



Verifica e previsione della regolarità del traffico ferroviario con un modello di simulazione basato sulle Reti di PETRI

Check and forecasting of railway traffic regularity by a PETRI Nets based simulation model

Prof. Ing. STEFANO RICCI, Dott. Ing. ANTONIO TIERI()*

SOMMARIO – L'articolo mostra le caratteristiche principali di un modello di simulazione dell'esercizio ferroviario. Utilizzando il formalismo grafico delle Reti di PETRI il modello simula il funzionamento degli enti elementari che costituiscono i sistemi di controllo dell'esercizio ferroviario attraverso la ricostruzione dei passaggi logici che li caratterizzano con l'ausilio delle Reti di PETRI estese, gerarchiche e colorate. Il modello può essere applicato negli studi di regolarità del traffico ferroviario su una linea e può inoltre essere usato efficacemente come supporto alle decisioni nella pianificazione di nuove linee e dei relativi orari di esercizio, nell'adeguamento di linee esistenti con mutate condizioni di traffico ed in fase di gestione del traffico per minimizzare i ritardi, essendo il modello dotato di un sistema automatico di precedenza.

SUMMARY – The article describes the main features of a railway operation simulation model. The model simulates the operation of the elementary devices composing the railway traffic management systems by describing their logical steps sequence by using the graphic formalism of extended and colored PETRI Nets. The model may be applied in railway traffic regularity studies as well as a decision support tool for the planning of new or upgraded lines and their timetable or for delays minimization in traffic management by its automatic trains overcome functionality.

1. Introduzione

La possibilità di aumentare la capacità di una tratta ferroviaria rispettando i livelli di regolarità promessi da chi gestisce il servizio di trasporto può essere verificata, o quanto meno stimata con buona approssimazione, in fase progettuale, attraverso modelli che simulino l'esercizio al fine di valutare eventuali criticità che impediscono il raggiungimento dei livelli accennati.

I modelli, per essere utili, dovrebbero simulare il processo rispettando quelle che sono le condizioni al contorno dettate dall'infrastruttura e dai sistemi di segnalamento e controllo. Inoltre dovrebbero poter essere facilmente implementabili su qualsiasi impianto, ovvero possedere una elevata flessibilità di rappresentazione.

Nell'articolo verranno presentati i risultati principali di una ricerca di dottorato [1], che ha avuto il fine di costruire un modello con le caratteristiche descritte, inserendosi in un percorso già avviato dall'area Trasporti del Dipartimento d'Idraulica, Trasporti e Strade (DITS) della "Sapienza" [2] [5] [6] [7] [9] [11] [13]. L'obiettivo finora perseguito è stato quello di simulare il funzionamento degli enti elementari che costituiscono i sistemi di controllo

1. Introduction

The possibility to improve the capacity of railway lines, within the regularity levels fixed by the railway undertaking that manage the transport service, may be verified, or at least estimated with good approximation, during the planning phase, by railway operation simulation models.

The models, to be useful, should simulate the process with all boundary conditions imposed by infrastructure, signaling and control systems. Moreover they should be applied on every railway plant, i.e. they should have high flexibility in the railway system representation.

In the article are presented the main results of a PhD research [1] finalised to the building of a model with defined features along a consolidated research path of Transport Area at Department of Hydraulics Transport and Roads of Sapienza University of Rome [2] [5] [6] [7] [9] [11] [13].

During the research, the same formalism was used to design a model for traffic regularity analysis, switching from a micro to a macro approach applied to railway op-

(*) Sapienza – Università di Roma – DITS – Area Trasporti.

(*) Sapienza – Università di Roma – DITS – Area Trasporti.

dell'esercizio ferroviario (deviatoi, circuiti di binario, segnali, ecc), attraverso la ricostruzione dei passaggi logici che li caratterizzano, utilizzando il linguaggio grafico delle Reti di PETRI. Queste si caratterizzano per la semplicità d'implementazione su un calcolatore, per la possibilità di seguire visivamente il processo durante il suo scorrere e per la possibilità di ricreare le tre forme fondamentali di istruzione condizionale in un linguaggio-macchina, ovvero IF-THEN-ELSE, WHILE-DO e CASE-DO, solo attraverso artifici grafici e dunque senza ricorrere alla programmazione di alto livello.

Durante la ricerca si è cercato di utilizzare lo stesso linguaggio per costruire un modello finalizzato alla verifica della regolarità della circolazione, dunque con un passaggio dal micro al macro rispetto alle ricerche precedenti sulle Reti di PETRI applicate all'esercizio. La ricerca ha dunque avuto come obiettivo la costruzione di un modello di simulazione dell'esercizio ferroviario in grado di analizzare la regolarità del traffico, in linea e in stazione, al fine di suggerire ipotesi di miglioramento del servizio.

Nel corso dei tre anni di ricerca l'obiettivo è stato ampliato, realizzando l'attuale versione del modello che può essere considerato uno strumento di supporto alle decisioni, non solo nella pianificazione dell'orario ferroviario, ma anche nella gestione ordinaria dei conflitti di circolazione. Quest'ultimo aspetto rappresenta, in termini temporali, l'ultimo risultato della ricerca; è da intendersi come risultato preliminare nell'utilizzo del modello come risolutore di conflitti di circolazione, in quanto necessita di ulteriori approfondimenti, ed è da considerarsi come il primo passo di un interessante filone di ricerca che ha come obiettivo finale la realizzazione di un modello di simulazione in grado di operare il cosiddetto *real time rescheduling*, ovvero la riprogrammazione dell'orario ferroviario in tempo reale a seguito di eventi che perturbano la circolazione.

2. Reti di Petri

2.1. Generalità

Le Reti di PETRI sono un linguaggio formale (meta-linguaggio) per la programmazione logica diffusamente utilizzato per risolvere problemi in automatica industriale, robotica e più in generale nella descrizione di processi che evolvono con eventi discreti in sistemi concorrenti (dettagli su teoria, algebra e formalismo grafico sono riportati nell'Appendice 1). Proprio per questa ragione da alcuni anni le Reti di PETRI sono utilizzate anche per l'analisi e la risoluzione di problemi legati all'esercizio ferroviario, in particolare nella descrizione del funzionamento dei sistemi di controllo e segnalamento. In realtà, rispetto alla formulazione originaria del linguaggio, a seconda del tipo di applicazione in cui sono state usate, molte varianti sono state realizzate. Tali varianti prendono genericamente il nome di Reti di PETRI

eration. Final goal of this research is structuring a simulation model based on PETRI Nets for line capacity and railway operation regularity, capable to support the effectiveness and efficiency evaluation of infrastructural and/or operational improvements to the railway system.

These are characterized by the simple implementation on a computer, the possibility to follow the process and the possibility to reproduce the three conditional instructions in machine language, IF-THEN-ELSE, WHILE-DO e CASE-DO, without high level programming but only graphical procedures.

The research aims at building a model with described characteristics, joining in already achieved simulation modules of the operation of single elements of railway control systems (switches, track circuits, signals, etc.), through the reconstruction of their logical connections, using PETRI nets graphic language.

During the three years, PhD research the objectives have been extended reaching the present version of the model, which can be considered a decision support tool, useful not only in timetable planning, but also in the ordinary management of traffic conflicts. This last aspect represents the last research development and the key premise for the model use as traffic conflicts solver, in view of the full building of a global simulation model capable to operate the *real time rescheduling*, that means the possibility to re-plan in real time the timetable due to the traffic perturbations.

2. PETRI Nets

2.1. Generalities

Le PETRI Nets are a formal language (meta-language) for the logic programming diffusely employed to solve problems in industrial automatics, robotics and, more in general, to describe processes evolving by discrete events in concurrent systems (details on theory, algebra and formalisms are reported in Appendix 1). For this reason since some years the PETRI Nets are employed for analysis and resolution of railway operation problems, in particular for the description of signalling and control systems operation. In comparison with the original formulation many variants have been introduced according to this applications. These variants took the name of *Extended (High Level)* PETRI Nets because each of them adds a certain property or functions to the original (*Low Level*) PETRI Nets. Moreover, is presently ongoing an international standardisation process promoted by the Technical University of Braunschweig.

