodate). Moreover, t_1 models the activity hours of the train, t_2 the hours of absence of the train, t_3 the average time of arrivals, t_4 (communication transition) the average time for loading/unloading a cargo. Similarly to Figure 3-4, it is possible to model a railway subnet with only outgoing ITUs. Moreover, in case of multiple rail lines with different destinations or in case of different incoming rates during the week, it is sufficient to connect multiple subsystems similar to that in Figure 3-5, with different values of the average time of the arrivals for the incoming loads. Finally, in Figure 3-6 it is represented the possibility of modeling the loading/unloading phase separately. In particular, the loading of ITUs on the train is enabled only after the unloading phase is ended. Accordingly, p_1 represents the presence of a train in the terminal, p_2 its absence, p_3 the capacity y of the train, p_4 the unloading of the train, t_1 the sojourn time of the train in the terminal, t_2 the absence time of the train from the terminal, t_3 (communication transition) the average unloading time for x ITUs, t_4 (communication transition) the average time for loading an ITU. Transition t_5 allows emptying p_3 at train departures.

d. Maritime or river ports and airports

Maritime and river ports, as well as airports, can be represented by the same subsystem, shown in Figure 3-7, with different meanings of places and transitions.

In case of maritime (river) ports, the model represents the transit, docking, sojourn, and load of a ship (barge), to be connected to the intermodal platform. Place p_1 indicates the absence of the vessel in the port, p_2 the docking of the vessel, p_3 its presence, p_4 the loaded cargo, p_5 the capacity C of the vessel. Further, t_1 is the required docking time, t_2 the dwell time in port, t_3 the sailing time, t_4 the enabling for load/unload, t_5 (communication transition) the loading/unloading time.

Similarly, for airports, Figure 3-7 represents the landing, length of stay and loading of an aircraft in the terminal. In such a case p_1 indicates the absence of the aircraft in the airport, p_2 the landing of the aircraft, p_3 its presence, p_4 the loaded cargo, p_5 the capacity C of the aircraft, t_1 is the required time for landing, t_2 the dwell time in airport, t_3 the flight duration,t_4 the enabling for loading/unloading, t_5 the average time for loading/unloading. As in the railway net, even for seaports, fluvial ports and airports, in case of multiple lines with different destinations or differences within the week, it is sufficient to connect similar subsystems, with different values of the average time of the arrivals for the incoming loads.

e. Access road

These subnets model the truck access roads to terminals. In particular, Figure 3-8 models the entrance into the terminal of straight trucks: place p_1 indicates the entrance of transportation means in the terminal, and p_2 its capacity C, transition t_1 (communication transition) the average arrival time and t_2 (communication transition) the average closing time.

In case of semi-trailer truck flows, the model is different from that in Figure 3-8, since trucks have to be disassembled for their ITUs to be transshipped. The corresponding alternative subnet model is shown in Figure 3-9, where place p_1 indicates the waiting ITUs, p_2 the waiting tractors, p_3 a cargo waiting for loading, p_4 the ITUs to be reassembled with the corresponding truck, p_5 (p_6) enables (inhibits) the freight to leave the terminal on the same arrival day, p_7 models the ITUs that will exit from the terminal the next day, p_8 the capacity C of the waiting area. t_1 (communication transition) is the average arrival time, t_2 the average waiting time, t_3 the average time for loading/unloading a cargo (communication transition), t_4 is the average arrival time both for ITUs and tractors available for reassembly, t_5 and t_6 model the delay times for the exit of semi-trailers, t_7 is the enabling for the immediate exit, t_8 (communication transition) is the average exit time of ITUs and tractors. When ITUs are left in the terminal by the semi-trailer trucks but tractors do not wait in the terminal for the next cargo, it is possible to use the simpler subnet in Figure 3-10 instead of the previous nets: place p_1 indicates the presence of the full semi-trailer truck in the terminal, p_2 the ITU left by the truck, p_3 the tractor leaving the terminal, transition t_1 (communication transition) average arrival time for the transportation means, t_2 the disjunction of the tractor from the ITU, t_3 (communication transition) average ITU's transportation time to the next subsystem, t_4 exit of the tractor. In a similar way, it is also possible to represent tractors that arrive in the terminal and load ITUs to deliver. This case is modelled in Figure 3-11, where p_1 represents incoming tractors without load, p_2 ITUs to be delivered, p_3 outgoing tractors with ITUs, t_1 (communication transition) the average arrival time of incoming tractors, t_2 the tractor connection to the ITU, t_3 (communication transition) the average arrival time of ITUs available to be delivered, t_4 semitrailers leaving the terminal.

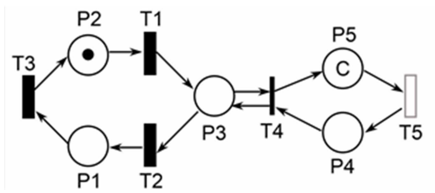


Figure 3-7. Seaport, river port or airport subnet model.

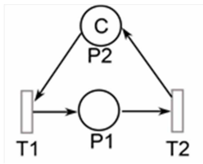


Figure 3-8. Access road subnet model for straight trucks.

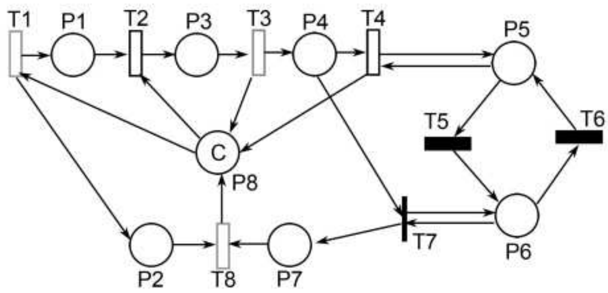


Figure 3-9. Access road subnet model for semi-trailer trucks with reassembly.

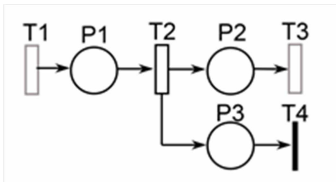


Figure 3-10. Access road subnet model for semi-trailer trucks leaving ITUs.

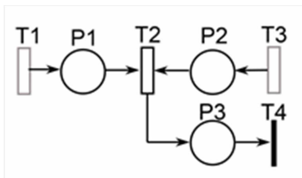


Figure 3-11. Access road subnet model for semi-trailer trucks loading ITUs.

f. Parking or yard storage area

External parking areas (for waiting trucks) and yard parking areas (to store ITUs) are modelled as in Figure 3-12, where p_1 indicates the entrance in the area, p_2 its capacity C, t_1 (communication transition) is the average arrival time and t_2 (communication transition) the average exit time.

g. Customs

These subnets model the presence of customs in the IFTT and are represented in Figure 3-13. The subnet inflows may refer to ITUs, straight trucks, or semi-trailer trucks. Each item is inspected by the customs and sent to the subsequent transport mode, or, in case of customs rejection, it is returned to the sender via the same means. Hence, p_1 (p_2) models the inspection of ITUs (straight trucks), p_4 the inspection of ITUs carried on semi-trailer trucks, p_6 the waiting tractors, p_7 the rejected ITUs and straight trucks, p_8 the capacity C of the custom area. Moreover, transition t_1 (t_2) is the average arrival time of ITUs (straight trucks), t_3 the average inspection time, t_4 the average exit time of accepted ITUs and straight trucks, t_5 the average exit time of rejected ITUs and straight trucks, t_6 the average arrival time of semi-trailer trucks, t_7 the average inspection time for semitrailers, t_8 the average exit time of accepted ITUs carried on semi-trailer trucks, t_9 the average exit time of rejected ITUs carried on semi-trailer trucks, t_{10} the average exit time of reassembled semi-trailer trucks. Note that t_1, t_2, t_4, t_5, t_6, t_8, t_{10} are all communication transitions.

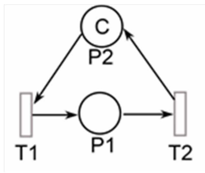


Figure 3-12. Parking or yard storage area subnet model.

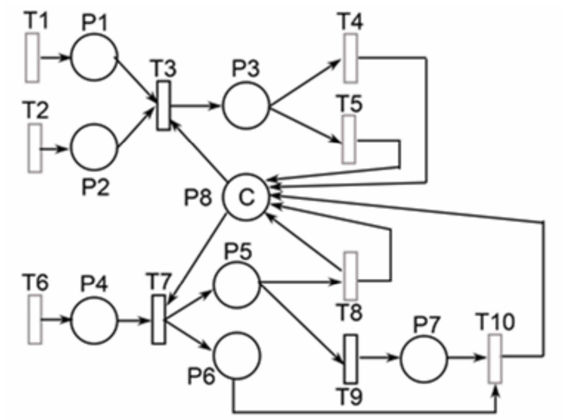


Figure 3-13. Customs area subnet model.

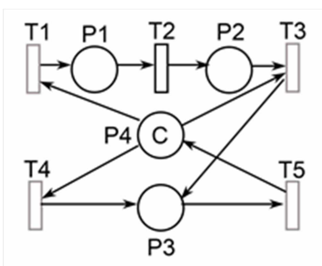


Figure 3-14. ITUs maintenance area.

h. ITUs maintenance area

These subnets model the terminal area dedicated to ITU maintenance. This area can be modelled as in Figure 3-14, where it is considered the possibility of executing either only routine maintenance or both routine and special maintenance. Hence, place p_1 represents ITUs waiting for special maintenance, p_2 ITUs under special maintenance, p_3 ITUs waiting for ordinary maintenance, p_5 ITUs under ordinary maintenance, p_4 the number of resources (C) available for the maintenance area. Moreover, transition t_1 models the average arrival time of ITUs in the special maintenance area, t_2 the average time for the special maintenance, t_3 the average waiting time for routine maintenance after the special one, t_4 the average arrival time of ITUs for routine maintenance, t_5 the average exit time of the ITUs from the maintenance area. Here t_1, t_3, t_4, t_5 are all communication transitions.

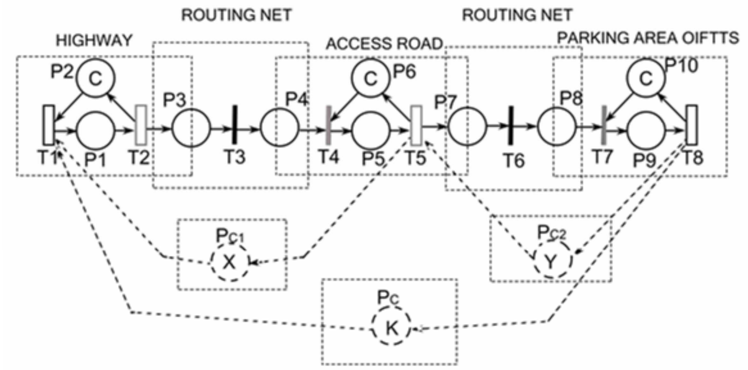


Figure 3-15. Example of TPN model of IFTT.

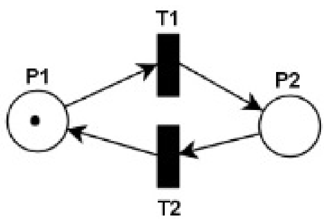


Figure 3-16. Opening/closing for hours subnet model.

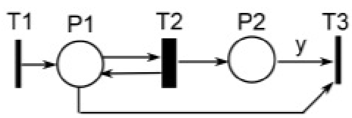


Figure 3-17. Opening/closing for days subnet model.

i. Example of TPN model of an IFTT

Figure 3-15 (solid lines only) shows the TPN model of a simple IFTT, including the last miles of a highway, an access road and a parking area. These are first modeled by the IFTTS respectively represented in Figure 3-2, Figure 3-8, and Figure 3-12. Hence, they are extended to obtain the matching OIFTTS, respectively by adding a sink place p_3 (highway), a source place p_4 and a sink place p_7 (access road) and a source place p_8 (parking area). The combination of the three subnets is allowed by two RNs and by the corresponding immediate transitions t_3 and t_6.

Finally, it is important to remark that in the access road and parking area subsystems the source transitions t_4 and t_7 are both immediate to allow the average arrival time of vehicles in the subnets being equal to the average exit time of the transportation means of the preceding subnets, i.e. t_2 and t_5.

j. Opening/closing of an IFTT subsystem

The cyclical opening/closing of any previously described IFTTS can be managed considering a subnet that allows controlling hours or days of activity/inactivity. Figure 3-16 depicts the model of the opening and closing of an IFTTS, specifying the hours of activity and inactivity of the terminal or of any external parking area. In this case, place p_1 (p_2) indicates when the subsystem is active (idle), and transition t_1 (t_2) models the activity (idling) time. As an alternative, Figure 3-17 shows the opening/closing of a subsystem depending on days, where p_1 represents the passing of a working/non-working day, p_2 is the counter of the number of working/non-working days, p_3 models the presence/absence of the means in the subnet, y (the weight of the arc from p_2 to t_3) is the number of working/non-working days, t_1 is the start of activity/inactivity, t_2 is the duration of the day, and finally t_3 is the end of activity/inactivity.

k. Checkpoint

To impose constraints on the IFTT behavior, checkpoints can be installed and modeled by GMECs controlling the TPN dynamics. The GMEC can be regarded as a supervisor specifying a state feedback control law. For instance, Figure 3-15 shows the case in which three checkpoints (dashed lines) are added to the IFTT example (solid lines) described at point *i* of the previous list. The number of incoming trucks is controlled by the control places p_{C1} (between the highway and access road), p_{C2} (between the access road and parking area), and p_{C3} (between source and sink transitions of the system). The control laws are:

$$M(p_1)+M(p_3)+M(p_4)+M(p_5)\leq M(p_{c1}) \ , \quad \text{(Eq. 3-1)}$$

$$M(p_7)+M(p_8)+M(p_9)\leq M(p_{c2})\ ; \quad \text{(Eq. 3-2)}$$

$$M(p_1)+M(p_3)+M(p_4)+M(p_5)+M(p_7)+M(p_8)+M(p_9)\leq M(p_{c3}) \quad \text{(Eq. 3-3)}$$

3.2.2 Case Studies

To evaluate its effectiveness and ease of application, the proposed model is applied to two IFTTs: the first is an example from the literature [85]; the second is a real case study. The TPN models are simulated in MATLAB using the HYPENS tool [170], and the performance indices are evaluated by multiple replications of simulation runs of 8,760 time units (one year, considering one hour per time unit) each. Note that, for sake of simplicity, in this work stochastic transitions are characterized only by exponential distributions since they require just one parameter and they are memoryless. This is a common choice in the literature, since the inter-arrival times of transportation means for the arrival processes are purely random and independent, they can be suitably characterized by means of the single parameter of an exponential distribution ([85]; [98]; [155]). Furthermore, in this work it is considered the single-server semantics and enabling memory policy [155]. Moreover, if transitions are in conflict, then one of them is randomly selected to fire. To evaluate the case studies behavior, two kinds of indices are taken into account: the utilization of the critical IFTT areas and the throughput of the IFTT subsystems interconnections.

A. A literature example

In this subsection the modelling framework is applied to test case-study taken from the literature. In particular, it refers to the work of Dotoli, *et al.* [85], and the case study is an intermodal terminal located in Trieste, Italy. The IFTT includes eight subsystems (Figure 3-18). Within the terminal, straight and semi-trailer trucks circulate, modelled as tokens of the TPN moving on two separate lines with the same capacity. The first module of the IFTT in Figure 3-18 is the tollbooth (Figure 3-3), differentiating the entrance frequency based on the vehicle type and day (weekdays/holidays). The highway subsystem (Figure 3-2) provides the trucks entrance into/exit from the terminal, while the railway subsystem (Figure 3-4) provides the entrance for straight trucks, whose arrival is differentiated between weekdays and holydays; hence, this subnet is obtained by joining two different models. The seaport subsystem (Figure 3-7) manages the arrival, departure, docking, loading and unloading of a ship that can carry both straight trucks and ITUs deposited by semi-trailer trucks. Finally, the terminal includes access roads, differentiated according to the type of trucks (Figure 3-8, Figure 3-9), and an opening/closing subnet (Figure 3-16). Hence, the TPN model of the IFTT in Figure 3-18 is determined connecting each TPN subsystem, obtaining the TPN in Figure 3-19, where each dashed box indicates a subsystem.

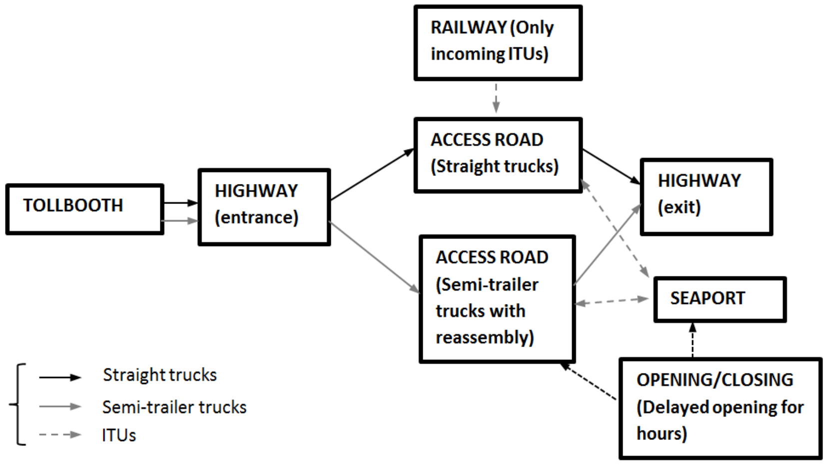


Figure 3-18. Scheme of the literature example [85].

Moreover, the TPN model includes seven RNs that allow routing the vehicles between the different terminal subsystems. Note that in the opening/closing subsystem of Figure 3-19, with respect to Figure 3-16 it is added a transition (t_{61}) to create a delay between the terminal opening and the start of the embarkation on the vessel. This transition is enabled only at the initial marking of the simulation by place p_{65}. In addition, places p_{17} (modelling the capacity of access road for straight trucks) and p_{42} (modelling the capacity of access road for semi-trailer trucks) are respectively assigned a capacity of A=40 and B=100 vehicles. Table 3-1 shows the firing times (in hours) and the meaning of the deterministic and stochastic transitions that model the flows of means within the TPN in Figure 3-19: values are assigned based on the terminal data in [85].

Implementing the net in Figure 3-19 in HYPENS [170] with replications of 8,760 time units each, the obtained computation time for each replication is of less than 8 minutes on a PC with an Intel Core 2 Duo - 2.80 GHz processor and 4 Gb of RAM. Hence, the approach can be applied to even larger and more complex IFTTs. The analysis of the terminal behavior is conducted by the evaluation of some proper performance indices ([171], [172]). In particular, at first it is evaluated the average free capacity of the IFTT, i.e., the average number of straight (semi-trailer) trucks FC_1 (FC_2) that may still be accommodated in the terminal, i.e., the marking of p_{17} (p_{42}). Second, it is evaluated the average free capacity FC_3 of the last portion of the entrance highway (the marking of p_{11}), i.e., the average number of vehicles that may still enter it. Third, it is estimated the average throughput $Tr(t_i)$ of suitable stochastic transitions $t_i \in T_E$, i.e., the average number of fires per time unit of t_{13}, t_{14}, t_{21}, t_{31} (chosen since they respectively represent the passage of vehicles from the highway to the terminal and from the terminal to the seaport). The performance indices are obtained from 1000 independent replications with a 95% confidence. At each replication, the delays of the stochastic transitions are randomly generated with respect to the associated probability distribution. The resulting half width of the confidence interval is about 1.5% in the worst case.

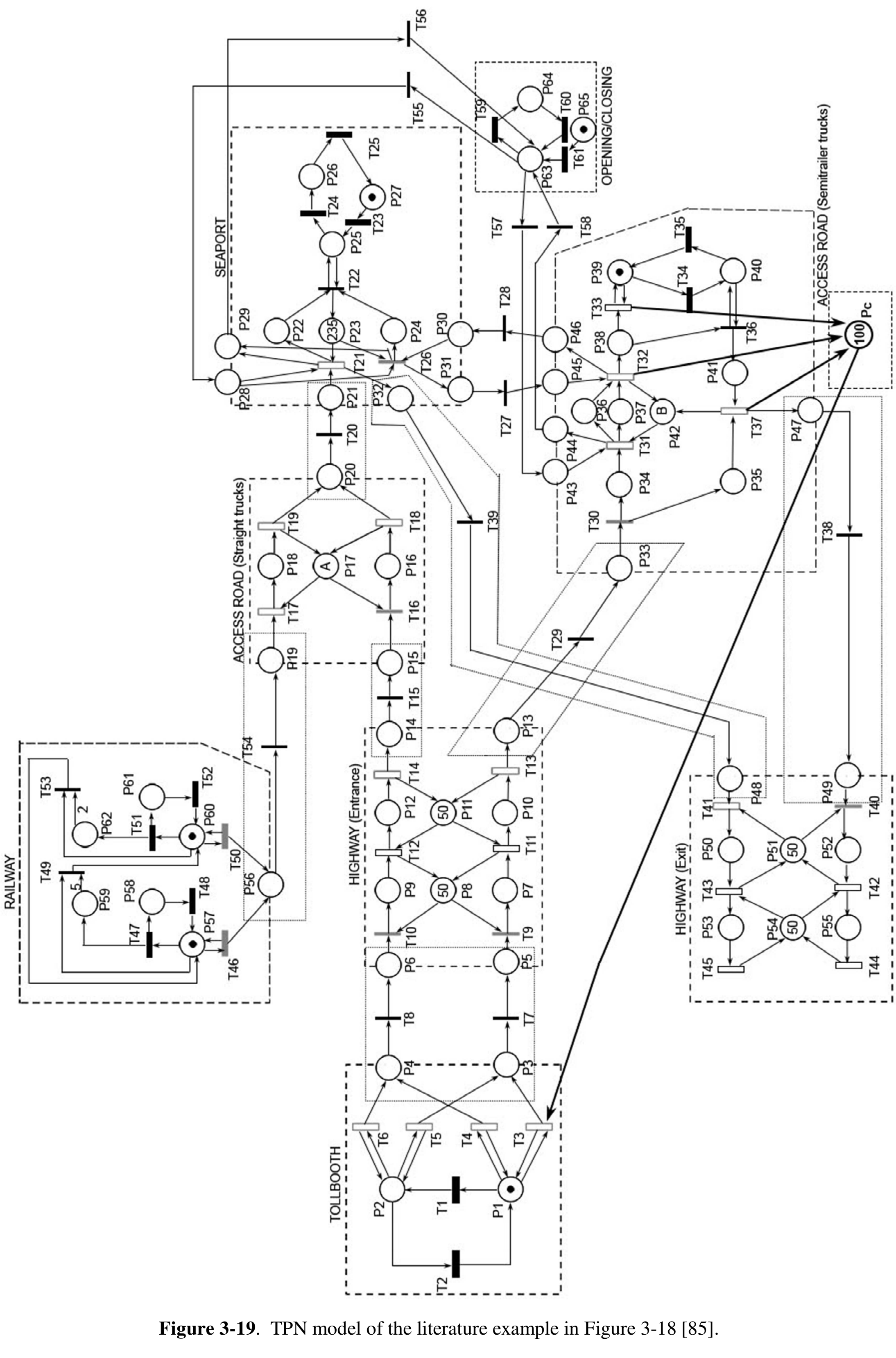


Figure 3-19. TPN model of the literature example in Figure 3-18 [85].

This confirms the accuracy of the estimates, although an increased number of replications would provide a narrower confidence interval. The choice of a high number of replications is particularly useful in case of real time simulation for decision support at the operational level, but it is of course determined by a compromise choice depending on the model complexity and the resulting simulation time.

Table 3-2 shows the simulation results. In particular, the Table collects in each row respectively the results corresponding to the as-is situation and four alternatives (Scenarios 1 to 4): the first three alternatives provide structural actions by creating new parking areas, while in the last scenario ICT tools are considered that can avoid oversaturation, through the exchange of information among the logistic actors, thus resulting in lower investments. Moreover, each column of Table 3-2 reports the obtained values of the defined IFTT performance indices, while the last column shows the corresponding run time.

Analyzing the as-is scenario (first row of Table 3-2), it is apparent that the access road for straight and semi-trailers trucks is oversaturated, which leads to congestion of the IFTT. Indeed, the relative free capacities FC_1 and FC_2 of p_{17} and p_{42} in Figure 3-19 have very low average values, just like the free capacity of the last portion of the highway FC_3 in p_{11}. Obviously, this also affects the average number of vehicles passing from the highway to the terminal and then to the port, as evidenced by the low throughput values (columns 5 to 8). To overcome the disadvantages of the as-is scenario, four different alternative solutions are considered: the increase of the capacity of p_{17} from A=40 to 190 (Scenario 1); increasing the capacity of p_{42} from B=100 to 450 (Scenario 2); increasing both these capacities by setting A=190 and B=450 (Scenario 3); the insertion of a supervisor, by means of a GMEC (shown in Figure 3-19 with bold lines) keeping the as-is capacities (Scenario 4).

The evaluated indices reported in Table 3-2 (last four rows) show that, by increasing the access roads capacities, the flow of vehicles within the system becomes more regular and congestion is avoided. In particular, in Scenario 1, considering the increase only of the straight trucks access road capacity (A in Figure 3-19), the performance indices still highlight an oversaturation of the semi-trailer access road (FC_2 equal to 3.98 in Table 3-2) and consequently of the incoming highway connected to the IFTT (FC_3 equal to 3.50). This leads to a high value of the average free capacity of the straight trucks access road: the area seems free, but this depends only on the slowing down of the highway flow.

In scenario 2 it is increased the semi-trailer trucks access road capacity (B in Figure 3-19). This results in the decongestion of the access roads and of the incoming highway, although the capacity of the straight trucks area remains too low. Hence, the best results are obtained in Scenario 3, i.e., by increasing the capacities of both p_{17} and p_{42} in Figure 3-19 (see second-last line of Table 3-2). Scenario 4 considers the control by a checkpoint of the entrance during weekdays of semi-trailer trucks into the terminal, using a monitor place between the highway tollbooth and the semi-trailer access road. It is assumed that ICT tools allow exchanging information among the logistics actors. Hence, it is assumed that by a suitable information, provided to the semi-trailer trucks owners, the semi-trailer trucks flow is forbidden until the highway of the terminal and the parking area of the terminal are no longer oversaturated, limiting pollution, decreasing travel costs and increasing road safety.

TABLE 3-1 - Meaning and firing times of transitions in the TPN of Figure 3-19.

Transition	Description	Firing time [h]
T_1	Weekdays	120.000
T_2	Holydays	48.000
T_3,T_4	Arrival of semi-trailer (straight) trucks on weekdays	0.590 (0.210)
T_5,T_6	Arrival of semi-trailer (straight) trucks on holydays	0.570 (0.290)
T_{11},T_{12}, T_{13} T_{14}, T_{42}, T_{43}, T_{44}, T_{45}	Flows of vehicles through the highway	0.170
T_{21}	Embarkation/disembarkation of straight trucks	0.017
T_{31}	Average embarkation time for semi-trailer trucks	0.001
T_{32}	Embarkation/disembarkation of semi-trailer trucks	0.220
T_{33}	Semi-trailer trucks exiting the terminal	0.670
T_{34}	Tractors waiting in the terminal	23.500
T_{35}	Reassembling tractors/cargo	0.900
T_{37}	Departure of semi-trailer trucks	0.210
T_{61}	Opening terminal delay	1.000
T_{59}	Opening time of the terminal	5.500
T_{60}	Closing time of the terminal	18.500
T_{23}	Presence of vessel in the seaport	0.500
T_{24}	Shipping time	17.000
T_{25}	Docking	6.500
T_{17}	Transition of straight trucks from rail to terminal	0.110
T_{18}, T_{19}	Straight trucks entering the port	0.100
T_{46}, T_{50}	Arrival of trains	2.000
T_{47}	Presence of train in the railway on weekdays	7.000
T_{48}	Absence of train in the railway on weekdays	17.000
T_{51}	Presence of train in the railway on holydays	4.000
T_{52}	Absence of train in the railway on holydays	20.000
T_{41}	Straight trucks exiting the terminal	0.100

TABLE 3-2 - Performance evaluation of the TPN of FIGURE 3-19.

Scenarios	FC_1	FC_2	FC_3	$Tr(T_{13})$	$Tr(T_{14})$	$Tr(T_{21})$	$Tr(T_{31})$	Run time [s]
As-is	19	3.21	3	0.03	0.07	6.28	0.06	434
1	143	3.98	3.50	0.03	0.07	6.28	0.06	648
2	6.8	382	38.90	1.70	3.80	7.90	3.80	1618
3	163	384	49	1.70	3.80	3.88	3.82	1605
4	16.04	13.82	15.26	0.36	0.79	5.66	0.79	748

Accordingly, the ICT control law is realized by preventing the TPN from evolving towards forbidden states, i.e., saturated access road and highway for semi-trailer trucks. Since these restrictions on the system behavior are logical predicates that do not depend on the time evolution, the control problem can be formulated using GMEC, i.e., constraining the weighted sum of markings in a place subset, as follows:

$$M(p_3)+M(p_5)+M(p_7)+M(p_{10})+M(p_{13})+M(p_{33})+M(p_{34})+M(p_{36})+M(p_{37})+M(p_{38}) \leq B$$

The constraint is imposed including in the net a control place, which has as initial marking $M(P_C)$=B. This enables the semi-trailer trucks to flow in the highway, according to the free space still available in the relative access road, avoiding congestion (last row of Table 3-2).

B. A real case study

The second case study concerns a real intermodal inland rail-road terminal located in Bari (Southern Italy) at the GTS - General Transport Service S.p.A. company, a leader in intermodal freight transport in Italy and Europe, owning about 1,800 containers of different types and 280 rail wagons. The current management of the logistics system is considered and some possible improvements are proposed.

In Figure 3-20 a scheme of the IFTT is presented. Semi-trailer trucks and trains circulate in the IFTT, the former through access roads, the second by a dedicated railway line. Trucks and trains transport ITUs that are stored and made available for the next transport mean in a dedicated yard storage area. During the week, the terminal can accommodate trucks from 6.30 a.m. to 6.30 p.m., while on Sunday the terminal is closed. The company manages semi-trailer trucks traffic as follows: vehicles that carry ITUs to the rail destinations of Piacenza and Bologna (Italy); vehicles that load ITUs to deliver in the port with destination Patras (Greece); vehicles that pick up ITUs to load from (deliver to) the initial (final) customer.

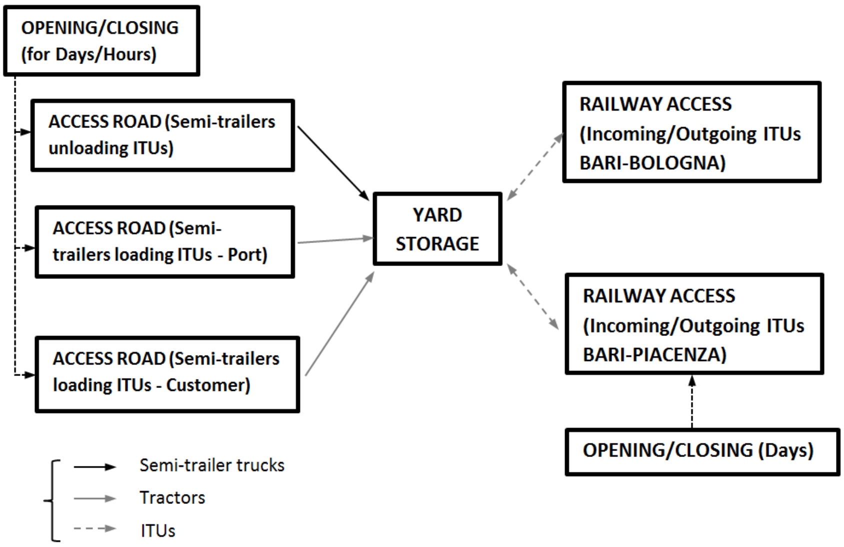


Figure 3-20 - Scheme of the real case study.

The rail traffic is classified into: trains from/to Piacenza, with capacity C_{TP}=34 ITUs, and trains from/to Bologna, with capacity C_{BT}=20 ITUs. Trains from/to Bologna circulate all week on alternate days, arriving in Bari at 7.30 a.m. and staying until 5.30 p.m. (the trains return to the terminal after 38 hours). Trains from/to Piacenza, instead, arrive every day at 7.30 a.m. and stay till 5.30 p.m., while there are no arrivals on Sunday. The ITUs delivered to the IFTT by road or rail are stacked in a yard storage with capacity C_{YS}=250 ITUs.

B.1. The terminal TPN model

This section presents the TPN model describing the IFTT in Figure 3-20. The TPN system $\langle TPN, \boldsymbol{M_0}\rangle$ of Figure 3-20 with TPN=(P, T, $\mathbf{Pre}, \mathbf{Post}$, F) models the structure and the dynamic evolution of the IFTT under the current management that is here called case as is. The TPN system in Figure 3-20 consists of the necessary subnet models described in Section 3.2.3, connected in an appropriate manner. The TPN digraph elements are specified as follows. The place set is $P = P_R \cup P_C \cup P_F$: set P_R models the system resources (i.e., access roads, rail, and GTS terminal); set P_C models the available capacities of the resources; set P_C contains places used to model conditions, to give priority and synchronize the main events of the system (time of day or day of the week, opening/availability and closure/unavailability of resources, etc.). In the TPN model, a token in a place $P_i \in P_R$ represents an ITU, semi-trailer or train in the system, a token in a place $P_i \in P_C$ is an available position in a resource and a token in a place $P_i \in P_F$ represents a condition that is verified. Moreover, the transition set of the net in Figure 3-21 is $T=T_S \cup T_D \cup T_I$. Exponential stochastic transitions in T_S model the input of vehicles into the IFTT, their flows and activities.

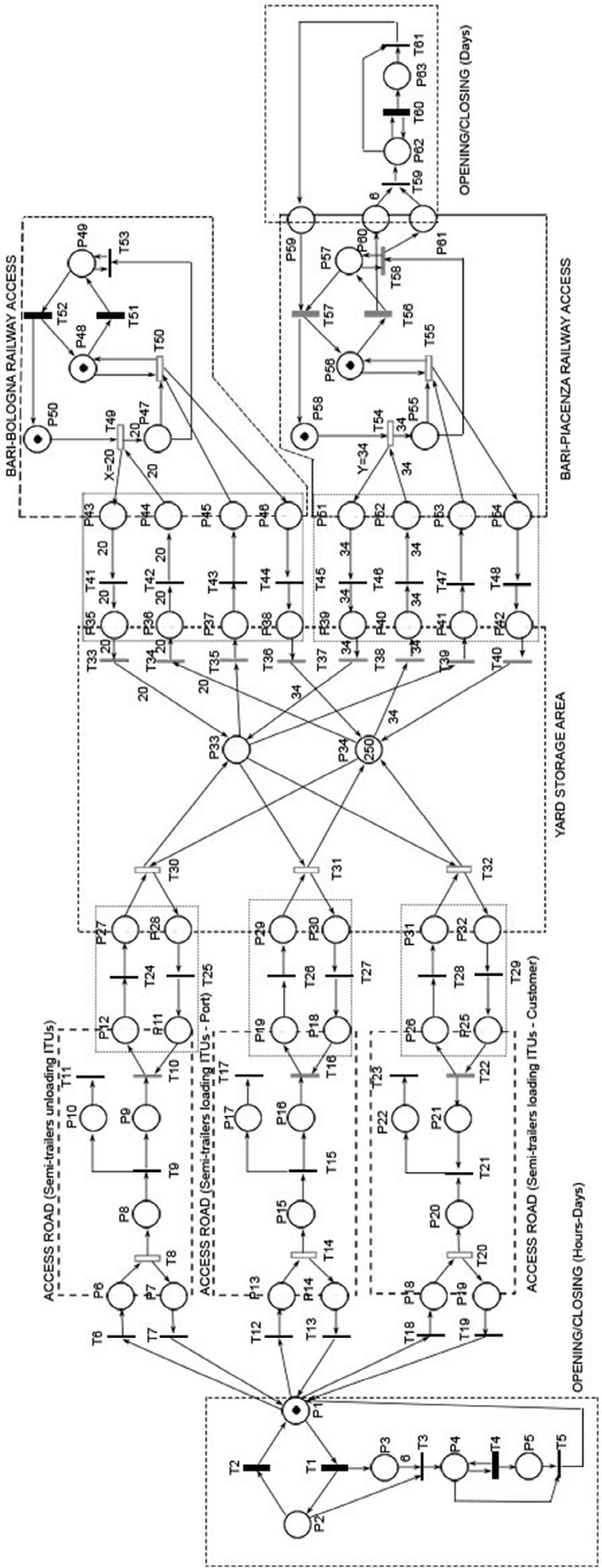


Figure 3-21. TPN model of the real case study in Figure 3-20.

Moreover, set T_D of deterministic timed transitions models the occurrence of deterministic events at particular times of the day, i.e., the terminal opening (transition t_1) and closing events (transition t_2), the weekly closing of the terminal (t_4) the weekly pause of the train arrival/leaving from/to Piacenza (t_{60}), the arrival (t_{51}-t_{56}) and the departure (t_{52}-t_{57}) of the trains, set T_I collects the TPN immediate transitions, i.e., t_3 modelling the start of the closure interval for the terminal, t_5 modelling the end of the closure interval for the terminal, t_{11}-t_{17}-t_{23} modelling the exit of trucks from the terminal, t_{53}-t_{58} reset of the train capacity at every departure, t_{59}-t_{61} start/end of the closure interval for train arrival/departure from/to Piacenza. Matrices **Pre** and **Post** and the initial marking $\boldsymbol{M_0}$ of the TPN system in Figure 3-21can be deduced from the edges and the token distribution shown in the figure. In particular, each place $p_i \in P_R$ can accommodate vehicles and, assuming that the system is empty at the initial marking, it holds $\boldsymbol{M_0}(p_i)$=0 for each $p_i \in P_R$. On the other hand, the initial marking of each $p_i \in P_C$ is set equal to the corresponding resource capacity. According to the terminal structure in Figure 3-20, the IFTT model in Figure 3-21 is formed by suitably connecting using five RNs eight subsystems among the following kinds of IFTTS: 1) access road for semi-trailer trucks unloading ITUs, 2) access road for semi-trailer trucks loading ITUs, 3) yard storage area, 4) railway with separate ITUs load/unload an the opening/closing management; 5) opening/closing of an IFTT subsystem. For each subnet the meaning of places and transitions are those listed in Section 3.4, and the firing times associated with stochastic and deterministic transitions are given in Table 3-3.

B.2. Simulation results

The IFTT dynamics is simulated and analysed using the data in Table 3-3. The aim is studying the system behaviour considering the actual management of the terminal and comparing it with possible scenarios and alternative solutions.

Table 3-3- Meaning and firing times of transitions of the TPN in Figure 3-21.

Transition	Description	Firing time [h]
t_1	Hours of activity of the terminal	12.00
t_2	Hours of closure of the terminal	12.00
t_4	Sunday closure	12.00
t_6	Average arrival time of full semi-trailer trucks, unloading ITUs in the terminal	0.34
t_7	Average time for ITU unloading	0.13
t_9	Average arrival time of tractors that load ITUs with destination final customer	0.46
t_{10}-t_{13}	Average time for semi-trailer assembling	0.13
t_{12}	Average arrival time of tractors that load ITUs with destination port	1.14
t_{15}	Stay time in terminal of the train Bari-Bologna	10.00
t_{16}	Absence time in terminal of the train Bari-Bologna	38.00
t_{17}	Average time for unloading cargo of Bari-Bologna train	2.00
t_{18}-t_{23}	Average time for ITU loading on the train	0.13
t_{20}	Stay time in terminal of the train Bari-Piacenza	10.00
t_{21}	Absence time in terminal of the train Bari-Piacenza	14.00
t_{22}	Average time for unloading cargo of Bari-Piacenza train	3.00
t_{26}	Weekly pause for Bari-Piacenza train	24.00

The indices evaluating the IFTT performance are [173], [99]:

1) the occupation of the yard storage area, evaluating the number of ITUs in the area, its average value O_{YS} and its maximum value;

2) the occupation of access roads, evaluating the number of semi-trailer trucks waiting for loading/unloading ITUs, its average value O_{AR} and its maximum value;

3) the average throughput $Tr(t_i)$ or average number of fires per time unit of some stochastic transitions $T_i \in T_S$.

Starting from the actual structure (scenario as-is), the system behavior is evaluated in eight additional scenarios, to test the model capability to represent different situations (see Table 3-4): in Scenarios 1-2-3 it is assumed, respectively, an increase of 20-30-50% in the number of empty ITUs exiting the terminal by semi-trailer trucks for subsequent loading of goods; in Scenario 4 the yard storage capacity C_{YS} is increased, from 250 to 375 ITUs; in Scenario 5 the times associated to loading/unloading ITUs, i.e. t_{30}, t_{31}, t_{32}, t_{50}, t_{55}, are halved; in Scenario 6 it is considered a reduction of 25% of the loading/unloading times of the ITUs (t_{30}-t_{31}-t_{32}-t_{50}-t_{55}); in Scenario 7 the traffic of ITUs is incremented of 100% and the times for loading/unloading ITUs are halved; finally, in Scenario 8 the loading/unloading times are doubled. The performance indices for each scenario are listed in Table 3-5 and Table 3-6. The average occupations O_{YS}, O_{AR1}, O_{AR2}, and O_{AR3} are respectively calculated for the yard storage area p_{33}, for the access road of semi-trailers leaving ITUs in the terminal p_{27}, for the access road of tractors carrying ITUs from terminal to the final customer p_{29}, for the access road of tractors carrying ITUs from terminal to port p_{31}. The occupation maximum values (minimum values are zero) are also in Table 3-5.

Throughputs are calculated for the loading/unloading of ITUs in the yard storage ($Tr(t_{30})$-$Tr(t_{31})$-$Tr(t_{32})$) and the loading on trains ($Tr(t_{50})$-$Tr(t_{55})$).

In the first row of Table 3-5 are shown the results for the case as-is. Comparing the average value of occupation of the yard storage O_{YS} (28.88 ITUs) with its maximum capacity C_{YS} (250 ITUs), and analyzing the average values of occupation of the access roads O_{AR1}- O_{AR2}- O_{AR3} (which amount to around one vehicle), the system appears not congested, highlighting a good management of the available resources.

In Scenarios 1-2-3 it is considered a 20-30-50% increase of the load of goods carried by straight trucks. As a consequence, the values associated with some parameters of the net are adapted to represent the relative Scenario, as reported in Table 3-4.

Table 3-4 - Scenarios for performance evaluation of TPN in Figure 3-21.

Sc.	T_8 [h]	T_{14} [h]	T_{20} [h]	T_{49} [h]	T_{54} [h]	X [ITU]	Y [ITU]	C_{YS} [ITU]
As-is	0.34	1.14	0.46	2.00	3.00	20	34	250
1	0.30	1.14	0.38	2.40	3.60	24	41	250
2	0.28	1.14	0.35	2.60	3.80	26	44	250
3	0.25	1.14	0.30	3.00	4.30	30	51	250
4	0.25	1.14	0.30	3.00	4.50	30	51	375
5	0.25	1.14	0.30	1.50	2.25	30	51	250
6	0.25	1.14	0.30	2.20	3.40	30	51	250
7	0.20	1.14	0.23	2.00	3.00	40	68	250
8	0.34	1.14	0.46	3.00	4.50	20	34	250

Table 3-5 - Performance indices of TPN in Figure 3-21 – avg. and max occupation.

Sc.	O_{YS} [ITUs]	Max $M(p_{33})$	O_{AR1} [veh]	Max $M(p_{27})$	O_{AR2} [veh]	Max $M(p_{29})$	O_{AR3} [veh]	Max $M(p_{31})$
As-is	28.88	115	0.67	9	0.30	9	1.24	24
1	48.49	174	0.77	113	0.21	6	1.16	23
2	54.20	200	0.86	9	0.20	9	1.03	34
3	240.94	250	9.20	72	0.16	4	0.85	12
4	339.60	370	2.35	30	0.16	4	0.82	10
5	16.75	125	0.46	8	0.52	11	2.38	45
6	33.73	180	0.71	12	0.20	7	1.77	47
7	26.27	170	0.59	8	0.16	6	1.88	47
8	234.38	250	8.17	68	0.24	5	0.83	11

Table 3-6 - Performance indices of TPN in Figure 3-21 - Throughput.

Scenarios	$Tr(t_{30})$ [veh/h]	$Tr(t_{31})$ [veh/h]	$Tr(t_{32})$ [veh/h]	$Tr(t_{50})$ [ITU/h]	$Tr(t_{55})$ [ITU/h]
As-is	2.73	0.78	2.06	1.46	2.56
1	3.07	0.80	2.43	1.75	3.02
2	3.30	0.82	2.63	1.89	3.27
3	3.72	0.81	3.10	0	0
4	3.63	0.82	3.05	0.38	0.48
5	3.62	0.82	3.05	2.16	3.72
6	3.69	0.81	3.08	2.18	3.77
7	4.61	0.79	4.00	2.90	5.12
8	2.71	0.80	2.03	0.16	0.15

Table 3-7 - Validation indices.