In the railway operation simulation it has been employed a particular typology of extended PETRI Nets, therefore in the following paragraphs will be described both the basic functions of PETRI Nets and their ex-

estese (o di alto livello), in quanto ognuna aggiunge una qualche proprietà o funzione alle Reti di PETRI originarie (di basso livello). Proprio per cercare di armonizzare i diversi modi di programmare con le Reti di PETRI, è attualmente in atto un processo di standardizzazione internazionale promosso dall'università di Braunschweig.

Poiché nel modello di simulazione ferroviario sviluppato è stato utilizzato un particolare tipo di Reti di PETRI estese, nei paragrafi seguenti si farà cenno al funzionamento base delle Reti di PETRI e alle estensioni utilizzate nel modello e rese disponibili dal software Faber®.

Tale software permette la realizzazione di reti che, rispetto alle tradizionali Reti di PETRI, hanno le seguenti proprietà:

- 1) sono reti *estese*, nel senso che associano alle transizioni, ai posti e alle marche delle azioni e delle facoltà ulteriori a quelle permesse ad un normale grafo di stato;
- 2) sono reti *gerarchiche*, nel senso che è possibile costruire un modello object-oriented attraverso la costruzione di classi di oggetto formate da specifiche sottoreti e la possibilità di istanziare tali classi in oggetti da posizionare in un Top Level;
- 3) sono reti *colorate*, nel senso che è possibile associare a ciascun oggetto un colore per ogni azione compiuta all'interno dell'oggetto, ovvero per ogni transizione che scatta.

2.2. Estensioni delle Reti di PETRI utilizzate nel modello

Le proprietà fondamentali delle Reti di PETRI estese in Faber® riguardano la formalizzazione delle marche e le azioni associabili ad una transizione e ad un posto.

In Faber® una marca può essere trattata sia come semplice indicatore di posto attivo, sia come record di dati "viaggiante" nella rete. Ciò è possibile perché ogni struttura algebrica formata dal disegno delle reti di PETRI nel software è la base per una programmazione in C o in C++ (in particolare nel modello di simulazione ferroviario si è utilizzato il C). In altre parole ad ogni rete disegnata in Faber® è associato uno *script* in C, le cui variabili sono rappresentate proprio dalle marche.

È inoltre possibile inserire nelle reti due particolari tipi di transizioni non semplici: le transizioni con predicato e le transizioni con azione.

Le transizioni con predicato sono semplici transizioni a cui sono aggiunte, tramite un'istruzione in codice una o più condizioni; il rispetto di queste ultime è vincolante per lo scatto della transizione. Rispetto ad una normale transizione la differenza è che l'attivazione di tutti i posti di input è una condizione solo necessaria ma non sufficiente allo scatto della transizione stessa. Il predicato è sempre una condizione di tipo booleano; essa valuta le caratteristiche dei posti di input della transi-

tensions used in the model and made available by Faber® software.

This software allows the following performances:

- 1) the nets are *extended*, in the sense of associating, to transitions, places and tokens, actions and further possibilities in addition to those allowed in a simple state-graph;
- 2) the nets are *hierarchical*, in the sense that it is possible to use an object-oriented approach through the construction of classes of objects by assembling specific subnets and the possibility of calling these classes to put on a top level;
- 3) the nets are *coloured*, in the sense that it is possible to associate one colour to each object for each action performed into the object, i.e. for each transition firing.

2.2. PETRI Net extensions used in the model

The fundamental properties of extended PETRI nets used in the simulation model concern the token formalization and the properties associated to the transitions and to the places.

In Faber® a token can be managed both like a simple indicator of an active place and like a data record travelling across the net. This is possible because each algebraic structure, created by PETRI net draw, in the software, is the base of a C or C++ code (in particular in the railway simulation model C language is used). In other words, a C script code is associated to each designed PETRI net and the tokens are the variables.

Is moreover possible to insert in the nets some particular types of no simple transitions: transitions with predicate and transition with action.

The *transitions with predicate* are simple transitions with an added instruction. This represent one or more conditions: if these are observed the transition can fire, otherwise not. In practice, respect to a simple transition the difference is that the activation of all the input places is a only necessary condition, but not sufficient to the transition firing. In the model the transition with predicate are drawn with a little rhomb into the rectangle. The predicate is ever a Boolean condition evaluating the characteristics, i.e. the parameter values, of the tokens in input places or the objects parameters.

The *transitions with action* are simple transitions that, by specific C code instructions, can execute one or more actions when the transition fires. The actions access data of tokens of output places and they can manage the tokens released into output places. By data tokens, parameters and local variables, it is possible to modify the values of input variables of a transition returning new output values. Furthermore an action al-

zione stessa, ovvero i valori delle variabili delle marche o dei parametri degli oggetti.

Le transizioni con azione sono semplici transizioni a cui, tramite una scrittura di codice in C, è possibile far eseguire azioni nel momento in cui la transizione scatta. L'azione ha accesso ai dati delle marche prelevati dai posti di input e li può scrivere nelle marche rilasciate nei posti di output. Con i dati delle marche dei posti di input e con parametri e variabili locali è possibile modificare i valori delle variabili in ingresso e restituire in uscita nuovi valori. Inoltre un'azione consente di effettuare altre operazioni accessorie, come trascrivere i dati di interesse in appositi fogli di output al fine di valutare i risultati ottenuti.

Un particolare tipo di transizioni con azione sono le transizioni temporizzate. Nelle Reti di PETRI il fattore tempo delle transizioni è spesso utilizzato per simulare il tempo occorrente a svolgere una determinata operazione di un sistema. Tale tempo può essere sia un numero dato (risultato di un calcolo deterministico), sia il risultato di una distribuzione di probabilità, a seconda dell'approccio che si desidera seguire nella modellazione, deterministico o stocastico. Nel modello si è seguito un approccio deterministico e due tipi di transizioni temporizzate sono state utilizzate che si distinguono per il modo con cui è applicato un certo ritardo D :

- *firing delay*, che implica che le marche presenti nei posti di input della transizione vengono eliminate solo dopo il tempo D ;
- *release delay*, che implica l'eliminazione immediata delle marche dai posti di input della transizione, nel momento stesso in cui questi siano attivi, e la creazione delle marche nei posti di output solo dopo il tempo D .

Dal punto di vista logico, nel primo caso significa assegnare un tempo di operazione agli stati fisico/logici rappresentati dai posti di input, nel secondo determinare una distanza temporale dall'attivazione di uno stato fisico/logico ad un altro.

Nel modello si è fatto largo uso delle priorità di scatto fra le transizioni. In pratica fra transizioni concorrenti, ovvero che hanno uno stesso posto come input, si può stabilire una gerarchia di scatto in maniera tale da impedire scelte casuali da parte del software.

In fig. 1 è riportata la simbologia utilizzata nel modello per i diversi tipi di transizione.

La composizione efficace di transizioni, predicati ed azioni consente di sviluppare Reti di PETRI estese che riproducano le tipiche istruzioni condizionali dei linguaggi di programmazione di alto livello, riducendo al minimo la "programmazione in C" in favore del "disegno in Reti di PETRI".

Infine, in accordo alla teoria base delle Reti di PETRI, ogni posto può contenere più marche contemporanea-

low to do complementary operations like printing data in reports to better evaluate the simulation results.

A particular type of transition with action are the *timed transitions*. In a PETRI net the time factor of the transitions is often used to simulate the time necessary to execute an operation of a system. This time can be both a fixed number (result of a deterministic calculation) or result of a probabilistic distribution, depending on the modeling approach, deterministic or stochastic.

In the railway model a deterministic approach and two types of timed transitions are used, different for the delay (D) functioning:

- *firing delay*, tokens present in the input places of a transition are destroyed after the time D ;
- *release delay*, when the input places are active the tokens are immediately destroyed and the creation of the new tokens in the output places happens after the time D .

From a logical point of view, in the first case it means to assign an operational time to the physical/logical states represented by input places, in the second to fix a temporal distance from the activation of a physical/logical state to another.

To avoid this last possibility, in the model the *transition with priority* is largely used. Practically, between concurrent transitions, i.e. transitions with the same input places, it is possible to establish a firing hierarchy so that random choices by software are stopped.

In figure 1 the symbols used in Faber for the different types of transition are shown.

The effective composition of transitions, predicates and actions allows to develop extended PETRI nets reproducing typical conditional instructions of high level programming languages, reducing *C programming* in favor of *PETRI net design*.

Finally, according to the basic theory of PETRI nets, each place can contain more tokens of the same type at the same time. In this case, to manage the queue in the places, there are three possibilities:



Fig. 1 – Tipologie di transizione e relativa simbologia. *Types of transition and symbols used.*
(Transizione = *Transition* - con predicato = *with predicate* - con azione = *with action*).

mente, purché siano dello stesso tipo. In questo caso si può gestire la coda nei posti secondo tre modelli:

- FIFO (first-in, first-out) se l'ultima marca aggiunta è anche l'ultima a lasciare la coda;
- LIFO (last-in, first-out) se l'ultima marca è anche la prima a lasciare la coda;
- *token delay*, se le marche lasciano il posto secondo un ritardo temporale prestabilito.

È inoltre possibile impostare in ciascun posto una marca al "tempo zero" della simulazione, iniziandola con gli attributi desiderati.

2.3. Struttura "object oriented" del modello

Il modello ha un'impostazione ad oggetti, in cui le classi di oggetto rappresentano gli enti elementari costituenti la linea. Quali elementi è necessario rappresentare è argomento del capitolo 3, mentre lo schema seguito per la modellazione è raffigurato nella fig. 2. In questa è possibile notare tre livelli di programmazione.

Partendo dal basso, il primo livello corrisponde alle varie Sotto-reti (spesso divise a loro volta in Pagine per mera insufficienza di spazio nella rappresentazione) che simulano ognuna una specifica funzione elementare svolta all'interno del sistema di esercizio. In questo caso, per funzione elementare si intende la possibilità di far compiere all'ente un'azione che ha influenza nella circolazione dei treni: ad esempio la verifica di libertà di un ente, la percorrenza di un treno sopra un circuito di binario, il cambio di posizionamento di un deviatore, l'accelerazione o la decelerazione di un treno sopra una sezione di blocco, la predisposizione di un itinerario per un treno che giunge in stazione, la verifica di un conflitto di circolazione fra due treni, ecc. Ciascuna Sotto-rete è costruita con la combinazione di posti e transizioni, siano esse semplici, con predicato, con azione o con entrambi. Dunque la marcatura di ciascun posto rappresenta uno stato logico che il sistema può assumere all'interno di una determinata funzione.

Il livello intermedio è rappresentato dalle Classi di oggetto, che non sono altro che la descrizione di un ente ferroviario attraverso la composizione delle sue funzioni elementari, dunque delle sue Sotto-reti. Sono Classi di oggetto i circuiti di binario, i deviatori, i segnali, i passaggi a livello, il radio block center, i pedali conta-assi, ecc. Il collegamento fra le varie Sotto-reti è assicurato da posti speciali, denominati Comuni, che permettono il passaggio di una marca da una Sotto-rete ad un'altra.

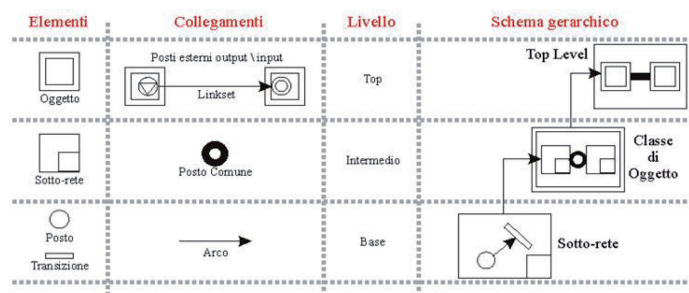


Fig. 2 – Struttura gerarchica utilizzata nel modello. *Hierarchical structure used in the model.*
(Elementi = Elements - Collegamenti = Links - Livello = Level - Schema gerarchico = Hierarchical scheme).

- FIFO (first-in, first-out), the last added token is also the last in queuing order;
- LIFO (last-in, first-out), the last added token is the first in the queuing order;
- 'token delay', the tokens leave the place according to a fixed temporal distance.

It is moreover possible to put in each place a starting token, by initializing it with desired attributes.

2.3. "Object oriented" model structure

The model is object oriented, with classes of objects representing the elementary devices composing the line.

The elements to be represented is a chapter 3 topic, while the modelling scheme is depicted in figure 2, where three levels of programming are noticed.

Starting from the bottom, the first level corresponds to the various sub-nets (often further divided in Pages for space insufficiency in the representation) each one simulating a specific elementary functions. In this case an elementary function is the possibility to let a device performing an action influencing the train traffic: i.e. the freedom check of a device, the train running on a track circuit, the movement of a switch, the acceleration or deceleration of a train on a block section, the route building for a train arriving in a station, the check of a traffic conflict between two trains. Each sub-net combines places and transitions (simple, with predicate, with action or both). Therefore the activation of each place represents a logical state, which the system may assume within a defined function.

The intermediate level is represented by the classes of objects, descriptions of devices by the composition of their elementary functions (sub-nets). Track circuits, switches, level crossings, radio block centres, axle counters are classes of objects. The link among the various sub-nets is ensured by special (common) places, which allow the token migration from a sub-net to another one.

The Top level is composed by a set of objects, suitably instantiated by the various classes, linked each other. The link elements between the objects are called *Linkset* and, by the external places of various objects, allow the token migration from an object to another.

2.4. Colored Nets and simulation window

In literature the term *Colored PETRI nets* corresponds to different meanings. Originally a net was defined colored if the tokens took different colors depending upon their attributes.

For extension, a colored PETRI net model includes elements which can acquire different colors depending upon the actions executed by the transitions. The elements may be both places and objects. In particular, when the output of a simulation will be introduced, the possible colored visualization for the Top Level objects in a specific window during the simulation will be more clear.

3. Architecture of railway operational model for regularity analysis

3.1. Basic scheme of the model

The model was built up by PETRI Nets for representing the objects composing the signalling and control systems. It has been employed a modular object oriented logic by decomposing the line into finite elements to be repeated and combined. This approach allows to describe any other line with similar signalling and control features.

As shown in figure 3 the model is composed by two modules (*Common* and *Management*), whose objects are includ-

Modulo Comune → Dipende solo dal sistema di segnalamento e controllo

- Circolo di Binario
- Deviazione
- Segnale
- Altri...

Modulo di Gestione → Dipende dal programma di esercizio (teorico e reale)

- Orario
- DCM

(Modulo Comune = *Common Module* - Modulo di Gestione = *Management Module* - Dipende solo dal sistema di segnalamento e controllo = *Depends upon signaling and control system only* - Dipende dal programma di esercizio (teorico e reale) = *Depends upon timetable (theoretical and real)* - Circuito di Binario = *Track Circuit* - Deviatore = *Switch* - Segnale = *Signal* - Orario = *Timetable* - DCM = *Dispatching Centre*).

simulare e, dunque, gli oggetti ad esso appartenenti sono non immediatamente “replicabili” su un'altra linea o sulla stessa ma con diverse condizioni di traffico. Il Layout del Top Level è dunque formato dalle varie istanze di oggetti dal modulo Comune, mentre le Funzioni Itinerario attingono sia agli oggetti del modulo Comune sia a quelli del modulo di Gestione.

Prima di vedere nel dettaglio i tre moduli è però opportuno introdurre le marche che entrano nelle classi di oggetto che li compongono.

3.2. Marche definite nel modello

Per poter far viaggiare le informazioni attraverso le Sotto-reti del modello, si è utilizzata l'importante proprietà delle Reti di PETRI estese che permette di definire le marche come vettori di informazioni configurati come variabili strutturate (STRUCT).

Poiché il modello simula sia la percorrenza fisica sia le condizioni che rendono possibile tale percorrenza, ovvero la registrazione della richiesta di formazione itinerario da parte di un treno, la verifica di esistenza, la verifica di libertà, il blocco e la liberazione degli oggetti che formano l'itinerario, tre diversi tipi di marche (TRENO, AM e NUL) sono stati utilizzati per non confondere il passaggio di informazioni riguardanti la percorrenza fisica del treno con quelle dovute alle pre-condizioni logiche perché tale percorrenza avvenga. Ognuno di questi tre tipi contiene le informazioni (variabili semplici) mostrate in fig. 4.