Performance Index	Meaning	PI	ρ	RPI
Tr(t_6)	Throughput of unloaded ITUs	2.94	0.04	2.98
Tr(t_9)	Throughput of exiting ITUs (to port)	0.88	0.03	0.88
Tr(t_{12})	Throughput of exiting ITUs (to customer)	2.17	0.04	2.18

In particular, the following IFTT parameters vary: the average number of trucks entering the terminal, i.e., the average interarrival time of trucks in the terminal (t_8, t_{14}, t_{20} in Figure 3-21); the number of ITUs carried by trains (X, Y in Figure 3-21); the average time needed for their loading (t_{49}, t_{54}). Note that the capacity of the yard storage area remains equal to 250 ITUs. As reported in Table 3-5 and Table 3-6, the system reacts well in the first two situations, i.e. O_{YS} is around one fifth of the maximum capacity (C_{YS}) and the access roads are occupied by at most one vehicle during working hours. When the number of carried ITUs is increased by 50%, the yard storage (O_{YS}) and the access road for semi-trailer trucks with full ITUs (O_{AR1}) become congested, revealing the limitation of the system to manage an increase of the volumes of full ITUs. To reduce the congestion some alternatives are evaluated. In Scenario 4, the C_{YS} is increased from 250 ITUs to 375 ITUs, but this does not produce any considerable improvement; O_{YS} and O_{AR1} do not show a substantial decrease. In Scenario 5 the times needed for loading/unloading ITUs, i.e., the firing times of t_{30}-t_{31}-t_{32}-t_{50}-t_{55}, are halved, assuming a value of 0.07 hours, with C_{YS} equal to its original 250 ITUs value. In this way, the number of ITUs waiting in yard storage area O_{YS} and the number of semi-trailers waiting for unloading O_{AR1}, drastically decrease. Scenario 6 considers a reduction of 25% of the loading/unloading times of the ITUs (t_{30}-t_{31}-t_{32}-t_{50}-t_{55} equal to 0.1 hours), without modifying C_{YS}, and the system still does not congest. In Scenario 7, the traffic of ITUs is doubled (as shown in Table 3-4; with regards to t_8, t_{14}, t_{20} and X, Y) and the times for loading/unloading ITUs are halved (t_{30}-t_{31}-t_{32}-t_{50}-t_{55} equal to 0.07 hours). The obtained performance indices values demonstrate that the terminal can manage well a large increase of ITUs handling (see the relative O_{YS}, O_{AR1}, O_{AR2}, O_{AR3}), but only if the resources needed for the loading/unloading are increased in such a way that the times associated to these activities can be halved. In Scenario 8, a doubling of the loading/unloading times is assumed (t_{30}-t_{31}-t_{32}-t_{50}-t_{55} equal to 0.26 hours), to represent a situation in which a technical failure or a shortage of staff occurs. The performance indices show now a congestion of the yard storage area (O_{YS}) and of the access road for full semi-trailer trucks (O_{AR1}), causing difficulties in the management of ITUs carried by trains. The throughputs in all scenarios completely reflect the remarks for each case.

As an example, Figure 3-22 (Figure 3-23) represents the evolution under Scenario 3 of the markings of places p_{27} (p_{33}), i.e., the variation over time of the occupation of the first access road, whose average O_{AR1} and peak values are in Table 3-5. The figures show that under this scenario the access road copes with the incoming flows, while the storage area is always congested, so that its occupation is often close to its capacity of 250 vehicles.

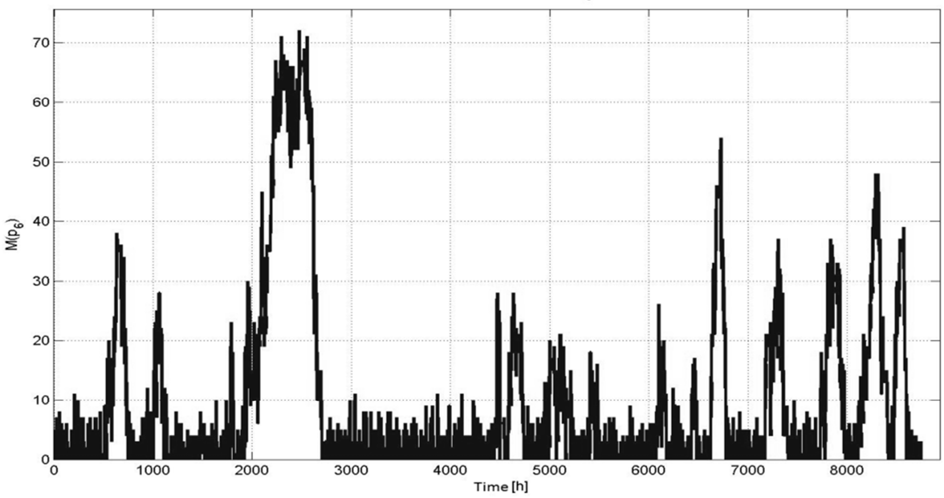


Figure 3-22. M(p_{27}) (occupation of access road) in Scenario 3.

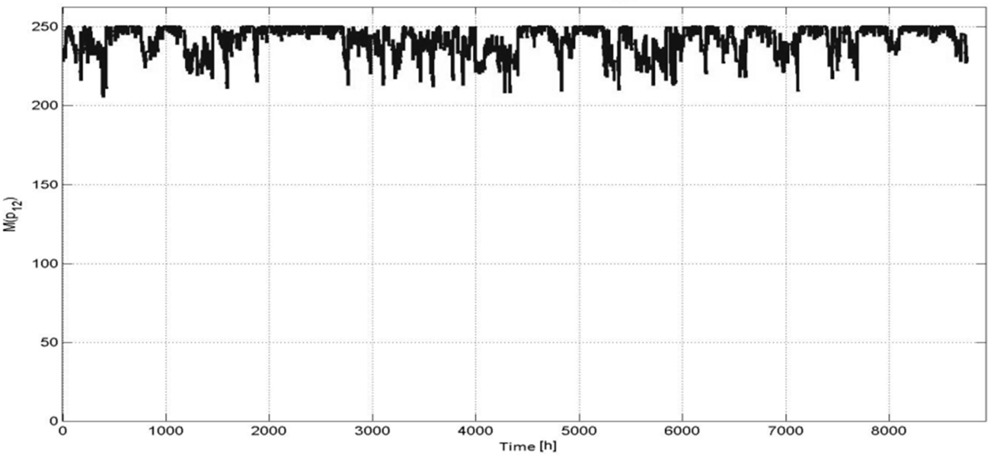


Figure 3-23. M(p_{33}) (occupation of yard storage area) in Scenario 3

Finally, it is to remark that an average computation time of 1 minute is obtained for each replication on a PC with an Intel Core 2 Duo-2.80 GHz processor and 4 Gb RAM. The performance indices are obtained from 1000 independent replications with a 95% confidence. The half width of the confidence interval is about 0.9% in the worst case.

B.3. The model validation

Validation shows how closely the model represents the real system and it may be achieved by applying the single mean test [174]. Specifically, real data are provided by the company and compared with some representative performance index of the model. The half width of the relative confidence interval is determined. Table 3-7 reports the performance indices obtained by the simulation with the relative half width of the confidence interval and the equivalent values computed by historical data provided by the company. Denoting by *PI* the generic performance index provided by the simulation, by *RPI* the corresponding index obtained by real data and ρ the relative half width of the confidence interval, Table 3-7 shows that for each considered performance index it holds:

$$PI - \rho \leq RPI \leq PI + \rho \quad \text{(Eq. 3-4)}$$

Hence, applying the single mean test [174], the results prove that the simulation closely represents the actual system.

Summing up, both case studies show that using the proposed model for IFTT analysis has a huge potential for verifying its efficient operation, allowing to synthetically measure the effective impact of new infrastructures, the criticality of failures or of increased traffic flows, etc.

3.3 Management of intermodal terminals by hybrid Petri nets

The purpose of this section is to use First-Order Hybrid Petri Nets (FOHPNs) to effectively manage IFTTs in a closed-loop fashion. In fact, FOHPNs offer the possibility to adapt and optimize the management of the terminal, depending on the desired functioning, setting a linear objective function to be optimized subject to linear constraints. This kind of control policy is not computable starting from a purely discrete Petri net model that is essentially useful to derive supervisory controllers. On the contrary, the proposed hybrid model allows combining both time-driven dynamics, proper of transportation means flows, and event-driven dynamics, such as: opening/closing of the terminal and scheduled arrivals/departures of transportation means. In particular, two optimal control policies are suggested for the terminal while coordinating the speeds associated with the continuous transitions of the net: one aiming at maximizing the outflows of the terminal, the other at minimizing the residual fluid (i.e., the ITUs) in the yard storage area. This allows taking offline decisions on the management of transfer speeds and the quantity of resources needed for ITUs transferring, in case of variation of commercial flows in the terminal. Moreover, the model can be used for deriving an online control approach to solve congestions or malfunctioning caused by unexpected events or abnormal increments of transportation means flows.

The proposed modeling and control technique is applied to the real logistics company in Bari (southern Italy) already considered in in the previous section. After validating the terminal model in an open-loop setting, the closed-loop dynamics is evaluated under different scenarios, including a situation of potential congestion. It is shown that, while the controlled system copes well with this case, when no control action is applied, the system enters a congested state.

The Hybrid PNs (HPNs) formalisms can be effectively taken into account in the field of transportation networks modeling and management, since they allow the efficient modeling of the traffic flows as fluids, while still keeping the discrete dynamics description and event-driven dynamics [15]. This fluidization allows reducing the computational effort for the system simulation and performance evaluation. In the context of HPNs formalisms, FOHPNs [17] are a framework in which the system design parameters are continuous, allowing solving integer linear programming problems to select in a closed-loop fashion suitable operational parameters that optimize appropriate performance indices. FOHPNs have been employed to model, simulate, analyze and control manufacturing systems [175], [176], [177]. Subsequently, applications of FOHPNs have appeared in the related literature on modeling and controlling freeway systems [59], [61].

While the above contributions applying HPNs to transportation systems regard vehicular and particularly urban traffic, to the best of the authors' knowledge, no contribution has been proposed in the related literature on the application of hybrid PNs to intermodal transportation and particularly IFTTs.

3.3.1 Elementary FOHPN Models for IFTTs

As already pointed out in the previous section an IFTT can be considered as a set of interconnected subsystems: *access road*; *railway access*; *yard storage area; opening/closing* subsystem. Hence, the FOHPN representing the terminal is made of subnets, each modeling the operations on containers in a particular subsystem. In the sequel, will be described the discrete/continuous/hybrid subnets that are used in the management of the considered case study. It is to remark that in this section are employed only continuous, immediate and deterministic timed transitions.

A. Access Road

The access road subnet represents the road section from which semitrailers enter/exit from the terminal. This subnet is represented in Figure 3-24 (a): it consists of two continuous transitions, T_{1C} and T_{2C}, and a continuous place P_{1C}. Transitions represent the flow of transportation means that enter/exit the terminal. The continuous place contains the fluid representing semitrailers.

B. Railway Access

The railway access subnet represents the arrivals/departures of trains in the terminal on dedicated rail lines. Trains arrive and depart from the terminal picking-up or delivering ITUs. In particular, when a train arrives in the terminal, ITUs are unloaded and subsequently stored based on their final destination, until the train is completely empty. Only at this point the loading operation starts, involving the ITUs stored in the yard zone having the train destination that are stored in the corresponding yard zone. Finally, the train leaves the terminal according to its timetable.

Figure 3-24 (b) shows the FOHPN model of the railway access line. When P_{1D} is marked, no train is in the railway access of the terminal. The firing of timed transition T_{2D} models the arrival of a train. Its time delay is equal to the number of hours in which no train is in the terminal access. When P_{2D} is marked, one train is in the terminal access and remains there for a number of hours that is equal to the time delay of transition T_{1D}. Continuous places P_{1C} and P_{2C} represent the number of ITUs allocated to different destinations. Such quantities coincide with the weights (X and Y) of the input arcs to places P_{1C} and P_{2C}. It is obviously possible to increase or decrease the number of destinations varying the number of continuous places. Continuous transitions T_{1C} and T_{2C} model the ITUs unloading operations that reduce the amount of fluid in P_{1C} and P_{2C}, and simultaneously increase the content of place P_{3C}. The unloading operation terminates when the train is empty. To impose that the loading operation starts only when the unloading operation is completely finished, two immediate transitions T_{3D} and T_{4D} and a discrete place P_{3D} are introduced in the model. Transition T_{3D} moves a token to place P_{3D} as soon as the content of P_{3C} increases of one unit. Transition T_{4D} fires as soon as the content of P_{3D} is equal to X+Y, i.e., the unloading operation is finished. The marking of place P_{4C} immediately becomes equal to X+Y and transition T_{3C} is enabled. Its firing models the partitioning of the available space in the train devoted to full (H) and empty (K) ITUs. Clearly, it is H+K=X+Y. At this point, the loading operation, corresponding to the firing of transitions T_{4C} and T_{5C}, may start. Obviously, this may happen provided that the train is in the terminal access, namely place P_{2D} is marked. Moreover, the enabling condition of T_{4C} and T_{5C} also depends on the availability of ITUs in the yard storage area that is not modelled in this subnet. Finally, immediate transitions T_{5D} and T_{6D} model an important constraint. The departure time of the train is fixed and, as explained above, is related to the firing of transition T_{1D}. It may happen, when the train is expected to leave, that it is not full at its maximum capacity. If such is the case, it is important to reset the current partitioning of the free capacity in full and empty ITUs, namely to reset to zero the content of places P_{5C} and P_{6C} in order to avoid affecting future operations.

C. Yard Storage Area

The yard storage area subnet represents the area dedicated to ITUs temporary storage. The subnet in Figure 3-24 (c) models the partitioning of the storage yard in n sub-areas where ITUs are stored depending on their destination. The content of places P_{1C} to P_{nC} models the number of ITUs in each subarea, while P measures the free yard capacity (W when the area is empty). Transitions T_{1C} to T_{2nC} represent the entrance/exit of ITUs to/from the areas. The net structure guarantees that the sum of tokens in continuous places is always equal to W. Therefore, the marking in Figure 3-24 (c) represents a situation where the yard is empty.

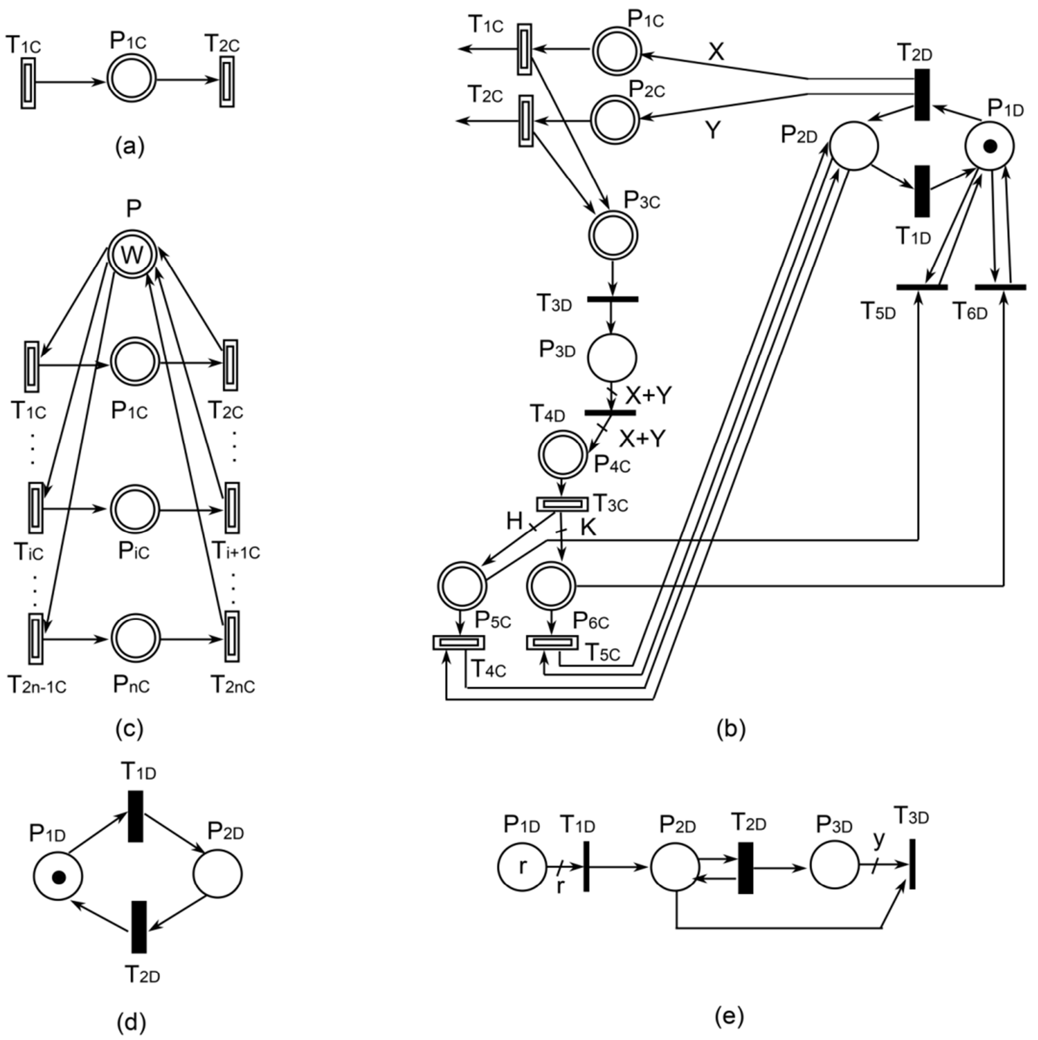


Figure 3-24. (a) Access road subnet. (b) Railway access subnet. (c) Yard storage area subnet. (d) Opening/closing "for hours" subnet. (e) Opening/closing "for days" subnet.

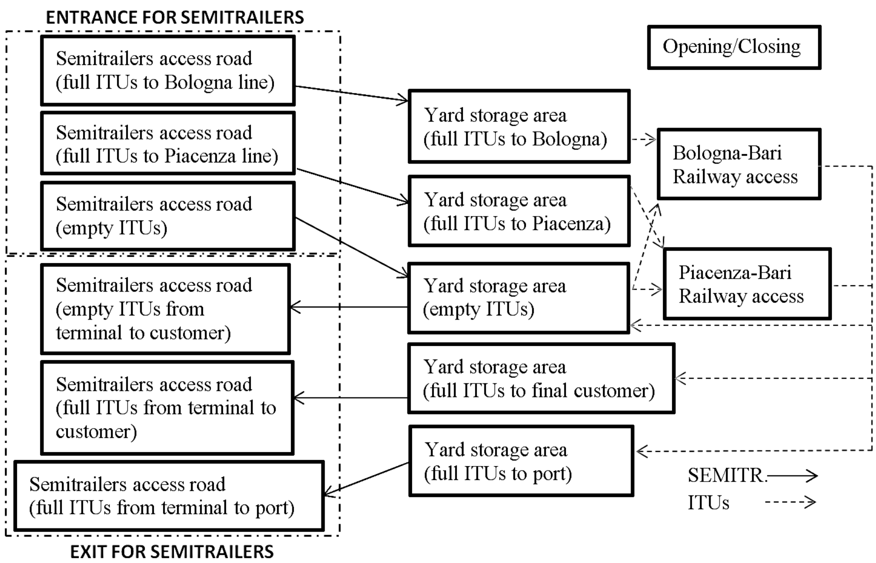


Figure 3-25. Logical scheme of the GTS IFTT

D. Opening/Closing Subsystems

The cyclical opening/closing of the previously described activities can be managed considering appropriate subnets that allow controlling hours or days of activity/inactivity.

Two different cases are here considered. Figure 3-24 (d) depicts the model that describes how activities vary during the different hours of a day. Place P_{1D} (P_{2D}) indicates when the subsystem is active (idle), and transition T_{1D} (T_{2D}) models the activity (idling) time. Figure 3-24 (e) shows the model that describes how activities vary during the different days of a week. As soon as r tokens arrive in P_{1D}, transition T_{1D} immediately fires adding one token in P_{2D}. A time delay of one day is associated with T_{2D}, and, provided that P_{2D} remains marked, T_{2D} fires after that a time interval corresponding to one day has elapsed. Therefore the content of P_{3D} is a measure of the number of days that have elapsed since the last firing of T_{1D}. As soon as the marking of P_{3D} becomes equal to y, transition T_{3D} fires empting places P_{2D} and P_{3D}, and thus resetting the count of the number of days since the last occurrence of T_{1D}. In the IFTT model, such a subnet is used to regulate weekly opening/closing phases. It is to notice that such a net can be clearly simplified if no counter of days is used and all time delays are measured in hours. However, in this work it is kept as in Figure 3-24(e) because it provides a more intuitive interpretation of the behavior in real systems, as the case study in the next section.

3.3.2 The Case Study and the Management Problem

The real case application is on the rail-road terminal of the Italian company, "GTS - General Transport Service S.p.A.," located in Bari (Southern Italy). Although the IFTT has been already introduced and described in Section 3.2, here a more detailed description of its features is provided, so as to properly contextualize the management problem. In particular, in the terminal many operations are combined so as to offer an efficient shipping service. Semi-trailer trucks and trains may circulate in the IFTT, where the former enter/exit the terminal through the access roads, while the latter use dedicated railway lines that link Bari to Bologna and Piacenza. During the week, the terminal can accommodate trucks from 6.30 a.m. to 6.30 p.m., while on Sunday the terminal is closed. Instead, the arrivals and departures on the railway lines follow a fixed timetable, and the rail traffic is classified into: trains from/to Piacenza, with capacity C_{PT}=34 ITUs, and trains from/to Bologna, with capacity C_{BT}=20 ITUs. Trains from/to Bologna circulate from Monday to Saturday and arrive to the terminal three days a week; they arrive in Bari at 7.30 a.m. and stay until 5.30 p.m. (the trains return to the terminal after 38 hours). Trains from/to Piacenza, instead, arrive every day at 7.30 a.m. and stay till 5.30 p.m., while there are no arrivals on Sunday. Each cargo of the trains from Bologna/Piacenza to Bari, contains ITUs divided in: full for the final customer (TB2=13 ITUs delivered from Bologna, TP2=24 ITUs delivered from Piacenza), empty (TB1=4 ITUs delivered from Bologna, TP1=0 ITUs delivered from Piacenza), and full/empty directed to the port (TB3=3 ITUs delivered from Bologna, TP3=10 ITUs delivered from Piacenza). Otherwise, the trains departing from Bari on the Bologna line have EB=14 empty ITUs and FB=6 full ITUs, the trains departing from Bari on the Piacenza line have EP=7 empty ITUs and FP=27 full ITUs. Once arrived at the terminal (both with semitrailers or trains) the ITUs can be stored in a dedicated yard storage area with a maximum capacity C_{YS}=250 ITUs. In this model, differently from the previous section, the yard is divided in sub-areas, such as: full ITUs for Bari/Bologna trains, full ITUs for Bari/Piacenza trains, full ITUs for the final customer, ITUs for the port, and empty ITUs for all destinations. The transferring of ITUs from one transportation mean to another is managed by two cranes that work in parallel all the working day long.

The IFTT logical scheme is represented in Figure 3-25, where arrows symbolize the flow of ITUs and transportation means in the terminal from one subsystem to another. Specifically, the IFTT subsystems are: daily/weekly opening/closing, access roads, yard storage areas, and railway accesses. The arrival/exit of the ITUs in/from the terminal, as already reported, is possible in two ways: the access roads and the railway access. In the former case, the ITUs are carried by semitrailer trucks that enter the terminal and unload ITUs in the yard. Subsequently, the semitrailers become immediately available to load a new cargo. In the latter case, trains arriving at the terminal can unload the cargo on the yard and are reloaded with the available ITUs.

A. The FOHPN Model of the Terminal

This section describes the FOHPN model of the GTS IFTT, represented in Figure 3-26 and obtained as the modular composition of the subnets described in the previous subsection.

The meaning and the delay time (the range mfs - MFS) associated with discrete (continuous) transitions are summarized in Table 3-8, Table 3-9, and Table 3-10. Note that in Figure 3-26 all places have initial marking equal to zero, except for $m_{P_{1D}} = 1$, which represents the situation of open terminal, $m_{P_{6D}} = 1$ and $m_{P_{9D}} = 1$, representing the absence of trains in the terminal, and $m_{P_{9C}} = 250$, indicating the maximum free available capacity in the yard.

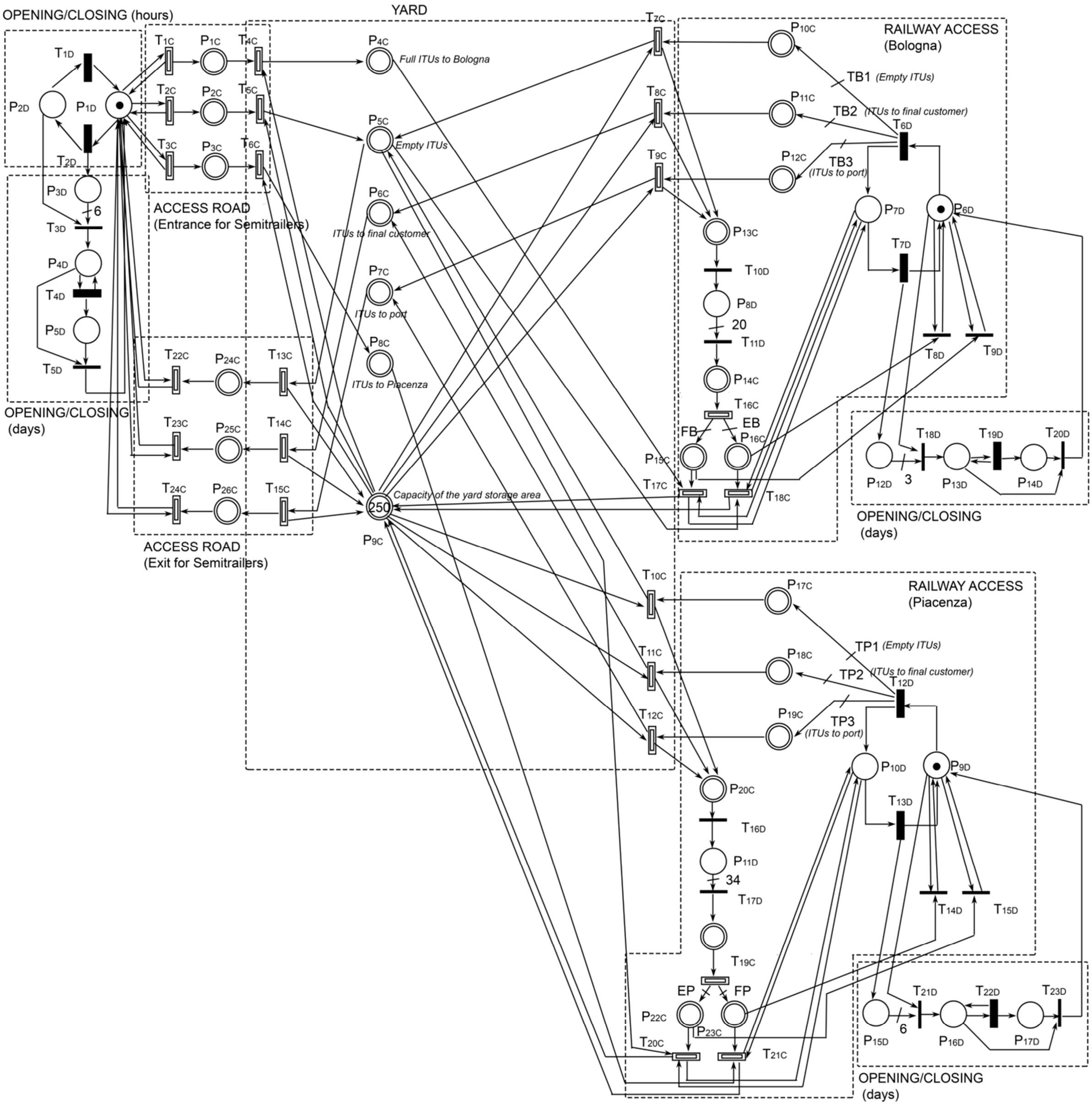


Figure 3-26. The GTS IFTT model in the FOHPN framework

Table 3-8. Time Delay of the Discrete (non-Immediate) Transitions in **Figure 3-26**

Transition	Description	Time delay [h]
T_{1D}	Hours of activity of the terminal	12.00
T_{2D}	Hours of closure of the terminal	12.00
T_{4D}	Sunday closure	12.00
T_{6D}	Absence time of the train Bari-Bologna	38.00
T_{7D}	Dwell time of the train Bari-Bologna	10.00
T_{12D}	Absence time of the train Bari-Piacenza	14.00
T_{13D}	Dwell time of the train Bari-Piacenza	10.00
T_{19D}	Weekly stop for the Bari-Piacenza train	24.00
T_{22D}	Weekly stop for the Bari-Bologna train	24.00

Table 3-9 mfs-MFS of the Continuous Transitions in **Figure 3-26.**

Transition	Description	Time delay [h]
T_{1C}	Average arrival rate of semitrailers with full ITUs to Bologna	0.25 – 0.25
T_{2C}	Average arrival rate of semitrailers with empty ITUs to Bologna and Piacenza	1.16 – 1.16
T_{3C}	Average arrival rate of semitrailers with full ITUs to Bologna	1.92 – 1.92
T_{4C}, T_{5C}, T_{6C}, T_{7C}, T_{8C}, T_{9C}, T_{10C}, T_{11C}, T_{12C}, T_{13C}, T_{14C}, T_{15C}, T_{17C}, T_{18C}, T_{20C}, T_{21C}	Average ITUs' transfer rate	12.5 –14.7
T_{16C}, T_{19C}	Availability rate of free capacity in the Bologna/Piacenza train after unloading	100 – 300
T_{22C}	Average exit rate of semitrailers with empty ITUs to customer	0 – 2
T_{23C}	Average exit rate of full semitrailers with full ITUs to customer	0 – 6
T_{24C}	Average exit rate of semitrailers with ITUs to port	0 – 2

Table 3-10 mfs-MFS of Continuous Transitions Characterizing the Scenarios.

	mfs – MFS of a subset of continuous transitions [ITUs/h]					
Scenario	T_{1C}	T_{2C}	T_{3C}	T_{22C}	T_{23C}	T_{24C}
1	0.25–0.25	1.16–1.16	1.92–1.92	0–2.00	0–6.00	0–2.00
2	0.37–0.37	1.74–1.74	2.88–2.88	0–3.00	0–9.00	0–3.00

Note that in Table 3-10 two sets of possible ranges are reported. As discussed in the section devoted to numerical simulations, such values depend on the considered operating scenario. These values have been argued based on the information on the system behavior and on a series of historical data relative to year 2014 provided by GTS.

B. The Terminal Management

In this sub-section two IFTT management strategies are presented based on FOHPNs and using two different objective functions.

On one hand, the aim is to maximize the outflows of the terminal using the following objective function:

$$J_1 = a^T \cdot v, \quad where\ a_j = \begin{cases} 1\ if\ t_j\ is\ an\ exogenous\ transition \\ 0\ if\ t_j\ is\ an\ endogenous\ transition, \end{cases} \qquad \text{(Eq. 3-5)}$$

v is the IFS vector, and the exogenous transitions are those associated with the transferring of ITUs from/to the storage area of the terminal and the endogenous transitions are the remaining ones.

On the other hand, the aim is the minimization of the residual fluid in the storage area by objective function [177]:

$$J_2 = a^T \cdot v, \quad where\ a_j = \begin{cases} \sum_{p \in P_{Yard}} C(p,t_j) if\ t_j \in P_{Yard}^{(c)} \cup {}^{(c)}P_{Yard} \\ 0 \qquad\qquad\qquad\qquad otherwise \end{cases} \qquad \text{(Eq. 3-6)}$$

$$J_2 = a^T \cdot v, where\ a_j = \begin{cases} \sum_{p \in P_{Yard}} C(p,t_j)\ if\ t_j \in P_{Yard}^{(c)} \cup {}^{(c)}P_{Yard} \\ \qquad 0 \qquad\qquad otherwise \end{cases}$$

and P_{Yard} is the set of places modeling the yard storage area.

C. Numerical Results

In this section the results of two series of simulations are reported. The outcomes are obtained by in the MATLAB environment using the HYPENS tool [170].

The first one aims at validating the model, namely showing that the proposed FOHPN model, when no control action is applied, actually reproduces the behavior of the real system, so that it can be used as a reference model to derive efficient control actions. The second series of simulations aims at showing the effectiveness of the proposed control laws both in standard operating conditions and in critical cases, when congestions in the yard may potentially occur.

To show that the model mimics the behavior of the terminal when no control action is applied it is considered the AS-IS scenario, called Scenario 1. It corresponds to the operating conditions (in terms of flows of ITUs, semitrailers, timetable of trains, and so on) in which the system actually works. To simulate it, it is assumed that all transitions that could be controlled, as described in the previous section, fire at a random speed. In more detail, the values of the instantaneous firing speeds, rather than being the result of an optimization problem, are taken randomly, according to a uniform distribution in their admissible ranges (reported in the first row of Table 3-10). To make the situation more realistic, the PN in Figure 3-26 is integrated with a discrete place with a deterministic transition in self-loop, completely disjoint from the rest of the net. The time delay of such a transition is taken equal to 0.25 hours, so that it fires with a frequency of 4/hour. This imposes that macro-events occur at least at a frequency equal to such a value, thus new values of continuous transitions are randomly selected with a sufficiently high frequency.

The duration of the simulations is considered to be of 8760 hours, corresponding to one year of activity and the initial marking corresponding to the empty terminal as represented in Figure 3-26. The single mean test is applied, taking into account 100 replications with a 95% confidence. Validation is carried out considering three performance indices: 1) the average free capacity of the yard (A_{YS}); 2-3) the number of entering (IN_{YS}) and exiting (OUT_{YS}) ITUs in/from the yard. These parameters values are computed via simulation and are compared with those provided by the company, denoted as: A_{YSReal}, IN_{YSReal}, and OUT_{YSReal}. The obtained results are summarized in the upper part of Table 3-11, where 1-OS (1-OR) denotes the scenario in open-loop via simulation (real terminal). It is to notice that the half width confidence interval of the indices values obtained via simulation is equal to 1%. Hence, it is possible to conclude that the FOHPN model mimics the real system behavior.

D.The Controlled System

The effectiveness of the proposed control laws is here shown by considering two different scenarios. Scenario 1 is the AS-IS scenario, already discussed in the previous subsection. In Scenario 2 it is supposed that the commercial flow of incoming ITUs is increased of 50%, to evaluate the effectiveness of the proposed management policies in a potentially congested situation. In particular, as summarized in Table 3-10 (second row) the firing speeds of transitions modeling the semitrailers entrance are increased of an amount equal to 50% with respect to Scenario 1, as well as the MFSs of transitions modeling the semitrailers exit.

Note that in Scenario 2 also the number of (empty and full) ITUs in the trains is increased of a 50% amount, so, with reference to Figure 3-26, it holds TB1=6, TB2=19, TB3=5, TP1=0, TP2=36, TP3=15, EB=21, FB=9, EP=11, FP=40.

Table 3-11. Performance Indices in Open-Loop.

Scenarios	Performance indices	
1-OS (Open-loop via simulation)	A_{YS} [ITUs]	232.16
	IN_{YS} [ITUs]	$5.79 \cdot 10^4$
	OUT_{YS} [ITUs]	$5.79 \cdot 10^4$
1-OR (Open-loop, real system)	A_{YS} [ITUs]	230
	IN_{YS} [ITUs]	$5.77 \cdot 10^4$
	OUT_{YS} [ITUs]	$5.77 \cdot 10^4$
2-OS (Open-loop via simulation)	A_{YS} [ITUs]	10.6
	IN_{YS} [ITUs]	1.256
	OUT_{YS} [ITUs]	1.246

Table 3-12 summarizes the performance indices used to validate the effectiveness of the control in the two closed-loop scenarios (1-CS and 2-CS). Note that here, together with the three indices used in the previous subsection, a fourth index is considered: the minimum value of the free capacity of the yard storage area (*min*($m_{P_{9C}}$)). The last two columns in Table 3-12 report, respectively, the performances indices values obtained by optimizing the outflow of the terminal (J_1) and the residual fluid in the storage area (J_2). The following considerations can be drawn.

In Scenario 1 under control (Scenario 1-CS in Table 3-12) the IFTT appears to be well organized and can well face the requests of the customers, having a yard storage area that never congests. In particular, it is possible to observe that A_{YS}, i.e., the average free capacity available in the yard, is equal to 236.71 ITUs in the case of maximization of the outflows (J_1) and is equal to 238.89 ITUs in the case of minimization of the ITUs in the yard (J_2), not too far from the maximum value equal to 250 ITUs. As expected, better results in terms of A_{YS} are obtained when minimizing J_2, even if differences are quite negligible in the two cases.

The value of the performance index *min*($m_{P_{9C}}$), shows that in both cases the system never congests. Finally, the number of entering ITUs (IN_{YS}) is equal to $4.33 \cdot 10^5$ when considering J_1, and is equal to $2.95 \cdot 10^5$ when considering J_2, so as the number of exiting ITUs (OUT_{YS}) is equal to $4.33 \cdot 10^5$ when dealing with J_1 and is equal to $2.95 \cdot 10^5$ when dealing with J_2. Such values reflect the expected results because in case of outflows maximization the goal is to transfer as many ITUs as possible, while in the second case the objective is to maximally reduce the number of ITUs waiting in the yard

It is also possible to compare such values of performance indices with those obtained when the system evolves in open-loop (1-OS and 1-OR in Table 3-11). In particular, it is possible to notice that basically the same values of free capacity in the yard are obtained in the open-loop and in the closed-loop case. However, in the closed-loop case, such values correspond to much higher values of flows (almost in the ratio 5 to 1).

In Scenario 2 (2-CS in Table 3-12) it is possible to observe a decrease in A_{YS} with respect to Scenario 1 both in the case of J_1 and in the case of J_2, even if differences are quite negligible. More significant differences occur in the values of *min*($m_{P_{9C}}$) with respect to the previous scenario. Obviously, also under this scenario a better performance with respect to such parameters is obtained by minimizing J_2. Conclusions similar to those in the previous scenario can be drawn looking at the other performance indices. As a result, it is possible to claim that both the considered management strategies allow coping with the higher traffic commercial flows, preventing congested situations and supporting the offline decision maker planning. To further validate the effectiveness of the control policies in such a critical scenario, the open-loop system has been simulated as described in the previous section with the only difference that the ranges of continuous transitions in Table 3-10 are taken from the second row of the table rather the first one. The resulting values of the performance indices are summarized in the last part of Table 3-11 (2-OS). As it can be seen the yard storage immediately reaches a congested state (its free capacity is almost null) so the flow of ITUs is negligible during most of the simulation time.

This section is concluded with some important remarks concerning the use of the model in real time. In particular, it is remarkable that the *simulation time,* i.e., the time needed to perform a simulation for the considered observation time window of one year, is only about 10 min for the system shown in Figure 3-26 on a PC equipped by an Intel Core 2 Duo-2.80 GHz processor and 4 GB RAM. Moreover, in order to assess the usability of the model in an online control, it is evaluated the *optimization time,* i.e., the average time needed by this technique to optimize the firing speeds to be associated with each continuous transition at each macro-event. The *optimization time* results equal to 0.0001 seconds which validates the applicability of the method in real time. Of course, the application of one of the two control policies to the system in real time should take into account that the system state should be monitored and (short) regular time intervals would need to be considered to allow simulating the terminal and calculating the optimal speeds for the closed-loop policy application in real time. For instance, the simulations relative to a time window of one week require about 21 seconds on the same machine. Hence, all the recalled indices show the applicability of the model both to take decisions offline and in real time applications.

Table 3-12. Performance Indices in Closed-Loop.

Scenarios	Performance indices	J_1	J_2
1-CS (Closed-loop, via simulation)	A_{YS} [ITUs]	236.71	238.89
	$min(m_{P9c})$ [ITUs]	223.94	208.79
	IN_{YS} [ITUs]	$4.33 \cdot 10^5$	$2.95 \cdot 10^5$
	OUT_{YS} [ITUs]	$4.33 \cdot 10^5$	$2.95 \cdot 10^5$
2-CS (Closed-loop, via simulation)	A_{YS} [ITUs]	232.6	235.19
	$min(m_{P9c})$ [ITUs]	119.35	119.97
	IN_{YS} [ITUs]	$4.55 \cdot 10^5$	$3.64 \cdot 10^5$
	OUT_{YS} [ITUs]	$4.55 \cdot 10^5$	$3.64 \cdot 10^5$

3.4 Intermodal terminal planning by TPNs and Data Envelopment Analysis

This section presents an innovative planning technique for IFTTs which is based on the integration of the TPNs modelling framework with the cross-efficiency Data Envelopment Analysis multi-criteria optimization technique. The proposed procedure allows solving some resource planning problems typically occurring in IFTTs. In effect, in the last decades, the use of decision making procedures utilizing Multi-Criteria Decision Making (MCDM) methods, as the DEA technique, has seen an increasing use in different application areas (see the review in [178]). Several MCDM approaches are available, each with its own advantages and disadvantages [178]. Most MCDM techniques are usually applied prior to decision making or project execution, while DEA is more often utilized for the evaluation of schemes already implemented [179], as is the case of resource planning in IFTTs. Furthermore, DEA acknowledges between its advantages, its ease of use and ability to quantify results, thus simplifying the analysis. The main advantages of exploiting the TPN modelling power joined with the cross-efficiency DEA technique is the ability to evaluate the efficiency of different possible alternatives in the resource planning when a variation in the commercial flows is foreseen. In particular, the focus here is on how to properly dimension the number of resources required to transfer ITUs, and the capacity and frequency of the transportation means, to address the nominal freight flows, as well as their eventual foreseen increments. In this section is evaluated the effectiveness of the alternatives based on how they influence some suitably chosen indices of the terminal performance, e.g., throughput, emissions, cost, etc. As a result, the presented technique allows decision makers to perform what-if analyses as well as to provide quick and effective information on how to address the typical problems related to the resource planning and management of intermodal terminals.

Simulations are conducted on the rail-road terminal of the GTS (already presented in the previous sections) using data provided by the company. Differently from the previous sections, here it is presented an extended model that drops some simplifying assumptions considered for the preceding applications (i.e., unlimited number of resources available to transfer ITUs, which is not the case of a real terminal, and a unique yard storage area which is also uncommon in a real terminal that typically exhibits a storage area partitioned into sub-areas devoted to the storage of ITUs with different destinations) and at the same time allows performing a realistic description of the ITUs transfer operations, without demanding a particularly high computational effort. More in detail, through Monte Carlo simulations, the terminal TPN model is evaluated by simulating multiple different scenarios: (1) the current situation based on real data provided by the GTS company, (2) some potentially critical situations (obtained by assuming an increase in the freight flows), and (3) other scenarios in which specific actions are to be implemented in order to cope with critical circumstances. On the basis of such simulations, conclusions can be automatically drawn on critical topics of the network, possible bottlenecks, under-utilized resources or resources to be supplemented using the DEA MCDM approach. It is important to remark that, although the GTS is still testing the proposed integrated methodology, some promising improvements in terms of usefulness in the resource planning decision making process are being observed.

3.4.1 Basics of Data Envelopment Analysis and cross-efficiency DEA

Thanks to its robustness and simplicity of application, the Data Envelopment Analysis approach [180] is a commonly adopted technique to compare a set of alternatives with considerably different and heterogeneous operating characteristics under multiple and conflicting criteria. In simple words, DEA is a technique aimed at determining the efficiency of each alternative in order to define a ranking among them.

In more detail, a set of F alternatives is to be evaluated on the basis of n conflicting criteria, quantified via appropriate performance indices divided into a subset of K criteria to be maximized and H criteria to be minimized, being $K+H=n$. Now, focusing on the i-th alternative (i.e., scenario in the next secions) (i=1,..., F), let $y_{k,i}$ be the value of the k-th performance index to be maximized (k=1,…,K) and $x_{h,i}$ be the value of the h-th performance index to be minimized (h=1,2,…,H). In particular, these values can be computed based on the knowledge of the system behavior in the i-th scenario, which could result from the observation of the real system or from numerical simulations carried out on a mathematical model of the system. Then, let $u_{k,i}$ and $v_{h,i}$ be the weighting coefficients associated, respectively, with the k-th performance index to be maximized and the h-th performance index to be minimized. Such weights, as explained in the following, are used to evaluate the efficiency of the i-th alternative.

The efficiency of the i-th alternative is typically defined as the ratio between the weighted sum of the values of the performance indices to be maximized and the weighted sum of the values of the performance indices to be minimized [180]:

$$E_i = \frac{\sum_{k=1}^{K} u_{k,i} \cdot y_{k,i}}{\sum_{h=1}^{H} v_{h,i} \cdot x_{h,i}} \qquad \text{(Eq. 3-7)}$$

where the non-negative weighting coefficients result from the solution of the following optimization problem:

$$\max \frac{\sum_{k=1}^{K} u_{k,i} \cdot y_{k,i}}{\sum_{h=1}^{H} v_{h,i} \cdot x_{h,i}} \quad \text{(Eq. 3-8)}$$

subject to (s.t.):

$$\frac{\sum_{k=1}^{K} u_{k,i} \cdot y_{k,i}}{\sum_{h=1}^{H} v_{h,i} \cdot x_{h,i}} \leq 1, \quad \text{(Eq. 3-9)}$$

$$u_{k,i}, v_{h,i} \geq 0, \quad k = 1,2,\ldots,K, \quad h = 1,2,\ldots,H. \quad \text{(Eq. 3-10)}$$

Note that constraint (Eq. 3-9) imposes that the efficiency cannot be greater than one. In particular, an alternative is considered efficient if and only if E_i=1, otherwise it is not efficient. Due to the nonlinearity of the above optimization problem, determining the weighting coefficients, and consequently evaluating the efficiency of the current scenario, is computationally demanding in the case of several criteria. A solution to overcome such a limitation consists in the so-called output-oriented method [180]. According to this method, the weighting coefficients are computed solving the following linear optimization problem:

$$\max \sum_{k=1}^{K} u_{k,i} \cdot y_{k,i} \quad \text{(Eq. 3-11)}$$

s.t.:

$$\sum_{k=1}^{K} u_{k,i} \cdot y_{k,i} - \sum_{h=1}^{H} v_{h,i} \cdot x_{h,i} \leq 0 \quad \text{(Eq. 3-12)}$$

$$\sum_{h=1}^{H} v_{h,i} \cdot x_{h,i} = 1 \quad \text{(Eq. 3-13)}$$

$$u_{k,i}, v_{h,i} \geq 0, \quad k = 1,2,\ldots,K, \quad h = 1,2,\ldots,H. \quad \text{(Eq. 3-14)}$$

Note that, due to constraint (Eq. 3-13), the optimal value of the above optimization problem still coincides with the value of the efficiency defined in (Eq. 3-7).