La marca TRENO è utilizzata nelle Sotto-reti dei circuiti di binario e dei deviatori che rappresentano la percorrenza dei treni. E' una variabile STRUCT definita da un numero di variabili semplici pari a $15+2x$, con x numero di stazioni (o di nodi) rappresentate nel modello. Otto di queste, quelle sulla sinistra della fig. 6, non sono propriamente variabili ma parametri fissi perché rappresentano caratteristiche intrinseche del treno da simulare; le altre (quelle sulla destra della marca TRENO in fig. 6), sono invece caratteristiche che possono subire variazioni durante la simulazione dell'esercizio.

La marca AM (Autorità di Movimento) entra in tutte le

MARCA: TRENO	
ID_Treno	Tempo assoluto
Categoria Treno	Progressiva
Direzione di Traffico	Distanza dalla prossima stazione
% Peso Frenato	Velocità corrente
Max velocità treno	Velocità max ammissibile in sezione entrante
Accelerazione media	Velocità max ammissibile in sezione successiva
Decelerazione media	Priorità in conflitto
Rango di velocità	Itinerario nella x stazione (sx)
	(...)
	Itinerario nella x stazione (dx)
	(...)

Nelle sottoreti che simulano la percorrenza

MARCA: AM	
ID_Treno	
Direzione di Traffico	
Itinerario nella x stazione (sx)	
(...)	
Itinerario nella x stazione (dx)	
(...)	

Nelle sottoreti che simulano la richiesta e la formazione degli itinerari

MARCA: NUL	
<u>Condizioni logiche booleane funzionali al sistema</u>	

Fig. 4 – Informazioni delle marche definite nel modello. *Information of tokens defined in the model.* (MARCA = TOKEN - Treno = Train - Categoria Treno = Train Category - Direzione di Traffico = Traffic direction - % Peso Frenato = Braken Mass % - Max velocità treno = Max train speed - Accelerazione media = Mean acceleration - Decelerazione media = Mean deceleration - Rango di velocità = Speed Class - Tempo assoluto = Absolute speed - Progressiva = Progressive - Distanza dalla prossima stazione = Distance to the next station - Velocità corrente = Present speed - Velocità max ammissibile in sezione entrante = Max allowed speed in previous section - Velocità max ammissibile in sezione successiva = Max allowed speed in following section - Itinerario nella x stazione = Route in station x - Nelle sottoreti che simulano la percorrenza = In sub-nets simulating running - Nelle sottoreti che simulano la richiesta e la formazione degli itinerari = In sub-nets simulating routes request and building - Condizioni logiche booleane funzionali al sistema = System related logical boolean conditions).

ed in a Top level, where the layout infrastructure and the route functions of trains running on the line are designed.

The *Common Module* contains all the classes of object that represent elements of the railway infrastructure; these classes depend only on signaling and control system of railway line and stations. The *Management Module* includes the classes representing the functions of traffic planning and management. All the objects are called on the Top level, the place where the infrastructure layout and the route functions, i.e. the logical conditions to form and to run the routes in the stations, are.

Before entering in modules details, the tokens composing in the various classes of objects should be introduced.

3.2. Tokens defined in the model

In order to allow the information movement across the model subnets, an important property of Extended PETRI nets is used, i.e. a token can be defined like a structural (STRUCT) variable becoming a real data record.

The model simulates both the physical train running and the conditions allowing this running (i.e. route recording, existence check, freedom check, interlocking and release of objects composing the route, therefore three different token typologies (TRENO, AM e NUL) have been used to avoid confusion between running and conditions. Each token ty-

Sotto-reti che rappresentano le funzionalità logiche che compie il treno diverse dalla percorrenza fisica. Dunque nelle Sotto-reti che rappresentano la verifica di libertà di un itinerario, il bloccamento, lo sbloccamento, le comunicazioni con i segnali, ecc. Anche in questo caso, all'aumentare del numero di stazioni aumentano le variabili semplici da inserire nella variabile **STRUCT**, essendo il numero complessivo $2+2x$. Si può notare come in questo caso l'unico elemento variabile sia proprio l'informazione sugli itinerari da percorrere.

L'ultimo tipo di marca utilizzata è la variabile semplice **NUL**, usata solo come attivatore di transizioni con condizioni logiche che esulano dalle azioni compiute dal treno, ma necessarie al funzionamento del modello.

3.3. Modulo Comune

Tutti gli enti fisici che compongono i sistemi di segnalamento e controllo dell'esercizio sono realizzati all'interno del modulo Comune. Le classi di oggetto rappresentative degli enti sviluppati durante la ricerca sono circuito di binario e deviatoio, le due fondamentali per poter compiere un'analisi di regolarità della linea scelta per calibrare il modello.

Le classi di oggetti, che si suddividono in diverse Sotto-reti a seconda delle funzioni che devono espletare, sono realizzate in maniera tale da contenere tutte le possibili opzioni che lungo una linea si potrebbero verificare, in maniera da non dover ridisegnare oggetti per situazioni particolari. I dettagli descrittivi delle singole classi sono riportate nell'Appendice 2. Così, ad esempio, la classe circuito di binario può essere utilizzata indistintamente in entrambi i sensi di marcia, come primo, ultimo o intermedio circuito di un itinerario, nelle operazioni di verifica di libertà, di bloccamento e di percorrenza di un itinerario. Si è in pratica cercato di realizzare classi di oggetti quanto più universali possibili, così da poter comporre un'intera linea ferroviaria semplicemente "clonando" e collegando con opportune interfacce i vari oggetti.

3.4. Modulo di Gestione

Il modulo di Gestione si compone di due classi di oggetto, **TT** e **DCM**. A differenza del modulo Comune, in quello di Gestione sono presenti classi di oggetto che dipendono dal programma di esercizio che si vuole simulare.

3.5. Top Level

Il Top Level è il luogo in cui si compongono i vari oggetti istanziati dalle classi precedentemente esposte. In esso, attraverso Sotto-reti in cui sono presenti oggetti, posti

polo contiene l'informazione mostrata in figura 4.

Il token **TRAIN** è utilizzato nelle sotto-reti dei circuiti di rotaie e interruttori che simulano il treno in movimento; è una variabile **STRUCT** definita da $15 + 2x$ variabili semplici (x = numero di stazioni) rappresentate nel modello. Otto di queste variabili (verdi a sinistra in figura 4) non sono vere variabili ma parametri fissi che rappresentano le caratteristiche proprie del treno da simulare; le altre (rosa in figura 4) sono, invece, caratteristiche che possono essere modificate durante la simulazione.

Il token **Movement Authority (AM)** entra in tutte le sotto-reti che rappresentano funzionalità logiche applicate al treno diverse dal movimento. Pertanto è presente nelle sotto-reti che rappresentano la verifica di libertà di una rotta, l'interlocking e il rilascio, la comunicazione con i segnali, ecc.. Anche in questo caso la quantità di variabili semplici da includere nella variabile **STRUCT** aumenta con il numero di stazioni. In questo caso l'elemento variabile è solo l'informazione sulle rotte da percorrere.

Il tipo di token più semplice è **NUL**, utilizzato solo per attivare alcune transizioni con condizioni indipendenti dalle azioni del treno, ma necessarie per il funzionamento del modello.

3.3. Common module

Tutti gli elementi fisici del sistema di segnalamento e controllo sono progettati nel modulo Comune. Gli elementi da rappresentare nel modello di regolarità sono: il circuito di rotaie e l'interruttore.

Queste classi di oggetti sono disegnate attraverso molte sotto-reti, ciascuna con una funzione particolare. Sono sviluppate per avere tutte le possibili opzioni lungo la linea; in questo modo è possibile utilizzare la stessa classe anche per situazioni diverse, senza ridisegnare gli oggetti per particolari condizioni. La descrizione dettagliata delle singole classi di oggetti è riportata nell'Appendice 2. Pertanto, per esempio, la classe circuito di rotaie può essere utilizzata indifferentemente in entrambi i sensi di marcia, come primo, ultimo o intermedio circuito di un itinerario, nelle operazioni di verifica di libertà, di bloccamento e di percorrenza di un itinerario. Si è cercato di realizzare classi di oggetti il più possibile universali, così da poter comporre un'intera linea ferroviaria semplicemente "clonando" e collegando con opportune interfacce i vari oggetti.