The limitation of the traditional DEA method is that it places no constraints, other than positivity, on weights, thus allowing the assessment of an alternative's efficiency using the set of weights that is most favorable to that alternative. For this reason, the classical DEA approach is often used only in a pre-evaluation phase [181], while authors proposed several procedures to better discriminate among alternatives [182]. The most common is the so-called cross-evaluation approach [183], which includes both a self- and a peer-evaluation of the alternatives, each of which is not only assessed by its own weights but also by those of all other alternatives. In more detail, the coefficients resulting from maximizing the efficiency of each alternative are also used to determine the efficiency of all the others. Thus, each alternative is measured via F relative efficiency values (each with respect to one of the others, including itself) and the resulting cross-efficiency is the mean value of them. Formalizing, a *cross-efficiency matrix* **CE**=$\{E_{f,i}\} \in (\Re^{+})^{FxF}$ is determined, whose value $E_{f,i}$ represents the efficiency of the f-th alternative calculated with the most favorable weights of the i-th competing alternative (obtained from optimization problem (Eq. 3-11)-(Eq. 3-14)):

$$E_{f,i} = \frac{\sum_{k=1}^{K} u_{k,i} \cdot y_{k,f}}{\sum_{h=1}^{H} v_{k,i} \cdot x_{h,f}} \quad with\ i = 1,2,...,\ F \qquad \text{(Eq. 3-15)}$$

and the *cross efficiency* of the i-th alternative is obtained as:

$$CE_i = \frac{1}{F}\sum_{f=1}^{F} E_{f,i}. \qquad \text{(Eq. 3-16)}$$

The above approach still contains a limitation, indeed the solution of the linear optimization problem (Eq. 3-11)-(Eq. 3-14) is not unique in general. This clearly does not affect the values of $E_{i,i}$ but has an impact on the values of $E_{f,i}$, $f \neq i$. To solve this issue, Doyle and Green [184] propose a second-level optimization procedure that should be executed for all the i-th alternatives after solving the problem (Eq. 3-11)-(Eq. 3-14). It consists in the solution of a second optimization problem for each alternative i (i=1,...F), which enables to compute a new set of weighting coefficients $u_{k,i}$, $v_{h,i}$, k=1,...,K, h=1,...,H, to be used in eq. (Eq. 3-15) to redefine the cross-efficiency matrix:

$$\max \sum_{k=1}^{K} u_{k,i} \left(\sum_{f=1, f\neq i}^{F} y_{k,f} \right) \qquad \text{(Eq. 3-17)}$$

s.t.:

$$\sum_{h=1}^{H} v_{h,i} \left(\sum_{f=1, f\neq i}^{F} x_{h,f} \right) = 1 \qquad \text{(Eq. 3-18)}$$

$$\sum_{k=1}^{K} u_{k,i} y_{k,i} - E_i \sum_{h=1}^{H} v_{h,i} x_{h,i} = 0 \qquad \text{(Eq. 3-19)}$$

$$\sum_{k=1}^{K} u_{k,i} \cdot y_{k,f} - \sum_{h=1}^{H} v_{h,i} \cdot x_{h,f} \leq 0, \quad f = 1,...,F;\ f \neq i \qquad \text{(Eq. 3-20)}$$

$$u_{k,i}, v_{h,i} \geq 0, \quad k = 1,2,...,K, \quad h = 1,2,...,H. \qquad \text{(Eq. 3-21)}$$

Note that this second optimization only leads to an update of the off-diagonal terms of the cross efficiency matrix. On the contrary, the diagonal elements do not change as a consequence of constraint (Eq. 3-21) which imposes that, for each i-th alternative, E_i is equal to the optimal efficiency resulting from the previous optimization problem (Eq. 3-11)-(Eq. 3-14).

Depending on the decision maker's preferences, it is possible to solve the problem under additional constraints on the evaluating criteria. This can be done by using the so-called assurance regions constraints [185], which apply additional constraints on the relative magnitude of the weighting coefficients to emphasize the importance of some criteria on the others.

Summing up, the main steps of the cross-efficiency DEA technique may be described as follows:

1. determine F scenarios or alternatives to compare. Collect data on them either by an experimental campaign on the real system or by a simulation campaign on a system model. Compute the performance index values $y_{k,i}$ and $x_{h,i}$ with i=1,…,F, k=1,…,K and h=1,…,H.
2. for each scenario i=1,…,F solve the linear optimization problem (Eq. 3-11)-(Eq. 3-14) and compute the efficiencies E_i based on the resulting weighting coefficients and eq. (Eq. 3-7).
3. for each scenario i=1,…,F solve the second-level optimization problem (Eq. 3-17)-(Eq. 3-21) and compute the cross-efficiencies CE_i based on (Eq. 3-15) and (Eq. 3-16).
4. order the F scenarios in a decreasing order of cross-efficiency values: the most (less) efficient one is the one with the greatest (lowest) cross-efficiency value.

3.4.2 State of the Art

This section addresses the IFTT resource planning. In particular, the complexity of IFTTs and the large number of sub-activities that compose their workflows lead to the definition of sub-models to be composed and controlled to provide an efficient and effective freight transport service.

The literature review of the above Sections 2.1.8 and 3.2 highlights various advantages in the use of TPN formalisms when tackling IFTTs decision problems (basics on TPNs are reported in Section 2.1.1). In more detail, TPNs can be effectively used to simply represent workflows of container terminals, evaluate the IFTTs performance via simulations and what-if analysis, and control the behavior of the system. The employment of TPNs for resource planning problems is then useful for the evaluation of resolution actions in case of critical situations. However, such a formalism should be integrated with techniques able to identify the most appropriate action among the multiple ones available. The requirement of such an integration is exactly the motivation of this part of the research. Indeed, the proposed procedure integrates a cross-efficiency Data Envelopment Analysis technique to the TPN modelling technique in order to evaluate and rank different possible alternatives to be taken in the resource planning in the case of an estimated increase in the commercial flows. The alternatives are compared with each other and ranked according to their impact on some suitably chosen indices on the terminal performance, e.g., throughput, emissions, cost, etc. Therefore, the presented technique allows decision makers to evaluate the IFTT performance, as well as performing what-if analysis and providing useful information on how to address the typical problems related to resource planning and management.

3.4.3 The Real Case Study

The considered real case study consists in the already presented rail-road terminal of the "GTS - General Transport Service S.p.A" (for details see Section 3.2.4.B). Figure 3-27 graphically describes the logical scheme of the GTS terminal, showing the flows of both ITUs and transportation means in the IFTT from one subsystem to another. Specifically, the following subsystems are depicted: daily/weekly opening/closing, access roads, yard storage areas, and railways. Semitrailer trucks can arrive at the terminal via the access roads, and the transported ITUs can be unloaded in the yard or directly on the corresponding train, whereupon the trucks become immediately free to load a new cargo and then they leave the terminal. Similarly, trains arriving at the terminal can unload the cargo on the storage area or on an available semitrailer truck. After the discharge, trains can be loaded with ITUs picked-up from the yard or from trucks. The current management of the terminal relies on knowledge-based decisions. Skilled personnel of the company organize and optimize flows of ITUs and the use of the available resources. Until now, no standard procedure or decision support system has been implemented to improve the system behavior. Therefore, TPNs are used to evaluate the performance of the terminal resources under different operating scenarios and, combined with the cross-efficiency DEA technique, to suggest the most appropriate resolution in case of criticalities. First it is considered the nominal terminal behavior, that is the scenario based on data provided by the GTS.

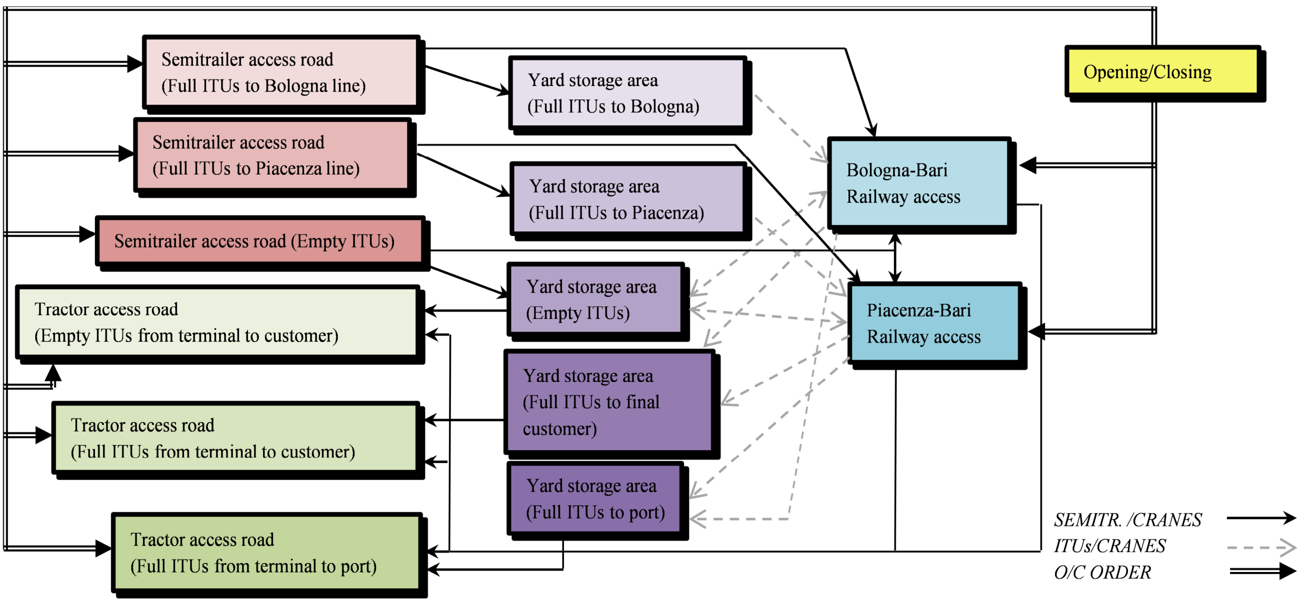


Figure 3-27. The logical scheme of the IFTT case study.

As specified in the introduction of this section, this case study has been already presented under some simplifying assumptions that are here removed to make the model much more realistic. More in detail, while Section 3.2 the number of cranes available to transfer ITUs is supposed to be infinite, here it is assumed finite, in addition the yard storage area is modelled as a whole, whereas in this section it is more realistically divided into sub-areas devoted to stock ITUs having different delivery destinations (as in Section 3.3). Finally, in Sections 3.2 and 3.3 it is assumed that, whenever ITUs are discharged from semitrailers, they are first stored in the yard, and then moved to the train, and vice-versa. On the contrary, it is more accurately modelled the fact that, when ITUs reach the terminal via a semitrailer, if the corresponding train is already available, no intermediate storage in the yard occurs, and ITUs are directly loaded on that train. The same happens for ITUs arriving via a train and to be loaded on a semitrailer.

After modelling the terminal in its current configuration and workflows (Scenario AS-IS), it is assumed that a 100% increase in the terminal commercial flows by road is expected for a duration of 90 days, due for instance to a new fruitful contract. The first goal is to evaluate if the actual resource planning is able to cope with such a critical situation, otherwise the amount of available resources should be properly modified. The possible actions consist in increasing the number of cranes used to transfer ITUs, the frequency of trains or the number of wagons for each train during the week, or a suitable combination of such actions. Note that, since the duration of the flows increase is assumed to be limited, no structural operation oriented at modifying the terminal infrastructure is taken into account. As a result, a number of possible alternative solutions able to cope with the planned increased flows are obtained. Furthermore, in order to evaluate the most suitable one among them, the cross-efficiency DEA technique is applied, thus providing the decision maker with a ranking of the feasible solutions in terms of the terminal performance and cost of the identified action.

3.4.4 The Timed Petri Net Model of the Case Study

Figure 3-28 shows the TPN model of the IFTT. It keeps the structure and colours of the logical scheme in Figure 3-27, while the subsystem blocks are replaced by the corresponding TPN subnets, which will be described in the following. It is to remark here that for sake of brevity some explanations on the system functioning are reported directly while describing the TPN model.

The basic operations performed in the terminal are the cyclical opening/closing necessary to control the activity/inactivity of the system for hours or for days. Figure 3-28 includes one opening/closing (hours) subnet: it is composed by 2 places (p_1, p_2) and 2 deterministic transitions (t_1, t_2). When p_1 (p_2) is marked, the terminal is opened (closed). The time of activity (inactivity) is represented by the time delay associated with t_1 (t_2) and is equal to 12 hours (in both cases). Three subnets for opening/closing (days) activities are also present in Figure 3-28. The first one is connected to the opening/closing (hours) subnet. The other two are connected to the Bologna and the Piacenza railway access. For sake of brevity, it is given a detailed description of the first one, which holds also for the second one, with an appropriate change of nomenclature. The presence of 6 tokens (modelling the 6 working days) in p_3 allows the immediate firing of transition t_3 and leads to the addition of one token in p_4. A time delay of one day (modelling the Sunday closure) is associated with t_4. Consequently, the firing of t_4 corresponds to the beginning of a new working week. It should be noticed that in the subnet connected to the Bologna railway access, the weight of the arc from place p_{24} to transition t_{27} is equal to 3, rather than 6. This depends on the fact that trains arrive at alternate days, consequently a new token in p_{24} means that 2 days are elapsed, rather than one day as in the Piacenza case. Therefore, 6 working days are represented by 3 tokens in p_{24}.

The yard storage area is modelled in Figure 3-28 by a subnet composed by the following seven places: p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}. The first six places model the six subareas of the yard, while the last one models the total residual capacity. Hence, at any time instant it holds: $\boldsymbol{M}(p_{10}) + \boldsymbol{M}(p_{11}) + \boldsymbol{M}(p_{12}) + \boldsymbol{M}(p_{13}) + \boldsymbol{M}(p_{14}) + \boldsymbol{M}(p_{15}) = C_{YS}=250$. In Figure 3-28 the initial marking of these places is as follows: $\boldsymbol{M}(p_{10})=6$; $\boldsymbol{M}(p_{11})=21$; $\boldsymbol{M}(p_{12})=27$; $\boldsymbol{M}(p_{13})=0$; $\boldsymbol{M}(p_{14})=0$; $\boldsymbol{M}(p_{15})=196$, which means that at the beginning of the activities the terminal is not completely empty.

The connection of the terminal with the road side is represented by two subnets in Figure 3-28. More in detail, two access roads are modelled, which correspond to the entrance and the exit of the semitrailers. The subnet modelling the access road for the entrance of semitrailers is composed of the following five places: p_5, p_6, p_7, p_8, p_9, and six transitions: t_6, t_7, t_8, t_9, t_{10}, t_{11}. The presence of a token in p_5 represents the consensus to a semitrailer to enter the terminal during its opening periods. Stochastic transitions t_6, t_7, and t_8 model the arrival of semitrailers containing ITUs directed to different destinations: Bologna (full ITU), any destination (empty ITU), and Piacenza (full ITU). Such ITUs may either be stored in the yard (*Case 1*), or loaded in the corresponding train without waiting in the yard (*Case 2*). More in detail, the two alternatives are as follows: *(Case 1)* the unload of the ITUs in the different storage areas is modelled by transitions t_9 t_{10}, and t_{11}, which represent the entrance of empty tractors in a waiting area (p_9); *(Case 2)* transitions t_{22}, t_{24}, t_{39}, and t_{41}, belonging to the subnets modelling the railway access, fire when the train and its cargo (empty and full ITUs) are ready to be loaded.

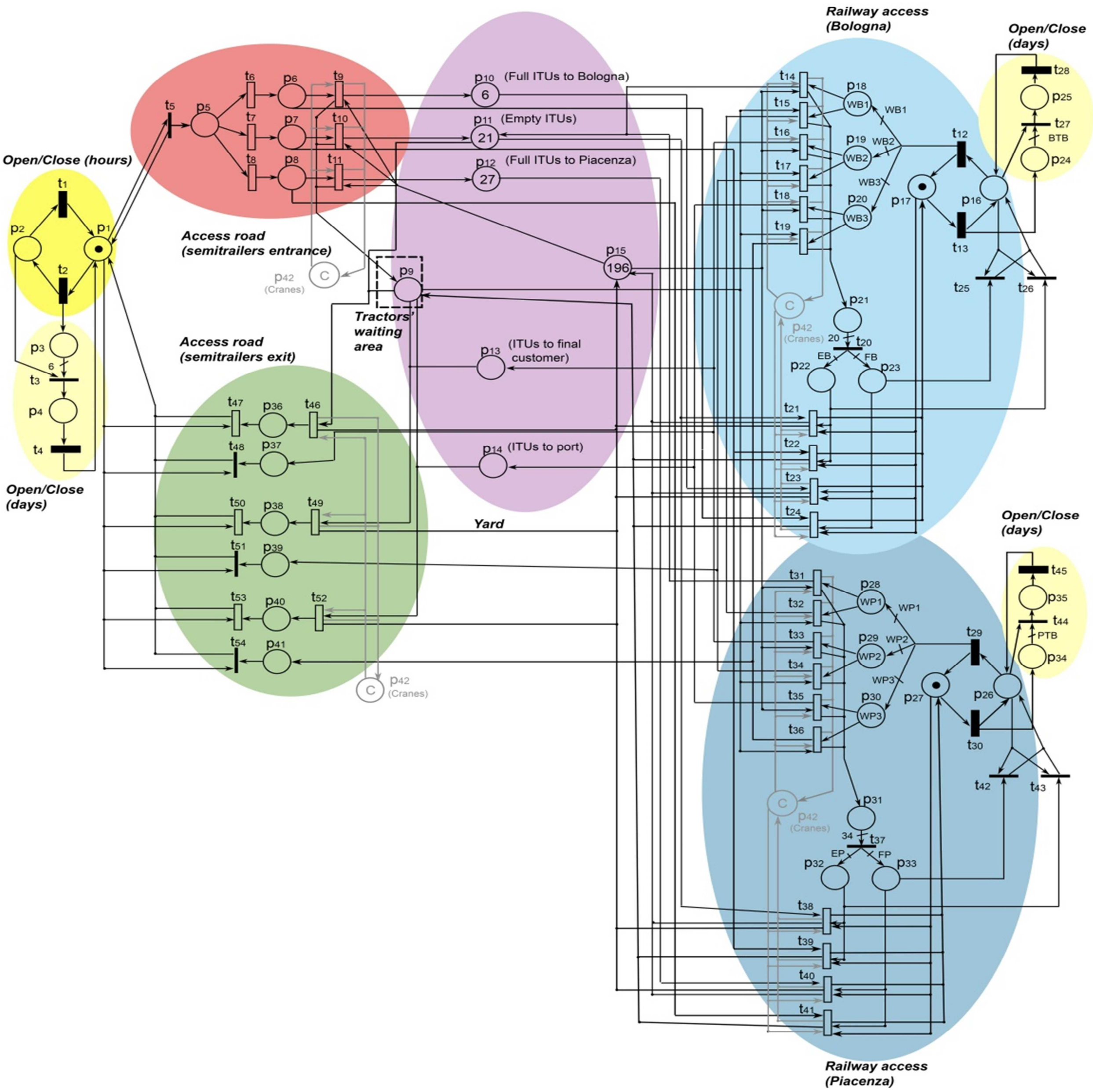


Figure 3-28. The TPN model of the IFTT case study in **Figure 3-27**.

Their firings remove tokens corresponding to semitrailers from places p_6, p_7, and p_8 and increase the content of p_9 corresponding to free tractors. The subnet modelling the access road for the exit of semitrailers is composed by the following six places: p_{36}, p_{37}, p_{38}, p_{39}, p_{40}, p_{41}, six stochastic transitions: t_{46}, t_{47}, t_{49}, t_{50}, t_{52}, t_{53}, and three immediate transitions: t_{48}, t_{51}, t_{54}. When the terminal is open, semitrailers exit from the terminal (t_{47}, t_{48}, t_{50}, t_{51}, t_{53}, t_{54}) only after ITUs and tractors are assembled (t_{46}, t_{49}, t_{52}). In more detail, transitions t_{47}, t_{50}, and t_{53} model the assembly of tracks using ITUs that are picked up from the yard (analogous to previous *Case 1*) while transitions t_{48}, t_{51}, and t_{54} model the assembly of tracks using ITUs that are picked up directly from the train (analogous to previous *Case 2*). In each of the above cases (*1* and *2*) three transitions are used to model three possible situations for the exiting semitrailers: the destination is the port and ITUs are full, the destination is the final costumer and ITUs are either full or empty. A remark should be made concerning place p_9 modelling the tractors' waiting area. No upper bound is imposed on the marking of this place although the capacity of the waiting area of tractors is obviously finite. The hypothesis holds, since time spent by tractors in such area is negligible and no accumulation may occur. The connection of the terminal with the railway side is represented by two subnets in Figure 3-28, one connecting the terminal to Bologna and the other one to Piacenza. For sake of brevity, in the following it is given a detailed description of the Bologna railway access, which holds also for the Piacenza railway access, with an appropriate change of nomenclature. Two places (p_{16} and p_{17}) and two transitions (t_{12} and t_{13}) are used to model the presence (p_{17}) or absence (p_{16}) of the train in the terminal. As soon as transition t_{12} (modelling the arrival of the train) fires, WB1, WB2 and WB3 tokens enter, respectively, places p_{18}, p_{19}, and p_{20}. Such quantities are equal to the number of ITUs (empty or full) addressed to different destinations. Places p_{18}, p_{19}, and p_{20} have two output transitions each. In particular, as in the case of the access roads, ITUs may be temporarily stored in the yard, or may immediately leave the terminal via a semitrailer. Every time one of the transitions t_{14}, t_{15}, t_{16}, t_{17}, t_{18}, and t_{19} fires, a new token is added to place p_{21}. As soon as the marking of p_{21} becomes equal to 20, which means that the train is now empty, the immediate transition t_{20} fires increasing the marking of places p_{22} and p_{23} of an amount respectively equal to EB and FB. This represents the fact that the empty train can accommodate EB empty ITUs, and FB full ITUs. Transitions t_{21}, t_{22}, t_{23}, and t_{24} model the loading of ITUs on the train. Finally, transitions t_{25} and t_{26} are used to reset to zero the free available capacity of the train in the case that the train has to leave the terminal without being full load.

Table 3-13. Time delay / average time delay of transitions in Figure 3-28.

Transition	Description	Time/average time delay [h]
t_1	Hours of activity of the terminal	12.00
t_2	Hours of closure of the terminal	12.00
t_4	Sunday closure	12.00
t_6	Average arrival time of semitrailers with full ITUs to Bologna	4.00
t_7	Average arrival time of full semitrailers with empty ITUs	1.7
t_8	Average arrival time of semitrailers with full ITUs to Piacenza	1.00
t_9, t_{10}, t_{11}, t_{14}, t_{15}, t_{16}, t_{17}, t_{18}, t_{19}, t_{21}, t_{22}, t_{23}, t_{24}, t_{31}, t_{32}, t_{33}, t_{35}, t_{36}, t_{38}, t_{39}, t_{40}, t_{41}, t_{46}, t_{49}, t_{52}	Average time needed by a crane to transfer an ITU	0.17
t_{13}	Dwell time of the train Bari-Bologna	10.00
t_{12}	Absence time of the train Bari-Bologna	38.00
t_{30}	Dwell time of the train Bari-Piacenza	10.00
t_{29}	Absence time of the train Bari-Piacenza	14.00
t_{28}	Weekly stop for the Bari-Bologna train	24.00
t_{45}	Weekly stop for the Bari-Piacenza train	24.00

This situation should be taken into account when modeling the system since trains time scheduling imposes mandatory constrains. Consequently, if such an undesirable case occurs, it is important to reset the capacity of the train in order to avoid affecting the next loading.

It is worth noting that in the net of Figure 3-28 is represented a grey colored place p_{42}. It symbolizes the number of cranes in the terminal for the ITUs transferring operations. Note that such a place is replicated only for the sake of clarity. Finally, Table 3-13 summarizes the (average) time delays of transitions in Figure 3-28 and describes their physical meaning.

3.4.5 Performance Evaluation and Resource Planning of the Case Study

The described TPN model is now used to simulate and analyze the behavior of the GTS IFTT. Firstly, the model is validated by using a Monte Carlo approach and the single mean test ([155]), then it is shown how, combined with the cross-efficiency DEA, it can be used for addressing the IFTT resource planning by evaluating different possible alternative scenarios.

A. The TPN model simulation in nominal conditions

In order to evaluate the terminal behavior, a series of performance indices are considered ([99]), which are listed and described in Table 3-14. In order to validate the presented TPN model, the values obtained via simulation (over 100 replications) for the performance indices listed in Table 3-14 are compared with the corresponding historical data over one year of activity provided by the company (that is, the Scenario AS-IS, as described in the following).

Table 3-14. The performance indices.

Index name	Index meaning
O_{YB}	Average occupation of the yard sub-area dedicated to ITUs with destination Bologna
O_{YP}	Average occupation of the yard sub-area dedicated to ITUs with destination Piacenza
O_{YFC}	Average occupation of the yard sub-area dedicated to ITUs with destination Final Customer
O_{YPO}	Average occupation of the yard sub-area dedicated to ITUs with destination Port
O_{YE}	Average occupation of the yard sub-area dedicated to empty ITUs
F_{YS}	Average free available space in the yard storage area (i.e., average number of ITUs that can be still accommodated in the yard)
min F_{YS}	Minimum value of free available space in the yard storage area
U_C	Average value of free cranes (i.e., a measure of the cranes utilization)
TH	Sum of the average throughput related to the railway outflows
TM	Time (in hours) before the occurrence of a congestion in the terminal

More in detail, denoting by *PI* the generic performance index provided by the simulation, by *RPI* the corresponding historical index, and ρ the relative half width of the confidence interval, for each considered performance index it is verified that it holds $PI - \rho \leq RPI \leq PI + \rho$, where ρ is equal to 0.1% of *PI* in the worst case, which demonstrates that the model well mimics the actual system. It is worth noting that the input variables for the TPN model (i.e., the time delay of each stochastically timed transition) is determined by means of statistical evaluations on the historical data about one year of activity provided by the company.

It is to recall here that each simulation covers one year of evolution of the IFTT. In particular, it is assumed that each time unit of the simulation corresponds to one hour of evolution of the system. All simulations are obtained using MATLAB HYPENS ([170]). On average, the computation time of each replication equals 80 seconds in the worst case on a PC with an Intel Core 2 Duo-2.80 GHz processor and 4 Gb RAM.

The main parameters characterizing the IFTT system and its inputs in nominal operating conditions (Scenario AS-IS) are summarized in Table 3-15 (second column). In particular:

- t_6, t_7, and t_8, model, respectively, the arrival of semitrailers with ITUs directed to the Bologna line, the arrival of semitrailers with empty ITUs, and the arrival of semitrailers with ITUs directed to the Piacenza line;
- t_{47}, t_{50}, and t_{53}, model, respectively, the exit of semitrailers with ITUs directed to the final customer, the exit of semitrailers with empty ITUs, and the exit of semitrailers with ITUs directed to the port;
- t_{12} and t_{13}, model, respectively, trains absence and permanence times;
- t_{28} models the weekly stop for the Bari-Bologna train;
- WB1, WB2, WB3, EB, FB, WP1, WP2, WP3, EP, and FP model the capacities of the trains. More in detail, WB1 (WP1) and WB2 (WP2) represent the number of empty and full ITUs from Bologna (Piacenza) for the final customer, while WB3 (WP3) represents the total number of both empty and full ITUs from Bologna (Piacenza) to the port, EB (EP) and FB (FP) represent respectively the number of empty and full ITUs directed to Bologna (Piacenza);
- C models the number of available cranes;
- BTB models the frequency of trains per week arriving from (to) Bologna to (from) Bari.

Table 3-15. Characteristic parameters of the TPN defining the considered scenarios.

Parameter \ Scenario	AS-IS	S_{100}	S_{100}-C_5	S_{100}-C_3-BL_4	S_{100}-C_3-BW_{25}	S_{100}-C_3-PW_{40}
t_6[h]	4.0	**2.0**	**2.0**	**2.0**	**2.0**	**2.0**
t_7[h]	1.7	**0.85**	**0.85**	**0.85**	**0.85**	**0.85**
t_8[h]	1.0	**0.5**	**0.5**	**0.5**	**0.5**	**0.5**
t_{12} [h]	38.0	38.0	38.0	**14.0**	38.0	38.0
t_{13} [h]	10.0	10.0	10.0	**10.0**	10.0	10.0
t_{47}[h]	6.0	6.0	6.0	6.0	6.0	6.0
t_{50}[h]	0.5	0.5	0.5	0.5	0.5	0.5
t_{53}[h]	1.1	1.1	1.1	1.1	1.1	1.1
t_{28} [h]	24.0	24.0	24.0	**72.0**	24.0	24.0
WB1 [ITUs]	4.0	4.0	4.0	4.0	4.0	4.0
WB2 [ITUs]	13.0	13.0	13.0	13.0	13.0	13.0
WB3 [ITUs]	3.0	3.0	3.0	3.0	3.0	3.0
EB [ITUs]	14.0	14.0	14.0	14.0	**17.0**	14.0
FB [ITUs]	6.0	6.0	6.0	6.0	**8.0**	6.0
WP1 [ITUs]	0.0	0.0	0.0	0.0	0.0	0.0
WP2 [ITUs]	24.0	24.0	24.0	24.0	24.0	24.0
WP3 [ITUs]	10.0	10.0	10.0	10.0	10.0	10.0
EP [ITUs]	7.0	7.0	7.0	7.0	7.0	**8.0**
FP [ITUs]	27.0	27.0	27.0	27.0	27.0	**32.0**
C [cranes]	2.0	2.0	**5.0**	**3.0**	**3.0**	**3.0**
BTB [days/week]	3.0	3.0	3.0	**4.0**	3.0	3.0

Table 3-16. Performance indices values under scenarios AS-IS and S_{100}.

Scenario	O_{YB} [ITUs]	O_{YP} [ITUs]	O_{YE} [ITUs]	O_{YFC} [ITUs]	O_{YPO} [ITUs]	F_{YS} [ITUs]	min F_{YS} [ITUs]	U_C [ITUs]	TH [ITUs/h]	TM [h]
AS-IS	20.85	3.91	1.33	1.28	0.93	221.70	179.64	1.15	0.18	8,760
S_{100}	64.68	183.81	0.53	0.02	0.01	0.95	0.00	1.99	0.18	460

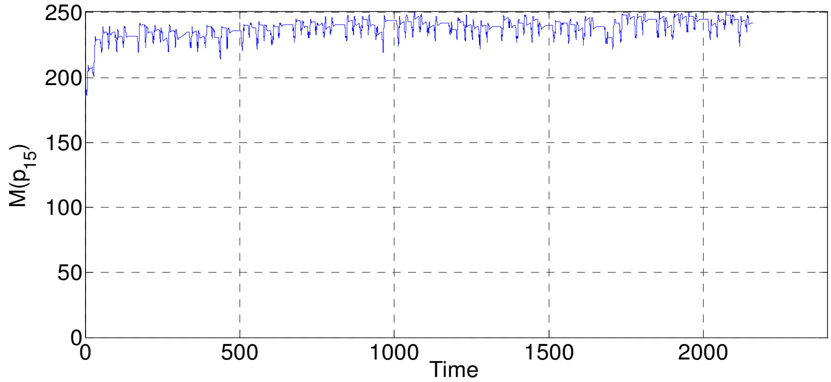


Figure 3-29. Yard free capacity $M(p_{15})$ in scenario AS-IS.

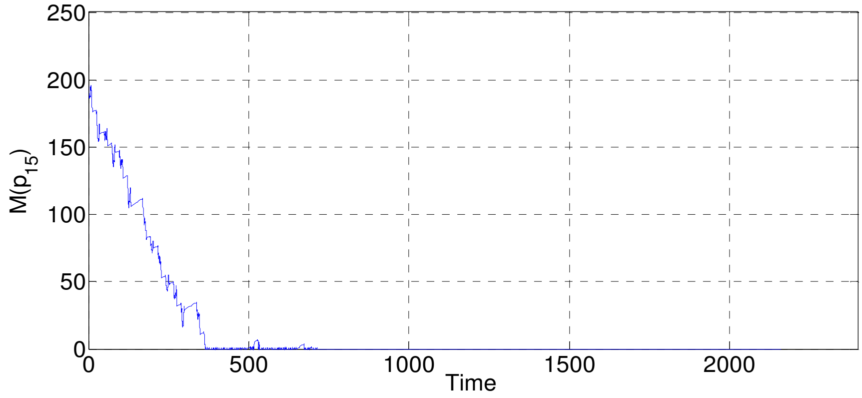


Figure 3-30. Yard free capacity $M(p_{15})$ in scenario S_{100}.

Table 3-15 (second row) summarizes the values of the obtained performance indices in nominal operating conditions (Scenario AS-IS). By analyzing such values, it is possible to conclude that the IFTT is well organized: the yard storage area never congests and the two cranes are sufficient to transfer the ITUs. In particular, it is possible to observe that F_{YS}, i.e., the average free capacity in the yard, is equal to 221.70 ITUs that is reasonable considering that its maximum value is C_{YS}=250 ITUs and is in accordance with Figure 3-29, which shows the evolution of the corresponding marking $\boldsymbol{M}(p_{15})$. Note that, in Figure 3-30 it is reported the marking evolution for 90 days of the terminal functioning to allow a visual comparison with the scenarios described in the next section.

Moreover, the last column in Table 3-16 reports the time elapsed (in hours) before a congestion occurs (TM), which in Scenario AS-IS equals 8,760 hours (or 365 days, which is the maximum value for the performed simulation), thus confirming that the terminal is well structured and organized for its actual flows.

B. The scenarios for the IFTT resource planning

As already stated, after modelling the terminal in its current configuration and workflows (Scenario AS-IS), it is assumed that a 100% increase in the commercial flows by road is expected for a duration of 90 days, due for instance to new contracts. It is to specify here that such an evaluation is performed with the specific purpose of analyzing the terminal behavior under heavy conditions and has been shared with the GTS board according to their estimates. In order to evaluate whether the actual resource planning in the IFTT is able to cope with such an increase in the flows it is analyzed the so-called Scenario S_{100} (i.e., the scenario with a 100% increase of means input flows by road). The characteristic parameters of Scenario S_{100} are reported in the third column of Table 3-15 (where the occurred changes with respect to the AS-IS scenario are highlighted in bold), while the obtained performance indices are in the third row of Table 3-16, showing that in this case the average free capacity of the yard is quite low. This is in agreement with Figure 3-30 reporting the evolution of the yard free capacity. Clearly, such a capacity drastically decreases, because of the significant increment in input flows via the street, reaching zero after 460 hours (about 19 days). In particular, it is possible to conclude that the most critical sub-areas are those corresponding to Bologna and Piacenza destinations, as it can be observed comparing indices O_{YB} and O_{YP}, in the third row of Table 3-16 with the original values in the second row. It is also possible to conclude that, since in S_{100} congestion is reached in less than 90 days, the current resource configuration and planning is unable to cope with the new estimated flows. Hence, additional resources are required in the IFTT, and the decision maker is required to analyze the impact of possible actions and to identify the most proper one, taking into account a cost-benefit analysis.

In the evaluated case study the possible actions consist in increasing either the number of cranes used to transfer ITUs, or the frequency of trains, or the number of wagons for each train during the week, or in carrying out a suitable combination of such actions. Among all these resolution actions, those which cause a congestion in the IFTT in less than 90 days are clearly useless in coping with the planned flows increase, while some others are unfeasible (for instance, in the case study there are already 6 trains a week for Piacenza, hence this value cannot be further increased). Hence, all these solutions are discarded and four different alternative scenarios to the AS-IS case are considered as summarized in Table 3-15 (columns 4 to 7).

In more detail, for each scenario in Table 3-15 the peculiarities of the analyzed case are pointed out using a representative nomenclature. In particular, letters C, BL, BW, and PW respectively stand for Cranes, Bologna Line, Bologna Wagons, and Piacenza Wagons; while the numbers that follow the above letters indicate the corresponding number of resources. Hence, the additional four scenarios are detailed as follows:

- S_{100}-C_5 denotes the scenario S_{100} (i.e., with a 100% increase of means input flows by road) in which the number of available cranes is equal to 5 (rather than 2);
- S_{100}-C_3-BL_4 denotes the scenario S_{100} in which the number of available cranes is equal to 3 (rather than 2) and the Bologna trains frequency per week is equal to 4 (rather than 3), assuming, without loss of generality, that such journeys are the first four days of each week;
- S_{100}-C_3-BW_{25} denotes the scenario S_{100} in which the number of available cranes is equal to 3 (rather than 2) and the number of wagons of each train from/to Bologna is 25 (rather than 20);
- S_{100}-C_3-PW_{40} denotes the scenario S_{100} in which the number of available cranes is equal to 3 (rather than 2) and the number of wagons of each train from/to Piacenza is 40 (rather than 34).

Among these scenarios, it is necessary to identify the best actions preventing the avoidance of the blocking situation caused by the 100% increase of the commercial flows. This evaluation could be carried out by comparing the marking evolution of the most critical resources of the terminal. However, as it emerges looking at figures from Figure 3-31to Figure 3-34, it would be non-trivial. More in detail, in such graphs it is reported the evolution of the resources that mostly suffer from the increase of the commercial flows. For each considered scenario, the evolution of $\boldsymbol{M}(p_{10})$, $\boldsymbol{M}(p_{12})$, $\boldsymbol{M}(p_{15})$, $\boldsymbol{M}(p_{42})$ are represented for 90 days of functioning of the terminal. As previously reported, $\boldsymbol{M}(p_{10})$ represents the occupation of the yard devoted to ITUs with destination Bologna, $\boldsymbol{M}(p_{12})$ represents the occupation of the yard devoted to ITUs with destination Piacenza, $\boldsymbol{M}(p_{15})$ the free available space in the yard storage area and $\boldsymbol{M}(p_{42})$ the cranes utilization. The values of such markings have different order of magnitude, hence, in the graphs the respective normalized trends are reported to allow an effective comparison of the scenarios. Furthermore, only four variables are considered to simplify the interpretation of the scenarios providing readable trends. Despite the adoption of these simplifying measures, an effective comparison of the scenarios appears still particularly difficult. In effect, in each graph the slopes and the trends of each variable are almost superimposable and do not give information decisive for a ranking of the alternatives.

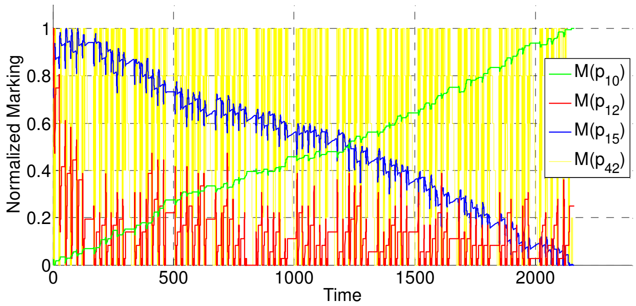


Figure 3-31. $\boldsymbol{M}(p_{10})$, $\boldsymbol{M}(p_{12})$, $\boldsymbol{M}(p_{15})$, $\boldsymbol{M}(p_{42})$ in scenario S_{100}-C_5.

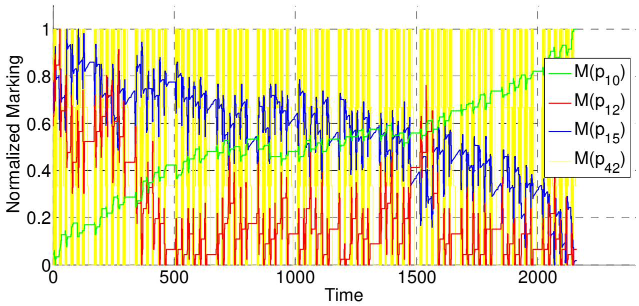


Figure 3-32. $\boldsymbol{M}(p_{10})$, $\boldsymbol{M}(p_{12})$, $\boldsymbol{M}(p_{15})$, $\boldsymbol{M}(p_{42})$ in scenario S_{100}-C_3-BW_{25}.

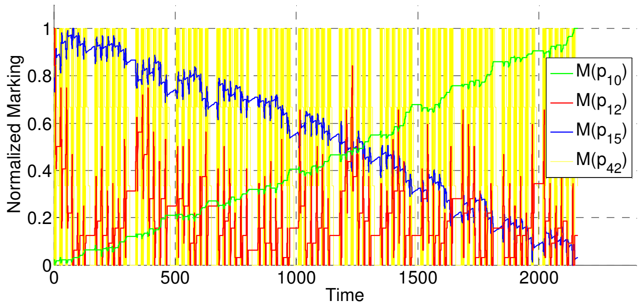


Figure 3-33. $\boldsymbol{M}(p_{10})$, $\boldsymbol{M}(p_{12})$, $\boldsymbol{M}(p_{15})$, $\boldsymbol{M}(p_{42})$ in scenario S_{100}-C_3-BL_4.

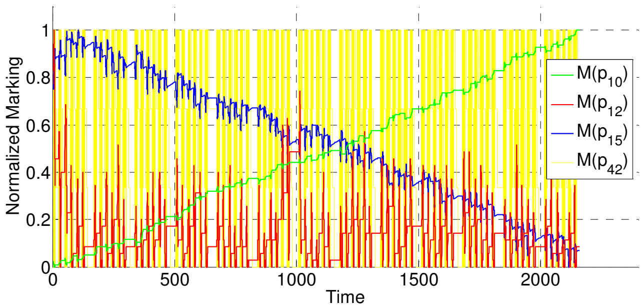


Figure 3-34. $\boldsymbol{M}(p_{10})$, $\boldsymbol{M}(p_{12})$, $\boldsymbol{M}(p_{15})$, $\boldsymbol{M}(p_{42})$ in scenario S_{100}-C_3-PW_{40}.

C. Using DEA for selecting the optimal IFTT planning

The cross-efficiency DEA technique is applied to choose among the possible alternatives. In particular, this multi-criteria decision making approach is here used to select one among the four scenarios discussed in the previous section, whose characteristic parameters are reported in Table 3-15 (columns 4 to 7). First, the TPN model is used to simulate the behavior of the IFTT and compute the performances indices reported in Table 3-17. Note that, with respect to Table 3-14 and Table 3-16, two additional indices are now considered, which clearly affect the evaluation of the best resource planning:

- Cost: the additional cost arising from the investment in the new resources with respect to the AS-IS scenario. These estimates are obtained considering, for the period of 90 days, the average rental cost for cranes and the salary of an expert operator in case of increase in the number of cranes, as well as the general conditions of contract for carriage of goods by rail provided by the Italian operator [Trenitalia] in case of increase in the number of trains per week or wagons per trains;
- CO_2: the increase in the release of greenhouse gases into the atmosphere due to the additional resources with respect to the AS-IS scenario. These values are estimated according to the procedure in [Ecotransit].

By analyzing the obtained performance indices values in the four scenarios (Table 3-17), it appears evident that the different scenarios cannot be easily compared. Furthermore, it is to observe that, while some of such indices should be minimized (O_{YB}, O_{YP}, O_{YE}, O_{YFC}, O_{YPO}, Cost, CO_2), others should be maximized (F_{YS}, min F_{YS}, U_C, TH, TM). Figure 3-35 reports the value of each performance index in the different scenarios normalized with respect to its maximum value. It is to remark that each of the considered scenarios is optimal by at least one performance index, while it is the worst one by other indices and is midway between other scenarios when further indices are considered. As a consequence, identifying the most efficient scenario is quite complex. Therefore, the need for the application of an automatic method such as the cross-efficiency DEA arises, especially when a variety of alternatives are available (not reported here for the sake of brevity).

Table 3-17. Performance indices values for the TPN under alternative resource planning.

Scenario	Nr	*Performance indices to be minimized*							*Performance indices to be maximized*				
		O_{YB} [ITUs]	**O_{YP}** [ITUs]	**O_{YE}** [ITUs]	**O_{YFC}** [ITUs]	**O_{YPO}** [ITUs]	**Cost** [€]	**CO_2** [t-CO_2eq]	**F_{YS}** [ITUs]	**min F_{YS}** [ITUs]	**U_C** [ITUs]	**TH** [ITUs/h]	**TM** [h]
		$x_{h=1,i}$	$x_{h=2,i}$	$x_{h=3,i}$	$x_{h=4,i}$	$x_{h=5,i}$	$x_{h=6,i}$	$x_{h=7,i}$	$y_{k=1,i}$	$y_{k=2,i}$	$y_{k=3,i}$	$y_{k=4,i}$	$y_{k=5,i}$
S_{100}-C_5	i=1	118.71	4.56	3.65	1.13	0.46	150,659.06	8.64	121.48	10.00	3.01	1.37	2,510
S_{100}-C_3-BL_4	i=2	100.00	6.15	4.17	0.64	0.40	108,269.69	31.80	138.62	18.00	1.29	1.42	3,432
S_{100}-C_3-BW_{25}	i=3	61.75	9.77	4.30	0.71	0.37	116,396.69	25.92	173.10	113.00	1.29	1.46	3,270
S_{100}-C_3-PW_{40}	i=4	110.66	5.11	3.82	0.67	0.14	244,749.29	59.76	129.40	5.00	1.31	1.39	2,222

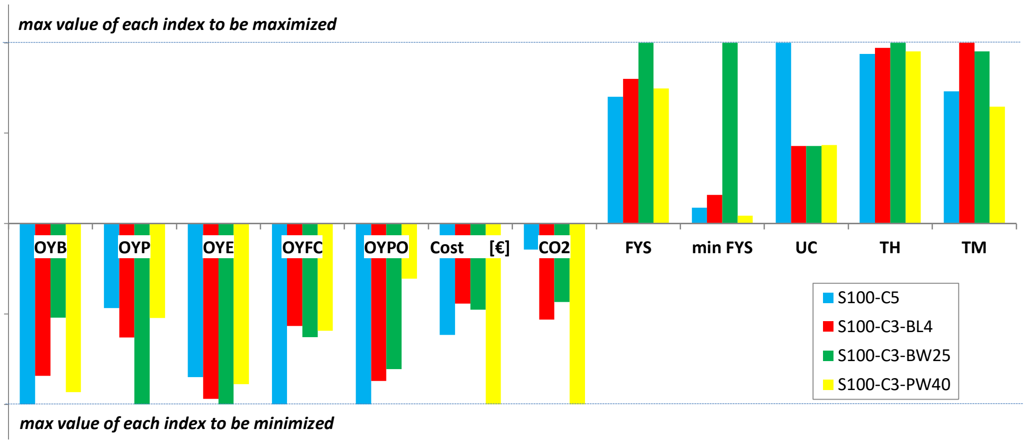


Figure 3-35. Graphical comparison of the obtained performance index values in the evaluated scenarios.

Using the notation in Section 3.4.1, the number of alternatives is F=4 and, for each of them, H=7 performance indices should be minimized (i.e., $x_{h,i}$ with h=1,…,7 and i=1,…,4, columns 3 to 9 in Table 3-17) and K=5 performance indices should be maximized (i.e., $y_{k,i}$ with k=1,…,7 and i=1,…,4, columns 10 to 14 in Table 3-17). In the evaluated case study, the decision maker imposes that the cost is to be considered as the most important among all the criteria. This simply requires the addition of some constraint in the cross-efficiency DEA formulation. In particular, for each scenario i, it is imposed the weighting coefficient $v_{6,i}$ of the cost index to be higher than or equal to twice the value of all other weights $v_{h,i}$ with h=1,…,7 and h≠6, and $u_{k,i}$, with k=1,…,5. To this aim, the cross-efficiency DEA optimization problems (Eq. 3-11)-(Eq. 3-14) and (Eq. 3-17)-(Eq. 3-21) detailed in Section 3.4.1 are solved subject to:

$$\begin{cases} v_{6,i} / v_{h,i} \geq 2 & \text{for } h = 1,2,3,4,5,7. \\ v_{6,i} / u_{k,i} \geq 2 & \text{for } k = 1,2,3,4,5. \end{cases} \qquad \text{(Eq. 3-22)}$$

Accordingly, are obtained the results in Table 3-18. The most efficient resource planning results to be the one performed in Scenario S_{100}-C_3-BW_{25}, which consists in increasing the number of cranes to 3 (rather than 2) and, at the same time, the number of wagons of each train from/to Bologna to 25 (rather than 20). Looking at Table 3-17, it is possible to conclude that the resource planning guarantees the correct functioning of the IFTT for at least 90 days more (last column). It also provides, with respect to the other scenarios, the highest value of F_{YS}, i.e., the highest average free available space in the yard storage area for ITUs arrangement, as well as the highest value of the min F_{YS} index. Hence, under such a scenario the yard storage area is never saturated and the number of ITUs managed in such an area varies in the range [113÷250] ITUs. Furthermore, it is possible notice that Scenario S_{100}-C_3-BW_{25} presents the second lowest values for the Cost and CO_2 parameters (Table 3-17, seventh and eighth column). These results highlight the effectiveness of the cross-efficiency DEA technique in identifying the most efficient resource planning alternative when the choice is neither trivial nor obvious.

Summing up, the proposed approach, by combining TPNs and the cross-efficiency DEA, allows decision makers to model the terminal's behaviour as well as to perform what-if analysis in order to rapidly obtain the evaluation and comparison of resolution actions to critical situations. Therefore, the presented technique represents a useful tool to properly address the typical problems related to resource planning and management of intermodal terminals.

Future developments will consider high-level Petri Nets to further refine the model and to allow solving more specific optimization problems. For instance, on the one hand the colored Petri Nets allow to associate features to tokens and to distinguish between different types of containers, as well as to optimize the processing time related to specific workflows, on the other hand the fuzzy Petri Nets are useful to describe imprecise information to implement knowledge based control strategies.

It will be also investigate how uncertainty on some parameters can be properly taken into account in the evaluation phase, e.g., by applying the stochastic or fuzzy cross-efficiency Data Envelopment Analysis techniques

Table 3-18. Cross-efficiency values for the different resource plannings in Table 3-17.

Scenario	Nr	CE_i (10^2)	rank
S_{100}-C_5	i=1	0.49	3
S_{100}-C_3-BL_4	i=2	0.52	2
S_{100}-C_3-BW_{25}	i=3	0.83	1
S_{100}-C_3-PW_{40}	i=4	0.26	4

.

Chapter 4

4 Modeling and control of railway traffic

The focus in this chapter is on the railway traffic rescheduling problem, whose proper resolution is fundamental for the improvement of the railway service performances in the context of a smart city. Indeed, in many countries increasing the market share of public transport, and especially railway transport, is considered as one of the solutions for mobility problems. Moreover, railway transport is a sustainable transportation mode. Thus, increasing the market share of railway transport is one of the top priorities of many governments. Basically, train rescheduling consists in retiming the offline scheduled traffic (i.e., the nominal timetable) to minimize undesired effects on the railway service when unpredictable events occur in the network (e.g., train delays, customer discomfort, energy consumption) [186], [187], [188], [189], [190]. Typically, unpredictable events are distinguished into ***disturbances*** (i.e., relatively small perturbations such as signal malfunctions or no-show of staff) and ***disruptions*** (i.e., large and particularly damaging external accidents such as trains or infrastructure breakdowns) [114], [190], [191]. Both cause the nominal timetable to become invalid because at least one train deviates from its original schedule. Generally, Train Dispatchers (TDs) manage disturbances mostly manually based on their experience and knowledge [114]. However, this highly time-consuming approach normally leads to suboptimal timetables. Such a manual procedure can be refined and speeded up by using automated real-time rescheduling procedures that may support TDs in determining in real-time suitable control actions and updating timetables while optimizing some traffic performance indices.

This chapter resumes the results of the research conducted during these three years of PhD course and published in [192] and in [193]. The contributions propose advanced techniques to support Train Dispatchers (TD) in the rescheduling process both in case of disturbances and disruptions. In particular, the purpose is to achieve a smart railway transportation system by combining the traditional management techniques with innovative mathematical models and control techniques. The first part of the chapter presents two automated procedures for real-time rescheduling in case of disturbance. The second part presents a novel methodology for real-time rescheduling in case of disruptions.

4.1 A Decision Support System for Real-Time Rescheduling of Railways

In this section a ***Decision Support System*** (DSS) for real-time management of railway networks is presented. The DSS employs a mathematical programming approach addressing traffic rescheduling under unexpected ***disturbances*** in a mixed- (single- and double-) tracked network. The DSS simulates the network behavior with the mathematical programming model based on the railway topology and constraints, rescheduling the timetable in real-time, detecting and solving conflicts in the network. The technique is applied to a real data set related to a large portion of a regional network in Southern Italy. This contribution is motivated by the emerging need for railway management systems to determine conflict-free timetables on-line and limit travelers' discomfort under disturbances. Decision Support Systems (DSS) [159] are needed to quantify the management effects and allow frequent, fast, and effective changes to the timetable under disturbances, while monitoring the timetable ability to absorb delays.

Railway (re)scheduling models may be of two types: cyclic (i.e., periodic within the day), or non-cyclic (i.e., aperiodic during the day). If trains in the network share similar characteristics and similar time planning, a cyclic timetable is preferred. However, this approach requires large computational efforts, so it is typically adopted for off-line scheduling. In rescheduling, instead, it is more appropriate to adopt aperiodic models to describe a limited dispatching area in which trains are to be rescheduled in a very short time. A key point of aperiodic models is the correct choice of the time horizon in order to provide the TD with feasible solutions in a short computation time [130]. If the time horizon is too small, only few trains are rescheduled and few conflicts can be detected and solved, whereas a too long time horizon leads to a larger number of trains running in the system, hence to more conflicts, higher complexity and larger computation times. Another strategic choice is between fixed- and variable-speed models [119]. The former assume that trains operate at their maximum speed, wherever possible, while the latter update train speed profiles to include the consequences of conflicts due to constraints imposed by the signaling system. In the related literature the majority of models are based on the simpler fixed speed method [119]. Some DSS have been developed to help TD quickly and effectively reschedule trains (see [194] for a discussion), but the accepted policy still consists in scheduling trains following their nominal order or according to some dispatching rules, such as a first come first served rules. Some works present an optimization framework to reschedule trains with different priorities, which can be computed statically or dynamically to include the needs of different stakeholders [195]. It is also possible to use a greedy algorithm, which performs a depth first search using an evaluation function to prioritize conflicts and perform the branch according to a set of criteria to obtain good rescheduling [140]. However, most of the models in the literature are not able to absorb a long delay on the network in a feasible time, so that railway timetables are typically poorly robust to long delays or tracks blocking [196], although more recent publications prove the feasibility of real-time rescheduling by means of "alternative graphs" in heavily disturbed large networks (see [195] and [194]).

In this this section it is presented a novel DSS for train rescheduling based on a fixed speed profile method. Here, the rerouting problem is disregarded since it requires a complete knowledge of the railway infrastructure that is seldom available. The DSS is based on a mathematical programming approach for traffic rescheduling of an *n*-tracked network [130], where *n* is an arbitrary number of parallel tracks. The method may be used for modeling aperiodic timetables, where rescheduling is made in a finite time horizon after which the model loses validity. The model allows the rescheduling of the railway traffic treating one disturbance at a time, hence it is not needed to establish any hierarchy between the occurring disturbances. The approach in [130], is here reformulated and adapted to the case of rescheduling in mixed- (both single- and double-) tracked network. Moreover, the approach is enhanced by proposing a DSS for railway real-time rescheduling that is able to improve the robustness of the planned rescheduled timetable by adopting an analytical method based on an allocating buffer time technique [197]. The DSS uses a three level rescheduling method: in the first level trains rescheduling is executed in the chosen time horizon; in the second one the rescheduled timetable is used to identify and solve the current trains conflict and all those that may subsequently occur. At this second level a meta-heuristic algorithm is applied, whose complexity depends on the number of trains in the network and on the railway infrastructure. The basic idea is to consider each train as a two dimensional array, in which one dimension is used to define the progressive station distances and the second one relates to the progressive arrival/departure times planned in each station according to the nominal timetable. The third level allows verifying the timetable robustness by means of a Monte Carlo approach and choosing the rescheduled timetable among the proposed alternatives. As a result, the hierarchical rescheduling allows obtaining in a short computation time a conflict-free and robust schedule.

The DSS is applied to a railway network in Southern Italy, constituted by single tracks with few double track segments, and such that in some stations only a train can stop or pass through. It is considered a real disturbance occurring in a weekday in the morning at rush hour and compare the DSS operation with the timetable determined using the model in [130], demonstrating the DSS effectiveness and efficiency.

4.1.1 The DSS Architecture

A railway is composed of stations and block sections separated by safety signals, to control traffic while imposing the planned train circulation [113]. Usually, buffer times are pre-allocated to absorb small disturbances, e.g., sudden decelerations or too long stops in stations. However, during the actual operation, some disturbances may occur and cause traffic disruption and changes of schedules from the nominal case. The proposed DSS is devoted to reschedule trains while respecting the existing restrictions without penalizing the overall traffic performance. A fixed speed aperiodic profile is assumed for each train [194]. The feasibility of the different speed profiles is later checked for each train.

The structure of a railway rescheduling DSS includes three sub-systems [194]: 1) Conflict Detection (CD): given the infrastructure status, timetable, rolling stock information, position and speed of each train, potentially conflicting train routes in a given period of traffic prediction are determined [195]; 2) Conflict Resolution (CR): given the actual train delays and predicted conflicts, a suitable and robust timetable is proposed [195], [194]; 3) Train Speed Coordination (TSC): starting form trains sequences and routes, the updated speed profiles are computed, respecting traffic regulations and minimizing delays and energy consumption, and transmitted to drivers [194]. With the fixed speed model, the DSS joins in one the CD and CR sub-systems, whose output is used by the TSC sub-system to automatically obtain the new speed profiles, showing them in a time chart [194]. In this way, the TD can communicate to drivers the traffic state, stopping trains to improve the safety level.

The proposed DSS structure is shown in Figure 4-1, and its composing sub-systems are detailed as follows:

- *TD-DSS Interface*: it is a user friendly Graphical User Interface (GUI) that allows to control in real-time the monitored dispatching area with a space-time graphical representation of trains. The GUI provides the TD with tools to launch the underlying DSS mathematical model, identifies the robustness timetable, and solves all possible network conflicts;
- *Rescheduling Mathematical Model*: this is the DSS underlying framework that allows modeling the network topology with capacity and safety constraints. The model is the MILP problem presented in [130] and revisited as detailed in the subsequent section. It is executed to determine the Rescheduled Timetable in the desired time horizon when a disturbance occurs;
- *Validation Model*: it is a reproduction of the DSS mathematical model aiming at improving the rescheduled timetable robustness. The sub-system simulates several instances of the nominal timetable perturbing it with a small buffer time in each station in the network [197].

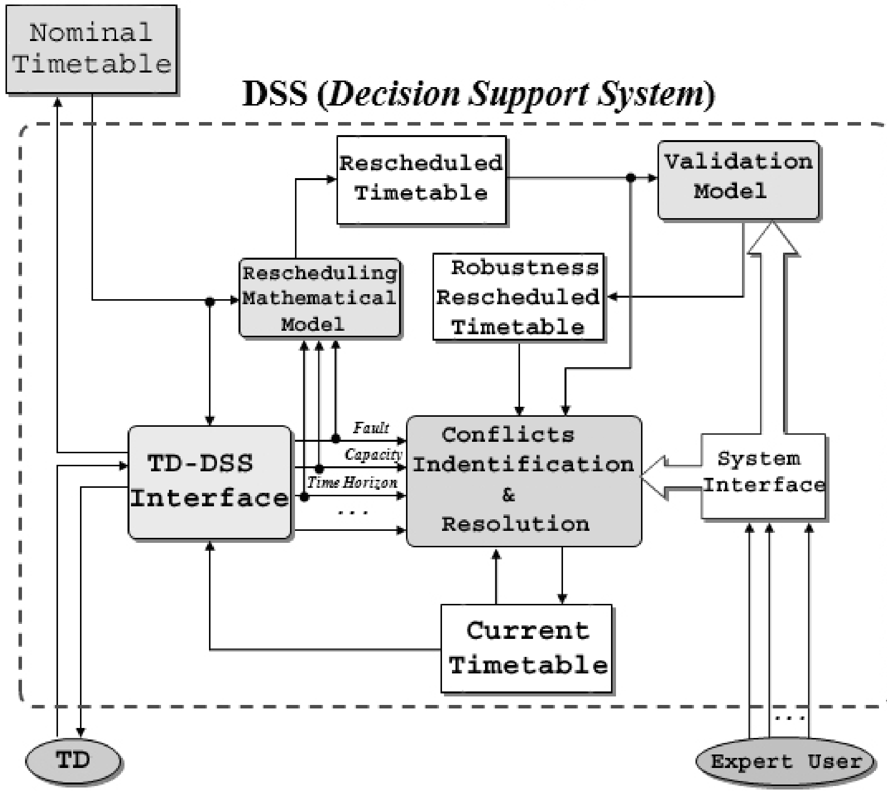


Figure 4-1 The architecture of the proposed DSS.

Therefore, at each instance a new timetable is determined. In this way, the validation model improves the timetable robustness without overly penalizing the quality of the final solution that depends on the maximum buffer time that may be associated to the nominal stop. The obtained timetable is the Robustness Rescheduled Timetable;

- *Conflicts Detection & Resolution.* To identify and solve all the possible conflicts due to the disturbance it is used an ad hoc heuristic procedure, which mimics a job shop scheduling problem aiming at minimizing the average knock-on delays [122], whose complexity depends on the number of trains and tracks in the network. The TD uses the obtained Current Timetable to reschedule trains.
- *System Interface*: it is a low-level interface independent from the *TD-DSS Interface* which allows upgrading the single sub-systems. It is for use only by expert users, e.g., the DSS designer or specialized programmers.

The DSS architecture is characterized by the independence of the validation model from the mathematical model, allowing some flexibility. In fact, the TD can choose not to launch the *Validation Model*, using the *Rescheduled Timetable* to identify the next conflict located at the end of the time horizon. This decreases the computational time and guarantees obtaining a conflict-free timetable in a short time. Therefore, the *Validation Model* may be used both off-line, after saving the output robustness timetable, or in real time, to solve conflicts.

4.1.2 The Revisited Rescheduling Model

The DSS employs a MILP rescheduling model revisiting an approach for *n*-tracked networks [130], whose notation is reported in Table 4-1. The model uses two main pieces of information on the network: its capacity and the number of tracks in each segment. The framework in [130] is revisited as regards the objective function, constraints, and time window solution. Rather than considering a fixed time window that must be very large to contain all possible conflicts arising after the disturbances (resulting in a large computation time), it is considered a short finite time. The model application is followed by a heuristic algorithm that allows calculating near-optimal solutions, so that the TD can choose and communicate the new status to drivers as soon as possible.

Table 4-1. The Model Notation

Name	Meaning
T	Set of trains to be rescheduled
i	Generic train in the network
g_i^{train}	Generic train length g_i^{train}
B	Set of segments with a start and an end point
j	Generic segment in the network
N_s	Number of segments in the railway network
E	Set of events of all trains
k	Generic event in the network
d_k	Generic event duration
$b_k^{initial}$, $e_k^{initial}$	Start and end times of event k in nominal timetable
b_k^{static}, e_k^{static}	Start and end times of event k that has already started when the disturbance occurs
o_k	Event point of origin, to investigate whether trains of k and $(k+1)$ events on j are in the same direction
h_k	Binary variable indicating whether there is a scheduled stop at a station during event k
$E^{connection}$	Set of all connected events or couples $(k,\bar{k})$
$g_{k,\bar{k}}^{connection}$	Connection time of connected events $(k,\bar{k})$
$K_i \subseteq E$	Sorted set of events of each train $i \in T$
n_i	Last event in K_i
$L_j \subseteq E$	Sorted set of events of each segment j
m_j	Last event in L_j
s_j	Binary variable indicating type of a segment j
δ_j	Buffer time added to nominal stop at j
Δ_M	Maximum buffer time added to nominal stop at j
P_j	Set of tracks in each segment j
t	Generic track in P_j
g_{jt}^{track}	Generic track size
Δ_j^M	Time units between two trains in opposite directions
Δ_j^F	Time units between two trains in the same direction
$c_i^{penality}$	Fixed penalty associated to train i with a total delay greater than a threshold w_i
c_i^{low}	Cost associated to each unit delay

Each train in the network can occupy only one track, so its occupation is exclusive. This restriction is implemented with a time shifting technique that reschedules the timetable of the penalized train with a buffer time, so that a train can be slowed or stopped to improve the traffic safety.

The model is based on the concept of event, which is a train request to use a track, with a start time, end time, and duration. If the rescheduled duration is larger than its nominal value, the difference is the delay associated to the event. If not, a recovery time in the rescheduled timetable is allowed to improve the nominal duration.

A. Problem decision variables and revisited model

The MILP formulation contains eight decision variables, divided into integer and binary and listed in the sequel. Integer variables , x_k^{end} , z_k are the actual starting and ending times and the delay of event . Binary variable q_{kt} indicates whether event k uses track t in segment j (q_{kt}=1) , with $k \in L_j, t \in P_j, j \in B$. Binary variable $\gamma_{k\hat{k}}$ indicates whether event k occurs before $\hat{k}$ ($\gamma_{k\hat{k}}$=1), with $k, \hat{k} \in L_j, j \in B, k < \hat{k}$. Binary variable $\lambda_{k\hat{k}}$ indicates whether event k is rescheduled with earlier start time than $\hat{k}$ ($\lambda_{k\hat{k}}$=1), with $k, \hat{k} \in L_j, j \in B, k < \hat{k}$. Binary variable ε_i indicates whether train $i \in T$ has a delay lower than its threshold w_i (ε_i =1). Binary variable ε_k indicates whether $k \in E$ ends with a delay larger than threshold w_k (ε_k =1).

The revisited mathematical model adopts the following novel objective functions with respect to [130]:

$$Min \sum_{k \in E} (c_k^{low} \cdot z_k) \quad \text{(Eq. 4-1)}$$

$$Min \sum_{k \in E} z_k \quad \text{(Eq. 4-2)}$$

The cost of the total delay for all events in the horizon is minimized in (Eq. 4-1). Alternatively, (Eq. 4-2) considers a delay.

The objective functions are subject to train constraints:

$$x_k^{end} = x_{k+1}^{begin}, \quad \forall k \in K_i - \{n_i\}, \ i \in T \quad \text{(Eq. 4-3)}$$

$$x_k^{end} \geq x_k^{begin} + d_k, \quad \forall k \in E \quad \text{(Eq. 4-4)}$$

$$x_k^{end} - x_k^{begin} \geq d_k - \varepsilon_k \delta_k \quad with: \qquad \text{(Eq. 4-5)}$$

$$\begin{cases} \delta_k > 0 \; if \, (d_k \geq d_{k_{trip}}, h_k = 0) \vee (d_k \geq d_{k_{stop}}, h_k = 1) \\ \delta_k = 0 \; if \, (d_k < d_{k_{trip}}, h_k = 0) \vee (d_k < d_{k_{stop}}, h_k = 1) \end{cases}$$

$$d_k > \varepsilon_k \delta_k, \quad \forall k \in E \qquad \text{(Eq. 4-6)}$$

$$x_k^{begin} \geq b_k^{initial}, \quad \forall k \in E; \; h_k = 1 \qquad \text{(Eq. 4-7)}$$

$$x_k^{begin} = b_k^{static}, \quad \forall k \in E; \; b_k^{static} > 0 \qquad \text{(Eq. 4-8)}$$

$$x_k^{end} = e_k^{static}, \quad \forall k \in E; \; e_k^{static} > 0 \qquad \text{(Eq. 4-9)}$$

$$x_k^{end} - e_k^{initial} \leq z_k, \quad \forall k \in E \qquad \text{(Eq. 4-10)}$$

Constraint (Eq. 4-3) specifies that each train event is succeeded by the next in the train ordered set of events. Constraint (Eq. 4-4) means that each event must use the assigned track at least for the time specified by the minimum parameter . Constraint (Eq. 4-5) is an alternative that is proposed to (Eq. 4-4), meaning that a train can decrease its planned running/stop time by δ_k time units if it is greater than the minimum reschedulable running ($d_{k_{trip}}$)/stop ($d_{k_{stop}}$) time. Therefore, δ_k is a recovery time in the timetable. Particularly, (Eq. 4-5) is applicable if constraint (Eq. 4-6) is verified, imposing that duration d_k must be greater than δ_k. This is necessary because if the duration of a rescheduled event is too short, then the updated timetable cannot ensure the initial stop and running times (e.g., $d_k = 0$ and $h_k = 1$ the planned stop is deleted and travelers do not have time to get off the train). Constraint (Eq. 4-7) enforces the restrictions related to planned stops and the consequential earliest possible start time. Constraints (Eq. 4-8) and (Eq. 4-9) ensure that events that have started but not yet ended before the disturbance will end as planned. Constraint (7) records the magnitude of the delay of events.

Technical constraints are also considered as follows:

$$\sum_{t \in P_j} q_{kt} = 1, \quad \forall k \in L_j, j \in B \qquad \text{(Eq. 4-11)}$$

$$q_{kt} + q_{\hat{k}t} - 1 \leq \lambda_{k\hat{k}} + \gamma_{k\hat{k}}, \quad \forall k, \hat{k} \in L_j, t \in P_j, j \in B; k < \hat{k} \qquad \text{(Eq. 4-12)}$$

$$x_{\hat{k}}^{begin} - x_k^{end} \geq \Delta_j^M \gamma_{k\hat{k}} - M(1 - \gamma_{k\hat{k}}), \qquad \text{(Eq. 4-13)}$$
$$\forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} \neq o_k$$

$$x_{\hat{k}}^{begin} - x_k^{end} \geq \Delta_j^F \gamma_{k\hat{k}} - M(1 - \gamma_{k\hat{k}}), \qquad \text{(Eq. 4-14)}$$
$$\forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} = o_k$$

$$x_k^{begin} - x_{\hat{k}}^{end} \geq \Delta_j^M \lambda_{k\hat{k}} - M(1 - \lambda_{k\hat{k}}), \qquad \text{(Eq. 4-15)}$$
$$\forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} \neq o_k$$

$$x_k^{begin} - x_{\hat{k}}^{end} \geq \Delta_j^F \lambda_{k\hat{k}} - M(1 - \lambda_{k\hat{k}}), \qquad \text{(Eq. 4-16)}$$
$$\forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} = o_k$$

$$\lambda_{k\hat{k}} + \gamma_{k\hat{k}} \leq 1 \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k} \qquad \text{(Eq. 4-17)}$$

$$g_i^{train} q_{kt} \leq g_{jt}^{track} \quad \forall k \in (K_i \cap L_j), t \in P_j, j \in B, i \in T \qquad \text{(Eq. 4-18)}$$

Constraint (Eq. 4-11) enforces that each event uses one track per segment, otherwise (Eq. 4-12) imposes that if two events are using the same track within a segment, then one of them must end and a required separation time must elapse until the next event starts on the same segment, so that either ((Eq. 4-13)-(Eq. 4-14)) or ((Eq. 4-15)-(Eq. 4-16)) hold. If trains meet, the minimum separation time is Δ_j^M , being *M* a large positive constant depending on the largest time horizon, otherwise if trains follow each other, the time is . Constraint (Eq. 4-17) enforces that $\gamma_{k\hat{k}}$ and $\lambda_{k\hat{k}}$ cannot be both 1, (Eq. 4-18) ensures that the track used at *k* is sufficiently long to accommodate the train.

The objective functions are subject to operator constraints.

$$z_{n_i} - w_i \le M\varepsilon_i \quad \forall i \in T \tag{Eq. 4-19}$$

$$z_k - w_k \le M\varepsilon_k \quad \forall k \in E \tag{Eq. 4-20}$$

$$x_{\hat{k}}^{begin} - x_k^{end} \ge g_{k\hat{k}}^{connection} \quad \forall k, \hat{k} \in E; (k, \hat{k}) \in E^{connection} \tag{Eq. 4-21}$$

$$x_k^{begin}, x_k^{end}, z_k \ge 0 \quad \forall k \in E \tag{Eq. 4-22}$$

$$\gamma_{k\hat{k}}, \lambda_{k\hat{k}} \in \{0,1\} \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k} \tag{Eq. 4-23}$$

$$q_{kt} \in \{0,1\} \quad \forall k \in L_j, t \in P_j, j \in B \tag{Eq. 4-24}$$

$$\varepsilon_i \in \{0,1\} \quad \forall i \in T \tag{Eq. 4-25}$$

$$\varepsilon_k \in \{0,1\} \quad \forall k \in E \tag{Eq. 4-26}$$

Constraint (Eq. 4-19) enforces a fixed penalty cost if a certain delay is exceeded and it becomes relevant only adopting an objective function that minimizes the sum of total delays z_{n_i} associated to the last event n_i of each train *i*. It is used the alternative (Eq. 4-20): if an event *k* has a delay z_k greater than threshold w_k, then $\varepsilon_k = 1$. Thus $\varepsilon_k \delta_k$ in (Eq. 4-5) is nonzero, so that is possible to recover δ_k time units during a trip or a stop. Constraint (Eq. 4-21) synchronizes connecting trains, (Eq. 4-22) imposes x_k^{begin}, x_k^{end}, z_k are positive, (Eq. 4-23)-(Eq. 4-26) impose $\gamma_{k\hat{k}}$, $\lambda_{k\hat{k}}$, q_{kt} , ε_i , ε_k are binary.

B. Validation model: Performance analysis

The rescheduling model is used to improve the timetable robustness to disturbances, using the following robustness index of the generic timetable *x* [197]:

$$R(x) = \sum_{i=1}^{|T|} \sum_{k=1}^{|K_i|} Buff_k \cdot Flow_k \cdot TT_k \cdot NSucT_k \cdot \frac{(|K_i| - k)}{|K_i|}, \; h_k = 1 \tag{Eq. 4-27}$$

In Eq. 4.27, $Buff_k$ is the buffer time associated to the stop events *k* of train *i*, $Flow_k$ is the average percentage of travelers that get on train *i* at the station of event *k*. Further, $NSucT_k$ is the number of trains that could be perturbed in the time horizon, TT_k is the percentage of tightness of track between stations *j* and *j+1*. Thus, given a timetable with *m* tracks, the tightest track has TT_k=1, the second tightest track has TT_k=(m-1)/m, and so on [197]. The value of *R(x)* does not return a direct measure of robustness for a timetable, but rather a value proportional to it. Therefore, it is useful to compare two timetables in order to evaluate which is more robust, having a higher *R(x)* [197].

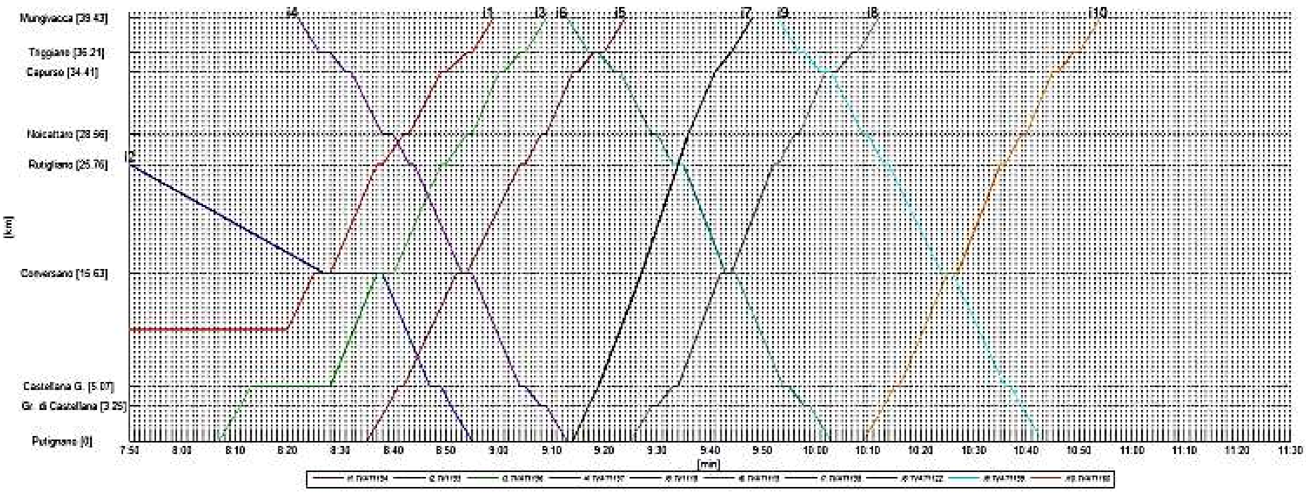
To improve the timetable robustness a Monte Carlo approach is adopted. In each simulation the procedure relaxes the nominal timetable with a pseudo-random buffer time that is added to each nominal stop time. This relaxation is considered in two different ways: 1) by a constant buffering, where $Buff_k$ is independent by the event duration (i.e., it is a random time); 2) by a proportional buffering, where $Buff_k$ is the 50% of the nominal event duration. When a simulation ends, the DSS launches the revisited mathematical model, determines the scheduled timetable *x*, calculates and stores in a database its robustness *R(x)*. Once all simulations are completed, the timetable with the maximum *R(x)* is the most robust one. In case of multiple timetables with the highest *R(x)* value, the system selects the first one.

C. Conflicts identification and resolution

The DSS includes a sub-system that identifies and solves conflicts after the chosen time horizon. It is based on a heuristics, and ensures each train arrives at its destination without crossings or coincidences, even if its final stop is after the end of the time horizon. Figure 2 shows the basic algorithm. Starting from the rescheduled timetable, the first crosses in the line are detected. First, single and double coincidences in stations are removed: all values in the time vector associated with each train are checked and, if a coinciding value between two trains is found, a time change is assigned to the train with the lowest index (that is, the train that has been for the longest time in the line). Hence, the first intersection in the line or station with a unit capacity is found, and the traveling time of two conflicting trains is calculated as the sum of all durations of runs and stops occurred up to the station before the crossing. If the time horizon is not extended, then priority is given to the train with the highest traveling time, shifting the time of the other train starting from the stop in the station located before the crossing. In case of new coincidences at the station, they are solved in the next step of the procedure. The model always gives priority to direct trains, to maximize the service quality. The computational time required by this procedure is linearly increasing with the number of trains in the time horizon and the number of stations with unit capacity.

4.1.3 The Case Study

The DSS is applied to a regional railway network in Apulia, Southern Italy, described in Table 4-2. The network has a low capacity so it is hard to manage traffic efficiently. Three different time horizons are considered (30, 40, and 50 minutes) for a real disturbance with a duration of 30 minutes occurring on a weekday at 7:50 a.m., a rush hour with many trains, affecting five trains in the maximum time horizon (50 min). Table 4-3 compares the rescheduling results obtained by the DSS with those by [130].



(a)

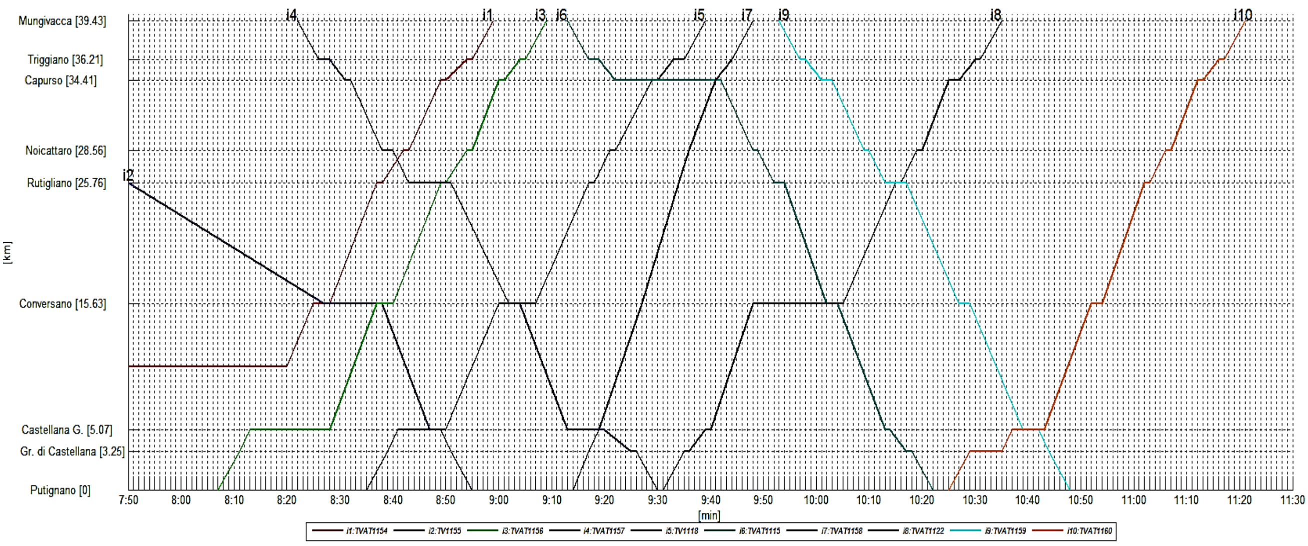


Figure 4-3. Planned (a) and rescheduled timetable (b) with 50 min time horizon.

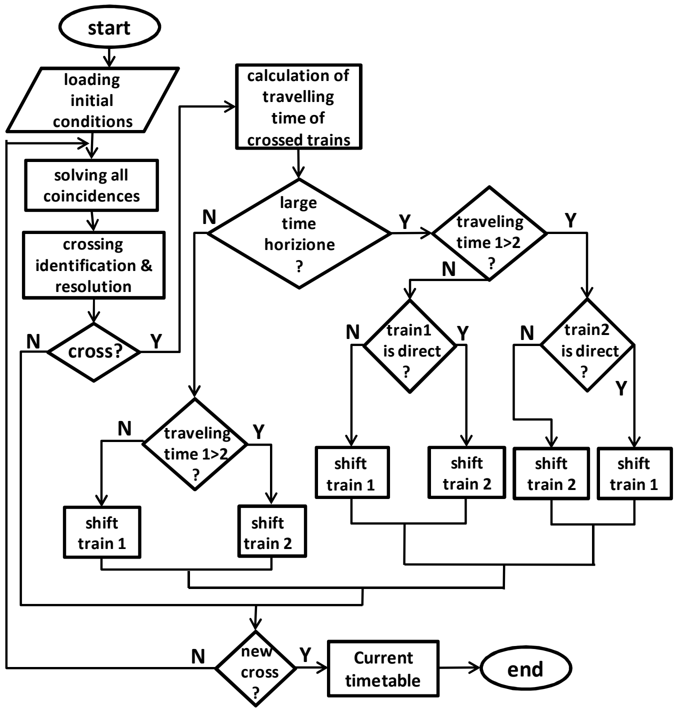


Figure 4-2. Heuristic procedure flow-chart.

Table 4-2. Infrastructure Data of the Considered Network.

Segment	Description	Type	s_j	P_j
j_1	Mungivacca-Triggiano	Single track	1	{t1}
j_2	Triggiano	Station	0	{t1,t2}
j_3	Triggiano-Capurso	Single track	1	{t1}
j_4	Capurso	Station	0	{t1,t2}
j_5	Capurso-Noicattaro	Single track	1	{t1}
j_6	Noicattaro	Station	0	{t1,t2}
j_7	Noicattaro-Rutigliano	Double track	1	{t1,t2}
j_8	Rutigliano	Station	0	{t1,t2}
j_9	Rutigliano-Conversano	Single track	1	{t1}
j_{10}	Conversano	Station	0	{t1,t2}
j_{11}	Conversano-Castellana	Single track	1	{t1}
j_{12}	Castellana	Station	0	{t1,t2}
j_{13}	Castellana-Grotte di Castellana	Single track	1	{t1}
j_{14}	Grotte di Castellana	Stop	0	{t1}
j_{15}	Grotte di Castellana-Putignano	Single track	1	{t1}

The table shows, for each time horizon, results for the total delay minimization (Eq. 4-2) and for the cost alternative (Eq. 4-1) with two cases, with different sets of c_k^{low} coefficients: the first one calculates the average travelers cost and the second one the average corporate cost (i.e., the penalty cost associated to the total delay made by trains). In all cases the DSS improves the timetabling obtained by [130]. Table 4-4 shows the large problem dimensions and short computational time to compute the rescheduling. Moreover, Table 4-5 shows the value of the parameters adopted to evaluate the scheduling robustness. Results demonstrate that with a constant buffering, the timetable is more robust than with a proportional one, although in the latter case the total delay is lower.

Table 4-3. Original Model and DSS Rescheduling Comparison

	Objective function					
	Total delay [min]		Average traveler cost [€]		Average corporate cost [€]	
Time horizon [min]	**Original model**	**Revisited model**	**Original model**	**Revisited model**	**Original model**	**Revisited model**
30	472	450	428.54	416.16	34.41	33.14
40	643	615	506.06	488.39	42.62	41.18
50	691	659	524.93	507.55	44.22	42.65

Table 4-4. Problem Computational Time and Dimensions

			Computational time [s] for each objective function		
Time horizon [min]	**Model**	**Constr. matrix**	**Total delay**	**Average travelers cost**	**Average corporate cost**
30	Original	116x99	1.06	1.07	1.05
	Revisited	135x118	1.10	1.10	1.13
40	Original	192x155	1.34	1.33	1.34
	Revisited	220x183	1.43	1.43	1.44
50	Original	289x222	1.68	1.68	1.69
	Revisited	326x259	1.81	1.89	1.83

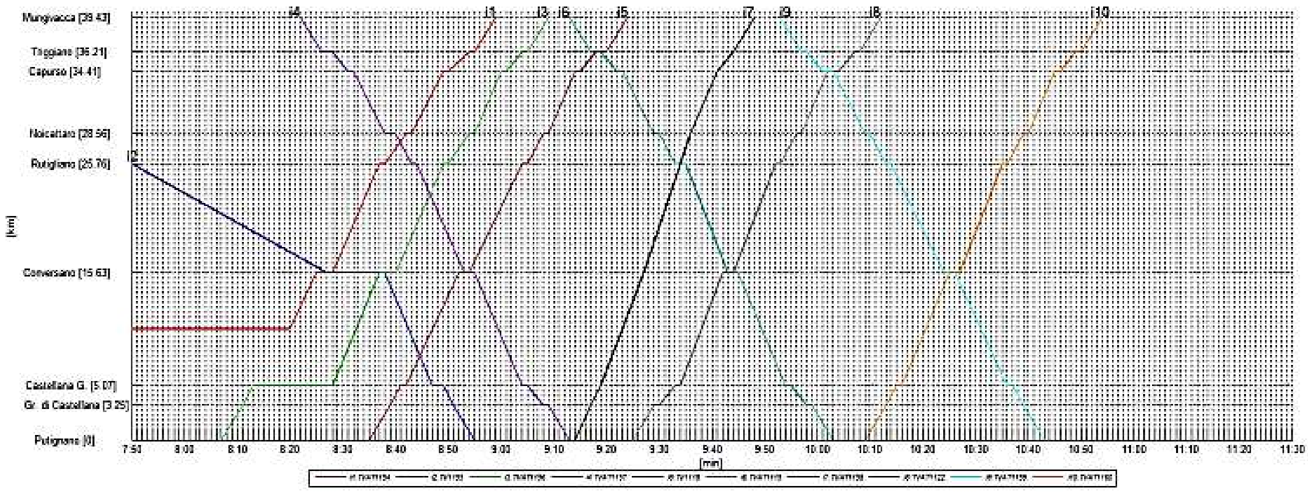
Table 4-5. Robustness Analysis (100 Simulations)

	30 min		40 min		50 min	
	Constant buffering	**Proportional buffering**	**Constant buffering**	**Proportional buffering**	**Constant buffering**	**Proportional buffering**
R(x) [/]	9.10	7.65	14.12	11.72	18.61	15.63
Total delay [min]	518	504	731	689	808	756
Comp. time [s]	135.28	136.72	178.19	180.41	229.46	240.10
Efficiency loss [%]	15.11%	12.00%	18.86%	12.03%	22.61%	14.71%

Table 4-6. Performance of Heuristic Procedure

Time horizon [min]	Nr. of conflicts	Computational time [s]
30	8	34.50
40	10	42.28
50	15	61.21

Hence, the buffering method choice depends on a compromise between operator preferences (i.e., robustness, with a constant buffering) and level of efficiency required by travelers (i.e., total delay, with a proportional buffering). Table 4-6 shows the detected conflicts and the computational time (by a 3.40 GHz processor and 8 GB RAM PC) needed to calculate the solution in each time horizon. Figure 4-3a and 4-3b show the planned and the rescheduled timetable in the 50 minutes horizon with a maximum number of 10 trains, where the first five i_i with i=1,…,5 are in the time horizon.



(a)

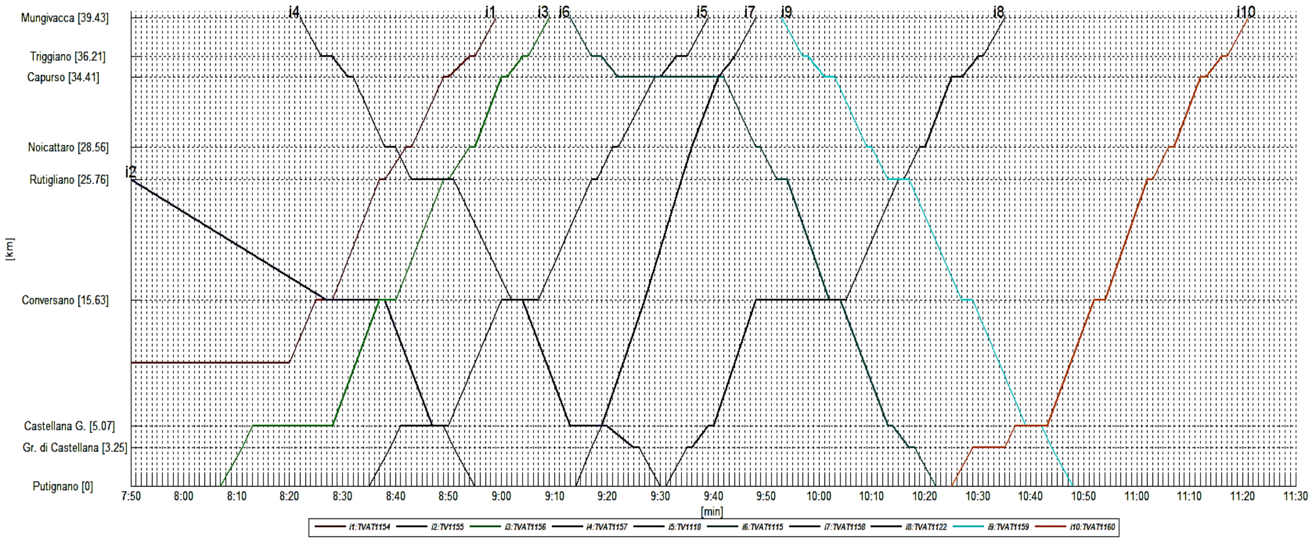


Figure 4-3. Planned (a) and rescheduled timetable (b) with 50 min time horizon.