3.4. Management module

Il modulo di Gestione è composto da due classi di oggetti, **TT** e **DCM**. A differenza del modulo Comune, nel modulo di Gestione sono presenti classi di oggetti che dipendono dal programma di esercizio da simulare.

3.5. Top Level

Il Top Level è il luogo in cui vengono composti tutti gli oggetti istanziati dalle classi precedentemente descritte. In esso, attraverso sotto-reti in cui sono presenti oggetti, posti



in the previous modules are stored. In this class there are two types of functions: the first one is dedicated to represent the physical layout of the railway line, with all the elements of the net; the second one is drawing the different route functions existing along the line and in the stations.

Lay-out representation

The lay-out is represented in the model by the objects instanced by the Common Module classes, linked by interfaces named Linkset. These are links between Portset, i.e. collections of input and output external places, allowing the coordinated migration of various tokens by one object to another one. In figure 5 a part of the sub-net LAYOUT of Roma-Formia line at Itri Station is represented.

In Faber© all the objects are represented in a similar shape, but in the model they have been highlighted in a red rectangle the tracks represented by TC class objects and in a blue rectangle the tracks representing a switch (SW objects).

The composition, with suitable links, of 4 track circuits objects and 2 switches objects (representing in the reality just one of them) defines the physical and logical switch element and the basic figure of the model: the coherent replication of this figure, shown in figure 6, composes the whole lay-out of the line.

The diagram shows a complex dependency graph with nodes and edges. A red box highlights a specific sub-graph structure, which is a tree-like structure with a root node and several child nodes, all connected by edges. The nodes are represented by small squares, and the edges are lines connecting them. The highlighted structure is a tree with a root node and several child nodes, all connected by edges. The nodes are represented by small squares, and the edges are lines connecting them. The highlighted structure is a tree with a root node and several child nodes, all connected by edges. The nodes are represented by small squares, and the edges are lines connecting them.

Basic figure to build the LAYOUT of the TOP LEVEL

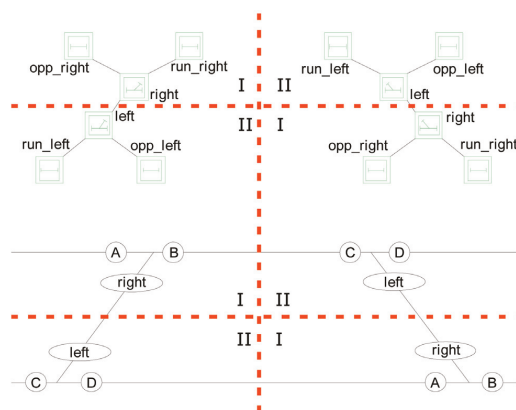


Fig. 7 – Proprietà dei collegamenti fra oggetti TC e SW. *Properties of the links between TC and SW objects.*

drante in alto a sinistra e quello in basso a destra sono fra loro simmetrici secondo un asse orizzontale. Stessa cosa vale per il quadrante in alto a destra con quello in basso a sinistra. Questa proprietà ha consentito di non duplicare le Portset e dunque un risparmio di tempo nella costruzione delle classi, ma soprattutto consentirà una costruzione meno onerosa di qualsiasi Top Level, poiché basta individuare qual è il deviatoio di destra e quale quello di sinistra per collegare gli unici sei Linkset utilizzati nel modello e riportati nella tabella 1.

Le Portset, ovvero i raggruppamenti di posti esterni, che rendono possibili i collegamenti sono, per la classe TC, le seguenti:

- TC_DX, collega al circuito di binario verso destra;
- TC_SX, collega al circuito di binario verso sinistra;
- TC_SW_OPP_DX, collega al deviatoio destro opposto (non percorribile);
- TC_SW_OPP_SX, collega al deviatoio sinistro opposto (non percorribile);
- TC_SW_RUN_DX, collega al deviatoio destro "di corsa" (percorribile);
- TC_SW_RUN_SX, collega al deviatoio sinistro "di corsa" (percorribile).

Analogamente, per la classe SW sono state realizzate altrettante Portset. Va tenuto presente che, per come è stata pensata questa classe, ad ogni deviatoio fisico corrispondono due oggetti deviatoi che possono essere individuati come deviatoio destro e deviatoio sinistro. Le Portset sono le seguenti:

- SW_DX, collega al deviatoio destro;
- SW_SX, collega al deviatoio sinistro;
- SW_TC_OPP_DX, collega al circuito di binario dal de-

TABELLA 1- TABLE 1

LINKSET E RELATIVE PORTSET FRA OGGETTI TC E SW
LINKSET AND PORTSET BETWEEN TC AND SW OBJECTS

Linkset	da/a from/to	a/da to/from
1	TC_DX	TC_SX
2	SW_DX	SW_DX
3	TC_SW_OPP_SX	SW_TC_OPP_SX
4	TC_SW_OPP_DX	SW_TC_OPP_DX
5	TC_SW_RUN_SX	SW_TC_RUN_SX
6	TC_SW_RUN_DX	SW_TC_RUN_DX

From the point of view of the logical connections the two types of switches are between them doubly mirror symmetrical, i.e. referring to the figure 7 the quadrant in the top to the left and that in the bottom to the right are symmetrical between them as regards an horizontal axis. The same result is true for the quadrants in the top to the right and in the bottom to the left. This property allows to halve the needed Port-sets and so a time saving during the class design but most of all it will allow a less expensive construction of a generic Top Level, because it is sufficient the individuation of the Right and Left Switches to connect the only 6 Link-sets used in the Top level. These Link-sets connect the explained Port-sets and are shown in table 1.

For the Track Circuit class the following Port-sets were designed:

- TC_DX connects to Track circuit to the right;
- TC_SX connects to Track Circuit to the left;
- TC_SW_OPP_DX connects to *opposed* (not passable) right Switch;
- TC_SW_OPP_SX connects to *opposed* (not passable) left Switch;
- TC_SW_RUN_DX connects to *running* (passable) right Switch;
- TC_SW_RUN_SX connects to *running* (passable) left Switch.

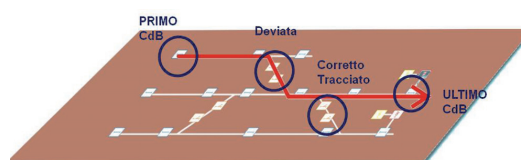


Fig. 8 – Input di una funzione itinerario nel modello. *Route function input in the model.*

(PRIMO CdB = FIRST TRACK CIRCUIT - Deviate = Reverse switch - Corretto Tracciato = Normal switch - ULTIMO CdB = LAST TRACK CIRCUIT).

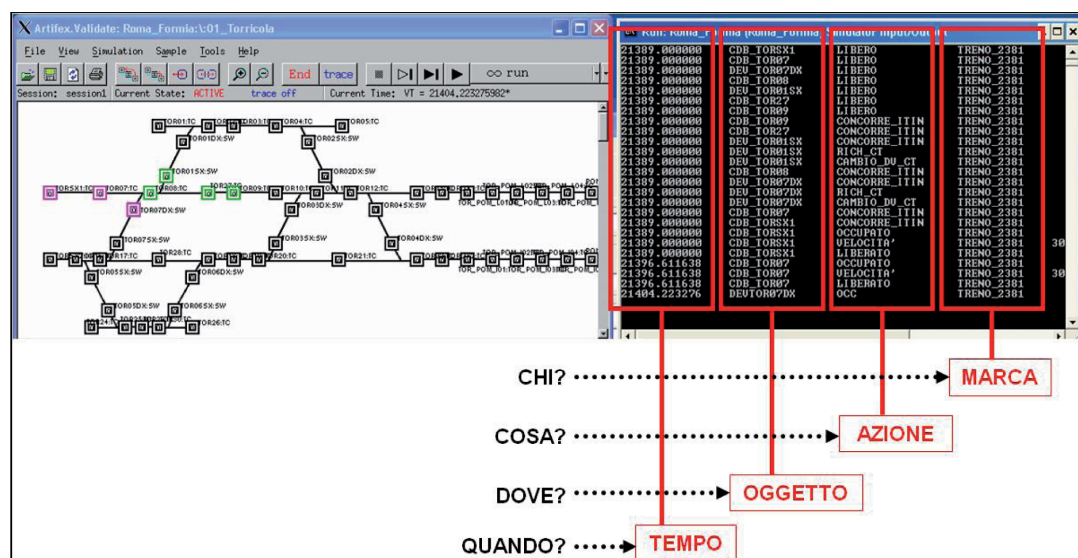


Fig. 9 – Finestra di simulazione e file di output. *Simulation window and output file.*

(CHI? = WHO? - COSA? = WHAT? - DOVE? = WHERE? - QUANDO? = WHEN? - MARCA = TOKEN - AZIONE = ACTION - OGGETTO = OBJECT - TEMPO = TIME).

viatoio destro opposto (non percorribile);

- SW_TC_OPP_SX, collega al circuito di binario dal deviatoio sinistro opposto (non percorribile);
- SW_TC_RUN_DX, collega al circuito di binario dal deviatoio destro "di corsa" (percorribile);
- SW_TC_RUN_SX, collega al circuito di binario dal deviatoio sinistro "di corsa" (percorribile).