4.2 A Decision Making Procedure for Robust Train Rescheduling based on Mixed Integer Linear Programming and Data Envelopment Analysis

This section presents a ***self-learning decision making procedure*** for ***robust*** real-time train rescheduling in case of ***disturbances***. The procedure is applicable to aperiodic timetables of mixed-tracked networks and it consists of three steps. The first two are executed in real-time and provide the rescheduled timetable, while the third one is executed offline and guarantees the self-learning part of the method. In particular, in the first step, a robust timetable is determined, which is valid for a finite time horizon. This robust timetable is obtained solving a Mixed Integer Linear Programming problem aimed at finding the optimal compromise between two objectives: the minimization of the delays of the trains and the maximization of the robustness of the timetable. In the second step, a merging procedure is first used to join the obtained timetable with the nominal one. Then, a heuristics is applied to identify and solve all conflicts eventually arising after the merging procedure. Finally, in the third step an offline cross-efficiency fuzzy Data Envelopment Analysis technique is applied to evaluate the efficiency of the rescheduled timetable in terms of delay minimization and robustness maximization when different relevance weights (defining the compromise between the two optimization objectives) are used in the first step. The procedure is thus able to determine appropriate relevance weights to employ when disturbances of the same type affect again the network. The railway service provider can take advantage of this procedure to automate, optimize, and expedite the rescheduling process. Moreover, thanks to the self-learning capability of the procedure, the quality of the rescheduling is improved at each reapplication of the method. The technique is applied to a real data set related to a regional railway network in Southern Italy to test its effectiveness.

As already introduced in the introductive part of this chapter, the efficient management of railway traffic is crucial for rail companies to provide their customers with a quality service [198]. In particular, both companies and customers are interested in an on-time service. The first can avoid sanctions applied if the accumulated delay overcomes the maximum imposed by contract, while the latter can benefit from a reliable service without loss of money and time. Moreover, an efficient service can improve customers' loyalty to the company and this produces, as a secondary effect, a reduction of road traffic in favor of a more sustainable railway transport.

Automated real-time rescheduling emerges as a promising technique to manage railway traffic in a smart and technologically advanced way when unexpected events affect the normal behaviour of the network. This technique refers, as the one of the previous Section 4.1, to disturbances and allows to refine the results of the manual rescheduling performed by Train Dispatchers. In particular, the proposed automated real-time procedures may support TDs in determining in real-time suitable control actions and updating timetables while optimizing some traffic performance indices [2], [7], [10]. Furthermore, in order to improve the quality of the automated rescheduling, it is important to evaluate the corresponding outputs, predict the consequences of different control actions, and keep trace of the most favorable one. In this work, this aim is achieved by including into a suitable decision making procedure both an automated real-time rescheduling process and an offline self-learning technique. Such a train rescheduling procedure can on the one hand reduce the workload of the TD, by automatically and autonomously performing a real-time rescheduling, and on the other hand improve the effectiveness of the rescheduling at each reapplication of the method.

More in detail, the presented decision making procedure is valid for railway systems using aperiodic timetables and presenting mixed (single- and double-) tracked networks. It is not considered any limitation on the dimension and the topology of the system. The procedure consists of three steps: in the first one, an optimal timetable over an appropriate Time Horizon (TH) is obtained by solving a

Mixed Integer Linear Programming (MILP) problem aimed at finding the best compromise between the minimization of delays of trains and the maximization of the robustness of the timetable. In the second step, a merging procedure is first used to join the optimal timetable over the chosen TH with the nominal one in the remaining time window of the timetable. Then, all possible conflicts arising after TH are iteratively solved by means of a heuristic procedure, which calculates a near-optimal rescheduling solution. Finally, the third step consists in an offline self-learning procedure aimed at predicting the results and effectiveness of alternative control actions and updating an external database with the most appropriate solution to use in the event that a disturbance of the same type occurs in the future. To this aim, a cross-efficiency fuzzy Data Envelopment Analysis method proposed for the performance evaluation of healthcare systems in [199] is applied to determine the efficiency of the rescheduled timetable (in terms of reduction of delays and maximization of robustness) and to rank alternatives according to their efficiency values.

The proposed decision making procedure is tested on a real case study that is a portion of a regional railway network where a set of trains is affected by a disturbance at a rush hour in a weekday. The railway network is located in Southern Italy and is constituted mostly by single tracks with few double track segments, and such that in some stations only a train can stop or pass through.

To confirm the effectiveness of the proposed procedure the outcomes of the technique are compared with those obtained with the traditional TD manual procedure. Moreover, the effects of the robustness maximization are accurately validated by statistically evaluating two indices: the number of conflicts of trains and the time delay caused by the occurrence of an additional disturbance both for the robust rescheduled timetable and for a rescheduled timetable obtained by only minimizing delays.

Summarizing, the main contributions of this section consist in:

(1) presenting an automatic decision making procedure for robust real-time train rescheduling in case of disturbances;

(2) stating and solving a MILP problem for mixed-tracked railway systems that simultaneously addresses the minimization of the delays caused by a disturbance and the maximization of the robustness of the timetable;

(3) integrating the resolution of the MILP problem with a heuristic procedure to speed-up the real-time rescheduling procedure;

(4) proposing an offline DEA-based self-learning procedure, which predicts and evaluates the consequences of alternative control actions to improve the effectiveness of the method at each reapplication.

A. Positioning of the approach with respect to the state of the art and managerial implications

In the related literature, only few studies have been developed in the context of decision making procedures for real-time train rescheduling (see for instance the discussions in [137] and [135]). These contributions mainly focus on automating the rescheduling procedure, while there is still a lack of self-learning approaches that can strongly improve the performance of the decision making and quickly predict the results of changes in control actions. As a matter of fact, nowadays railway companies are still seeking for automatic solutions to improve traffic management in order to enhance timeliness and reliability of railway services.

Exhaustive discussions on automated real-time rescheduling can be found in [190], [200], [118], and [201], showing that, in the relevant literature, three main classes of computer-based rescheduling approaches can be broadly identified: simulation models, heuristic procedures, and mathematical optimization models, or a combination of them. The most common framework to reschedule railway traffic is MILP [198], [201], [202].

Traditionally, in the related literature, rescheduling optimization models consider a single-objective function. However, the nature of train rescheduling problem is intrinsically multi-objective due to multiple conflicting interests of the involved stakeholders and to social and environmental issues. Hence, multi-objective approaches generally produce better rescheduling alternatives. To actually provide a significant support to the TD, rescheduling approaches in railway networks should at least possess two features, namely, timeliness and robustness [118]. The former consists in the minimization of delays caused by the occurrence of a disturbance, while the latter is the ability of the timetable to absorb such disturbances, to tolerate a certain degree of uncertainty, and/or to cope with unexpected troubles without significant modifications [203], [204], [205].

When disturbances arise, they cause primary delays to the directly affected train that typically propagate to other trains as secondary delays. Consequently, one of the major requirements of an effective rescheduling is to limit the spread over the network of secondary delays, which in turn requires that: (1) primary delays can be absorbed, (2) few primary delays result in small secondary delays, and (3) secondary delays can quickly disappear thanks to simple dispatching operations, without spreading over the network [205], [206].

Usually, robustness is achieved by adding some time reserves in the timetable to allow for flexibility when rescheduling traffic to prevent delays' further spreading. Such time reserves can be classified into recovery times and buffer times. The former are time reserves computed offline by increasing the travel time with respect to the minimum one (i.e., traveling at the maximum speed), while the latter are time reserves over the minimum separation time between consecutive train paths [190], [207]. In other words, recovery times are introduced mainly to reduce primary delays, while buffer times are defined to limit the propagation of secondary delays. Most studies ensure the robustness of timetables to small disturbances in the offline scheduling process by perturbing the nominal timetable with observed or simulated disturbances (see for instance [204], [208], [197], [209], [210], [211], and [212]). On the contrary, as reported in [204], few contributions aim at increasing the timetable robustness in the real-time rescheduling process, although it can be more effective than the offline process, which is an open-loop control process. Such a lack is mainly due to the high combinatorial complexity of multi-objective MILP problems that often make computation times too high for the real-time requirements. In order to overcome such a limitation, a good compromise can be achieved by combining the optimal rescheduling by MILP techniques with a heuristics that simplifies and speeds up the rescheduling procedure [200].

Another important aspect to take into account is that of iteratively evaluating and improving the effectiveness of the real-time rescheduling. To this aim, due to the recalled multi-objective nature of the problem and thanks to the large amount of available data, Multi-Criteria Decision Making (MCDM) techniques can represent an efficient tool. In fact, in the last decades, the use of MCDM methods has extensively increased in different application areas (see the review in [178]). Several MCDM approaches are available, each with its own advantages and disadvantages [178]. Most MCDM techniques are usually applied prior to decision making or project execution, while DEA is more often utilized for the evaluation of schemes already implemented [179]. The ease of use of the DEA and its ability to quantify results make this technique an efficient tool for simplifying data analysis. Moreover, thanks to the possibility of combining DEA with the fuzzy set theory (see [199]), the TD may take into account uncertainties affecting the rescheduling process.

In the above context, this work presents a three-steps decision making procedure, which is characterized by three main novelties that are able to significantly alleviate the TD real-time work: (1) a rescheduling technique consisting in solving a bi-objective MILP optimization problem with the twofold aim of minimizing delays and maximizing the robustness of the timetable, (2) a heuristics to speed up the rescheduling process so as to cope with the real-time requirements, (3) an offline self-learning procedure based on the fuzzy DEA technique to evaluate and rank the effectiveness of alternative rescheduling actions.

The MILP approach in the first step of the decision making procedure is formulated on the basis of the techniques of the previous Section 4.1. In particular, the railway traffic is still considered as a sequence of events that can be assigned when trains, technical, and operator constraints are fulfilled. However, here it is presented a bi-objective formulation that, as already stated, aims at minimizing delays at each station and maximizing the overall robustness of the timetable. With respect to other existing contributions (see also [186], [114], [200], [118], [117]), here the focus is on preventively reducing the impact of additional disturbances by incorporating the robustness maximization directly in the rescheduling problem statement, rather than providing a rescheduling procedure coping with disturbances after their occurrence. Furthermore, here it is combined the simplicity of heuristics -to reduce the computational complexity- with the precision of MILP.

In addition, this decision making procedure provides the offline self-learning procedure to allow a performance evaluation of the obtained solution. Moreover, the DEA technique has been used in the railway scheduling context only in [213], and in a setting very different from the present one. In fact, contrarily to [213], the DEA technique is here considered in a fuzzy setting to cope with data imprecision and uncertainty and the focus is on the robustness concept, which is disregarded in [213].

Summing up, the procedure is useful for railway companies (to provide their customers with on-time services, reduce sanctions or penalties, and avoid possible errors caused by a manual rescheduling) and for passengers (to reduce waiting times and delays or to limit travel discomforts).

4.2.1 Basics on the fuzzy cross-efficiency DEA technique

The fuzzy cross-efficiency DEA is an evolution of the cross-efficiency DEA already presented in Section 3.4.1. Whereas the cross DEA technique allows obtaining a ranking of the **deterministic** or **crisp** cross-efficiencies of a set of F alternatives, the fuzzy cross DEA allows to take into account some uncertainties that typically affect the performance indices of the alternatives. Hence, by modeling uncertain performance index values by triangular fuzzy numbers [214], a generalization to the previous approach has been introduced in [199], leading to the so-called fuzzy cross-efficiency DEA. In the sequel are briefly described the main steps of such an approach.

Consider a set of F alternatives, whose cross-efficiency has to be evaluated with respect to n conflicting criteria divided into a subset of W criteria to be maximized and a subset of H criteria to be minimized. Criteria are quantified via appropriate performance indices. Now, let $y_{w,f}$ be the value of the w-th performance index to be maximized (w=1,…,W) and $x_{h,f}$ be the value of the h-th performance index to be minimized (h=1,…,H), both evaluated when the f-th alternative is active (f=1,...,F). Then, let $u_{w,f}$ and $v_{h,f}$ be the weighting coefficients associated, respectively, with the w-th performance index to be maximized and the h-th performance index to be minimized. First, the focus is on the performance indices to be minimized. Instead of using a single deterministic value $x_{h,f}$ associated with the generic h-th criterion and the generic f-th alternative, a triple $\tilde{x}_{h,f} = (x^{o}_{h,f}, x^{m}_{h,f}, x^{p}_{h,f}) \in \Re^3$ -i.e., a triangular fuzzy number- is defined, whose entries are, respectively, the most optimistic, the modal, and the most pessimistic estimate of the f-th alternative performance index under the h-th criterion. Analogously, the triple $\tilde{y}_{w,f} = (y^p_{w,f}, y^m_{w,f}, y^o_{w,f}) \in \Re^3$ is defined, whose entries are, respectively, the most optimistic, the modal, and the most pessimistic estimate of the f-th alternative under the w-th criterion to be maximized. Both triples represent triangular fuzzy numbers, i.e., they can assume different real values with a degree of possibility in [0,1], according to suitable triangular membership functions defined in [199]. Hence, for each f-th alternative it is required to determine a cross-efficiency $\tilde{CE}_f$ which is now a fuzzy variable represented by the triple (CE^p_f, CE^m_f, CE^o_f). In particular, following [199], the triple defining $\tilde{CE}_f$ is computed using an approach that is based on three main goals: (1) the maximization of the modal value $\tilde{CE}^m_f$, (2) the minimization of the distance of the pessimistic value $\tilde{CE}^p_f$ from the modal value $\tilde{CE}^m_f$, and (3) the maximization of the distance of the optimistic value $\tilde{CE}^o_f$ from the modal value $\tilde{CE}^m_f$. In more detail, for each alternative f, a *Positive Ideal Solution* (PIS) is defined as the ideal solution with cross-efficiency triple $(CE^{PIS}_{1,f}, CE^{PIS}_{2,f}, CE^{PIS}_{3,f})$ that simultaneously satisfies goals (1)-(3), i.e., such that:

$$F^{PIS}_{1,f} = \max CE^m_f, \quad F^{PIS}_{2,f} = \min[CE^m_f - CE^p_f], \quad F^{PIS}_{3,f} = \max[CE^o_f - CE^m_f] \qquad \text{(Eq. 4-28)}$$

so that the PIS cross-efficiencies are obtained as:

$$\begin{cases} CE^{PIS}_{1,f} = F^{PIS}_{1,f} - F^{PIS}_{2,f} \\ CE^{PIS}_{2,f} = F^{PIS}_{1,f} \\ CE^{PIS}_{3,f} = F^{PIS}_{1,f} + F^{PIS}_{3,f} \end{cases} \qquad \text{(Eq. 4-29)}$$

The PIS is determined solving three problems similar to (Eq. 3-17)-(Eq. 3-21), where the objective functions are in turn the three equations in (Eq. 4-28), and constraints are obtained by appropriately modifying (Eq. 3-18)-(Eq. 3-21) to cope with the fuzzy character of the performance indices (not report here for the sake of brevity but refer to [199] for details). Since in practice such a maximally efficient alternative as the PIS can never be obtained, as goals (1)-(3) can never be reached simultaneously by the same weight set, from the PIS of the f-th alternative a compromise fuzzy cross-efficiency value (CE^p_f, CE^m_f, CE^o_f) is determined. The corresponding weight set is determined solving a fuzzy multi-objective linear programming problem by introducing an auxiliary variable and by using a procedure proposed in [215] for a generic fuzzy multi-objective programming problem and adapted to the fuzzy cross-efficiency DEA case in [199].

Finally, the crisp cross efficiencies DCE_f for each alternative f=1,…,F are determined by defuzzifying the obtained triple using the well-known center of the area method [214]:

$$DCE_f = \frac{CE_f^p + CE_f^m + CE_f^o}{3}. \quad \text{(Eq. 4-30)}$$

The resulting crisp values obtained for each alternative may then be used to define a ranking among them.

4.2.2 The Proposed Decision Making Procedure

The proposed decision making procedure is schematically illustrated in the flowchart in Figure 4-4 and consists of three main steps.

Once a disturbance occurs, the TD queries the database that contains the following information:

1) The *Time Horizon* (*TH*) in which the optimization procedure (Step 1) has to be performed. This value depends on the current disturbance and on the network complexity and infrastructure.

2) The *nominal timetable in TH*, namely the timetable originally scheduled offline over the time horizon.

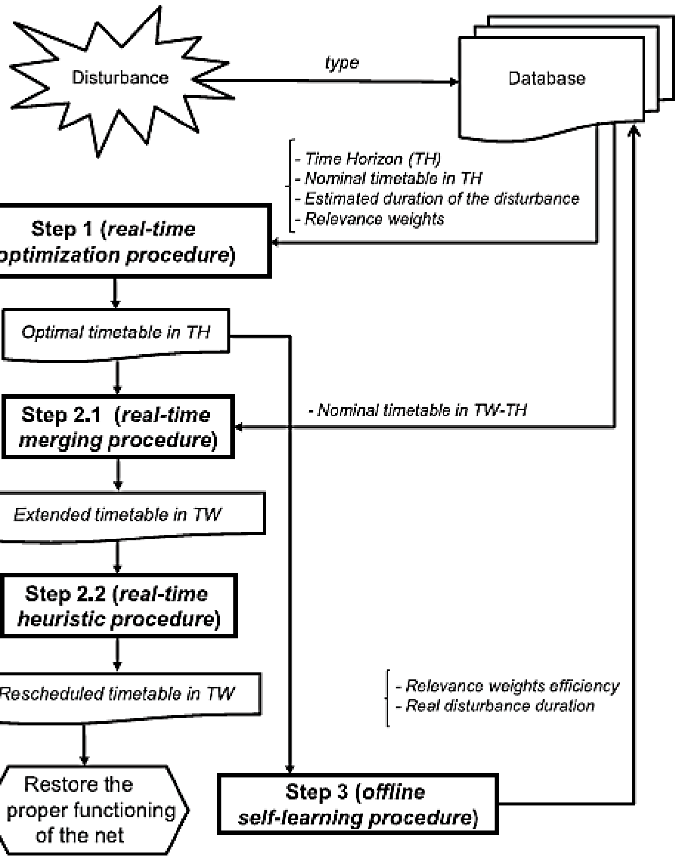


Figure 4-4. A flow-chart representation of the decision making procedure.

3) An *estimate of the duration of the disturbance* calculated using statistical methods, as usually happens in railway networks [117], or other more sophisticated techniques (see for instance [216]).

4) The most suitable *relevance weights* to be used in the objective function of the optimization procedure in Step 1. Again, these values are provided by the self-learning procedure in Step 3.

After the database querying, the real-time optimization procedure in Step 1 is executed (see Subsection 4.2.2A for more details), obtaining the optimal timetable in *TH* as the best compromise between two objectives (suitably weighted by the relevance weights): the minimization of delays of trains and the maximization of the robustness of the timetable. The resulting timetable is guaranteed to be adherent to the nominal timetable, conflict-free (i.e., without unfeasible train coincidences in stations or crossings at single-track lines) within *TH*, and robust with respect to additional small disturbances.

The real-time merging procedure in Step 2.1 (see Subsection 4.2.2B) provides an extended timetable in the whole scheduling *Time Window* (*TW*). This is done combining the optimal timetable in TH with the nominal one in the remaining time horizon of length *TW-TH* (also obtained by the database queering). Note that the time interval *TW* is the overall time window for which the aperiodic rescheduling has to be performed, so it is at most one day. Obviously, the absence of conflicts is only guaranteed in *TH*, which cannot be too large in order to ensure an optimal rescheduling in real-time. Therefore, an analysis of conflicts eventually arising in the remaining interval of length *TW-TH* is performed. This is done at Step 2.2 via the real-time heuristic procedure detailed in Subsection 4.2.2C, which allows obtaining in real-time a conflict-free rescheduled timetable in *TW*.

Finally, Step 3 is applied. It consists in an offline self-learning procedure aimed at updating the database in order to better select the relevance weights in the case of future applications of the technique for disturbances of the same type. Indeed, since the decision making procedure is based on estimates of the duration of the disturbance, it is important to assess the effectiveness of the adopted solution, so as to obtain a more accurate rescheduling in the event that a disturbance of the same type occurs again. To this aim, Step 3 of the presented technique applies a cross-efficiency fuzzy Data Envelopment Analysis to evaluate the efficiency of the obtained solution (in terms of delay at each station and robustness of the timetable) compared to that of other possible solutions obtained by varying the relevance weights in the objective function of the optimization procedure (see Subsection 4.2.2D for more details on this step). As output of Step 3, the most suitable relevance weights along with the type and duration of the real disturbance are provided and stored in the database.

An important remark should be made. The real-time rescheduling is based on real-time information about the railway network state. The TD is continuously updated about the evolution of the system. Therefore, if the duration of the real disturbance is overestimated, then the rescheduled timetable is still valid. On the contrary, if the disturbance is underestimated, then, thanks to the short computation times of the whole technique, the TD can restart the procedure with updated information about the duration of the disturbance and the state of the network, thus obtaining a more appropriate rescheduling.

A. Step 1: The real-time optimization procedure

Step 1 aims at finding, in the given time horizon *TH*, the optimal compromise between the minimization of delays in the railway network, due to the occurrence of a generic disturbance, and the maximization of the robustness of the resulting timetable to secondary delays and to the occurrence of additional disturbances. This is achieved by stating and solving a MILP problem. Before introducing the mathematical formulation of the problem, it is briefly recalled some background on railway networks and provided the basic notation used in the rest of the work (see also Table 4-7).

Railway networks are usually divided into connected segments, which can be of two types (see Figure 4-5): line segments and station segments [186], [201]. The former include tracks linking stations, the latter include tracks in stations. Each segment can be composed by single or parallel tracks, and each track can be occupied by at most one train at a time.

As an example, Figure 4-5 provides the representation of a generic infrastructure: station A (segment j_A) comprises two tracks (tc_1 and tc_2), station C (j_C) three tracks (tc_1, tc_2, tc_3), and they are connected by a line segment (i.e., a rail connection) j_B, which is constituted by a single track.

The optimization procedure in Step 1 is based on the notion of *event*, which is a train request to occupy a track for a well-defined time interval. The occurrence of a disturbance deviates the involved train from its nominal behaviour. This implies that the end of the affected event has to be necessarily shifted according to the duration of the occurred disturbance before the application of Step 1.

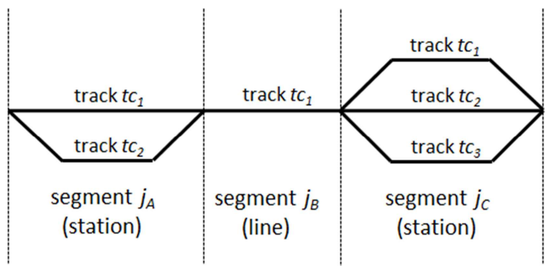


Figure 4-5 An example of a generic railway line (adapted from [186]).

Table 4-7. Summary of the notation.

Parameters	***Physical meaning***
T	Set of trains to be rescheduled in *TH*
i	Generic train in T
g_i^{train}	Length of train $i \in T$
B	Set of segments of the network
j	Generic segment in B
Pj	Set of tracks composing segment j
tc	Generic track in P_j
$g_{j,tc}^{track}$	Length of generic track tc of segment j
E	Set of events related to all trains in *TH*
k	Generic event in E
$K_i \subseteq E$	Ordered set of events of the generic train $i \in T$ (according to the nominal timetable)
n_i	Last event in K_i
$L_j \subseteq E$	Ordered set of events of the generic segment $j \in B$ (according to the nominal timetable)
$E^{connection} \subseteq E \times E$	Set of couples $(k, \hat{k})$ formed by connected events (i.e., $\hat{k}$ must not start until k has ended)
$g_{k,\hat{k}}^{connection}$	Minimum exchange time between connected events
h_k	Binary variable indicating whether event k corresponds to a 'stop in station' or not
$R_{i,k}$	Robustness index of train i within *TH* with respect to event k
(α,β)	Relevance weights
$Flow_k$	Average percentage of passengers getting on train involved at a 'stop in station' event k
TT_k	Percentage of tightness of track at event k
$NSucT_k$	Number of trains that can be perturbed by the delayed train after the event k in *TH*
d_k	Nominal duration of event k
d_k^{trip}, d_k^{stop}	Minimum nominal duration for trip and stop times of event k
δ_k	Recovery time for event k in the nominal timetable
$b_k^{nominal}, e_k^{nominal}$	Start and end times of event k in the nominal timetable
$b_k^{static}, e_k^{static}$	Start and end times of event k that has already begun when the disturbance occurs
o_k	Point of origin of event k, indicating whether trains associated with events k and k+1 are traveling or not in the same direction
Δ_j^M	Safety time between two trains travelling in opposite directions at segment j
Δ_j^F	Safety time between two trains travelling in the same direction at segment j
M	Large positive constant (i.e., the length of the largest possible *TH*)
w_k	Fixed threshold to activate the binary variable ε_k
$Buff_{\max}$	Upper bound of variable $Buff_k$, $k \in E$
Decision variab.	***Physical meaning***
z_k	Real variable indicating the delay of the generic event k
$Buff_k$	Real variable indicating the buffer time associated with the generic 'stop in station' event k
x_k^{begin}	Real variable indicating the start time of event k
x_k^{end}	Real variable indicating the end time of event k
$q_{k,j,tc}$	Binary variable indicating whether event k uses track tc of segment j or not
$\lambda_{k,\hat{k}}$	Binary variable indicating whether event k is rescheduled to occur after $\hat{k}$
$\gamma_{k,\hat{k}}$	Binary variable indicating whether event k occurs before $\hat{k}$ as in the nominal timetable
ε_k	Binary variable indicating if train i reaches (or not) the stop in station event k with a delay larger than w_k time units

As already specified, the primary delay causes secondary delays on the other trains in the timetable. In this work, it is called *delay of the generic event k* (and denote it z_k) the difference between the end of the event after its rescheduling (due to the occurrence of a disturbance) and its nominal end. Furthermore, it is called $Buff_k$ the *buffer time* added to event k, i.e., an extra time suitably added to the estimated duration of event k corresponding to stop in station.

By an analogy with traditional risk management procedures [217], the primary aim of introducing buffer times in the timetable is to increase the flexibility and achieve a specific goal, i.e., to let the network absorb secondary delays and possible additional disturbances. However, the introduction of buffer times can further increase the delay of events associated with a stop in station and ultimately the delay propagation in the whole network. For this reason, in order to provide a better service for passengers, the first goal of the optimization procedure is to minimize what here is called *cumulative delay*, which is defined as the sum on the whole network of the extended delays. The extended delay of an event is calculated as the sum of the delay associated with the event and the corresponding buffer time, when the event is a stop in station, otherwise it corresponds to the delay of the event. The second goal is to minimize the effect of possible additional disturbances, which is realized as explained before, by adding proper buffer times to the events corresponding to stop in station. Their effect on the robustness of the obtained timetable can be quantified by a suitable index R, which has to be maximized [204], [197].

Summarizing, given the set of trains T (whose generic element is indexed by i), the set of events E (whose generic element is indexed by k), and being $K_i \subseteq E$ the ordered set of events associated with train i, the objective function aims at minimizing the cumulative delay of the network while maximizing its robustness:

$$Min\ [\alpha \cdot \sum_{k \in E} (z_k + Buff_k \cdot h_k) - \beta \cdot R]. \qquad \text{(Eq. 4-31)}$$

In (Eq. 4-31) α and β represent the relevance weights assigned to normalize and balance the two terms of the objective function, and these two parameters are chosen based on the TD preferences stored in the database or according to the suggestions resulting from Step 3. Furthermore, the binary parameter h_k specifies whether the event k is associated with a station segment (h_k =1) or with a line segment (h_k =0). In fact, the buffer time $Buff_k$ can be added to the delay z_k only in case of a stop in station event. Therefore, ($z_k + Buff_k \cdot h_k$) represents the extended delay. The robustness of the network is quantified in (Eq. 4-31) by index R defined as follows [197]:

$$R = \sum_{i=1}^{|T|} \sum_{k=1}^{|K_i|} R_{i,k},$$

$$\text{where } R_{i,k} = Buff_k \cdot Flow_k \cdot TT_k \cdot NSucT_k \cdot \frac{(|K_i| - k)}{|K_i|} \cdot h_k, \text{ for } k = 1,\ldots,|K_i|. \qquad \text{(Eq. 4-32)}$$

In eq. (Eq. 4-32) $Flow_k$ is the average percentage of travelers getting on the *i*-th train at event *k*. Note that, given an event *k* according to the considered notation, it is univocally determined which is the train involved in it. According to [197], parameter $TT_k = (|K_i| - (k-1))/|K_i|$ is the percentage of tightness of tracks between stations, where the tightest track is defined as the longest distance track, i.e., given $|K_i|$ events for the *i*-th train, the tightest track has $TT_1 = 1$, the second tightest track has $TT_2 = (|K_i| - 1)/|K_i|$, the third one has $TT_3 = (|K_i| - 2)/|K_i|$, and so on. The parameter $NSucT_k$ is the number of trains in the time horizon that could be perturbed by the delayed train *i* after the event *k*.

The minimization of (Eq. 4-31) is performed under the following constraints:

<u>*Train constraints:*</u>

$$x_k^{end} = x_{k+1}^{begin}, \quad \forall k \in K_i - \{n_i\},\ i \in T \qquad \text{(Eq. 4-33)}$$

$$x_k^{end} - x_k^{begin} - Buff_k \cdot h_k \geq d_k - \varepsilon_k \delta_k\ ,\ \text{ if } d_k > \varepsilon_k \delta_k, \quad \forall k \in E$$
$$\text{where:} \begin{cases} \delta_k = 0 \text{ if } (d_k < d_{k_{trip}}, h_k = 0) \vee (d_k < d_{k_{stop}}, h_k = 1) \\ \delta_k > 0 \text{ if } (d_k \geq d_{k_{trip}}, h_k = 0) \vee (d_k \geq d_{k_{stop}}, h_k = 1) \end{cases} \text{ and } \begin{cases} \varepsilon_k = 1 \text{ if } \ z_k + Buff_k > w_k \\ \varepsilon_k = 0 \ \text{ otherwise} \end{cases} \qquad \text{(Eq. 4-34)}$$

$$x_k^{begin} \geq b_k^{\text{nominal}}, \quad \forall k \in E:\ h_k = 1 \qquad \text{(Eq. 4-35)}$$

$$x_k^{begin} = b_k^{static}, \quad \forall k \in E:\ b_k^{static} > 0 \qquad \text{(Eq. 4-36)}$$

$$x_k^{end} = e_k^{static}, \quad \forall k \in E:\ e_k^{static} > 0 \qquad \text{(Eq. 4-37)}$$

$$x_k^{end} - Buff_k \cdot h_k - e_k^{\text{nominal}} \leq z_k, \quad \forall k \in E \qquad \text{(Eq. 4-38)}$$

<u>*Technical constraints*</u>*:*

$$\sum_{tc \in P_j} q_{k,j,tc} = 1, \quad \forall k \in L_j,\ j \in B \qquad \text{(Eq. 4-39)}$$

$$q_{k,j,tc} + q_{\hat{k},j,tc} - 1 \leq \lambda_{k,\hat{k}} + \gamma_{k,\hat{k}}, \quad \forall k, \hat{k} \in L_j, tc \in P_j, j \in B; k < \hat{k}$$
$$x_{\hat{k}}^{begin} - x_k^{end} \geq \Delta_j^M \gamma_{k,\hat{k}} - M(1 - \gamma_{k,\hat{k}}), \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} \neq o_k \qquad \text{(Eq. 4-40)}$$

$$x_{\hat{k}}^{begin} - x_k^{end} \geq \Delta_j^F \gamma_{k,\hat{k}} - M(1 - \gamma_{k,\hat{k}}), \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} = o_k \qquad \text{(Eq. 4-41)}$$

$$x_k^{begin} - x_{\hat{k}}^{end} \geq \Delta_j^M \lambda_{k,\hat{k}} - M(1 - \lambda_{k,\hat{k}}), \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} \neq o_k \qquad \text{(Eq. 4-42)}$$

$$x_k^{begin} - x_{\hat{k}}^{end} \geq \Delta_j^F \lambda_{k,\hat{k}} - M(1 - \lambda_{k,\hat{k}}), \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k}, o_{\hat{k}} = o_k \qquad \text{(Eq. 4-43)}$$

$$\lambda_{k,\hat{k}} + \gamma_{k,\hat{k}} \leq 1 \quad \forall k, \hat{k} \in L_j, j \in B; k < \hat{k} \qquad \text{(Eq. 4-44)}$$

$$g_i^{train} q_{k,j,tc} \leq g_{j,tc}^{track} \quad \forall k \in (K_i \cap L_j), tc \in P_j, j \in B, i \in T \qquad \text{(Eq. 4-45)}$$

Operator preferences:

$$z_k + Buff_k \cdot h_k - w_k \le M\varepsilon_k \quad \forall k \in E \qquad \text{(Eq. 4-46)}$$

$$x_{\hat{k}}^{begin} - x_k^{end} \ge g_{k,\hat{k}}^{connection} \quad \forall k,\hat{k} \in E; k,\hat{k} \in E^{connection} \qquad \text{(Eq. 4-47)}$$

Variables constraints:

$$x_k^{begin}, x_k^{end}, z_k \ge 0 \quad \forall k \in E \qquad \text{(Eq. 4-48)}$$

$$\gamma_{k,\hat{k}}, \lambda_{k,\hat{k}} \in \{0,1\} \quad \forall k,\hat{k} \in L_j, j \in B; k < \hat{k} \qquad \text{(Eq. 4-49)}$$

$$q_{k,j,tc} \in \{0,1\} \quad \forall k \in L_j, tc \in P_j, j \in B \qquad \text{(Eq. 4-50)}$$

$$\varepsilon_k \in \{0,1\} \quad \forall k \in E \qquad \text{(Eq. 4-51)}$$

$$Buff_k \in [0, Buff_{\max}] \quad \forall k \in E \qquad \text{(Eq. 4-52)}$$

- *Train constraints:*

- Given the ordered set of events K_i associated with train *i*, whose last element is n_i, constraint (Eq. 4-33) forces event (*k*+1) to begin as soon as event *k* ends.
- Constraint (Eq. 4-34) states that the duration of the rescheduled event *k*, increased of the corresponding buffer time $Buff_k$, has to be higher than, or equal to, the offline scheduled duration d_k (both for trips, i.e., when h_k=0, and stops, i.e., for h_k=1) when it is lower than the minimum offline scheduled duration (i.e., $d_{k_{trip}}$ for trips or $d_{k_{stop}}$ for stops). However, when the offline scheduled duration is equal to, or higher than, the minimum offline scheduled duration, it is decreased by a factor $\varepsilon_k \cdot \delta_k$ (where the binary variable ε_k is equal to 1 if the extended delay of the considered event *k* is higher than a fixed threshold w_k, or 0 otherwise). The constant δ_k is a recovery time established in the nominal timetable.
- Constraint (Eq. 4-35) imposes that the beginning of the rescheduled event *k* has to be higher than or equal to the beginning chosen offline in the nominal timetable (i.e., b_k^{nominal}), when the event is a stop in station event (i.e., if h_k=1).
- Constraint (Eq. 4-36) states that the beginning of event *k* has to be equal to its static value (b_k^{static}) if the static value is higher than zero, that is, when event *k* has started before the occurrence of the disturbance. Similarly, constraint (Eq. 4-37) states that the end of event k has to be equal to its static value (e_k^{static}) if the static value is higher than zero, that is, when event *k* has started before the occurrence of the disturbance.

- Constraint (Eq. 4-38) imposes that the difference between the end of the rescheduled event *k* delayed of the corresponding buffer time $Buff_k$ and its offline scheduled end (as established in the nominal timetable, i.e., e_k^{nominal}) has to be lower than or equal to the corresponding delay z_k.

- *Technical constraints:*

- Constraint (Eq. 4-39) imposes that a single track cannot be occupied by more than one train at the same time.

- In constraint (Eq. 4-40), given two events k and $\hat{k}$ related to a track *tc*, when *k* occurs before $\hat{k}$ (as in the offline scheduling) track *tc* must be reserved to event *k*, while when *k* occurs after $\hat{k}$ (as may happen in case of rescheduling), track *tc* must be reserved to event $\hat{k}$ (that is, either $\gamma_{k,\hat{k}}$ is equal to 1 or $\lambda_{k,\hat{k}}$ is equal to 1).

- In constraint (Eq. 4-41), in case of trains travelling in opposite direction (i.e., when $o_{\hat{k}} \neq o_k$), any event $\hat{k}$ subsequent to event *k* and requiring the same track used by *k*, has to start when a Δ_j^M time interval has elapsed after the end of event *k*. Similarly, in constraint (Eq. 4-42), in case of subsequent trains (i.e., if $o_{\hat{k}} = o_k$), any event $\hat{k}$ subsequent to *k* and requiring the same track used by *k* has to start only when a Δ_j^F time interval has elapsed after the end of *k*.

- In constraint (Eq. 4-42), in case of trains travelling in opposite direction (i.e., $o_{\hat{k}} \neq o_k$) any event *k* subsequent to $\hat{k}$ and requiring the same track used by $\hat{k}$ has to start when a Δ_j^M time interval has elapsed after the end of $\hat{k}$. Similarly, in constraint (Eq. 4-43), in case of subsequent trains (i.e., $o_{\hat{k}} = o_k$), any event *k* subsequent to $\hat{k}$ and requiring the same track used by $\hat{k}$ has to start when a Δ_j^F time interval has elapsed after the end of $\hat{k}$.

- Constraint (Eq. 4-44) imposes that an event *k* can either occur after or before a generic event $\hat{k}$.

- Constraint (Eq. 4-45) imposes that the length of a generic train *i* should not exceed the length of the track it occupies.

- *Operator preferences:*

- Constraint (Eq. 4-46) imposes that, when an event *k* is rescheduled, the sum of its buffer time $Buff_k$ and of its delay z_k (i.e., the extended delay of the event *k*) minus a threshold w_k is lower than or equal to a large positive constant *M*.

- According to constraint (Eq. 4-47), event $\hat{k}$ cannot start if *k* is not ended and a constant time of connection $g_{k,\hat{k}}^{connection}$ between the two events has not elapsed.

➢ *Variables constraints:*

- Constraint (Eq. 4-48) states that the beginning and the end of a generic event *k*, as well as the corresponding delay z_k, are non-negative variables.

- Constraint (Eq. 4-49) imposes that $\gamma_{k,\hat{k}}$, i.e., the variable used to specify if event *k* occurs before $\hat{k}$ (value 1) or not (value 0), and $\lambda_{k,\hat{k}}$, i.e. the variable used to specify if *k* is rescheduled to occur after $\hat{k}$ (value 1) or not (value 0), are binary variables.

- Constraint (Eq. 4-50) imposes that $q_{k,j,tc}$, i.e., the variable used to specify if the event *k* uses track *tc* of segment *j* (value 1) or not (value 0), is a binary variable.

- Constraint (Eq. 4-51) imposes that ε_k, i.e., the variable used to specify if the delay of event *k* is higher than a fixed threshold w_k (value 1) or not (value 0), is a binary variable.

- Constraint (Eq. 4-52) states that $Buff_k$ can assume real values ranging between 0 and $Buff_{max}$ minutes, where the upper bound is suitably chosen depending on the average duration of the stop in station events.

The above optimization problem has at most $n_{var} = 8 \cdot |E|$ variables and $n_{const} = |K_i| \cdot |T| + 9 \cdot |E| + 2 \cdot |L_j| \cdot |B| + 4 \cdot |L_j|^2 \cdot |B| + |K_i \cap L_j| \cdot |P_j| \cdot |B| \cdot |T| + |E^{connection}| + |L_j| \cdot |P_j| \cdot |B|$ constraints.

It is to remark that the above problem statement, although inspired by [130], has several important differences: first of all, in the first part of the objective function it is considered the term (z_k+$Buff_k \cdot h_k$), i.e., the extended delay in case of stop in station events, rather than the single delay z_k; second, it is considered an additional term in the objective function that measures the robustness of the timetable to further disturbances; third, all the necessary constraints are included to model robustness.

The above real-time optimization procedure returns the optimal timetable in *TH* and guarantees that no conflict occurs in *TH*. However, given the dimension of the problem, the requirement to obtain a rescheduled timetable in a short computation time imposes the time horizon *TH* to be limited and in general shorter than the time window *TW* in which the overall timetable has to be defined (which may typically last up to 24 hours). Therefore, the following Step 2.1 is applied to extend the rescheduling to the whole time window.

B. Step 2.1: The real-time merging procedure

The merging procedure can be easily described as follows: all the events not included in *TH* but present in the nominal timetable and necessary for each rescheduled train to reach its final destination are shifted according to the optimal timetable in *TH.* All remaining events keep their nominal scheduling. The extended timetable in *TW* ends when the last train trip affected by the initial disturbance has reached its final destination. As previously discussed, the optimal timetable in *TH* is conflict-free within the time horizon. On the contrary, the merging procedure can lead to new conflicts arising after *TH* because of the presence of shifted events. The heuristic procedure in Step 2.2 allows identifying and solving such possible new conflicts.

C. Step 2.2: The real-time heuristic procedure

The real-time heuristic procedure is summarized in Figure 4-6. This heuristics mimics a job-shop scheduling problem [130], which aims at minimizing secondary delays while identifying and solving conflicts. In simple words, given the extended timetable in *TW*, the procedure detects the first conflict eventually arising after the time horizon, solves it, and goes on iteratively identifying and solving all conflicts until the last perturbed train has reached its final destination. More in detail, first, unfeasible coincidences in stations are removed (*PHASE1*). To this aim, all time values associated with each train (i.e., its arrival and departure time at each station) are checked: if a coinciding value between two trains is found, a waiting time is assigned to the train with a lower priority in order to maximize the service quality. Priority is given to direct trains or to trains with the highest traveling time. Note that different priorities may also be adopted according to the railway company policy, for instance favoring trains that on average transport more passengers. The waiting time in case of coincidences is assigned according to the following rule: to avoid simultaneous arrivals (in case of trains travelling in the same direction) and/or arrival-departure (in case of trains travelling in opposite directions) of two trains in a station, the train with lower priority has to wait in its previous station for the minimum waiting time (that is, the time required to allow passengers to get on and off from the other train, as established by the nominal timetable). Subsequently, both trains are allowed to depart, and, in case they are traveling in opposite directions and a crossing on a single-track line is thus generated, this will be solved in the next phase of the heuristics.

Once all unfeasible coincidences are solved, the first eventual crossing in a single-track line segment is identified (*PHASE2*). In this case the train with lower priority has to wait in the station located before the crossing point until the arrival of the highest priority train plus a safety time (as established by norms). The proposed heuristics is iterative, so that, in case of new unfeasible coincidences, they are solved going back to PHASE1. The procedure ends when the last perturbed train has reached its final destination and all conflicts are solved. Therefore, at the end of Step 2.2, the TD obtains the rescheduled timetable in TW and can restore the proper functioning of the network.

```
Algorithm: Heuristic procedure in Step 2.2
Input: ET = Extended Timetable in TW;
Set RT = Rescheduled Timetable in TW = ET;
run PHASE1
for (all couples of Trains (i_a, i_b) in RT)
    while (unfeasible coincidence) == true
        if (Travelling Time(i_a) > Travelling Time(i_b) OR i_a is direct)
            stop i_b in the previous station;
            departure(i_b) = departure(i_b) + waiting time;
        else
            stop i_a in the previous station;
            departure(i_a) = departure(i_a) + waiting time;
run PHASE2
for (all couples of Trains (i_a, i_b) in RT)
    while (crossing at a single-track line) = true
        if (Travelling Time(i_a) > Travelling Time(i_b) OR i_a is direct)
            stop i_b in the previous station;
            departure(i_b) = arrival(i_a) + safety time;
        else
            stop i_a in the station;
            departure(i_a) = arrival(i_b) + safety time;
for (all couples of Trains (i_a, i_b) in RT)
        if (unfeasible coincidence) = true
            go back to PHASE1
        else
            update RT
Output RT.
```

Figure 4-6. The pseudo-code summarizing the heuristic procedure in Step 2.2.

D. Step 3: The offline self-learning procedure

The proposed decision making procedure requires as input of Step 1 the relevance weights (α,β) to be used in (Eq. 4-31) that are stored in the database. The goal of Step 3 is that of identifying offline the most suitable weights, according to some conflicting criteria and based on the real duration of the disturbance. In such a way, the new weights are stored in the database and fruitfully used if disturbances of the same type occur in the future.