Infine, nella Sotto-Rete LAYOUT del Top Level è presente anche una pagina, che rappresenta i collegamenti fra alcuni oggetti della linea e il DCM; le marche treno uscenti dagli oggetti di classe TC della linea vengono intercettate dal DCM che le confronta, assegna priorità e le restituisce ad altri oggetti di classe TC posti all'ingresso della stazione successiva. Ovviamente i collegamenti con gli oggetti sono assicurati da altrettanti specifici Linkset.

Funzioni itinerario

Costruito nella maniera descritta il LAYOUT è possibile completare il Top Level con l'individuazione e la costruzione degli itinerari.

Nel modello per itinerario s'intende non l'intero percorso di un treno ma un tratto di questo percorso delimitato da due circuiti di binario e per il quale viene richiesta un'apposita autorizzazione al movimento. Un percorso di un treno sarà la composizione di più itinerari, ciascuno con il circuito di binario in uscita coincidente con quello in entrata nel

Likewise, for the Switch class as many Port-sets were designed. It is important that this class, for the way which is projected, for each physical switch there are two objects Switches named Right and Left Switch. The Port-sets are the following:

- SW_DX connects to the Right Switch;
- SW_SX connects to the Left Switch;
- SW_TC_OPP_DX connects to the Track circuit from the *opposed* (not passable) Right Switch;
- SW_TC_OPP_SX connects to the Track circuit from the *opposed* (not passable) Left Switch;
- SW_TC_RUN_DX connects to the Track circuit from the *running* (passable) Right Switch;
- SW_TC_RUN_SX connects to the Track circuit from the *running* (passable) Left Switch.

Finally, in the LAYOUT sub-net of the Top Level it is available a page representing the links among some lines objects and DCM; the train token leaving the TC class objects are intercepted by DCM, which compares them, define priorities and gives back them to other TC class objects located at the entering door of next station. Obviously the links with the objects are ensured by specific Link-sets.

On the basis of the built up LAYOUT it is possible to complete the Top Level with routes identification and building.

successivo. Dunque nel modello un itinerario è univocamente identificato da tre informazioni, mostrate in fig. 8: il circuito di binario di partenza, quello di arrivo e le posizioni di tutti i deviatoli necessari a comporre l'itinerario stesso.

Occorre dunque provvedere alla realizzazione di tante pagine quanti sono gli itinerari possibili (o voluti) in cui ricevere le informazioni di via libera o via impedita e in cui collegare i circuiti di binari in entrata e in uscita dall'itinerario; in più bisogna utilizzare almeno una pagina per ogni stazione per definire la posizione di tutti i deviatoli ed un'altra in cui l'informazione di annullamento degli itinerari possa essere "spedita" ad ogni oggetto della stazione. Le pagine degli itinerari sono simili a quella mostrata più avanti (fig. 11), quando verrà mostrato un esempio del funzionamento del modello; differiscono fra loro solo per la classe di oggetto da cui entrano le marche, che può essere TT o TC, a seconda che l'itinerario sia d'ingresso in linea o successivo ad un altro. In più, per comodità di realizzazione, più itinerari possono essere uniti in una stessa pagina, dividendosi successivamente in quella che gestisce la posizione dei deviatoli.

3.6. Output della simulazione

L'output della simulazione è composto da una finestra di simulazione e da un file di testo.

Nel simulatore è possibile selezionare ogni Pagina desiderata e controllare il passaggio delle marche e l'evoluzione del sistema attraverso i valori assunti dalle variabili di sistema e dai colori che sono stabiliti in fase di costruzione delle reti.

Nel file di testo ci sono le informazioni volute dall'analista e in particolare fondamentali risultano il tempo di simulazione, l'identificativo dell'oggetto in cui si compie un'azione, l'azione stessa e la marca che la compie.

La fig. 9 mostra un istante della fase di simulazione, con sulla sinistra della finestra un treno che percorre un tratto di linea (in fucsia) bloccato (in verde) e a destra gli eventi di questo passaggio trascritti nel file di testo.

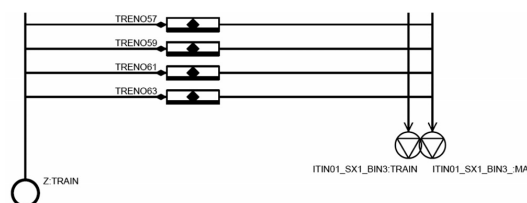


Fig. 10 – Parte della Sotto-rete a_01_Torricola di classe TT. Part of the Subnet a_01_Torricola of the class TT.

In the model the route is not the whole path of a train within the station, but just the one determined by two track circuits, for which a specific Movement Authority is required. A train path will include more routes, linked by final and initial track circuits. Therefore in the model one route is univocally identified by three points highlighted in figure 8: initial and final track circuits and positions of all switches necessary to compose the route itself.

It is necessary to prepare as many pages as many the possible routes are, where to receive freedom or occupation information and to connect entering and exiting track circuits; moreover it is necessary to use at least one page per station, where the switches positions are defined, and another one, where the route cancellation can be addressed to each station object. The routes pages are similar to those shown in figure 11, where the model operation is exemplified; the objects classes of entering tokens maybe TT or TC, according to the route location at the start or along the line. Moreover, many routes may be represented in a single page by just differentiating them in the pages where the differences by switches positions are highlighted.

3.6. Simulation output

The simulation environment includes a visual simulator and an output text file.

In the simulator it is possible to select every page or subnet and to check the tokens movements and the system evolving through the colors set up during net building.

In the text file is included the information desired by programmer and in particular the time of simulation, the ID object code, the action and the token.

In figure 9 it is possible to see a simulation window: a station (Torricola) during the running (fuchsia objects) of a train on a locked route (green objects).

4. Model functioning: running example

4.1. The train in the subnet

In order to better show the potentiality of this new module as decision support tool for the increase of traffic regularity, the *journey* of a train across some subnets, with or without the automatic overtaking management will be shortly described.

4.2. "Normal" mode

Step 1: Train generation

The train journey starts in an object of Timetable class, in the sub-net that represents the train generation.

4. Funzionamento del modello: esempio di percorrenza

4.1. Il treno nelle Sotto-reti

Per illustrare il comportamento d'insieme dei vari componenti finora illustrati, in questo capitolo è brevemente descritto il "viaggio" di un treno all'interno di alcune Sotto-reti, in assenza e in presenza dell'oggetto che gestisce le precedenza automatiche.

4.2. Modalità "normale"

Passo 1: Generazione del treno

Il viaggio del treno inizia nella Sotto-rete dell'oggetto ORARIO che gestisce la generazione dei treni.

In particolare nel posto START viene posta una "marca di partenza" che attiva la transizione con azione (lato destro in grassetto) GEN_TRENI in cui si assegna un numero progressivo alla marca in uscita verso il posto TRENI_GEN. A questo punto lo scatto della transizione ZZ permette l'indirizzamento verso le Sotto-reti in cui si definiscono le variabili del treno, collegate con il posto comune Z e la chiusura del ciclo verso il posto START che fa ricominciare la generazione per il treno successivo. In figura si nota come la transizione ZZ abbia sia scritte di codice in azione (lato destro in grassetto) sia in predicato (rombo nero all'interno). L'azione serve a copiare il progressivo verso i posti di output, mentre il predicato ad arrestare ad un numero voluto la generazione di treni. In più è stata inserita una transizione concorrente WW, nel caso non si vogliano far partire tutti i treni, ad esempio per verificare cosa accade fra due treni che partono uno dopo l'altro, così da non dover ripetere tutta la simulazione.

Passo 2: "Costruzione" del treno

Sempre nell'oggetto ORARIO, il posto comune Z è collegato con tutte le sottoreti in cui sono inseriti i treni che partono dalle varie stazioni.

In fig. 10 è rappresentata una porzione della sottorete contenente i treni che partono dalla stazione di Torricola. In questo caso per partenza s'intende l'entrata nel modello e non la generica partenza di un treno da una stazione. Quindi è più corretto dire che in fig. 10 vi sono i treni entranti in linea dalla stazione porta di Torricola.

Supponiamo ora di voler simulare il viaggio del treno EuroStar 9373, diretto da Roma a Formia, sessantunesimo treno dalle 7 del mattino di un giorno feriale.

Dal posto Z le varie marche generate, che per ora sono ciascuna una variabile STRUCT con inizializzato solo un numero progressivo al posto dell'ID_TRENO, possono entrare in una delle transizioni in figura. In particolare un pre-

In particolare in the place START (see figure A3 in Appendix 2) there is a starting token that activates the GEN_TRENI transition with actions (right side in bold), in which a progressive number of tokens is placed out in TRENI_GEN. At this point the firing of ZZ transition allows the addressing towards the subnets that define the train variables, linked with the Z common place (in bold), and the cycle closing toward START place so that the next train can be generated. The same figure shows the ZZ transition have code script both in action window (right side in bold) and in predicate (diamond inside the rectangle). The action copies the progressive number from input to output places and the predicate stops the train generation after a fixed looping number. Moreover the concurrent WW transition is used if it is not required the start of all trains, for instance to check what happens between two nearby trains starting one after the other, so it is not necessary to repeat the whole simulation.

Step 2: Train "building"

In the ORARIO object, the Z common place is linked with all subnets in which the starting trains from different stations are.

In figure 10 a small part of the sub-net is represented. In this case "start" means the entrance to the model and not the generic train starting from the station, which is in any case managed inside the Top Level. Therefore it is more correct to say that in figure the next trains coming in the Roma-Formia line at Torricola station are represented.

It is now supposed to simulate the journey of the ES train 9373, from Roma to Formia, 61st train running starting at 7.00 of a working day.

From the Z place the different tokens so far generated, that for now are STRUCT variables initialized with a progressive number like ID_TRAIN only, can run into the transitions in figure. In particular a predicate for each transition identifies the progressive number and captures the right token. In our example the token with ID_TRAIN = 61 will activate the transition TRENO61 in figure. This transition does not fire immediately, but with a firing delay correspondent to entry time in the Roma-Formia line of ES train 9373. After this time the transition will fire performing all its actions, i.e. the train *building* through the initialization of the variables to put in the TRAIN token and the MA token. In particular the software will change the ID_TRAIN from 61 to 9373, will insert the ES category in CAT variable, the direction "towards Formia", the entry location (in meters) along the line and the information about routes that the train will be run along the line itself.

Now the two tokens TRAIN and MA will be released to the output places of Timetable class shown in figure 4 (with a reverse triangle inside a circle). These places are

dicato per ogni transizione individua il numero progressivo e cattura la marca “giusta”. Nel nostro esempio la marca con ID_TRENO uguale a 61 attiverà la transizione TRENO61 in figura. Tale transizione non scatterà immediatamente ma ha inserito un ritardo corrispondente all’orario di entrata nella linea dell’EuroStar 9373. Passato tale tempo, la transizione scatterà, compiendo tutte le azioni in essa scritte, ovvero la “costruzione” del treno attraverso l’inizializzazione delle variabili da inserire nella marca TRENO e nella marca AM. In particolare il programma cambierà l’ID_TRENO da 61 a 9373, inserirà la direzione “verso Formia”, la progressiva d’ingresso nella linea, la categoria del treno e l’informazione su quali itinerari dovrà seguire in ingresso e in uscita nelle stazioni che incontrerà lungo la linea.

A questo punto le due marche, TRENO e AM, verranno depositate nei posti di esterni di output presenti in figura (con un triangolo all’interno). Questi sono collegati sul TOP LEVEL alla prima funzione itinerario che il treno 9373 dovrà percorrere.

Passo 3: Primo itinerario del treno

Sul TOP LEVEL, come detto, sono disegnati il layout della linea e tutte le funzioni itinerario che i vari treni dovranno utilizzare per poter compiere il loro viaggio. Ricordiamo che una funzione itinerario è univocamente determinata una

linked on the Top Level to the first route function that train 9373 will run.

Step 3: First route of train

On the Top Level, like anticipated, all the route functions that the different trains must use to run their trip on the line are stored. A route function is univocally defined when the following information are available: first track circuit, last track circuit and position (normal or reverse) of all the switches between these track circuits. On the Top Level all the objects are represented through two concentric squares and the class is named as <ObjectName:ClassName>.

The page in figure 11 manages the first route of train 9373, i.e. the entry route in Torricola station. From the object START of the class Timetable the tokens TRAIN and MA, being in the previous output places, enter in two input places of the class Top Level shown in figure (two concentric circles) linked to the START object. The ways of two tokens originate here.

The MA token enters in the subnet shown in figure 5 through the input place ITIN01_SX1_B3_ and, if it satisfies the condition to have the same route code of this sub-net, activates the ITIN01SX1B3_ transition. This cre-

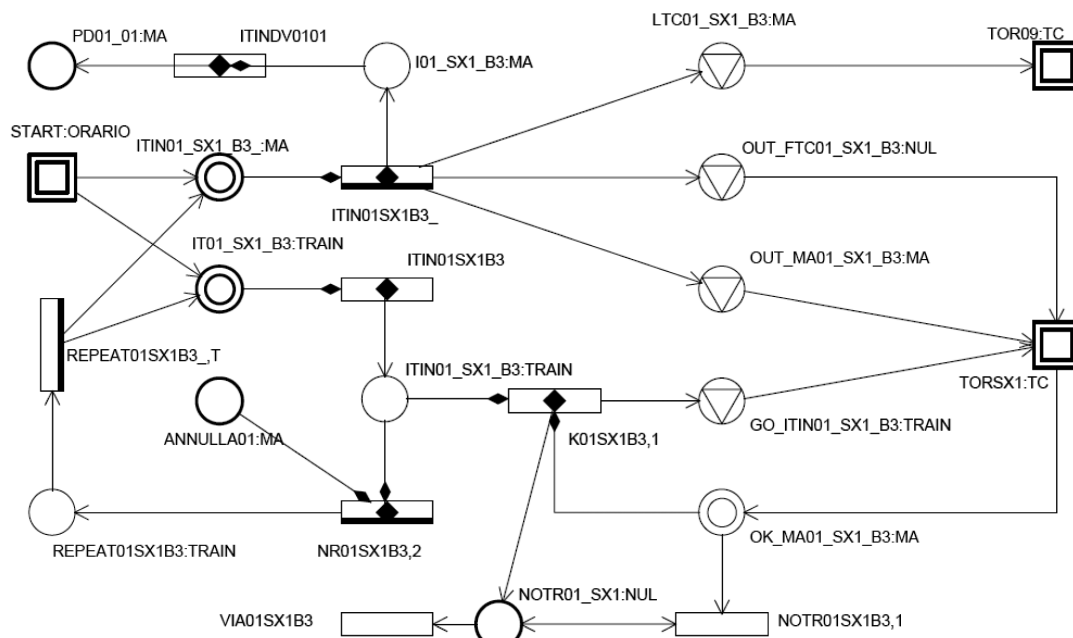


Fig. 11 – Pagina SX1B3 della Sotto-rete IT_01 del TOP LEVEL. SX1B3 page of the IT_01 sub-net of the TOP LEVEL.

volta noti: primo circuito di binario, ultimo circuito di binario e posizione dei deviatori fra questi due circuiti di binario. Sul TOP LEVEL tutti gli oggetti sono rappresentati con due quadrati concentrici e la classe è indicata nel modo seguente: <<NomeOggetto:NomeClasse>>.