The proposed procedure is based on the concept of fuzzy cross-efficiency Data Envelopment Analysis [199]. It consists in a generalization of the traditional DEA, whose employment for train rescheduling has been proposed up to now only in [213] but in a deterministic setting. Since the decision making procedure is based on statistical evaluations and data affected by imprecisions and uncertainty, the fuzzy DEA approach is here chosen which is more appropriate in such context. In particular, it allows effectively representing by means of fuzzy numbers the uncertainty affecting performance indices. In the sequel it is shown how the fuzzy cross-efficiency DEA technique is used to define the offline self-learning procedure.

A series of F couples of relevance weights (α,β) is generated. For each of them, the optimization procedure of Step 1 is executed considering the real duration of the occurred disturbance. Thus, different timetables are provided, and their effectiveness is evaluated on the basis of some conflicting criteria: the delay at each station (to be minimized) and some robustness indices (to be maximized). The fuzzy cross-efficiency DEA allows us to select the most efficient couple of weights according to such conflicting criteria, as described in Section 4.2.1.

The first robustness index to be maximized is R defined in (Eq. 4-32) and is a measure of the robustness of the overall timetable. The other robustness indices are the *Weighted Average Distance* (*WAD*) calculated for each train $i \in T$ [208]:

$$WAD_i = \frac{\sum_{k \in K_i} \frac{(2 \cdot k - 1) \cdot Buff_k \cdot h_k}{2 \cdot N_i^{st}}}{\sum_{k \in K_i} Buff_k \cdot h_k}, \quad i \in T, \qquad \text{(Eq. 4-53)}$$

where N_i^{st} is the number of stations encountered by the i-th train along its trip within the time horizon. In simple words, index WAD_i describes the buffer time distribution along the trip for the i-th train and can assume values between 0 and 1. For example, a value of $WAD_i = 0.5$ corresponds to the fact that on average an equal amount of buffer times is allocated in the first half and in the second one of the train trip, while values smaller (bigger) than 0.5 relate to a shift in the buffer times distribution towards the beginning (end) of the train trip. Usually, it is preferable to have time reserves concentrated early on the line (i.e., a small WAD_i value) [209]. However, if disturbances occur later on the line, the time reserves located previously to the occurrence may be of no use. Hence, authors in [196] state that the average amount of time reserves should be allocated on the middle part of a line, with a slight shift to the beginning.

Using the fuzzy cross-efficiency DEA technique, a *Defuzzified Cross-Efficiency* (DCE_f) value is computed for each couple of weights considered (f=1,…,F). Based on such values, the F alternatives are ranked and the most appropriate couple of relevance weights is determined.

Problems in Step 3 (see Section 4.2.1) have a number of variables equal to $3 \cdot (W+H) \cdot F$ and a number of constraints equal to $3 \cdot (2 \cdot F+W+H)$.

After performing Step 3, the database of the decision making procedure is updated with the relevance weights related to the most efficient alternative and with the type and duration of the occurred disturbance, so that, the next time a disturbance of the same type affects the railway traffic, Step 1 of the procedure receives in input more appropriate weights.

4.2.3 A Real Case Study

Before introducing the case study, it is to point out that for sake of clarity in this section it is adopted a notation that slightly differs from the one in Section 4.2.2A. In more detail, as regards trains, segments, and tracks, the respective identifying number is used as subscript of the corresponding symbol, instead of reporting the number assignment. For example, when referring to the first train, it is written i_1 instead of i=1, similarly the first segment is identified as j_1, and so on.

A. The considered railway network

In this section it is illustrated and tested the proposed approach on the real case study presented in Section 4.2.3. The choice of this network is not casual, it is voluntarily considered for its simple topology, namely a corridor, even if the approach can be applied to any other network, so as to allow replications of the test. Furthermore, such topology characterized by the presence of single tracks, allows evaluating the effectiveness of the procedure in case of strict constraints. Indeed, in many stations only a train at a time can stop or pass through. As a result, the efficient management of the railway traffic flow in the considered case study requires special attention.

Hence it is considered a portion of a regional railway network, namely "Ferrovie del Sud Est" (hereinafter FSE), located in Apulia (Southern Italy). The interested reader can refer to Table 4-2 fo more details. It includes the stations between the sites of Mungivacca (i.e., segment j_0) and Putignano (i.e., segment j_{16}), and its traffic is currently managed by a TD using a CTC (Centralized Traffic Control) system, which provides a centralized control for signals and switches within a limited territory, using a single control console. The operative system is installed in j_0 that is an independent station, not controlled by the CTC, as is j_{16}, whereby these stations are not studied here. The line is single-tracked except for segment j_7, which is double-tracked. Moreover, segment j_{14} is a unitary capacity station.

Railway directions are described as even and odd, respectively corresponding to trains going north and south through the network; in the considered network each day there are 24 even trains and 22 odd trains. A safety time $\Delta_j^M = 3$ minutes is assumed for two trains traveling in opposite directions of the same j-th segment, and a safety time $\Delta_j^F = 1$ minute for trains in the same direction of the segment. Due to the average duration of the stop in station events, the $Buff_{max}$ parameter is here set to 4 minutes.

Data on several months have been exploited, showing that the proposed methodology outperforms the current technique used by FSE, which is largely based on the manual rescheduling approach (that is, based on the TD's personal experience and considering the existing physical and safety constraints). In the next subsection it is detailed the application of the proposed decision making procedure to a typical scenario among the available data sets. To confirm the effectiveness of the proposed procedure, the approach results are validated firstly by comparing them with those obtained by solving the same case study manually (i.e., by the TD), and then by evaluating the robustness of the solution when a random additional disturbance on the same line is observed.

A real data set is considered referring to a train going from j_{16} to j_0 that stops along the line at segment j_{11} for the occurring of a disturbance, which consists in a track unavailability (e.g., for the presence of an obstacle on the line) and occurring at 7:50 am, a rush hour with high level of traffic. Based on historical data, the estimated duration of such a disturbance is 15 minutes, and, according to the network topology, the most suitable time horizon is *TH*=50 minutes, as confirmed by some preliminary assessments that have been carried out (not reported here for the sake of brevity). Therefore, the considered disturbance affects five trains in *TH*.

Finally, based on the estimated order of magnitude of the two terms in (Eq. 4-31), the relevance weights assumed for the real-time optimization procedure are initialized at: α=1 and β=100.

The MILP problem is solved by the GLPK tool in the MATLAB environment since they are largely used for the resolution of such problems; clearly, other optimizers (such as, for instance, CPLEX) can be successfully used to this aim. The results obtained by applying Step 1 of the proposed procedure are reported in Table 4-8 , showing the cumulative delay of the network (i.e., the first term in (Eq. 4-31)), the average delay at stations, and the overall robustness (i.e., the second term in (Eq. 4-31)) in the chosen time horizon. By applying the real-time merging procedure in Step 2.1, the extended timetable in TW is obtained, as shown in Figure 4-7 by means of a Cartesian graph that represents the railway schedule in time and space. This diagram shows stop and running events for all the considered trains, reporting the corresponding duration and direction. In particular, the time line is plotted on the x-axis, while the railway line (i.e., the space) is plotted on the y-axis.

Table 4-8. Results from the real-time optimization procedure (Step 1).

Index	Value
Cumulative delay [min]	242.00
Average delay at stations [min]	7.14
Robustness index R	2.08

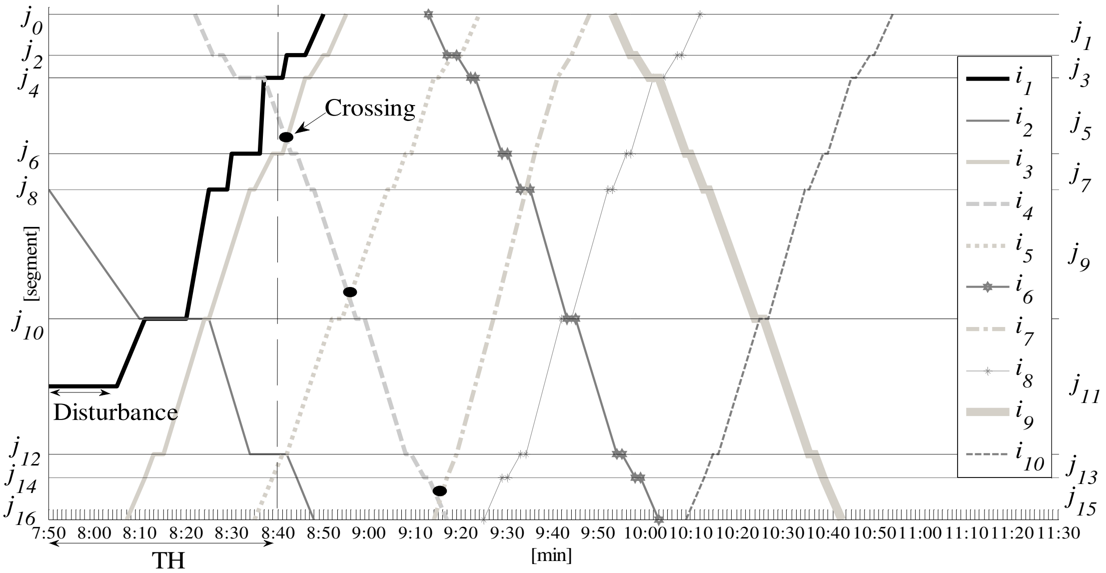


Figure 4-7. Extended timetable in *TW* (after Step 2.1).

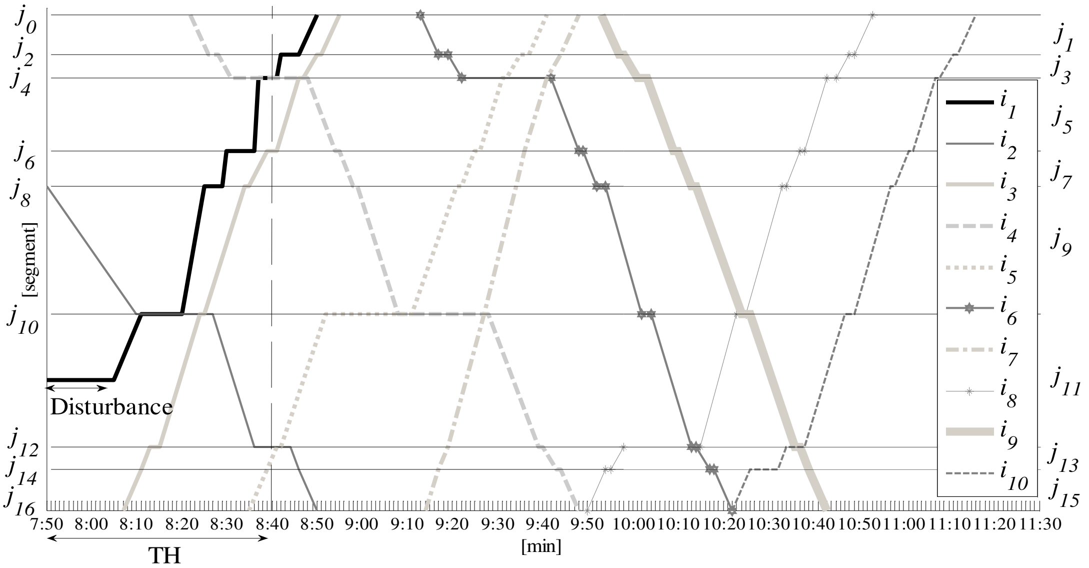


Figure 4-8. Rescheduled timetable in *TW* (after Step 2.2).

The station segments (reported on the left margin of Figure 4-7) are represented by lines parallel to the x-axis, while line segments (reported on the right margin of Figure 4-7) are the portion of the graph comprised between two consecutive stations. Stop events are represented as segments on station lines, whose length corresponds to the duration of the event. Running events are represented by oblique lines whose orientation indicates the train direction and whose slope represents the corresponding speed. For instance, the running events are oriented from bottom to up for even trains that travel from South to North (i.e., to the station on the upper end of the y-axis, that is, segment j0). By analyzing Figure 4-7, it is worth noting that the application of the optimization procedure allows obtaining a conflict-free timetable within the chosen TH. However, applying the merging procedure in Step 2.1, train conflicts may happen at time instants exceeding the considered time horizon. In fact, Figure 4-7 shows that trains i3 and i4 would cross at segment j5, i4 and i5 at j9, i4 and i7 at j15. All these crossings are at single-track lines, and hence they are unfeasible. Therefore, in order to identify and solve all conflicts arising after the chosen time horizon (up until a total of 24 hours), the heuristic procedure in Step 2.2 is applied, calculating a near-optimal solution, so that the TD can choose and communicate the new status to drivers as soon as possible. The corresponding rescheduled timetable in TW obtained after the heuristic procedure is graphically reported in

Figure 4-8. It is to remark that all conflicts on single tracks are solved. It is also to highlight that, at the end of Step 2.2, the TD can restore the proper functioning of the network by applying the rescheduled timetable in TW. In the evaluated case study this is possible in about 1 minute from the disturbance occurrence.

Table 4-9. Data for Step 3 of the decision making procedure: parameters to be minimized (a) and maximized (b).

Altern.# (α;β)		*Parameters to be minimized*										
		z_2 *[min]*	z_4 *[min]*	z_6 *[min]*	z_8 *[min]*	z_{10} *[min]*	z_{13} *[min]*	z_{15} *[min]*	z_{22} *[min]*	z_{24} *[min]*	z_{26} *[min]*	z_{28} *[min]*
1	o	11.70	14.40	13.50	13.50	12.60	0.00	0.00	9.00	9.00	9.00	8.10
	m	13.00	16.00	15.00	15.00	14.00	0.00	0.00	10.00	10.00	10.00	9.00
(1;1)	p	14.95	18.40	17.25	17.25	16.10	0.00	0.00	11.50	11.50	11.50	10.35
2	o	11.70	14.40	13.50	13.50	12.60	9.90	13.50	0.90	0.00	0.00	0.00
	m	13.00	16.00	15.00	15.00	14.00	11.00	15.00	1.00	0.00	0.00	0.00
(1;100)	p	14.95	18.40	17.25	17.25	16.10	12.65	17.25	1.15	0.00	0.00	0.00
3	o	11.70	18.00	21.60	21.60	20.70	9.90	13.50	0.90	3.60	3.60	2.70
	m	13.00	20.00	24.00	24.00	23.00	11.00	15.00	1.00	4.00	4.00	3.00
(1;500)	p	14.95	23.00	27.60	27.60	26.45	12.65	17.25	1.15	4.60	4.60	3.45
4	o	11.70	18.00	21.60	25.20	24.30	12.60	13.50	3.60	6.30	6.30	5.40
	m	13.00	20.00	24.00	28.00	27.00	14.00	15.00	4.00	7.00	7.00	6.00
(1;1000)	p	14.95	23.00	27.60	32.20	31.05	16.10	17.25	4.60	8.05	8.05	6.90
5	o	11.70	18.00	21.60	25.20	24.30	12.60	13.50	3.60	6.30	7.20	6.30
	m	13.00	20.00	24.00	28.00	27.00	14.00	15.00	4.00	7.00	8.00	7.00
(1;2000)	p	14.95	23.00	27.60	32.20	31.05	16.10	17.25	4.60	8.05	9.20	8.05
6	o	11.70	18.00	21.60	25.20	24.30	12.60	13.50	3.60	6.30	9.90	9.00
	m	13.00	20.00	24.00	28.00	27.00	14.00	15.00	4.00	7.00	11.00	10.00
(1;3000)	p	14.95	23.00	27.60	32.20	31.05	16.10	17.25	4.60	8.05	12.65	11.50

(a)

Altern.# (α;β)		*Parameters to be maximized*			
		WAD_1 *[%]*	WAD_2 *[%]*	WAD_3 *[%]*	R
1	o	0.12	0.00	0.14	1.65
	m	0.10	0.00	0.13	1.44
(1;1)	p	0.09	0.00	0.11	1.30
2	o	0.12	0.29	0.14	2.41
	m	0.10	0.25	0.13	2.09
(1;100)	p	0.09	0.23	0.11	1.88
3	o	0.35	0.29	0.37	2.80
	m	0.30	0.25	0.33	2.44
(1;500)	p	0.27	0.23	0.29	2.19
4	o	0.46	0.29	0.29	2.87
	m	0.40	0.25	0.25	2.49
(1;1000)	p	0.36	0.23	0.23	2.24
5	o	0.46	0.29	0.29	2.87
	m	0.40	0.25	0.25	2.49
(1;2000)	p	0.36	0.23	0.23	2.24
6	o	0.46	0.29	0.43	2.87
	m	0.40	0.25	0.38	2.50
(1;3000)	p	0.36	0.23	0.34	2.25

(b)

The offline self-learning procedure in Step 3 is applied to allow a performance evaluation of the obtained solution so as to properly update the database in the case that a similar disturbance occurs in the future. Table 4-9 reports the considered fuzzy inputs for Step 3. In particular, Table 4-9.a shows the parameters to be minimized (i.e., the delay at each station in *TH*), while Table 4-9.b reports those to be maximized (i.e., the index WAD_i for trains involved in *TH* and the overall robustness *R*). Such values are reported for different possible solutions obtained by varying the relevance weights in the objective function of the optimization procedure in Step 1 and taking into account the real duration of the occurred disturbance.

According to the self-learning procedure presented in Section 4.2.2 and detailed in Section 4.2.1, each parameter reports the corresponding optimistic (*o*), modal (*m*), and pessimistic (*p*) value, where optimistic and pessimistic values are obtained from the modal one (resulting from Step 1) considering the standard deviation of real data and assuming that the pessimistic value is closer to the modal one with respect to the optimistic value (the reader is referred to Section 4.2.1 for details on the fuzzy triples).

Note that a WAD_i value is not assigned to trains i_4 and i_5. This is due to the two conflicting goals of the optimization procedure. In fact, on the one hand, the robustness index WAD_i for trains i_4 and i_5 is equal to zero, since, in the considered time horizon, there are no subsequent trains that can be perturbed (i.e., the $NSucT_k$ parameter in (Eq. 4-32) is equal to zero). On the other hand, the delays minimization leads the MILP problem to set buffer times equal to zero (i.e., their lower bound). Hence, WAD_i cannot be determined for i=4,5. Also note that in Table 4-9 only six different alternatives are considered. This is due to the fact that, for small variations of the relevance weights, there are few relevant changes in the examined parameters. Hence, for the sake of brevity, in Table 4-9 are reported only few rescheduling alternatives, choosing among those that are significantly different from each other.

Table 4-10 shows the percentage value of the obtained defuzzified cross-efficiencies DCEf and the final ranking for the considered alternative solutions. By analyzing Table 4-10 it is important to note that the self-learning procedure confirms that the considered relevance weights (that is, the alternative numbered as #2) represent the best compromise between the two conflicting objectives in Step 1 (i.e., the minimization of all delays and the increase in the robustness). In fact, alternative #2 has the highest efficiency value. It is also worth noting that the obtained results are consistent with the order of magnitude of the two terms in (Eq. 4-31) (see Table 4-8). By increasing the ratio between the two relevance weights, it can be observed that the efficiency value decreases; in fact, although the robustness value increases, at the same time there is a greater rise in delays at the stations, thus leading to lower efficiency values. On the contrary, alternative #1 (i.e., with α=1 and β=1) has the worst efficiency; in fact, with such values of the relevance weights the first term of the objective function (Eq. 4-31) is too predominant over the second one, and thus the obtained timetable robustness is very limited.

Finally, Table 4-11 summarizes the computation times of the different phases of the decision making procedure when using a 3.40 GHz processor and 8 GB RAM PC. As it can be seen, the most burdensome part is Step 2.2 that requires almost 60 seconds. This is definitely consistent with an application in real time.

B. Validation of the proposed robust real-time rescheduling

By analyzing the data in Table 4-9.a, it can be observed that the initial delay is absorbed without spreading over the network (especially in alternative #2), thus providing a first confirmation of the fact that the obtained timetable is robust according to the robustness characteristic features provided in [218] and [206], despite the actual values of the relevance weights.

Table 4-10. Results of Step 3 of the proposed decision making procedure.

Alternative # ($\alpha;\beta$)	DCE_f **[%]**	**Ranking**
1 (1;1)	6.65%	6
2 (1;100)	68.01%	1
3 (1;500)	49.57%	2
4 (1;1000)	37.94%	3
5 (1;2000)	33.23%	4
6 (1;3000)	24.21%	5

Table 4-11. Problems dimensions and computation times.

Step	Problem dimension	Computation time [s]
Step 1	296x326 (variables x constraints)	1.90
Step 2.2	14 (solved conflicts)	59.80
Step 3	198x81 (variables x constraints)	11.90

In the following, to further validate the presented decision making procedure and to evaluate its impact on the company's performance, the obtained results are firstly compared with those obtained by manually solving the same case study already considered. Although there are no specific compulsory rules, usually TDs evaluate if, as a consequence of the occurred disturbance, any crossing or unfeasible coincidences arise, and, if so, they solve them one at a time, by applying almost the same logic already described in the heuristic procedure of Step 2.2. Hence, the results obtained by applying the proposed robust real-time rescheduling (i.e., Step 1 + Step 2.1 + Step 2.2) are compared with those obtained by a manual rescheduling, (i.e., applying the heuristics in Step 2.2 to the whole time window *TW*). Table 4-12 reports the cumulative delay (first row) and the average delay at stations (second row) resulting from the application of the decision making technique (column 2) and from the application of the manual rescheduling (column 3).

It is worth noting that by applying the proposed procedure the cumulative delay is equal to 242 minutes, while with the manual rescheduling it results equal to 1,284 minutes. Consequently, the presented rescheduling technique ensures a reduction of the cumulative delay for the case study higher than 80%. Moreover, when comparing the average delay at stations, the resulting reduction equals about 50%. It is also to be noticed that, by performing a manual rescheduling, errors or larger delays may arise, and the effectiveness of the adopted control actions can not be properly evaluated.

Furthermore, in order to validate the robustness of the obtained solution, the optimal timetable in *TH* (that is, the timetable obtained by applying Step 1 of the proposed decision making procedure) is compared with that obtained by only minimizing the cumulative delay (Section 4.2). To this aim, it is assumed that an additional small disturbance (e.g., a lamp failure on a colour light signal) occurs in the network after the initial primary disturbance has been coped with. To test the robustness of the rescheduled timetable under different scenarios, different characteristics are considered of the additional disturbance in terms of duration, affected train, and occurrence time. In particular, 100 replications are performed by randomly generating different duration values of the additional disturbance (in the range $[1,Buff_{max}]=[1,4]$ minutes), a different train directly affected by such a disturbance, and a different time of occurrence. Therefore, for the above rescheduled timetables are determined the average delay at stations and the number of arising conflicts that are solved by the subsequent heuristics. Table 4-13 reports for different durations of the additional disturbance the obtained results in terms of percentage of conflict-free timetables and average delay at stations within *TH*, as well as the corresponding percentage savings. By analyzing Table 4-13 it can be noticed that, when the duration of the additional disturbance is limited (i.e., below 2 minutes), keeping into account the robustness maximization allows obtaining remarkable savings both in terms of the percentage of conflict-free timetables and of the average delay at stations. On the contrary, when the duration of the new disturbance is greater than 2 minutes, the obtained results show no savings (or even a worsening) in terms of percentage of timetables without conflicts. Nonetheless, in such cases it is worth noting that the average delay at station is still substantially improved when the rescheduling is devoted to taking into account both the delays minimization and the robustness maximization.

This section is concluded by remarking that both railway companies and their passengers can take advantages from the presented decision making procedure. Indeed, in case of disturbances, rail companies need to provide their customers with a quality service, reducing delays or limiting travellers' discomfort. The proposed procedure trades off between two conflicting objectives. From the one hand it takes into account the minimization of the delay associated with each intermediate event, instead of the overall delay accumulated by the single train at the end of its route, thus providing lower intermediate passengers' waiting times, while limiting the waiting time over the entire journey. On the other hand, thanks to its ability to provide in real-time a robust rescheduled timetable, the method also allows reducing the spreading over the network of secondary delays and eventual additional primary delays. Moreover, the TD is provided with an automatic support tool that allows to rapidly restore the proper functioning of the network.

Table 4-12. Comparison of the decision making procedure with a manual rescheduling.

Index	Step 1 + Step 2	Manual Rescheduling	Reduction [%]
Cumulative delay [min]	242.00	1,284.00	81.20%
Average delay at stations [min]	7.14	14.00	49.00%

Table 4-13. Robustness validation with an additional stochastic disturbance in *TH*=50'.

Duration of additional disturbance [min]	Percentage of conflict-free timetables [%]			Average delay at stations within *TH* [min]		
	Delays min. + robustness max.	Delays minim.	Saving [%]	Delays min. + robustness max.	Delays minim.	Saving [%]
1.0	77.00%	31.00%	148.40%	0.04	0.19	79.00%
1.5	100.00%	50.00%	100.00%	0.00	0.14	100.00%
2.0	73.00%	27.00%	170.40%	0.03	0.25	88.00%
2.5	46.15%	53.84%	-14.30%	0.14	0.20	30.00%
3.0	18.18%	18.18%	0.00%	0.41	0.72	43.10%
3.5	46.15%	46.15%	0.00%	0.55	0.82	32.90%
4.0	15.38%	46.00%	-66.60%	0.88	0.89	1.10%

Finally, thanks to the self-learning procedure, the quality of the rescheduling is improved at each reapplication of the method. This way, companies can avoid sanctions (i.e., ticket refunds or penalties established by contract for late running), and customers can benefit from a reliable service without loss of money and time. Moreover, an efficient service can produce, as a secondary effect, a substantial shift of passengers (and freight) from other transportation modes to rail, thus helping reducing road traffic congestion and environmental impact.

4.3 A Bi-level Solving Algorithm for the Real-time Rescheduling Problem in Case of Disruption

The introductive section of this chapter highlighted the importance of introducing advanced technologies to support train dispatchers in minimizing losses and waste due to the occurrence of expected events affecting the nominal behavior of the railway traffic. As already explained, unexpected events can be classified into disturbances and disruptions, which can cause undesired effects both to the company and customers. Differently from Section 4.1 and 4.2, this section focuses on ***disruptions***, e.g., partial or full track blockade at a track section, which can lead to a large decrease in network capacity and require the application of severe actions, such as the cancelation and turning of trains, to avoid large delays spreading throughout the network. Actually, as already recalled, the approach used by train dispatchers is based on experience and if the traffic deviates from the nominal timetable TDs intervene by reordering, turning, cancelling ad rerouting trains minimizing passengers discomforts. This process becomes more complex in case of disruptions because these require the rescheduling of resource duties. In these cases they use contingency plans and emergency timetables to manage the traffic. This manual management is largely time-consuming and often leads to suboptimal outcomes as only a limited number of solutions can be reviewed for a rapid decision-making process.

The research on real-time train rescheduling is then focusing on the development of automated decision-support systems that can quickly model specific situations, calculate optimal solutions, and suggest the most appropriate decision to be implemented in a real-time context. Due to the scale, the complexity, and the short resolution time constraint, these problems remain challenging.

This section proposes a bi-level algorithm for the resolution of the real-time rescheduling problem in case of full-blockade. The railway traffic is modeled as a discrete event system and the rescheduling problem is set in a mixed integer linear programming fashion. The constraint model allows the representation of the essential characteristics of a disruption and permits the calculation of new feasible control actions to restore the nominal traffic conditions. More in detail, two constrained models of the disrupted area are presented, i.e., a macroscopic and a mesoscopic model. The first considers a high-level representation of the system, while the second one includes also specific control actions in the disrupted stations (i.e., platform assignment and train ordering at stations). The application of this algorithm, which considers both the macroscopic and mesoscopic models of the rescheduling problem, leads to a twofold advantage. On one hand it strongly reduces the required computation time with respect to the resolution of the full mesoscopically described problem. On the other hand, it allows the inclusion of the station scheduling model, which encounters the capacity limitations in the disrupted stations, that otherwise are neglected in the macroscopic formulation of the problem. Obviously, the amount of available platforms in station poses a limitation to the number of short-turns that can be performed and increases the number of cancelled train runs. Nevertheless, the obtained results are more realistic, and the limitations of the macroscopic description of the problem are overcome.

In the literature on rescheduling of railway traffic, depending on the research purposes, railway networks can be modeled on different scales and with different level of detail. A classification of the most frequently used models can be done according to the considered level of detail as:

- *Macroscopic models*: a high-level representation in which stations are nodes and the connecting tracks are links between the nodes. No details are provided on block sections and signaling, as well as on actions in stations. The results of such modeling technique can be departures and arrival times, and possible routes. Then, further refinements are required before the application of the dispatching results.
- *Microscopic models*: a low-level representation of the railway system, which includes at least block sections and switch locations in the network. Due to the complexity of the model computational complexity quickly increases and becomes an issue for large-scale systems.
- *Mesoscopic models*: a middle-level representation of the railway system, which includes elements from both the macroscopic and the microscopic modeling techniques. Certain parts of the network, like stations can be detailed whereas other parts are macroscopically modeled, such as links between stations. Then, mesoscopic models allow to obtain more detailed results with respect to macroscopic models, but risk to suffer from a high-computational complexity if the solving algorithms are not properly tailored to the specific issue.

One more classification regards the type of disruption affecting the railway traffic. In particular, two main types of disruption can be considered:

- *Partial blockade*: not all tracks included in a certain track section are blocked. It means that on the considered track section trains can circulate, but the capacity of the track section is decreased.
- *Full blockade*: all tracks of a certain track section are blocked. Consequently, no traffic is allowed on the track section affected by disruption.

In literature, both macroscopic and microscopic models are used for either full or partial blockades. In [152] a conflict detection and resolution algorithm is developed for a single track line layout with bi-directional traffic. A macroscopic model is applied defining stations as nodes where trains can overtake and line in between as single track where no overtake can take place. Disruptions are defined as time slots in which a track between two stations cannot be used. The rescheduling actions that can be performed are the retiming and the reordering of trains. The objective function to be minimized is the weighted sum of the difference between the actual arrival time at the destination and the scheduled arrival time at the destination for all trains. The MILP is solved for four different disruptions for a system of 6 trains and 5 stations using the GLPK solver. All instances could be solved to optimality within a second.

More advanced approaches to control disruptions have not been part of research until very recently in [219] and [220]. In the first contribution a constraint optimization problem is developed directed to partial and full track blockades. Since every disruption is unique, the emergency timetables currently in use will most likely not be optimal in every situation. The underlying idea in the research is therefore to adjust the nominal timetable to provide an optimal train service during the disruption. The adjusted timetable must deviate as little as possible from the nominal one to minimize passengers inconvenience. The research focuses on finding a stable cyclic timetable during a disruption that utilizes the available infrastructure optimally. Only the phase in which the disruption persists is considered and transitional effects into and after the disruption are neglected. For a partial and full blockade two separate integer problems are formulated based on alternative graphs similar to the ones used by D'Ariano [207]. The objective function aims at minimizing the number of cancelled trains and delays as well as balancing the number of trains in both directions and in time. The algorithms are applied on two real-world cases of the Dutch railways. The timetables from the optimization are compared to the emergency timetable of the NS. With an allowable delay of zero minutes, the same timetable as the emergency timetable is found. However, when allowing a delay of five minutes for all trains major differences start to occur as less trains need to be cancelled and the frequencies of the train lines increase.

4.3.1 A Macro and a Mesoscopic MILP model of the rescheduling problem

The railway traffic and network are here considered as a discrete-event system with a limited number of resources. In particular, a railway network can be described as a directed graph where nodes are the states of the system, directed arcs are its state evolution due to the occurrence of an event. Each node represents the presence (absence) of one or more trains in a station, depending on the capacity limits of the considered station, i.e., the number of available tracks (limited resources) where trains can dwell or be shunted. Two consecutive nodes can be connected by one or two arcs, standing for uni- or bi-directional train traffic; the weight on the arcs represents the number of tracks devoted to the corresponding traffic direction. The transition between two consecutive nodes via an arc represents the occurrence of a train run (i.e., the event). The paths in the graph, connecting origin nodes to destination nodes via a sequence of arcs and intermediate nodes represent the railway lines. Trains provide the transportation service for each line and are the entities moving in the system; the respective departures and arrivals are constrained by safety rules as well as performance requirements The movement of a train between two consecutive stations, i.e., the train run, is defined by a specific running time, while its stay in a station, that can be both a stop or a shunting action, is respectively defined by a dwell time, or a shunting time. In the nominal functioning of the network, each arrival and departure of the trains circulating in the system follows an offline scheduled timetable, i.e., the nominal timetable. In case of unexpected events, nominal timetables become useless and have to be properly rescheduled ensuring passengers a safe and fast transportation service.

Considered the above, the railway management can be represented as a constrained optimization problem, and the railway network as an event-driven system, whose evolution is determined by the occurrence of train runs. More in detail, here two MILP models based on the work in [221] are considered to represent the rescheduling problem in case of full blockade between two consecutive stations. Both of the MILP models can be used to represent the nominal and the disrupted functioning of the network, but one provides a macroscopic description of the system while the other a mesoscopic one. The models include recovery actions to reschedule the disrupted traffic; the macroscopic model allows cancelation of train runs, short-turnings, and shunting actions; while the mesoscopic model allows in addition the representation of capacity limits for disrupted stations (i.e., number of available platforms and respective assignment of trains) and ordering of trains on platforms. As already reported, a full blockade consists in the complete block of the railway traffic between two consecutive stations in both directions. It causes the interruption of all the lines passing through the blocked tracks. Consequently, the trains of the interrupted lines can travel up to one of the two stations involved in the blockade and then have to be short-turned on their way back. More in detail, during the disruption, trains enter the two stations only in one direction, then are short-turned and used to perform outgoing train runs on the opposite direction that otherwise have to be canceled. Hence, short-turning is used to give continuity to the transportation service in the stations involved by the disruption. Differently, shunting actions consist in moving trains to/from shunting yards when they cannot be used for short-turning.

The objective of both the proposed optimization models is to minimize the number of cancelations of train runs and the offset from the nominal timetable in order to provide an effective service also during the disruption. The concept of train run is at the basis of the models. A ***train run*** is represented as a ***pair departure-arrival*** (d_i, a_i) with $i \in T$, where T is the set containing the indices of all train runs. The railway network can be divided into two main regions, one includes the area around the blocked tracks where traffic is directly affected by the blockade, and the other one includes the rest of the network. Then, set T is divided into two subsets: T_{ND} and T_D. Set T_{ND} consists of all indices of train runs that are not influenced by the disruption and can be performed as nominally scheduled or eventually might suffer from short secondary delays. Set T_D consists of all indices of train runs that are directly affected by the disruption and can be canceled, or can be used for short-turning or shunting actions. The tracks of the network are represented by tr_e, with $e \in E$, where E is the set containing the indices of all tracks. Each train run is associated to a track, the set of the indices of all train runs not affected by the disruption and associated to the same track is represented by $T_e \subset T_{ND}$, with $e \in E$, while the set of all train runs affected by the disruption and associated to the same track is represented by $\overline{T_e} \subset T_D$, with $e \in E$.

The dynamics of the system is described by constraints that synchronize the departures and arrivals. The constraints are divided in the following sets:

- Timetable constrains

$$\begin{aligned} d_i &\geq r_i^d \quad \forall i \in T \\ a_i &\geq r_i^a \quad \forall i \in T \end{aligned} \qquad \text{(Eq. 4-54)}$$

where r_i^d and r_i^a are the ***nominal departure and arrival times*** of the i-th train run, as scheduled offline, while d_i and a_i are the departure and the arrival times if the i-th train run, belonging to lines that can be rescheduled.

- Running time constraints

$$\begin{aligned} a_i &\geq d_i + \tau_i^{rt} && \forall i \in T_{ND} \\ a_i &\geq d_i + \tau_i^{rt} + \beta c_i && \forall i \in T_D,\ c_i \in \{0,1\},\ \beta \ll 0 \end{aligned} \qquad \text{(Eq. 4-55)}$$

where τ_i^{rt} is the ***minimal nominal train run duration*** of the i-th train run. The running time constraint has two different formulations depending on the considered train run. If the train run belongs to T_{ND} (first inequality) then the arrival time has to be larger than or equal to the sum of the corresponding departure time and minimal nominal running time. Otherwise, if the train run belongs to T_D, then a binary ***cancelation variable*** c_i will be associated to the train run along with a large negative constant β. If the variable c_i assumes value 1, there is no longer coupling between a_i and d_i, and the train run is canceled. Note that, the difference between the nominal departure and arrival times of a train run is often larger than the pure ***running time*** τ^{rt}

$$r_i^a - r_i^d \geq \tau_i^{rt} \qquad \text{(Eq. 4-56)}$$

Equation 4.56 includes in this manner a recovery time in the run, which allows a train to sustain small delays.

- Continuity constraints

$$
\begin{aligned}
&d_{i+1} \geq a_i + \tau^{dw}_{(i,i+1)} && \forall i, i+1 \in T_{ND} \\
&d_{i+1} \geq a_i + \tau^{dw}_{(i,i+1)} + \beta c_i && \forall i+1 \in T_{ND}, \forall\ i \in T_D, \\
& && c_i \in \{0,1\},\ \beta \ll 0 \\
&d_{i+1} \geq a_i + \tau^{dw}_{(i,i+1)} + \beta c_{i+1} && \forall i \in T_{ND}, \forall i+1 \in T_D, \\
& && c_{i+1} \in \{0,1\},\ \beta \ll 0
\end{aligned}
\quad \text{(Eq. 4-57)}
$$

where $\tau^{dw}_{(i,i+1)}$ is the ***dwell time*** of the train in the station connecting train run *i* to train run *i+1*. More in detail, the departure of the (*i+1*)-th train run has to be larger than or equal to the sum of the arrival of the *i*-th train run and the $\tau^{dw}_{(i,i+1)}$ dwell time in station, when both train runs belong to set T_{ND}. Otherwise, if one or both train runs belong to T_D a cancelation variable is included in the constraints along with a large negative constant β. When the cancelation variable has value 1, there is no longer coupling between a_i and d_{i+1}. Note that, the dwell times in the timetable are the absolute minimum wait times at stations and the difference

$$r^d_{i+1} - r^a_i \geq \tau^{dw}_{(i,i+1)} \quad \text{(Eq. 4-58)}$$

can be larger than the dwell time providing a buffer for short delays.

- Headway time constraints

$$
\begin{aligned}
&d_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)}) + d_l && \forall k, l \in T_e \\
&d_l \geq \tau^h_{(k,l)} + \beta u_{(k,l)} + d_k && \forall k, l \in T_e \\
&d_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_k) + d_l && \forall l \in T_e, \forall k \in \overline{T_e} \\
&d_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_l) + d_l && \forall l \in \overline{T_e}, \forall k \in T_e \\
&d_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_l + c_k) + d_l && \forall l, k \in \overline{T_e} \\
&d_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_k) + d_k && \forall l \in T_e, \forall k \in \overline{T_e} \\
&d_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_l) + d_k && \forall l \in \overline{T_e}, \forall k \in T_e \\
&d_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_l + c_k) + d_k && \forall k, l \in \overline{T_e}
\end{aligned}
\quad \text{(Eq. 4-59)}
$$

$$a_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)}) + a_l \qquad \forall k,l \in T_e$$
$$a_l \geq \tau^h_{(k,l)} + \beta u_{(k,l)} + a_k \qquad \forall k,l \in T_e$$
$$a_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_k) + a_l \qquad \forall l \in T_e, \forall k \in \overline{T_e}$$
$$a_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_l) + a_l \qquad \forall l \in \overline{T_e}, \forall k \in T_e$$
$$a_k \geq \tau^h_{(k,l)} + \beta(1 - u_{(k,l)} + c_l + c_k) + a_l \qquad \forall k,l \in \overline{T_e}$$
$$a_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_k) + a_k \qquad \forall l \in T_e, \forall k \in \overline{T_e}$$
$$a_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_l) + a_k \qquad \forall l \in \overline{T_e}, \forall k \in T_e$$
$$a_l \geq \tau^h_{(k,l)} + \beta(u_{(k,l)} + c_l + c_k) + a_k \qquad \forall k,l \in \overline{T_e}$$

where $\tau^h_{(k,l)}$ is the ***headway time*** between two departures (arrivals) of two generic train runs, respectively *k* and *l,* that run on the same track; while $u_{(k,l)}$ is the binary ***headway ordering variable***. As in the previous constraints, two different formulations are presented depending on the type of train run. If the train runs belong to $T_e \subset T_{ND}$ no cancellation variable is present in the formulation and two alternatives are considered: (1) if $u_{(k,l)}$ assumes value 1, then the departure (arrival) of train run *k* has to be higher than or equal to the sum of the departure (arrival) of train run *l* and the headway time $\tau^h_{(k,l)}$, otherwise (2) if $u_{(k,l)}$ assumes value 0 the departure of train run *l* has to be higher than or equal to the sum of the departure of train run *k* and the headway time $\tau^h_{(k,l)}$. In simple words, it holds that in case (1), the *k*-th train run departs (arrives) after than the *l*-th departure (arrival) with a corresponding time delay of at least the headway time $\tau^h_{(k,l)}$; while in case (2) it holds the reverse. If one or both of the train runs belong to $\overline{T_e} \subset T_D$, a cancelation variable c_i is included in the constraints along with a large negative constant β. When the cancelation variable has value 1, there is no longer coupling between d_k (a_k) and d_l (a_l).

- Short-turn constraints

Consider a station $s \in S_{ST}$, where S_{ST} is the set of stations where trains can be short-turned during the full blockade, and two train runs *i* and *j* with the respective proceeding train run *q*(*i*) and preceding train run *p*(*j*). If *q*(*i*) and *p*(*j*) are canceled ($c_{q(i)} = 1$ and $c_{p(j)} = 1$), the arrival a_i with $i \in I_s$ (where is I_s the set of incoming train runs in station *s* during the full blockade) can be combined in station *s* with the departure d_j with $j \in O_s$ (where O_s is the set of outgoing train runs from *s* during the full blockade) so that continuity is ensured to the transportation service (Figure 4-9 (b)).

The short-turn is then modeled with the following constraint:

$$d_j \geq a_i + \tau^{turn}_{(i,j)} + \beta(3 - c_{p(j)} - c_{q(i)} - b_{(i,j)})$$
$$\forall i \in I_s, \forall j \in O_s, \forall s \in S_{ST}, b_{(i,j)} \in \{0,1\}, \beta \ll 0 \qquad \text{(Eq. 4-60)}$$

$$\left.\begin{array}{ll} \sum_{i \in I_s} b_{(i,j)} + c_j + (1 - c_{p(j)}) = 1 & \forall j \in O_s \\ \sum_{j \in O_s} b_{(i,j)} + c_i + (1 - c_{q(i)}) = 1 & \forall i \in I_s \end{array}\right\} \forall s \in S_{ST} \qquad \text{(Eq. 4-61)}$$

where $\tau^{turn}_{(i,j)}$ is the ***short-turn time***, i.e., the time necessary for the short-turn, and $b_{(i,j)}$ is the binary ***short-turn variable***. The inequality constraint (Eq. 4-60) imposes that train run *j* and *i* can be connected only if both are not canceled and if the short-turn variable $b_{(i,j)}$ is equal to one, then the departure d_j can take place only after the arrival a_i is occurred and $\tau^{turn}_{(i,j)}$ is elapsed. Furthermore, the equality constraints (Eq. 4-61) impose that each arrival should be assigned to a unique departure and vice versa. It has to be noticed that when the station *s* is on the border of the disrupted region, then train run *i* and *j* belong to set T_{ND} and consequently $c_{p(j)}$ and $c_{q(i)}$ are equal to zero.

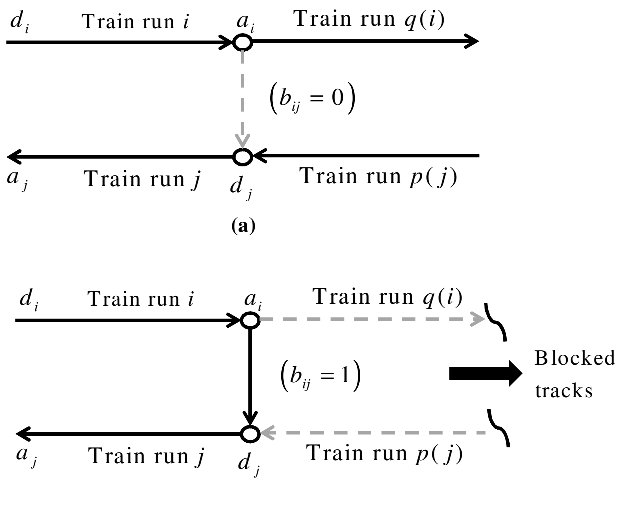


Figure 4-9. (a) Nominal traffic in station *s*. (b) Short-turn in station s.

- Shunting constraints

As already stated, if the considered station has a shunting area for the rolling stock, then shunting actions can be used when trains are not available for short-turning and there is an unbalance between arrivals and departures. Note that short-turns are always preferable to shunting actions, because they require fewer resources, in terms of staff, time, and money. Consider a station $s \in S_{ST,S}$ where $S_{ST,S}$ is the set of stations where trains can be short-turned or shunted from/to the shunting yard. The following ***shunting variable*** is introduced: $y^{in}_i \in \{0,1\}$, $i \in I_s$ which is used to assign to a planned departure the rolling stock in the shunting yard, and $y_j^{out} \in \{0,1\}$, $j \in O_s$ which is used for shunting the rolling stock of an arriving train, which will not continue its travel because of a cancelation. Then Eq. 4-62 holds:

$$\left.\begin{array}{l} y_i^{in} + (1 - c_{q(i)}) + c_i + \sum_{j \in O_S} b_{(i,j)} = 1 \quad \forall i \in I_s \\ y_j^{out}(1 - c_{p(j)}) + c_j + \sum_{i \in I_S} b_{(i,j)} = 1 \quad \forall j \in O_s \end{array}\right\} \forall s \in S_{ST,S} \qquad \text{(Eq. 4-62)}$$

- Capacity constraints

In the following constraints the capacity limit of the stations at each end of the disruption is taken into account. In particular, a ***short-turn on platform variable*** $b_{p,(i,j)} \in \{0,1\}$ is introduced, with $i \in I_s$, $j \in O_s$, and $p \in P_s$, where P_s is the set of platforms for the considered station $s \in S_{ST,S}$. Then the following capacity constraint holds:

$$b_{(i,j)} = \sum_{p \in P_S} b_{p,(i,j)}, \text{ with } s \in S_{ST,S} \qquad \text{(Eq. 4-63)}$$

which means that if an assignment (i, j) is set, it can be assigned to only one platform of the station *s*.

- Ordering constraints

If arrival a_i *with* $i \in I_s$ is connected to departure d_j with $j \in O_s$ and assigned to platform $p \in P_s$ ($b_{p,(i,j)} = 1$) and arrival a_k with $k \in I_s$ is connected to departure d_l with $l \in O_s$ and assigned also to platform $p \in P_s$ ($b_{p,(k,l)} = 1$), it is necessary to decide their order. Then, the ***ordering on platform variable*** $\omega_{(x,y)}$ is introduced, whose value is set to one when the arrival a_y with $y \in I_s$, has to be scheduled after the departure d_x *with* $x \in O_s$.

Hence, the ordering constraints are modeled as follows:

$$\begin{array}{l} a_i \geq d_l + \tau_{(i,l)}^{ord} + \beta(1 - \omega_{(l,i)}) \\ a_k \geq d_j + \tau_{(k,j)}^{ord} + \beta(1 - \omega_{(j,k)}) \end{array} \qquad \text{(Eq. 4-64)}$$

$$\begin{array}{c} 1 + \beta(2 - b_{p,(i,j)} - b_{p,(k,l)}) - \omega_{(l,i)} - \omega_{(j,k)} = 0 \\ \forall i,k \in I_s, \forall j,l \in O_s, p \in P_s, \forall s \in S_{ST,S} \\ \omega_{(l,i)}, \omega_{(j,k)}, b_{p,(i,j)}, b_{p,(k,l)} \in \{0,1\}, \beta \ll 0 \end{array} \qquad \text{(Eq. 4-65)}$$

where $\tau_{(i,l)}^{ord}$ is the ***minimum ordering time*** imposed between arrival a_i and departure d_l, the same holds for $\tau_{(k,j)}^{ord}$. The platform ordering constraint (Eq. 4-65) imposes that if both (i, j) and (k,l) short-turns hold and are assigned to the same platform p, then only one of the corresponding ordering variables $\omega_{(l,i)}$ and $\omega_{(j,k)}$ can assume the value 1.

Given the constraints of the railway system, both MILP rescheduling models can be written in the standard form as follows:

$$\begin{aligned} &\text{minimize} && f = \mathbf{g}^{\mathbf{T}}\mathbf{x} \\ &\text{subject to} && \mathbf{A}\mathbf{x} \leq \mathbf{b} \end{aligned} \qquad \text{(Eq. 4-66)}$$

with **g** a constant ***weight vector*** and **x** the ***decision variables vector***. The elements of the weight vector **g** can assume different values depending on the purpose of the optimization. Here, it is considered the minimization of the delays spreading over the network as well as the minimization of the cancelations and, if included, of the shunting actions. The decision variables vector x can be composed by all or a subset of the previously described decision variables (i.e., departure, arrival, cancelation, headway, short-turn, shunting, and ordering variables) depending on the level of detail of the model (i.e., macroscopic or mesoscopic model).

The **mesoscopic model** is written as follows:

$$\begin{aligned} &\text{minimize} && f_1 = \mathbf{g}_1^{\mathbf{T}}\mathbf{x}_1 \\ &\text{subject to} && \mathbf{A}_1\mathbf{x}_1 \leq \mathbf{z}_1 \end{aligned} \qquad \text{(Eq. 4-67)}$$

where $\mathbf{x}_1 = [\mathbf{d} \quad \mathbf{a} \quad \mathbf{c} \quad \mathbf{u} \quad \mathbf{b} \quad \mathbf{y} \quad \boldsymbol{\omega}]$ is the decision variables vector and includes the departure, arrival, cancellation, headway ordering, short-turn, short-turn on platform, and ordering on platform variables; and $\mathbf{g}_1 = [\mathbf{1} \quad \mathbf{1} \quad \boldsymbol{\lambda} \quad \mathbf{0} \quad \mathbf{0} \quad \boldsymbol{\gamma} \quad \mathbf{0}]$ is the constant vector with $\boldsymbol{\lambda} \gg \mathbf{0}$ in correspondence of the canceling variables, and $\boldsymbol{\gamma} \gg \mathbf{0}$ in correspondence of the shunting variables, so as to minimize cancellations and shunting actions. Note that each element of the vector $\mathbf{g}_1$ is still a vector and has the same dimension of the corresponding decision variable vector in $\mathbf{x}_1$.

The objective function can be rewritten as:

$$f_1 = \sum_{i \in T} d_i + \sum_{i \in T} a_i + \lambda \cdot \sum_{j \in T_D} c_j + \gamma \cdot \sum_{s \in S_{ST,S}} \left(\sum_{k \in I_s} y_k^{in} + \sum_{l \in O_s} y_l^{out} \right) \qquad \text{(Eq. 4-68)}$$

The constraints set includes equations from Eq. 4-54 to Eq. 4-65, i.e., the timetable, running time, continuity, headway time, short-turn, shunting, capacity, and ordering constraints. Consequently, the coefficient matrix **A** and the constant vector **z** are obtained by rewriting the recalled constraints in the standard form.

The **macroscopic model** is written as follows:

$$\begin{array}{ll} \text{minimize} & f_2 = \mathbf{g}_2{}^{\mathbf{T}}\mathbf{x}_2 \\ \text{subject to} & \mathbf{A}_2\mathbf{x}_2 \le \mathbf{z}_2 \end{array} \quad \text{(Eq. 4-69)}$$

where $\mathbf{x}_2 = [\mathbf{d} \quad \mathbf{a} \quad \mathbf{c} \quad \mathbf{u} \quad \mathbf{b_2}]$ is the decision variables vector and includes the departure, arrival, cancellation, headway ordering, short-turn, and ordering on platform variables; and $\mathbf{g}_1 = [\mathbf{1} \quad \mathbf{1} \quad \boldsymbol{\lambda} \quad \mathbf{0} \quad \mathbf{0}]$ is the constant vector with $\lambda \gg 0$ in correspondence of the canceling variables. Note that the decision variables vector $\mathbf{x}_2$ includes a subset of the decision variables included in vector $\mathbf{x}_1$.

The objective function can be rewritten as:

$$f_2 = \sum_{i \in T} d_i + \sum_{i \in T} a_i + \lambda \cdot \sum_{j \in T_D} c_j \quad \text{(Eq. 4-70)}$$

The constraints set includes equations from Eq. 4-54 to Eq. 4-62, i.e., the timetable, running time, continuity, headway time, short-turn, and shunting constraints. Consequently, the number of constraints is reduced with respect to the problem in Eq. 4-67. Also in this case the coefficient matrix **A** and the constant vector **z** are obtained by rewriting the recalled constraints in the standard form.

4.3.2 The Bi-level Solving Algorithm for the Real-Time Rescheduling in case of Disruption

The bi-level algorithm aims at finding a feasible timetable for the mesoscopic rescheduling problem in a short computation time, so that the resolution time can be suitable for a real-time control environment. The idea of finding an efficient solving algorithm, which can ensure a short computation time for the resolution of the recalled problem, derives from the result presented in [221]. In particular, in [221] the mesoscopic rescheduling problem is solved for a real case-study regarding a national railway network affected by a full blockade between two consecutive stations. In the presented work, the computation time required to find a feasible solution to the recalled problem largely overcome (i.e., more than 15 minutes) the time constraints required by a real-time control environment.

The algorithm here proposed consists in two consecutive steps (i.e., Step 1 and Step 2) in which the two MILP problems presented in the previous section are solved and the reschedule timetable for the mesoscopic problem is obtained. In particular, in Step 1 the macroscopic MILP problem in Eq. 4-69 is solved and the optimal decision variables vector $\overline{\mathbf{x}}_2$ is obtained. Then, in Step 2 the mesoscopic MILP problem of Eq 4-67 is considered and the results of Step 1 (i.e., $\overline{\mathbf{x}}_2$) are used to simplify its resolution, by reducing the respective search space. In particular, the mesoscopic model is modified by adding to the constraint set a number of $n+m$ equality constraints that (1) assign value 1 to a subset n of the cancellation variables vector $\mathbf{c}$ of vector $\mathbf{x}_1$ and (2) assign value 0 to a subset m of the short-turn variables vector $\mathbf{b}$ of vector $\mathbf{x}_1$. The n variables of vector $\mathbf{c}$ in $\mathbf{x}_1$ that are set to 1, correspond to the n variables of vector $\mathbf{c}$ in $\mathbf{x}_2$ that assume value 1 in the result vector $\overline{\mathbf{x}}_2$**,** while the m variables of vector $\mathbf{b}$ in $\mathbf{x}_1$ that are set to 0, correspond to the m variables of vector $\mathbf{b_2}$ in $\mathbf{x}_2$ that assume value 0 in the result vector $\overline{\mathbf{x}}_2$**.** In other words, the cancellations of train runs assigned in Step 1 (i.e., cancellation variables set to 1) are kept in the optimization problem of Step 2 and the short-turns of train runs that have been excluded in Step 1 (i.e., short-turn variables set to 0) are kept unfeasible in the optimization problem of Step 2.

4.3.3 A Real Case Study

In this section is presented the case study that is used for the application of the rescheduling algorithm in case of disruption. In particular, it regards a disruption in the Dutch railway network on the track section between the stations Dordrecht and Lage-Zwaluwe, which is part of one of the three routes from the north to the south Netherlands. The trains that travel between these stations pass over the Mordijk bridge that is often affected by disruptions due to recurrent adverse weather conditions that block the normal railway traffic. Figure 4-10 represents the main lines of the Dutch railway network considered in the case study and the disruption is signaled with a red cross over the train section between stations Dordrecht (Dor) and Lage-Zwaluwe (Lzw). At station Dordrech trains of interregional and regional lines dwell whereas in station Lage-Zwaluwe only regional trains dwell. Directly south to Lage-Zwaluwe there is a junction for trains travelling to and from stations Roosendaal and Breda. The disruption at the bridge is a full blockade, meaning that both tracks are unavailable and trains from the south cannot travel further the station Lage-Zwaluwe. Although normally only regional trains stop at Lage-Zwaluwe, in the case of a full-blockade it will also be the end point of interregional trains as passengers will travel by bus from this station to Dordrecht during the disruption. The trains will be turned at Lage-Zwaluwe and return to their starting destination. Trains starting from the north will have Dordrecht as endpoint and will be turned here as passengers will continue their trip to Lage-Zwaluwe by bus. The turning of trains at stations Dordrecht and Lage-Zwaluwe will lead to local deviations from the nominal timetable that can cause secondary delays for the rest of the network. The case study aims at calculating the rescheduled timetable for the disruption period that is feasible for the entire network and therefore for trains running in the network are taken into account. The timetable that is used is taken from part of the national timetable and consists of all train lines that run during the afternoon of a weekday. During rush hours in the morning and evening some lines run at higher frequencies and some extra lines are introduced to temporary increase the transport capacity of the network but these addictions are not taken into account here.

Figure 4-10. Main lines of the Dutch railway network. The disrupted train section between Dordrecht and Lage-Zwaluwe is signaled with a red cross. Rot=Rotterdam, Dor=Dordrecht, Lzw= Lage Zwaluwe, Rsd=Rosendaal, Bre=Breda, Ehv=Eindhoven

Table 4-14. Lines affected by the disruption.

Line	Origin	Destination	Times/hour	Blocked
IC1900	Den Haag	Venlo	2	y
IC2151	Amsterdam CS	Vlissingen	2	y
IC2249	Amsterdam CS	Dordrecht	2	n
SPR5000	Den Haag CS	Breda	2	y
SPR5100	Den Haag CS	Roosendaal	2	y
IC9240INT	Roosendaal	Amsterdam CS	1	y

The train lines that are directly affected by the blockade at the defined location as well as the lines that have stations Dordrecht and Lage-Zwaluwe as final destination are listed in Table 4-14. All lines in Table 4-14 run twice every hour in both directions except for the international line IC9240 that runs from and to Brussels but has Roosendaal as its origin/destination in the Netherlands. This international line is not considered in this case study as it would not be turned around in case of a disruption but rerouted instead to reach its destination.

Due to capacity limitations at the turning stations Dordrecht and Lage-Zwaluwe or because trains cannot be turned for a return trip, trains of affected lines might need to be cancelled before reaching their final destination. The current approach by manual dispatchers is often to partially cancel trains instead of cancelling them completely to minimize the inconvenience for passengers. In the case study the same approach will be used and in a certain area around the disruption trains of the affected lines will have the option of being cancelled. Further from the disruption, trains from these lines must keep running just as trains from all other lines. In Figure 4-10 the network is shown depicting in white the track sections where trains can be cancelled and in black the tracks on which cannot. Certain lines perform consecutive train runs on white track sections and if a train of these lines is cancelled then all these consecutive runs are cancelled together.

A. Modelling definitions and assumptions

The following assumptions and definitions are considered when modeling the system:

1. The end time of the disruption is known in advance. In reality the end time of the disruption may be unknown until the cause has been found. In that case an infinite disruption time can be assumed and a feasible timetable must be calculated without an end time. When the end time is known, the timetable must be recalculated once more to transition the traffic back towards the nominal timetable.

2. All train runs taking place before the disruption are assumed to be on time. In reality delays might be presented in the network at the start of the disruption.
3. The train runs that make up the lines from the timetable are all allocated to a certain track number. Although there are two or more tracks between all stations considered in this work is not possible for trains to change tracks. Train runs must always take place on the track they are scheduled to according to the timetable
4. Rolling stock cannot be exchanged between trains from different lines but can be split o combined for specific train runs according to the timetable.
5. Consecutive train runs of directly affected lines can be cancelled on the white tracks. Trains can therefore be cancelled at stations Rotterdam, Roosendaal, Breda, and Eindhoven. Train runs of these lines on other tracks cannot be cancelled and must always continue.
6. Trains running on the same track must hold a safe distance to each other which is enforced by headway constraints. The standard headway time between two trains is three minutes.
7. Trains that are turned at stations Dordrecht and Lage-Zwaluwe will have a minimum turning time of five minutes before their departure.
8. Trains arriving at stations are able to reach every platform from every track.
9. Rolling stock of all train lines can arrive at all platforms of the stations adjacent to the disruption.
10. Although rerouting trains might be an option in some situations, it is not considered in this work and therefore the international train IC9240 is not taken into account in the problem.
11. Train runs taking place at the blocked tracks at the start of the disruption continue their run as normal. Planned departures on these tracks after the start of the disruptions are cancelled.
12. The objective of the optimization is the minimization of the sum of delays and number of canelled trains in the network, and the minimization of shunting actions if included.

B. Results

The bi-level solving algorithm is then applied to the considered case study. The application of this algorithm, which considers both the macroscopic and mesoscopic models of the rescheduling problem, leads to a twofold advantage. On one hand it strongly reduces the required computation time with respect to the resolution of the full mesoscopically described problem. On the other hand, it allows the inclusion of the station scheduling model, which encounters the capacity limitations in stations Dordrecht and Lage-Zwaluwe, that otherwise are neglected in the macroscopic formulation of the problem. Obviously, the amount of available platforms in station poses a limitation to the number of short-turns that can be performed and increases the number of cancelled train runs. Nevertheless, the obtained results are more realistic, and the limitations of the macroscopic description of the problem are overcome.

The algorithm is performed for a scenario having a time window of 300 minutes and a rescheduling horizon of 200 minutes, with a disruption starting at minute t_{ds}=100 and ending 120 minutes later at minute t_{de}=220. The dimensions of the two MILP problems is reported in Table 4-15.

Table 4-15. MILP problems dimensions.

Macroscopic MILP model	
Nr. of constraints	29104
Nr. of variables	9719
Mesoscopic MILP model	
Nr. of constraints	53207
Nr. of variables	12237

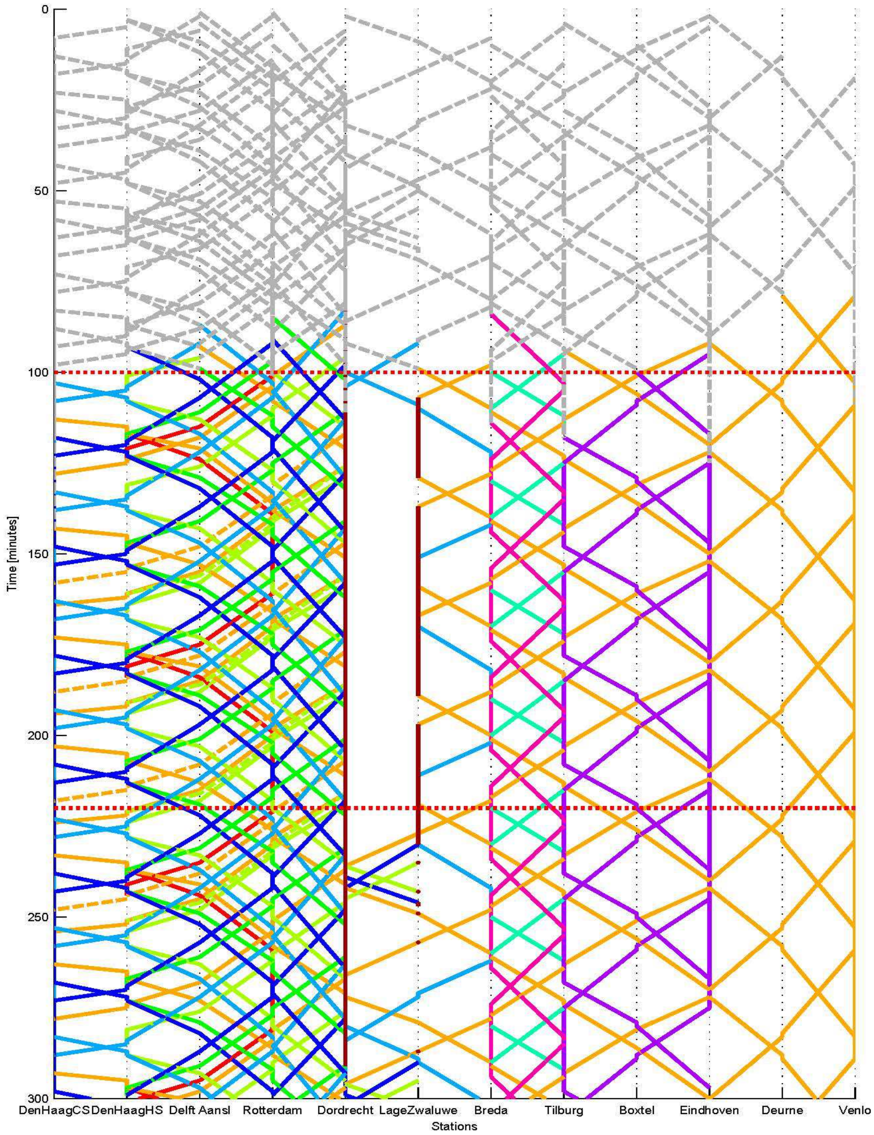


Figure 4-11. The rescheduled graphical timetable

Figure 4-11 shows a time-distance diagram of all tracks on the route from stations Den Haag CS to Venlo. All train lines that use tracks on this route are depicted in the figure, include all affected lines in the disruption region. Trains that overtake each other are allowed because their lines that are crossing are running on track sections that have four tracks between stations. Train runs that take place between minute 0 and 100 are shown in grey and lie in the past, they do not participate to the rescheduling process. Short-turnings and then assignments between arrival and departure events are shown in brown.

Figure 4-12 and Figure 4-13 depict the platform schedules corresponding to the solution in Figure 4-11 resulting from the bi-level resolution. Figure 4-12 shows the schedule for the six platforms of station Dordrecht and Figure 4-13 the schedule for the six platforms of station Lage-Zwaluwe. For both stations a feasible schedule is found with no overlapping between consecutive dwell periods. The minimum platform headway time of three minutes is respected between the train dwells at each platform. It should be noticed that the rescheduling process is not limited to the disruption but continues until the nominal timetable is again suitable. This is because a transition period after the disruption is considered during which train runs can still be cancelled and assigned.

The final solution is found with the Gurobi solver in 90 s on average, on an Intel Quadcore 2.4Ghz and 8 Gb Ram (still reducible avoiding output in the command window), which is strongly lower than the 18 minutes necessary for the resolution of the only mesoscopic MILP problem and acceptable for the real-time rescheduling purposes.

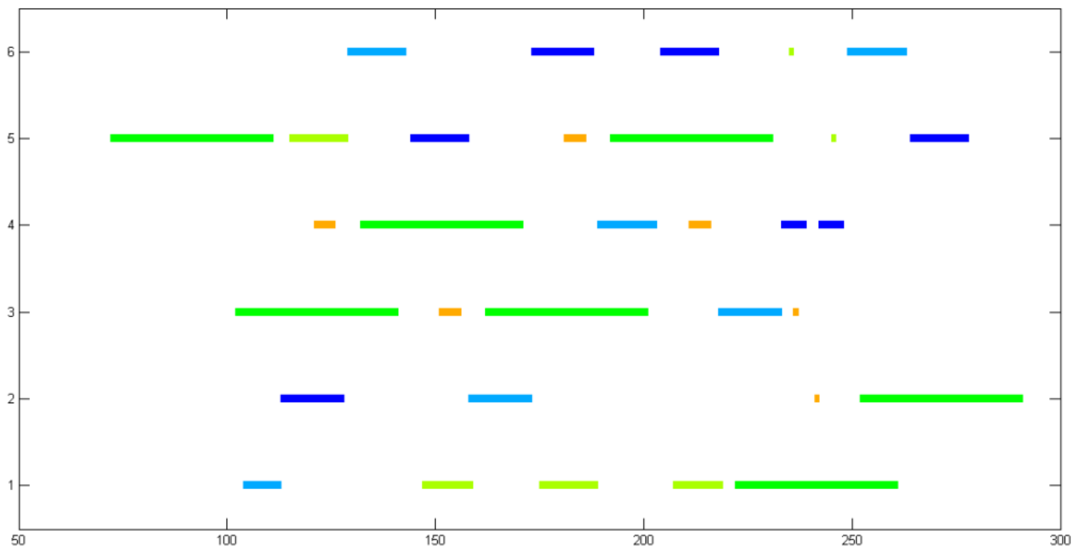


Figure 4-12. The platform assignment in station Dordrecht

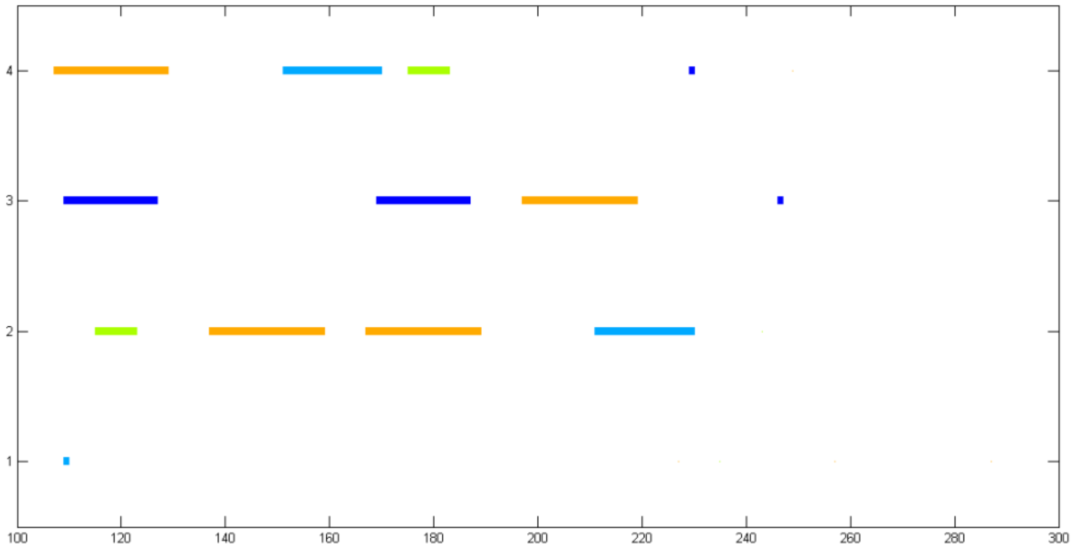


Figure 4-13. The platform assignment in station Lage Zwaluwe

Chapter 5

5 Conclusions and future works

This PhD thesis presents the results of my PhD research on Information and Communication Technologies (ICTs) in the context of Smart Cities, with particular attention to the study, design, and the development of advanced models and control techniques for intermodal freight transport terminals and railway transport.

In the first part of this work, the viability of discrete event methods for smart transportation systems is discussed. On one hand, it is provided a review of contributions on Petri nets for freight transportation systems. Papers are classified according to the addressed managerial problem, namely strategic/tactical or operational decision-making level problem, and the adopted PN formalism. It is also debated the approaches' feasibility, discussing contributions and limitations, and identifying future research directions to enhance the successful application of PNs in freight logistics and transportation systems under a smart transportation system perspective. The conducted investigation clearly shows that Petri nets are a valuable mathematical instrument for the resolution of the main managerial problems of freight transportation systems, however in some cases the literature contributions are limited (e.g. in the case of air transport). Moreover, it emerges a lack of contributions regarding the employment of fuzzy, continuous and hybrid (discrete and continuous) PNs, as well as a potential in the integration with other tools, e.g., genetic algorithms and multi-agent systems. On the other hand, an overview on the discrete event MILP models for railway systems is provided, highlighting their practical relevance in solving the railway traffic rescheduling problem, which is one of the most frequently encountered problems at the operational decision-making level.

In Chapter 3, it is demonstrated how the Petri net formalism can be used to model, simulate, analyze, and control intermodal freight transport terminal. First, a general, modular, and systematic modeling framework for IFTTs is proposed, to be used by decision makers in IFTT performance evaluation and optimization at operational level. Using a modular bottom-up approach, the subsystems constituting a generic intermodal terminal are identified. All subsystems are modeled by TPN modules and can be interconnected into a complete model by means of a systematic technique, allowing representing the whole IFTT and investigating the overall system dynamics. The model effectiveness is shown by means of two case studies - one from the literature and a real case study - evaluating the terminal efficiency in terms of performance indices and bottlenecks identification in a short computation time. Hence, simulating the proposed model in a computer-based environment, it turns out to be a decision support tool to assess the overall terminal management strategy, e.g., to assess the feasibility of alternative options when a new potential market is considered. In the subsequent section, a modeling and management framework for IFTTS based on first-order hybrid Petri nets is presented. This allows evaluating the terminal performance in closed-loop, while optimizing the system dynamics by simply solving linear programming problems. The approach may be used by the terminal decision maker offline, to take decisions about the terminal resources, or online, to solve congestions/malfunctions. The effectiveness of the method is validated on a real case study under different scenarios. Finally, it is shown how TPNs can be integrated with the DEA multi-criteria optimization technique to support decision makers in solving some of the most common IFTTs resource planning problems. In particular, such techniques are combined to dimension the number of resources used to transfer the intermodal transport units and to dimension the capacity/frequency of the transportation means employed for the intermodal delivery service. The timed Petri net model of a rail-road terminal is used to carry out Monte Carlo simulations considering various solving alternatives in case of critical scenarios and a cross-efficiency DEA technique is applied to rank the alternatives in terms of their impact on the terminal performance. Future developments can consider high-level Petri Nets to further refine the techniques and to allow solving more specific optimization problems. It will be also fundamental to investigate how uncertainty on some parameters can be properly taken into account, e.g., by applying the stochastic or fuzzy cross-efficiency Data Envelopment Analysis techniques.

Chapter 4 presents innovative techniques for the railway traffic rescheduling. First, a DSS for real-time management of a mixed-tracked railway network is presented. The DSS employs a rescheduling MILP model combined with a heuristic procedure, which extends the schedule after the optimization time horizon to guarantee the absence of conflicts in a short computation time. In the subsequent section a self-learning decision making procedure is proposed for real-time rescheduling of railway traffic under disturbances in mixed-tracked railway networks with acyclic. The technique combines a MILP model and a heuristic procedure to provide in real-time a feasible rescheduled timetable. Then, an offline self-learning procedure based on a cross-efficiency fuzzy DEA allows a performance evaluation of the obtained solution in order to properly update the database of timetables to be used in case of future disturbances of the same type. The technique is useful for both railway companies (to provide their customers with on-time services, reduce sanctions or penalties, and avoid possible errors caused by a manual rescheduling) and passengers (to reduce waiting times and delays or to limit travel discomforts). Future developments will consider the extension of the real-time rescheduling model to the case of multiple and overlapping disturbances occurrence. In the last section of Chapter 4, an innovative bi-level solving algorithm for real-time rescheduling of railway traffic in case of disruption is proposed. The rescheduling problem is modelled as a MILP model whose objective is to minimize delays, cancellation, and if allowed also shunting of trains in the network, respecting safety constraints in a short computation time. The innovative bi-level algorithm solves sequentially two optimization MILP problems. The first level optimization considers a macroscopic MILP model of the disrupted network in which it is ideally assumed that the stations involved by the disruption have infinite capacity and no platform constraint is necessary. The second level optimization considers a mesoscopic MILP model that includes the results of the first level optimization together with capacity constraints (i.e., including shunting-actions, short-turns with platform assignment, and ordering actions on platforms). The application of this algorithm, which considers both the macroscopic and mesoscopic models of the rescheduling problem, leads to a twofold advantage. On one hand it strongly reduces the required computation time with respect to the resolution of the full mesoscopically described problem. On the other hand, it allows the inclusion of the station scheduling model, which encounters the capacity limitations in disrupted stations, that otherwise are neglected in the macroscopic formulation of the problem. Obviously, the amount of available platforms in station poses a limitation to the number of short-turns that can be performed and increases the number of cancelled train runs. Nevertheless, the obtained results are more realistic, and the limitations of the macroscopic description of the problem are overcome. Further developments will consider the inclusion of the bi-level algorithm in a distributed optimization framework based on model predictive control, which shall solve the mesoscopic rescheduling problem in a distributed optimization perspective.

Appendix

The aim of this section is to offer, to the interested reader, clear guidelines to autonomously perform the previously presented simulations and eventually to extend and improve the proposed research work on intermodal terminals, using the Petri net formalism. In particular, some more details are here provided to model, analyze, control, and manage intermodal terminals by using the already recalled HYPENS tool [170] in the Matlab environment.

It has to be noticed that the adoption of the HYPENS tool for the simulation of the analyzed rail-road terminal is motivated by its various advantageous features. First and foremost, HYPENS is an open source tool that allows to simulate timed discrete, continuous, and hybrid Petri nets; second, it has been developed in Matlab, hence it allows designers and users to take advantage of several functions and structures already defined in Matlab, such as optimization routines, stochastic functions, matrices and arrays; third, the tool can also be easily interfaced with other Matlab programs and be used for analysis and optimization via simulation; last but not least, the large set of plot functions available in Matlab allow one to represent the results of the simulation in a clear and intuitive way.

The next subsections provide details on the procedures to be performed in Matlab and HYPENS in order to:

1. modularly model a generic intermodal terminal using the TPNs or the FOHPNs formalism;
2. evaluate the performance of the modeled intermodal terminal;
3. control the behavior of the terminal under study using FOHPNs.

In particular, the structure of the corresponding Matlab programs is provided together with a description of the newly defined and HYPENS functions. Subsection A focuses on the necessary procedure for modeling and performance evaluation of the GTS company rail-road terminal modelled via TPNs, whereas Subsection B focuses on modeling and control of the terminal via FOHPNs.

A. TPN modeling and performance evaluation of IFTTS using Matlab and HYPENS.

Section 3.2 presents a modular modeling technique based on the TPN formalism, which can be used to represent and analyze a generic IFTT. The approach considers IFTTs as discrete event systems composed by a set of interacting subsystems, each of which representable via TPNs. In particular, eleven basic subsystems are identified for the representation of a generic IFTT: (1) highway; (2) tollbooth; (3) railway; (4) maritime/river port or airport; (6) access road; (7) parking or yard storage area; (8) customs; (9) ITUs maintenance area, (10) opening/closing of an IFTT subsystem; (11)

checkpoint. For each of the above listed subsystems it is provided the corresponding TPN subnet. The TPN of the whole system is than obtainable as a bottom-up composition of the TPN models of the considered subsystems, connected via a standard interfacing net. The TPN model then can be used to perform simulation and analysis of the terminal under study.

The case studies presented in Section 3.2 are both modeled and analyzed following this procedure. In particular, the modular bottom-up approach of the technique is applied also for the implementation of the corresponding Matlab program. In effect, the aim of the developed research is to propose a general modeling technique and consequently a program for simulations, which can be reusable without loss of effectiveness. Considering the real case-study, i.e., the rail-road terminal of the Italian GTS company, five types of subsystems are selected for the representation of the whole terminal: 1. access road for semi-trailer trucks unloading ITUs, 2. access road for semi-trailer trucks loading ITUs, 3. yard storage area, 4. railway with separate ITUs load/unload; 5. opening/closing for days/hours. These basic subsystems, properly represented via TPNs and customized, are then connected to convert in a TPN model the logical scheme of the intermodal terminal reported in Figure Appendix 1. In the Matlab environment, the described modeling phase is implemented by defining specific functions for the definition of each of the required subnets in the TPN formalism. Then, the subnets obtained are connected into a unique network, which is used for the simulation and performance analysis of the behavior of the terminal using a Montecarlo approach.

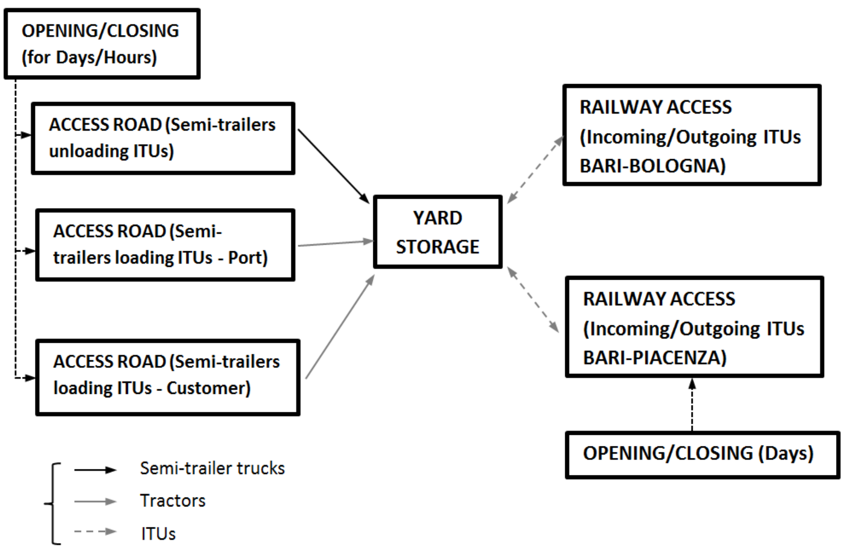


Figure Appendix 1. – Logical scheme of the GTS rail-road terminal.

Structure of the Matlab program

The structure of the program used for the rail-road IFTTs **modeling** and **simulation** in Section 3.2.2 is as follows:

Evolution.m is the main script-file and it runs in sequence the following functions:

1. OpenTerminal.m;
2. OpenDaysTerminal.m;
3. In_Semitrailers.m;
4. Out_Semitrailers.m;
5. Yard.m;
6. Line_Bologna.m;
7. Line_Piacenza.m;
8. OpenDaysPiacenza.m;
9. OpenDaysBologna.m;
10. Terminal.m;
11. Make_HPN.m
12. Simulator_HPN.m

Follows the explanation of the above listed functions.

Note that, since the program is based on the use of the HYPENS tool that considers hybrid PNs, all the newly defined functions usually include empty variables related to the continuous and hybrid parts of the network that do not exist for the considered TPN model.

OpenTerminal

Purpose

This function defines the opening/closing for hours subnet that enables the daily opening and closing of the IFTT.

Synopsis

```
Net_array_OpenTerminal=OpenTerminal();
```

Description

Parameters defined in function OpenTerminal are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_OpenTerminal= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}.
```

where

PreCC	Preincidence matrix for continuous places and transitions (empty).
PreCD	Preincidence matrix for continuous places and discrete transitions (empty).
PreDC	Preincidence matrix for discrete places and continuous transitions (empty).
Pre_DD	Preincidence matrix for discrete places and discrete transitions.
PostCC	Postincidence matrix for continuous places and transitions (empty).
PostCD	Postincidence matrix for continuous places and discrete transitions (empty).
PostDC	Postincidence matrix for discrete places and continuous transitions (empty).
Post_DD	Postincidence matrix for discrete places and discrete transitions.
M0C	Initial marking of continuous places.
M0D	Initial marking of discrete places.
vel	Matrix of speed vectors associated to continuous transitions (empty).
v	Vector of discrete transitions type (1=deterministic, 2=stochastic exponential).
D	Matrix of the time parameter associated to transitions.
s	Vector of the number of servers associated to discrete transitions.
alpha	Vector of priorities associated to discrete transitions.

Note that, in matrix **D** the time parameter associated to deterministic transitions is a constant value while the parameter characterizing the stochastic exponential transitions is the average value associated to the exponential distribution characterizing the transition.

OpenDaysTerminal, OpenDaysBologna, OpenDaysPiacenza

Purpose

The function OpenDaysTerminal defines the opening/closing for days subnet that enables the weekly opening and closing of the IFTT. Note that, in the same function, properly customized, is used for the opening/closing for days of the railway access from/to Bologna/Piacenza to the IFTT (i.e., by properly setting the parameters of the subnet).

Synopsis

```
Net_array_OpenDaysTerminal=OpenDaysTerminal();
```

Description

Parameters defined in function OpenDaysTerminal are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_OpenDaysTerminal= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

In_Semitrailers

Purpose

This function defines the subnet representing the flow of outgoing semitrailers in the IFTT. Note that, in the program this function is used as many times as is the number of subnets representing outgoing semitrailers.

Synopsis

```
Net_array_In_Semitrailers=In_Semitrailers();
```

Description

Parameters defined in function In_Semitrailers are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_In_Semitrailers= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

Out_Semitrailers

Purpose

This function defines the subnet representing the flow of incoming semitrailers in the IFTT. Note that, in the program this function is used as many times as is the number of subnets representing incoming semitrailers.

Synopsis

```
Net_array_In_Semitrailers=In_Semitrailers();
```

Description

Parameters defined in function In_Semitrailers are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_In_Semitrailers= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

Yard

Purpose

This function defines the subnet representing the yard storage area of the IFTT.

Synopsis

```
Net_array_Yard=Yard();
```

Description

Parameters defined in function Yard are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_Yard= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

Line_Bologna and **Line_Piacenza**

Purpose

The function Line_Bologna defines the subnet representing the flow of incoming/outgoing ITUs by railway in the IFTT from/to Bologna. Note that, in the program the same function properly customized is used to define the subnet representing incoming/outgoing ITUs by railway in the IFTT from/to Piacenza (i.e., by properly setting the parameters of the subnet).

Synopsis

```
Net_array_In_Semitrailers=In_Semitrailers();
```

Description

Parameters defined in function Line_Bologna are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_Line_Bologna = {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

Terminal

Purpose

This function connects by the routing nets all the subnets of the considered IFTT.

Synopsis

```
Net_array_Terminal=Terminal(Net_array_OpenDaysTerminal,
Net_array_OpenDaysTerminal, Net_array_InSemitrailers, Net_array_OutSemitrailers,
Net_array_Yard, Net_array_LineBologna, Net_array_LinePiacenza,
Net_array_OpenDaysBologna, Net_array_OpenDaysPiacenza);
```

Description

Parameters defined in function Terminal are loaded into the workspace. The function has no inputs but only outputs. More in detail:

```
Net_array_Terminal= {PreCC,PreCD,PreDC,Pre_DD,PostCC,PostCD,
PostDC,Post_DD,M0C,M0D,vel,v,D,s,alpha}
```

where it still holds the meaning of the previously described variables.

Make_HPN

Purpose

This function of the tool HYPENS creates the TPN of the whole network.

Synopsis

```
[Pre,Post,M,vel,v,D,s,alpha]=make_HPN(net,1);
```

Description

Parameters defined in function make_HPN are loaded into the workspace. The function has in input the net vector, i.e., the net array of the whole network, and produces as output the matrices and vectors of the network where the continuous and discrete parts have been assembled. For more details on the function refer to the HYPENS manual [170].

Simulator_HPN

Purpose

This function of the tool HYPENS can simuate the dynamics of the TPN over a defined time interval.

Synopsis

```
evol=simulator_HPN(Pre,Post,M,vel,v,D,s,alpha,time);
```

Description

Parameters defined in function simulator_HPN are loaded into the workspace. The function has in input the matrices and vectors of the network, and the simulation interval, then, it produces in output the cell array `evol` containing information over the system evolution. For more details on the function refer to the HYPENS manual [170].

The rail-road IFTTs **performance evaluation** in Section 3.2.2 is implemented by the function

MonteCarlo

Purpose

This function iteratively performs the simulation of terminal behavior over a fixed number of replications.

Synopsis

```
Av_param=MonteCarlo(replications);
```

Description

The function has in input the number of replications to be performed using the evolution.m function. The results of each replication are then used to calculate the average value of the parameters necessary for the performance evaluation of the terminal.

B. FOHPN modeling, performance evaluation, and control of IFTTS using Matlab and HYPENS.

The structure of the Matlab program used for modeling, performance evaluation, and control of IFTTS via FOHPNs is similar to the one previously described in Subsection A. Also in this case a modular approach is considered for the definition of the whole network; each subnet is obtained by proper functions and then connected to the others in a whole so as to simulate the dynamic behavior of the system under analysis. The control of the system is obtained by properly defining the weights of the objective functions to be optimized at each macroevent of the network. This is obtained by using an overloaded version of the simulator_HPN function.

Simulator_HPN

Synopsis

```
evol=simulator_HPN(Pre,Post,M,vel,v,D,s,alpha,time,J)
```

Description

Parameters defined in function simulator_HPN are loaded into the workspace. The function has in input the matrices and vectors of the network, the simulation interval, and the weight vector of the optimization function J. Then, it produces in output the cell array `evol` containing information over the system evolution. For more details on the function refer to the HYPENS manual [170].

.

Bibliography

[1] A. García-Ortiz, S. M. Amin, and J. R. Wootton, "Intelligent transportation systems-Enabling technologies," *Math. Comput. Model.*, vol. 22, no. 4–7, pp. 11–81, Aug. 1995.

[2] RITA, "National ITS Architecture - Theory of Operations, Research and Innovation Technology Administration,." US Department of Transportation, Washington, D.C, 2007.

[3] NJDOT, "Smart Transportation Guidebook." New Jersey Department of Transportation and Pennsylvania Department of Transportation, 2008.

[4] IBM, "Intelligent Transport: How cities can improve mobility." IBM Global Service, 2009.

[5] A. Debnath, M. Haque, H. Chin, and B. Yuen, "Sustainable Urban Transport," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2243, no. 2243, pp. 38–45, Dec. 2011.

[6] T. Goldman and R. Gorham, "Sustainable urban transport: Four innovative directions," *Technol. Soc.*, vol. 28, no. 1–2, pp. 261–273, Jan. 2006.

[7] G. Santos, H. Behrendt, and A. Teytelboym, "Part II: Policy instruments for sustainable road transport," *Research in Transportation Economics*, vol. 28, no. 1. Elsevier, pp. 46–91, Jan-2010.

[8] M. M. Haque, H. C. Chin, and A. K. Debnath, "Sustainable, safe, smart-three key elements of Singapore's evolving transport policies," *Transp. Policy*, vol. 27, pp. 20–31, May 2013.

[9] G. Akhras, "Smart Materials and Smart Systems for the Future," *Can. Mil. J.*, pp. 25–32, 2000.

[10] G. Cavone, M. Dotoli, and C. Seatzu, "A Survey on Petri Net Models for Freight Logistics and Transportation Systems," *IEEE Trans. Intell. Transp. Syst.*, 2017.

[11] C. Degano and A. Di Febbraro, "On using Petri nets to detect and recover from faulty behaviours in transportation facilities," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2002, vol. 1, pp. 31–36.

[12] D. Rama and J. D. Andrews, *IEE proceedings. Intelligent transport systems.*, vol. 10, no. 1. Institution of Electrical Engineers, 2016.

[13] M. Dotoli, N. Epicoco, M. Falagario, and G. Cavone, "A Timed Petri Nets Model for Intermodal Freight Transport Terminals," *IFAC Proc. Vol.*, vol. 47, no. 2, pp. 176–181, 2014.

[14] M. Dotoli and M. P. Fanti, "An urban traffic network model via coloured timed Petri nets," *Control Eng. Pract.*, vol. 14, no. 10, pp. 1213–1229, Oct. 2006.

[15] A. DiFebbraro, D. Giglio, and N. Sacco, "Urban Traffic Control Structure Based on Hybrid Petri Nets," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 224–237, Dec. 2004.

[16] H. Alla and R. David, *Continuous and Hybrid Petri Nets*, vol. 8. Springer, 1998.

[17] F. Balduzzi, A. Giua, and G. Menga, "First-order hybrid Petri nets: a model for optimization and control," *IEEE Trans. Robot. Autom.*, vol. 16, no. 4, pp. 382–399, 2000.

[18] F. Balduzzi, A. Giua, and G. Menga, "First-order hybrid Petri nets: a model for optimization and control," *IEEE Trans. Robot. Autom.*, vol. 16, no. 4, pp. 382–399, 2000.

[19] A. Gudelj, M. Krčum, and E. Twrdy, "Models and Methods for Operations in Port Container Terminals," *PROMET - Traffic&Transportation*, vol. 22, no. 1, pp. 43–51, Jan. 2010.

[20] K. H. Kim and H. B. Kim, "The optimal sizing of the storage space and handling facilities for import containers," *Transp. Res. Part B Methodol.*, vol. 36, no. 9, pp. 821–835, Nov. 2002.

[21] A. Imai, E. Nishimura, M. Hattori, and S. Papadimitriou, “Berth allocation at indented berths for mega-containerships,” *Eur. J. Oper. Res.*, vol. 179, no. 2, pp. 579–593, Jun. 2007.

[22] R. Garrido and F. Allendes, “Modeling the Internal Transport System in a Containerport,” *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1782, pp. 84–91, Jan. 2002.

[23] J. Liu, Y. Du, P. Li, and Y. Hong, “Petri-net Modeling of Container-port Work Flo,” *Inf. Technol. J.*, vol. 12, no. 9, pp. 1845–1850, Sep. 2013.

[24] A. Di Febbraro and N. Sacco, “Sensitivity analysis of the performances of seaport container terminals,” in *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, 2013, pp. 691–696.

[25] L. Yaxiong, P. Lefei, and L. Xinxue, “Modeling and simulation of full logistics process of the port based on Petri nets,” in *2010 2nd IEEE International Conference on Information Management and Engineering*, 2010, pp. 24–27.

[26] Wenyuan Wang, Xiangqun Song, Chen Tang, and Yu Liu, “System simulation of capacity for container terminal based on Stochastic Petri Net,” in *2008 IEEE International Conference on Automation and Logistics*, 2008, pp. 2889–2892.

[27] H. Zhang and Z. Jiang, “Modeling and Analysis of Container Terminal Logistics System by Extended Generalized Stochastic Petri Nets,” in *2006 IEEE International Conference on Service Operations and Logistics, and Informatics*, 2006, pp. 310–315.

[28] P. Legato, R. Trunfio, and F. Meisel, “Modeling and solving rich quay crane scheduling problems,” *Comput. Oper. Res.*, vol. 39, no. 9, pp. 2063–2078, Sep. 2012.

[29] R. Trunfio, “A timed petri net model for the quay crane scheduling problem,” 2014, pp. 441–447.

[30] M. Kefi, O. Korbaa, K. Ghedira, and P. Yim, “Formalising an agent-based container stacking model via petri nets,” 2006, vol. 12, no. PART 1.

[31] D. Kezic, P. Matic, and N. Racic, “P - Invariant based Petri net traffic controller,” in *2009 17th Mediterranean Conference on Control and Automation*, 2009, pp. 1096–1101.

[32] J. Zou, Y. Wei, and W. Wu, “Petri nets model of emergency control system for shipping,” in *Proceedings of 2012 9th IEEE International Conference on Networking, Sensing and Control*, 2012, pp. 295–300.

[33] Shiting Li, Qi Cai, and Bo Zhu, “Research on maintenance processes modeling techniques for ships based on Petri nets,” in *2009 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS)*, 2009, pp. 197–200.

[34] J. Bjørk and A. M. Hagalisletto, “Challenges in simulating railway systems using Petri Nets,” 2005.

[35] A. M. Hagalisletto, J. Bjork, I. C. Yu, and P. Enger, “Constructing and Refining Large-Scale Railway Models Represented by Petri Nets,” *IEEE Trans. Syst. Man Cybern. Part C (Applications Rev.*, vol. 37, no. 4, pp. 444–460, Jul. 2007.

[36] D. Wu and E. Schnieder, “Scenario-Based Modeling of the On-Board of a Satellite-Based Train Control System With Colored Petri Nets,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 11, pp. 3045–3061, Nov. 2016.

[37] M. S. Durmus and M. T. Söylemez, “Railway signalization and interlocking design via automation Petri nets,” *Asian Control Conf. ASCC 2009, 7th. IEEE*, pp. 1558–1563, 2009.

[38] M. S. Durmuş and M. T. Söylemez, “Automation Petri Net Based Railway Interlocking and Signalization Design,” in *Int. Symposium on Innovations in Intelligent Systems and Applications .*, 2009.

[39] U. Yildirim, M. Seçkin DURMUŞ, and M. Turan Söylemez, “Fail-Safe Signalization and Interlocking Design for a Railway Yard: An Automation Petri Net Approach,” in *International Symposium on Intelligent and*

Manufactu ring Systems (IMS 2010), 2010.

[40] M. T. Söylemez, M. S. Durmuş, U. Yildirim, S. Türk, and A. Sonat, "The application of automation theory to railway signalization systems: The case of Turkish National Railway Signalization Project," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2011, vol. 18, no. PART 1, pp. 10752–10757.

[41] M. T. Söylemez, M. S. Durmuş, U. Yıldırım, S. Türk, and A. Sonat, "The Application of Automation Theory to Railway Signalization Systems: The Case of Turkish National Railway Signalization Project," in *IFAC Proceedings Volumes*, 2011, vol. 44, no. 1, pp. 10752–10757.

[42] S. Vanit-Anunchai, "Verification of railway interlocking tables using coloured Petri nets.," in *Workshop on Practical Use of Coloured Petri Nets and the CPN Tools.*, 2014.

[43] A. Giua and C. Seatzu, "Modeling and Supervisory Control of Railway Networks Using Petri Nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 3, pp. 431–445, Jul. 2008.

[44] Y.-S. Weng, Y.-S. Huang, Y.-L. Pan, and M. Jeng, "Design of traffic safety control systems for railroads and roadways using timed Petri nets," *Asian J. Control*, vol. 17, no. 2, pp. 626–635, Mar. 2015.

[45] M. P. Fanti, A. Giua, and C. Seatzu, "Monitor design for colored Petri nets: An application to deadlock prevention in railway networks," *Control Eng. Pract.*, vol. 14, no. 10, pp. 1231–1247, Oct. 2006.

[46] S. Vanit-Anunchai, "Modelling Railway Interlocking Tables Using Coloured Petri Nets," in *International Conference on Coordination Languages and Models*, 2010, pp. 137–151.

[47] W. Guo, H. Zhao, and G. Zhou, "Verification and analysis of RSSP-1 protocol based on colored petri nets," in *Proceedings of the 2012 2nd International Conference on Business Computing and Global Informatization, BCGIN 2012*, 2012, pp. 592–595.

[48] W. JIA, B. MAO, T. HO, H. LIU, and B. YANG, "Bottlenecks Detection of Track Allocation Schemes at Rail Stations by Petri Nets," *J. Transp. Syst. Eng. Inf. Technol.*, vol. 9, no. 6, pp. 136–141, Dec. 2009.

[49] Y. H. Cheng and L. A. Yang, "A Fuzzy Petri Nets approach for railway traffic control in case of abnormality: Evidence from Taiwan railway system," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 8040–8048, May 2009.

[50] K. M. Ng, M. B. I. Reaz, and M. A. M. Ali, "A review on the applications of petri nets in modeling, analysis, and control of urban traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 858–870, Jun. 2013.

[51] Y. Qu, L. Li, Y. Liu, Y. Chen, and Y. Dai, "Travel routes estimation in transportation systems modeled by Petri Nets," in *Proceedings of 2010 IEEE International Conference on Vehicular Electronics and Safety*, 2010, pp. 73–77.

[52] T. Aized and J. S. Srai, "Hierarchical modelling of Last Mile logistic distribution system," *Int. J. Adv. Manuf. Technol.*, vol. 70, no. 5–8, pp. 1053–1061, Feb. 2014.

[53] H. Franke and W. Dangelmaier, "Decentralized management for transportation-logistics: A multi agent based approach," *Integr. Comput. Aided. Eng.*, vol. 10, no. 2, pp. 203–210, Jan. 2003.

[54] Y. Yan, H. Liu, and Y. Zhang, "Evaluation emergency transport performances in response for hazardous materials road transportation accident based on SPN: A case of jiangsu," in *Proceedings - 2010 International Conference on Optoelectronics and Image Processing, ICOIP 2010*, 2010, vol. 1, pp. 509–513.

[55] G. Centrone, W. Ukovich, M. P. Fanti, and G. Iacobellis, "A colored Petri net model of motorways for risk evaluation of HAZMAT transportation," in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2011, pp. 562–567.

[56] M. P. Fanti, G. Iacobellis, and W. Ukovich, "A risk assessment framework for hazmat transportation in highways by colored petri nets," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 45, no. 3, pp. 485–495, Mar. 2015.

[57] I. Kabashkin, “Heuristic Based Decision Support System for Choice of Alternative Routes in the Large-Scale Transportation Transit System on the Base of Petri Net Model,” in *Procedia Engineering*, 2016, vol. 134, pp. 359–364.

[58] J. J. Julvez and R. K. Boel, “A continuous petri net approach for model predictive control of traffic systems,” in *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 2010, vol. 40, no. 4, pp. 686–697.

[59] M. Dotoli, M. P. Fanti, and G. Iacobellis, “A freeway traffic control model by first order hybrid Petri nets,” in *IEEE International Conference on Automation Science and Engineering*, 2011, pp. 425–431.

[60] M. Dotoli, “Freeway traffic control via route guidance: An approach based on a First Order Hybrid Petri Nets model,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2012, vol. 45, no. 9, pp. 350–355.

[61] M. P. Fanti, G. Iacobellis, A. M. Mangini, and W. Ukovich, “Freeway traffic modeling and control in a first-order hybrid petri net framework,” *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 1, pp. 90–102, Jan. 2014.

[62] I. Demongodin, *Modeling and Analysis of Transportation Networks Using Batches Petri Nets with Controllable Batch Speed.* 2009.

[63] J. Ding, T. Chen, and T. Xu, “Propagated analysis of airport delays based on timed Petri nets,” in *CIS 2009 - 2009 International Conference on Computational Intelligence and Security*, 2009, vol. 1, pp. 608–614.

[64] B. Yang and L. I. Wei-Hong, “Capability evaluation of air cargo export handling system using Stochastic Petri Net,” in *2010 International Conference on Logistics Systems and Intelligent Management (ICLSIM)*, 2010, pp. 1589–1593.

[65] R. Davidrajuh and B. Lin, “Exploring airport traffic capability using Petri net based model,” *Expert Syst. Appl.*, vol. 38, no. 9, pp. 10923–10931, Sep. 2011.

[66] C. Lee, H. C. Huang, B. Liu, and Z. Xu, “Development of timed Colour Petri net simulation models for air cargo terminal operations,” *Comput. Ind. Eng.*, vol. 51, no. 1, pp. 102–110, Sep. 2006.

[67] H. Oberheid and D. Söffker, “Cooperative Arrival Management in Air Traffic Control - A Coloured Petri Net Model of Sequence Planning,” in *Applications and Theory of Petri Nets*, vol. 5062, 2008, pp. 348–367.

[68] B. Werther, C. Möhlenbrink, and M. Rudolph, “Colored Petri Net Based Formal Airport Control Model for Simulation and Analysis of Airport Control Processes,” in *Digital Human Modeling*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 1027–1036.

[69] P. Podgórski and J. Skorupski, “Aircraft taxi route choice in case of conflict points existence,” in *Communications in Computer and Information Science*, 2016, vol. 640, pp. 366–377.

[70] M. Jamal and N. A. Zafar, “Formalizing air traffic control system using agent-based Mobile Petri Nets,” in *2015 International Conference on Information and Communication Technologies (ICICT)*, 2015, pp. 1–6.

[71] H. Chou and C. Chang, “Petri-Net-Based Strategy To Synthesize the Operating Procedures,” pp. 114–123, 2005.

[72] J. W. Lai, H. H. Chou, and C. T. Chang, “Petri-net based integer programs for synthesizing optimal material-transfer procedures in pipeline networks,” *J. Chinese Inst. Eng. Trans. Chinese Inst. Eng. A/Chung-kuo K. Ch’eng Hsuch K’an*, vol. 29, no. 2, pp. 337–346, Mar. 2006.

[73] X. Yi, W. Si, and X. Wu, “Analysis of emergency response for accident of oil and gas pipeline based on Stochastic Petri Net,” 2016.

[74] O. Gursesli and A. A. Desrochers, “Modeling infrastructure interdependencies using Petri nets,” in *SMC’03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference*

Theme - System Security and Assurance (Cat. No.03CH37483), 2003, vol. 2, pp. 1506–1512.

[75] H. Ren, Z. Wang, and W. Shi, “Performance modeling of asynchronous pipelines - An overview,” in *Procedia Engineering*, 2012, vol. 29, pp. 3788–3793.

[76] Y. Guo, X. Meng, D. Wang, T. Meng, S. Liu, and R. He, “Comprehensive risk evaluation of long-distance oil and gas transportation pipelines using a fuzzy Petri net model,” *J. Nat. Gas Sci. Eng.*, vol. 33, pp. 18–29, Jul. 2016.

[77] N. Wu, M. C. Zhou, and F. Chu, “A petri net-based heuristic algorithm for realizability of target refining schedule for oil refinery,” *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 4, pp. 661–676, Oct. 2008.

[78] N. Q. Wu, F. Chu, C. Chu, and M. C. Zhou, “Short-term schedulability analysis of multiple distiller crude oil operations in refinery with oil residency time constraint,” *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 39, no. 1, pp. 1–16, Jan. 2009.

[79] N. Wu, F. Chu, C. Chu, and M. Zhou, “Hybrid petri net modeling and schedulability analysis of high fusion point oil transportation under tank grouping strategy for crude oil operations in refinery,” *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 40, no. 2, pp. 159–175, Mar. 2010.

[80] N. Wu, C. Chu, F. Chu, and M. Zhou, “Schedulability analysis of short-term scheduling for crude oil operations in refinery with oil residency time and charging-tank-switch-overlap constraints,” *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 190–204, Jan. 2011.

[81] N. Q. Wu, M. C. Zhu, L. P. Bai, and Z. W. Li, “Short-term scheduling of crude oil operations in refinery with high-fusion-point oil and two transportation pipelines,” *Enterp. Inf. Syst.*, vol. 10, no. 6, pp. 581–610, Jul. 2016.

[82] N. Q. Wu, M. C. Zhou, and Z. W. Li, “Short-Term Scheduling of Crude-Oil Operations: Enhancement of Crude-Oil Operations Scheduling Using a Petri Net-Based Control-Theoretic Approach,” *IEEE Robot. Autom. Mag.*, vol. 22, no. 2, pp. 64–76, Jun. 2015.

[83] H. Kadri and B. Zouari, “Shortest path search in dynamic reliability space: Hierarchical coloured petri nets model and application to a pipeline network,” in *Proceedings - UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, UKSim 2014*, 2014, pp. 254–259.

[84] H. Kadri and B. Zouari, “A High-level Petri Nets Approach for Multi-Objective Optimization in Pipeline Networks,” in *Proceedings of the 4th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, 2014, pp. 211–218.

[85] M. Dotoli, M. P. Fanti, A. M. Mangini, G. Stecco, and W. Ukovich, “The impact of ICT on intermodal transportation systems: A modelling approach by Petri nets,” *Control Eng. Pract.*, vol. 18, no. 8, pp. 893–903, Aug. 2010.

[86] M. Dotoli, H. Zgaya, C. Russo, and S. Hammadi, “A Multi-Agent Advanced Traveler Information System for Optimal Trip Planning in a Co-Modal Framework,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2397–2412, Sep-2017.

[87] F. Eng-Larsson and C. Kohn, “Modal shift for greener logistics – the shipper’s perspective,” *Int. J. Phys. Distrib. Logist. Manag.*, vol. 42, no. 1, pp. 36–59, Jan. 2012.

[88] K. Filipova, T. Stojadinova, and V. Hadjiatanasova, “Application of Petri nets for transport streams modeling,” *Facta Univ. - Ser. Archit. Civ. Eng.*, vol. 2, no. 4, pp. 295–306, 2002.

[89] A. Di Febbraro, G. Porta, and N. Sacco, “A Petri net modelling approach of intermodal terminals based on Metrocargo© system,” in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2006, pp. 1442–1447.

[90] P. Kemper and M. Fischer, "Modeling and Analysis of a Freight Terminal with Stochastic Petri Nets," *IFAC Proc. Vol.*, vol. 33, no. 9, pp. 267–272, Jun. 2000.

[91] G. Maione and M. Ottomanelli, "A preliminary Petri net model of the transshipment processes in the Taranto container terminal," *Emerg. Technol. Fact. Autom. 2005. ETFA 2005. 10th IEEE Conf.*, vol. 1, p. 7 pp.-pp.171, 2005.

[92] G. Maione, A. M. Mangini, and M. Ottomanelli, "A Generalized Stochastic Petri Net Approach for Modeling Activities of Human Operators in Intermodal Container Terminals," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 4, pp. 1504–1516, Oct. 2016.

[93] G. Cavone, M. Dotoli, and C. Seatzu, "Resource planning of intermodal terminals using timed Petri nets," in *2016 13th International Workshop on Discrete Event Systems, WODES 2016*, 2016, pp. 44–50.

[94] I. Kabashkin, "Modelling of Regional Transit Multimodal Transport Accessibility with Petri Net Simulation," in *Procedia Computer Science*, 2015, vol. 77, pp. 151–157.

[95] C. Degano and A. Di Febbraro, "Modelling automated material handling in intermodal terminals," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, 2001, vol. 2, pp. 1023–1028.

[96] C. Degano, A. Di Febbraro, and P. Fornara, "Fault diagnosis in an intermodal container terminal," *IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA*, vol. 2, pp. 433–440, 2001.

[97] C. Degano and A. Pellegrino, "Multi-agent coordination and collaboration for control and optimization strategies in an intermodal container terminal," *Engineering Management Conference, 2002. IEMC '02. 2002 IEEE International*, vol. 2. IEEE, pp. 590–595 vol.2, 2002.

[98] M. Dotoli, N. Epicoco, M. Falagario, and G. Cavone, "A Timed Petri Nets Model for Performance Evaluation of Intermodal Freight Transport Terminals," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 842–857, Apr. 2016.

[99] C. Silva, C. Guedes Soares, and J. Signoret, "Intermodal terminal cargo handling simulation using Petri nets with predicates," *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.*, vol. 229, no. 4, pp. 323–339, Nov. 2015.

[100] G. Cavone, M. Dotoli, and C. Seatzu, "Management of Intermodal Freight Terminals by First-Order Hybrid Petri Nets," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 2–9, Jan. 2016.

[101] F. Mahi, A. N. S. Moh, F. Debbat, and M. F. Khelfi, "Modelling and control of a multimodal transportation system using hybrid Petri nets with fuzzy logic," *Int. J. Syst. Control Commun.*, vol. 5, no. 3/4, p. 255, 2013.

[102] J. Yan and L. Li, "Fault-tolerant controller design for automated guided vehicle systems based on Petri nets," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, 2012, pp. 1531–1536.

[103] A. Gudelj, D. Kezić, and S. Vidačić, "Planning and optimization of AGV jobs by petri net and genetic algorithm," *J. Inf. Organ. Sci.*, vol. 36, no. 2, pp. 99–122, 2012.

[104] C. Liu and P. Ioannou, "Petri Net Modeling and Analysis of Automated Container Terminal Using Automated Guided Vehicle Systems," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1782, pp. 73–83, Jan. 2002.

[105] T. Nishi and R. Maeno, "Petri net decomposition approach to optimization of route planning problems for AGV systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 3, pp. 523–537, Jul. 2010.

[106] T. Nishi and Y. Tanaka, "Petri net decomposition approach for dispatching and conflict-free routing of bidirectional automated guided vehicle systems," *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 42, no. 5, pp. 1230–1243, Sep. 2012.

[107] M. Dotoli and M. P. Fanti, “Coloured timed Petri net model for real-time control of automated guided vehicle systems,” *Int. J. Prod. Res.*, vol. 42, no. 9, pp. 1787–1814, May 2004.

[108] N. Wu, “Resource-Oriented Petri Nets in Deadlock Avoidance of AGV Systems,” *System*, vol. 1, pp. 64–69, 2001.

[109] N. Wu and M. Zhou, “Modeling and Deadlock Control of Automated Guided Vehicle Systems,” *IEEE/ASME Trans. Mechatronics*, vol. 9, no. 1, pp. 50–57, Mar. 2004.

[110] N. Q. Wu and M. C. Zhou, “Shortest routing of bidirectional automated guided vehicles avoiding deadlock and blocking,” *IEEE/ASME Trans. Mechatronics*, vol. 12, no. 1, pp. 63–72, Feb. 2007.

[111] E. Roszkowska, “Undirected colored Petri net for modelling and supervisory control of AGV systems,” *Wodes’02 Sixth Int. Work. Discret. Event Syst. Proc.*, pp. 135–142, 2002.

[112] D. Giglio, “A petri net model for an open path Multi-AGV system,” *ICINCO 2014 - Proc. 11th Int. Conf. Informatics Control. Autom. Robot.*, vol. 2, pp. 734–745, 2014.

[113] M. Dotoli, F. Sciancalepore, N. Epicoco, M. Falagario, B. Turchiano, and N. Costantino, “A periodic event scheduling approach for offline timetable optimization of regional railways,” in *10th Int Conf Net Sens Contr*, 2013, pp. 849–854.

[114] V. Cacchiani *et al.*, “An overview of recovery models and algorithms for real-time railway rescheduling,” *Transp. Res. Part B Methodol.*, vol. 63, pp. 15–37, May 2014.

[115] M. Dotoli, N. Epicoco, M. Falagario, B. Turchiano, G. Cavone, and A. Convertini, “A Decision Support System for real-time rescheduling of railways,” in *2014 European Control Conference, ECC 2014*, 2014.

[116] Y. Bavafa-Toosi, C. Blendinger, V. Mehrmann, A. Steinbrecher, and R. Unger, “A new methodology for modeling, analysis, synthesis, and simulation of time-optimal train traffic in large networks,” *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 1, pp. 43–51, Jan. 2008.

[117] B. Adenso-Díaz, M. Oliva González, and P. González-Torre, “On-line timetable re-scheduling in regional train services,” *Transp. Res. Part B Methodol.*, vol. 33, no. 6, pp. 387–398, Aug. 1999.

[118] W. Fang, S. Yang, and X. Yao, “A Survey on Problem Models and Solution Approaches to Rescheduling in Railway Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6. pp. 2997–3016, Dec-2015.

[119] J.-F. Cordeau, P. Toth, and D. Vigo, “A Survey of Optimization Models for Train Routing and Scheduling,” *Transp. Sci.*, vol. 32, no. 4, pp. 380–404, Nov. 1998.

[120] J. Jacobs, “Reducing delays by means of computer-aided ‘on-the-spot’ rescheduling,” *Adv. Transp.*, vol. 15, pp. 603–612, May 2004.

[121] L. Chen, C. Roberts, F. Schmid, and E. Stewart, “Modeling and Solving Real-Time Train Rescheduling Problems in Railway Bottleneck Sections,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1896–1904, Aug. 2015.

[122] I. Şahin, “Railway traffic control and train scheduling based on inter-train conflict management,” *Transp. Res. Part B Methodol.*, vol. 33, no. 7, pp. 511–534, Sep. 1999.

[123] M. T. Isaai and N. P. Cassaigne, “Predictive and reactive approaches to the train-scheduling problem: a knowledge management perspective,” *IEEE Trans. Syst. Man Cybern. Part C (Applications Rev.*, vol. 31, no. 4, pp. 476–484, 2001.

[124] M. J. Dorfman and J. Medanic, “Scheduling trains on a railway network using a discrete event model of railway traffic,” *Transp. Res. Part B Methodol.*, vol. 38, no. 1, pp. 81–98, Jan. 2004.

[125] M. Mazzarello and E. Ottaviani, “A traffic management system for real-time traffic optimisation in railways,” *Transp. Res. Part B Methodol.*, vol. 41, no. 2, pp. 246–274, Feb. 2007.

[126] J. Törnquist, “Railway traffic disturbance management-An experimental analysis of disturbance complexity, management objectives and limitations in planning horizon,” *Transp. Res. Part A Policy Pract.*, vol. 41, no. 3, pp. 249–266, Mar. 2007.

[127] P. Pellegrini, G. Marliere, R. Pesenti, and J. Rodriguez, “RECIFE-MILP: An Effective MILP-Based Heuristic for the Real-Time Railway Traffic Management Problem,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2609–2619, Oct. 2015.

[128] F. Corman, A. D’Ariano, M. Pranzo, and I. A. Hansen, “Effectiveness of dynamic reordering and rerouting of trains in a complicated and densely occupied station area,” *Transp. Plan. Technol.*, vol. 34, no. 4, pp. 341–362, Jun. 2011.

[129] P. Pellegrini, G. Marliere, and J. Rodriguez, “Real Time Railway Traffic Management Modeling Track-Circuits,” p. 12p, Jan. 2012.

[130] J. Törnquist and J. A. Persson, “N-tracked railway traffic re-scheduling during disturbances,” *Transp. Res. Part B Methodol.*, vol. 41, no. 3, pp. 342–362, Mar. 2007.

[131] M. Almodóvar and R. García-Ródenas, “On-line reschedule optimization for passenger railways in case of emergencies,” *Comput. Oper. Res.*, vol. 40, no. 3, pp. 725–736, Mar. 2013.

[132] J. L. Espinosa-Aranda and R. García-Ródenas, “A demand-based weighted train delay approach for rescheduling railway networks in real time,” *J. Rail Transp. Plan. Manag.*, vol. 3, no. 1–2, pp. 1–13, Feb. 2013.

[133] L. Meng and X. Zhou, “Robust single-track train dispatching model under a dynamic and stochastic environment: A scenario-based rolling horizon solution approach,” *Transp. Res. Part B Methodol.*, vol. 45, no. 7, pp. 1080–1102, Aug. 2011.

[134] M. Boccia, C. Mannino, and I. Vasilyev, “The dispatching problem on multitrack territories: Heuristic approaches based on mixed integer linear programming,” *Networks*, vol. 62, no. 4, pp. 315–326, Dec. 2013.

[135] A. D’Ariano, “Innovative decision support system for railway traffic control,” *IEEE Intell. Transp. Syst. Mag.*, vol. 1, no. 4, pp. 8–16, 2009.

[136] A. R. Albrecht, D. M. Panton, and D. H. Lee, “Rescheduling rail networks with maintenance disruptions using Problem Space Search,” *Computers and Operations Research*, vol. 40, no. 3. Pergamon, pp. 703–712, Mar-2013.

[137] J. Törnquist, “Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms,” *Algorithmic MeThods Model. Optim. Rail- ways*, vol. 2, p. 23p, 2006.

[138] J. Tornquist and J. A. Persson, “Train Traffic Deviation Handling Using Tabu Search and Simulated Annealing,” *Proc. 38th Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 0, no. C, pp. 1–10, 2005.

[139] J. Törnquist, “Greedy algorithm for railway traffic re-scheduling during disturbances: A Swedish case,” *IET Intell. Transp. Syst.*, vol. 4, no. 4, pp. 375–386, 2010.

[140] J. Törnquist Krasemann, “Design of an effective algorithm for fast response to the re-scheduling of railway traffic during disturbances,” *Transp. Res. Part C Emerg. Technol.*, vol. 20, no. 1, pp. 62–78, Feb. 2012.

[141] H. Grahn and J. T. Krasemann, “a Parallel Re-Scheduling Algorithm for Railway Traffic Disturbance Management – Initial Results,” *Proc. 2nd Model. Tech. ITS Conf. Leuven*, vol. 46, pp. 0–3, 2011.

[142] R. Acuna-Agost, P. Michelon, D. Feillet, and S. Gueye, “SAPI: Statistical Analysis of Propagation of Incidents. A new approach for rescheduling trains after disruptions,” *Eur. J. Oper. Res.*, vol. 215, no. 1, pp. 227–243, Nov.

2011.

[143] Y. H. Min, M. J. Park, S. P. Hong, and S. H. Hong, "An appraisal of a column-generation-based algorithm for centralized train-conflict resolution on a metropolitan railway network," *Transp. Res. Part B Methodol.*, vol. 45, no. 2, pp. 409–429, Feb. 2011.

[144] M. M. Dessouky, Q. Lu, J. Zhao, and R. C. Leachman, "An exact solution procedure to determine the optimal dispatching times for complex rail networks," *IIE Trans. (Institute Ind. Eng.*, vol. 38, no. 2, pp. 141–152, Feb. 2006.

[145] T. J. J. van den Boom and B. De Schutter, "DYNAMIC RAILWAY NETWORK MANAGEMENT USING SWITCHING MAX-PLUS-LINEAR MODELS," *IFAC Proc. Vol.*, vol. 39, no. 12, pp. 343–348, Jan. 2006.

[146] J. Rudan, B. Kersbergen, T. Van Den Boom, and K. Hangos, "Performance analysis of MILP based model predictive control algorithms for dynamic railway scheduling," in *2013 European Control Conference, ECC 2013*, 2013, pp. 4562–4567.

[147] L. Wang, Y. Qin, J. Xu, and L. Jia, "A fuzzy optimization model for high-speed railway timetable rescheduling," *Discret. Dyn. Nat. Soc.*, vol. 2012, pp. 1–22, Dec. 2012.

[148] N. Kliewer and L. Suhl, "A note on the online nature of the railway delay management problem," in *Networks*, 2011, vol. 57, no. 1, pp. 28–37.

[149] L. Gély, G. Dessagne, and C. Lérin, "Modelling train re-scheduling with optimization and operational research techniques," pp. 1–9, 2006.

[150] S. Mu and M. Dessouky, "Scheduling freight trains traveling on complex networks," *Transp. Res. Part B Methodol.*, vol. 45, no. 7, pp. 1103–1123, Aug. 2011.

[151] R. M. Lusby, J. Larsen, M. Ehrgott, and D. M. Ryan, "A set packing inspired method for real-time junction train routing," *Comput. Oper. Res.*, vol. 40, no. 3, pp. 713–724, Mar. 2013.

[152] S. Narayanaswami and N. Rangaraj, "Modelling disruptions and resolving conflicts optimally in a railway schedule," *Comput. Ind. Eng.*, vol. 64, no. 1, pp. 469–481, Jan. 2013.

[153] P. Pellegrini, G. Marlière, and J. Rodriguez, "Optimal train routing and scheduling for managing traffic perturbations in complex junctions," *Transp. Res. Part B Methodol.*, vol. 59, pp. 58–80, Jan. 2014.

[154] D. R. Kraay and P. T. Harker, "Real-time scheduling of freight railroads," *Transp. Res. Part B*, vol. 29, no. 3, pp. 213–229, Jun. 1995.

[155] A. Zimmermann, *Stochastic discrete event systems: Modeling, evaluation, applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[156] G. Cavone, M. Dotoli, and C. Seatzu, "Management of Intermodal Freight Terminals by First-Order Hybrid Petri Nets," *IEEE Robot. Autom. Lett.*, 2016.

[157] G. Cavone, M. Dotoli, N. Epicoco, and C. Seatzu, "Intermodal terminal planning by Petri Nets and Data Envelopment Analysis," *Control Eng. Pract.*, vol. 69, 2017.

[158] W. DeWitt and J. Clinger, "INTERMODAL FREIGHT TRANSPORTATION," *Transp. New Millenn.*, 2000.

[159] M. Dotoli, S. Hammadi, K. Jeribi, C. Russo, and H. Zgaya, "A multi-agent Decision Support System for optimization of co-modal transportation route planning services," in *52nd IEEE Conference on Decision and Control*, 2013, pp. 911–916.

[160] F. Maggioni, G. Perboli, and R. Tadei, "The multi-path traveling salesman problem with stochastic travel costs: Building realistic instances for city logistics applications," in *Transportation Research Procedia*, 2014, vol. 3, pp. 528–536.

[161] G. Perboli, R. Tadei, and L. Gobbato, "The Multi-Handler Knapsack Problem under Uncertainty," *Eur. J. Oper. Res.*, vol. 236, no. 3, pp. 1000–1007, Aug. 2014.

[162] M. Dotoli, N. Epicoco, M. Falagario, D. Palma, and B. Turchiano, "A train load planning optimization model for intermodal freight transport terminals: A case study," in *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*, 2013, pp. 3597–3602.

[163] A. Caris, C. Macharis, and G. K. Janssens, "Decision support in intermodal transport: A new research agenda," *Comput. Ind.*, vol. 64, no. 2, pp. 105–112, Feb. 2013.

[164] I. Harris, Y. Wang, and H. Wang, "ICT in multimodal transport and technological trends: Unleashing potential for the future," *Int. J. Prod. Econ.*, vol. 159, pp. 88–103, Jan. 2015.

[165] A. Perego, S. Perotti, and R. Mangiaracina, "ICT for logistics and freight transportation: a literature review and research agenda," *Int. J. Phys. Distrib. Logist. Manag.*, vol. 41, no. 5, pp. 457–483, Jun. 2011.

[166] K. Alicke, "Modeling and optimization of the intermodal terminal Mega Hub," in *Container Terminals and Automated Transport Systems: Logistics Control Issues and Quantitative Decision Support*, Berlin/Heidelberg: Springer-Verlag, 2005, pp. 307–323.

[167] V. Boschian, M. Dotoli, M. P. Fanti, G. Iacobellis, and W. Ukovich, "A metamodeling approach to the management of intermodal transportation networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 3, pp. 457–469, Jul. 2011.

[168] Y. Du, W. Tan, and M. Zhou, "Timed compatibility analysis of web service composition: A modular approach based on petri nets," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 594–606, Apr. 2014.

[169] A. Giua, F. DiCesare, and M. Silva, "Generalized mutual exclusion contraints on nets with uncontrollable transitions," in *[Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics*, 1992, pp. 974–979.

[170] F. Sessego, A. Giua, and C. Seatzu, "HYPENS: a Matlab tool for timed discrete, continuous and hybrid Petri nets," *Lect. Notes Comput. Sci. Springer-Verlag*, vol. 5062, pp. 419–428, 2008.

[171] K. G. Murty, J. Liu, Y. W. Wan, and R. Linn, "A decision support system for operations in a container terminal," *Decis. Support Syst.*, vol. 39, no. 3, pp. 309–332, May 2005.

[172] N. Viswanadham, *Analysis of Manufacturing Enterprises: An Approach to Leveraging Value Delivery Processes for Competitive Advantage*. Springer US, 1999.

[173] C. I. Liu, H. Jula, and P. a Ioannou, "Design, simulation, and evaluation of automated container terminals," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 12–26, Mar. 2002.

[174] A. L. Law, "Simulation modeling and analysis," *New York McGraw*, 2007.

[175] F. Balduzzi and A. Di Febbraro, "Combining fault detection and process optimization in manufacturing systems using first-order hybrid Petri nets," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, 2001, vol. 1, pp. 40–45.

[176] F. Balduzzi, A. Giua, and C. Seatzu, "Modelling and simulation of manufacturing systems with first-order hybrid Petri nets," *Int. J. Prod. Res.*, vol. 39, no. 2, pp. 255–282, Jan. 2001.

[177] A. Inaba, T. Suzuki, and S. Okuma, "Realization of fault tolerant manufacturing system and its scheduling based on hierarchical Petri net modeling," *2003 IEEE Int. Conf. Robot. Autom. (Cat. No.03CH37422)*, vol. 3, pp. 3959–3964, 2003.

[178] M. Velasquez and P. T. Hester, "An Analysis of Multi-Criteria Decision Making Methods," *Int. J. Oper. Res.*, vol. 10, no. 2, pp. 56–66, 2013.

[179] R. Markovits-Somogyi, “Ranking Efficient and Inefficient Decision Making Units in Data Envelopment Analysis,” *Int. J. Traffic Transp. Eng.*, vol. 1, no. 4, pp. 245–256, 2011.

[180] A. Charnes, W. W. Cooper, and E. Rhodes, “Measuring the efficiency of decision making units,” *Eur. J. Oper. Res.*, vol. 2, no. 6, pp. 429–444, Nov. 1978.

[181] C. Araz and I. Ozkarahan, “Supplier evaluation and management system for strategic sourcing based on a new multicriteria sorting procedure,” *Int. J. Prod. Econ.*, vol. 106, no. 2, pp. 585–606, Apr. 2007.

[182] L. Angulo-Meza and M. P. Estellita Lins, “Review of methods for increasing discrimination in data envelopment analysis,” in *Annals of Operations Research*, 2002, vol. 116, no. 1–4, pp. 225–242.

[183] T. Sexton, R. Silkman, and A. J. Hogan, “Data envelopment analysis: Critique and extensions.,” *Meas. Effic. An Assess. Data Envel. Anal.*, vol. 1986, no. 32, pp. 73–105, 1986.

[184] J. R. Doyle and R. H. Green, “Cross-Evaluation in DEA - Improving Discrimiation Among DMUs,” *INFOR*, vol. 33, no. 3, pp. 205–222, Aug. 1995.

[185] M. Khalili, A. S. Camanho, M. C. A. S. Portela, and M. R. Alirezaee, “The measurement of relative efficiency using data envelopment analysis with assurance regions that link inputs and outputs,” *Eur. J. Oper. Res.*, vol. 203, no. 3, pp. 761–770, Jun. 2010.

[186] M. Dotoli, N. Epicoco, M. Falagario, A. Piconese, F. Sciancalepore, and B. Turchiano, “A real time traffic management model for regional railway networks under disturbances,” in *IEEE International Conference on Automation Science and Engineering*, 2013, pp. 892–897.

[187] X. Guo, J. Wu, H. Sun, R. Liu, and Z. Gao, “Timetable coordination of first trains in urban railway network: A case study of Beijing,” *Appl. Math. Model.*, vol. 40, no. 17–18, pp. 8048–8066, Sep. 2016.

[188] E. Hassannayebi, S. H. Zegordi, and M. Yaghini, “Train timetabling for an urban rail transit line using a Lagrangian relaxation approach,” *Appl. Math. Model.*, vol. 40, no. 23–24, pp. 1339–1351, Dec. 2016.

[189] S. Van Aken, N. Bešinović, and R. M. P. Goverde, “Designing alternative railway timetables under infrastructure maintenance possessions,” *Transp. Res. Part B Methodol.*, vol. 98, pp. 224–238, Apr. 2017.

[190] X. Li, B. Shou, and D. Ralescu, “Train rescheduling with stochastic recovery time: A new track-backup approach,” *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 44, no. 9, pp. 1216–1233, Sep. 2014.

[191] T. Dollevoet, D. Huisman, L. G. Kroon, L. P. Veelenturf, and J. C. Wagenaar, “Application of an iterative framework for real-time railway rescheduling,” *Comput. Oper. Res.*, vol. 78, pp. 203–217, Feb. 2017.

[192] M. Dotoli, N. Epicoco, M. Falagario, B. Turchiano, G. Cavone, and A. Convertini, “A Decision Support System for real-time rescheduling of railways,” in *2014 European Control Conference, ECC 2014*, 2014, pp. 696–701.

[193] G. Cavone, M. Dotoli, N. Epicoco, and C. Seatzu, “A decision making procedure for robust train rescheduling based on mixed integer linear programming and Data Envelopment Analysis,” *Appl. Math. Model.*, vol. 52, pp. 255–273, Dec. 2017.

[194] A. D’Ariano, M. Pranzo, and I. A. Hansen, “Conflict resolution and train speed coordination for solving real-time timetable perturbations,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 208–222, Jun. 2007.

[195] F. Corman, A. D’Ariano, D. Pacciarelli, and M. Pranzo, “Bi-objective conflict detection and resolution in railway traffic management,” *Transp. Res. Part C Emerg. Technol.*, vol. 20, no. 1, pp. 79–94, Feb. 2012.

[196] M. J. C. M. Vromans, R. Dekker, and L. G. Kroon, “Reliability and heterogeneity of railway services,” *Eur. J. Oper. Res.*, vol. 172, no. 2, pp. 647–665, Jul. 2006.

[197] M. A. Salido, F. Barber, and L. Ingolotti, “Robustness in railway transportation scheduling,” in *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, 2008, pp. 2833–2837.

[198] E. Castillo, I. Gallego, J. M. Ureña, and J. M. Coronado, “Timetabling optimization of a mixed double- and single-tracked railway network,” *Appl. Math. Model.*, vol. 35, no. 2, pp. 859–878, Feb. 2011.

[199] M. Dotoli, N. Epicoco, M. Falagario, and F. Sciancalepore, “A cross-efficiency fuzzy Data Envelopment Analysis technique for performance evaluation of Decision Making Units under uncertainty,” *Comput. Ind. Eng.*, vol. 79, pp. 103–114, Jan. 2015.

[200] F. Corman and L. Meng, “A review of online dynamic models and algorithms for railway traffic management,” *IEEE Trans Intell Transp Sys*, vol. 16, no. 3, pp. 1274–1284, 2015.

[201] X. Xu, K. Li, L. Yang, and J. Ye, “Balanced train timetabling on a single-line railway with optimized velocity,” *Appl. Math. Model.*, vol. 38, no. 3, pp. 894–909, Feb. 2014.

[202] Y. Gao, L. Yang, and S. Li, “Uncertain models on railway transportation planning problem,” *Appl. Math. Model.*, vol. 40, no. 7–8, pp. 4921–4934, Apr. 2016.

[203] E. V. Andersson, A. Peterson, and J. Törnquist Krasemann, “Quantifying railway timetable robustness in critical points,” *J. Rail Transp. Plan. Manag.*, vol. 3, no. 3, pp. 95–110, Aug. 2013.

[204] M. A. Salido, F. Barber, and L. Ingolotti, “Robustness for a single railway line: Analytical and simulation methods,” *Expert Syst. Appl.*, vol. 39, no. 18, pp. 13305–13327, Dec. 2012.

[205] E. Hassannayebi, S. H. Zegordi, M. R. Amin-Naseri, and M. Yaghini, “Train timetabling at rapid rail transit lines: a robust multi-objective stochastic programming approach,” *Oper. Res.*, vol. 17, no. 2, pp. 435–477, Jul. 2017.

[206] L. G. Kroon, D. Huisman, and G. Maróti, “Optimisation models for railway timetabling,” in *Railway timetable & traffic, Eurailpress*, 2008.

[207] A. D’Ariano, D. Pacciarelli, and M. Pranzo, “Assessment of flexible timetables in real-time traffic management of a railway bottleneck,” *Transp. Res. Part C Emerg. Technol.*, vol. 16, no. 2, pp. 232–245, Apr. 2008.

[208] L. G. Kroon, R. Dekker, and M. J. C. M. Vromans, “Cyclic railway timetabling: A stochastic optimization approach,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007, vol. 4359 LNCS, pp. 41–68.

[209] M. Fischetti, D. Salvagnin, and A. Zanette, “Fast Approaches to Improve the Robustness of a Railway Timetable,” *Transp. Sci.*, vol. 43, no. 3, pp. 321–335, Aug. 2009.

[210] C. Liebchen, M. Schachtebeck, A. Schöbel, S. Stiller, and A. Prigge, “Computing delay resistant railway timetables,” *Comput. Oper. Res.*, vol. 37, no. 5, pp. 857–868, May 2010.

[211] R. Larsen, M. Pranzo, A. D’Ariano, F. Corman, and D. Pacciarelli, “Susceptibility of optimal train schedules to stochastic disturbances of process times,” *Flex. Serv. Manuf. J.*, vol. 26, no. 4, pp. 466–489, Dec. 2014.

[212] P. Jovanović, P. Kecman, N. Bojović, and D. Mandić, “Optimal allocation of buffer times to increase train schedule robustness,” *Eur. J. Oper. Res.*, vol. 256, no. 1, pp. 44–54, Jan. 2017.

[213] M. Samà, C. Meloni, A. D’Ariano, and F. Corman, “A multi-criteria decision support methodology for real-time train scheduling,” *J. Rail Transp. Plan. Manag.*, vol. 5, no. 3, pp. 146–162, Nov. 2015.

[214] H.-J. Zimmermann, *Fuzzy Set Theory—and Its Applications*. Springer Netherlands, 2001.

[215] M. Jiménez and A. Bilbao, “Pareto-optimal solutions in fuzzy multi-objective linear programming,” *Fuzzy Sets Syst.*, vol. 160, no. 18, pp. 2714–2721, Sep. 2009.

[216] L. Oneto *et al.*, “Dynamic delay predictions for large-scale railway networks: Deep and shallow extreme learning machines tuned via thresholdout,” *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 47, no. 10, pp. 2754–2767, Oct. 2017.

[217] X. Hu, N. Cui, E. Demeulemeester, and L. Bie, "Incorporation of activity sensitivity measures into buffer management to manage project schedule risk," in *European Journal of Operational Research*, 2016, vol. 249, no. 2, pp. 717–727.

[218] T. Dewilde, P. Sels, D. Cattrysse, and P. Vansteenwegen, "Defining robustness of a railway timetable," in *Proc 4th Int Semin Railw Oper Model Anal, RailRome2011*, 2011, pp. 1–20.

[219] I. Louwerse and D. Huisman, "Adjusting a railway timetable in case of partial or complete blockades," *Eur. J. Oper. Res.*, vol. 235, no. 3, pp. 583–593, Jun. 2014.

[220] L. P. Veelenturf and L. G. Kroon, *Disruption management in passenger railways : models for timetable, rolling stock and crew rescheduling*. Erasmus Universiteit Rotterdam, 2014.

[221] L. L. Blenkers, "Railway disruption management Lex Blenkers," 2015.