La Pagina in fig. 11 gestisce il primo itinerario che il treno 9373 dovrà compiere, ovvero quello in entrata nella stazione di Torricola. Dall'oggetto ORARIO della classe TT le marche TRENO e AM presenti nei posti di output visti in precedenza entrano nei due posti di input della classe TOP LEVEL in figura (due cerchi concentrici) collegati con l'oggetto ORARIO. A questo punto le strade delle due marche si dividono.

Analizziamo prima il percorso della marca AM. Questa marca entra nella sottorete in figura nel posto di input ITIN01_SX1_B3_ e, se soddisfa la condizione di avere il codice itinerario corrispondente a quello della sottorete, attiva la transizione ITIN01SX1B3_. Questa transizione genera in output tre marche AM uguali a quella di input ed una di tipo NUL, ognuna con una precisa funzione.

La prima, che poi giunge al posto comune PD01_01, ha il compito di andare in un'altra Pagina della Sotto-rete IT_01 in cui sono presenti i deviatori che il treno incontrerà lungo l'itinerario. In tale Pagina la marca AM, che possiede il codice dell'itinerario che il treno 9373 deve percorrere per entrare nella stazione, ordinerà la posizione di tutti i deviatori che l'itinerario incontra.

Sempre con riferimento alla figura 11 la seconda marca generata dalla transizione ITIN01_SX1_B3_ viaggia, attraverso il posto di output LTC01_SX1, verso l'ultimo circuito di binario dell'itinerario, in questo caso TOR09, entrando nella sottorete in cui si effettua la verifica di libertà dell'ente per comunicare l'arresto della verifica all'ultimo circuito di binario dell'itinerario.

La terza e quarta marca uscenti da ITIN01_SX1_B3_ sono invece indirizzate verso il primo circuito di binario TORSX1 e hanno due diverse funzioni: la terza, di tipo NUL e uscente dal posto di output OUT_MA01_SX1_B3, entra nella sottorete di TORSX1 che effettua il bloccamento dell'ente per arrestarlo, in quanto primo circuito di binario dell'itinerario, dando origine alla procedura di percorrenza del treno, la quarta marca, di tipo AM e uscente dal posto di output OUT_MA01_SX1_B3, inizia la procedura di verifica di libertà dell'itinerario dal primo circuito di binario TORSX1.

Contemporaneamente la marca TRENO sarà arrivata al posto semplice ITIN01_SX1_B3 e resterà in attesa che si verifichino uno dei due eventi simulati dalle transizioni di cui il posto è input: l'autorizzazione a procedere con la percorrenza o l'annullamento e il ripetersi di tutta la procedura descritta in questo passo 3. Nel primo caso dal primo circuito di binario sarà uscito nel posto esterno di input OK_MA01_SX1_B3 una marca con lo stesso ID_TRENO di quello in attesa in ITIN01_SX1_B3, il che significa che il primo ente dell'itinerario, e dunque anche tutti gli altri, è bloccato e pronto per ricevere il treno. A questo punto scatterà la transizione K01SX1B3 e, attraverso il posto esterno di out-

ates in its output places three MA tokens and one NUL token, each one with a particular job.

The first one arriving to the common place PD01_01, must enter another sub-net of the Top Level, where all switches encountered by the train along the route are. In this sub-net the token MA, having the entry route code in Torricola station of train 9373, will order the position of all switches included in this route.

Referring to the same figure, the second token created by ITIN01SX1B3_ transition goes, through the output place LTC01_SX1, towards the last track circuit (in this case TOR09), entering in the subnet of elements freedom check to communicate its stop because TOR09 is the last track circuit.

The third and fourth tokens getting out from ITIN01SX1B3_ are instead addressed to the first track circuit TORSX1 and do two different jobs. The third, type NUL, goes out from Top Level at output place OUT_MA01_SX1_B3 and enters in the TORSX1 subnet, that manages the element locking to stop, because TORSX1 is the first track circuit, and to allow in this way the procedure for the train running. The fourth token, type MA, exits from Top Level class by OUT_MA01_SX1_B3 and starts the procedure of freedom check of the TORSX1 track circuit.

At the same time the token TRAIN is arrived at the simple place ITIN01_SX1_B3 waiting for the happening of one between two events simulated by the transitions that this place can activate: the running authorization or the cancellation and the replication of the whole procedure described in step 3. In the first case from the first track circuit a token with the same ID_TRAIN of that waiting in ITIN01_SX1_B3 will be put in the input place OK_MA01_SX1_B3; this means that the first element of route, and therefore everyone, is locked and ready to receive the track occupation by train. Now the transition K01SX1B3 will be fired and, through the output place GO_ITIN01_SX1_B3, the train starts its trip across the subnet of the first track circuit TORSX1.

If, instead, any element of the route will be occupied or locked, so a sub-net of the Top Level will receive this information and, in our example, a MA token will appear in the common place ANNULLA01; then NR01SX1B3 transition will fire and a new procedure for route function, after a fixed time, will start. This process will continue until the running authorization will be received from the first track circuit.

Step 4: Route freedom check

As soon as the token is arrived in the first track circuit, the MA token performs the check of freedom of the element. This event is reported in the output file through a script referring that the track circuit is free and, at the same time, the object in the simulation window takes the blue color during the check. In case the object is not free

put GO_ITIN01_SX1_B3, il treno comincerà il suo viaggio attraverso la sottorete che simula la percorrenza nel primo circuito di binario TORSX1.

Se invece uno qualsiasi degli enti che compongono l'itinerario dovesse risultare occupato o bloccato, allora una Pagina sul TOP LEVEL riceverebbe l'informazione e, nel nostro esempio, una marca apparirebbe nel posto comune AN-NULLA01; ciò farebbe scattare la transizione NR01SX1B3 e inizierebbe il ripetersi, dopo un tempo prestabilito di tutta la procedura. Tale ripetizione continuerebbe fino all'arrivo dell'autorizzazione alla percorrenza nel primo circuito di binario.

Passo 4: Verifica di libertà

Una volta entrata nel primo circuito di binario, la marca AM effettua la verifica di libertà dell'ente. Tale operazione viene visualizzata in sede di output con una scrittura di codice che avverte che il circuito di binario è libero e con l'accensione e lo spegnimento in azzurro dell'oggetto libero. In caso di oggetto non libero la procedura si annulla e anche questo viene segnalato sul foglio di output.

Finita la procedura nel primo circuito di binario, la verifica di libertà continua nell'ente successivo, circuito di binario o deviatore, potendosi spostare le marche fra un ente e l'altro grazie a degli appositi collegamenti fra oggetti, che possono unire sia un singolo posto di output a un singolo posto di input sia due interfacce composte da diversi posti esterni.

Passo 5: Bloccamento

La procedura di bloccamento degli enti ha un funzionamento simile a quello della verifica di libertà ma procede in senso inverso. Infatti, una volta verificata la libertà dell'ultimo circuito di binario, questo è anche il primo che viene bloccato. A ritroso, tutti gli altri verranno bloccati fino al primo. Dal punto di vista del modello la procedura è con la sequenza appena descritta, ma poiché alcuni sistemi di segnalamento e controllo hanno il bloccamento istantaneo di tutti gli enti coinvolti, ciò è facilmente riproducibile non imponendo nessun tempo di scatto alle transizioni che simulano il bloccamento.

L'operazione viene visualizzata con una riga sul foglio di output che segnala che un certo treno ha bloccato un ente e con il colore verde che resta acceso durante tutto il tempo in cui l'oggetto non è libero.

Una volta bloccato da una marca AM con un certo ID_TRENO, un ente non può più essere interessato da marche diverse e annullerà qualsiasi operazione eccetto quella di percorrenza di una marca TRENO con lo stesso ID_TRENO.

Passo 6: Percorrenza

Con l'intero itinerario bloccato, l'operazione di percorrenza può iniziare e dunque la marca TRENO può entrare

the procedure is cancelled and also this event is reported on the output text file.

As soon as the first track circuit procedure is completed, the check of freedom proceeds to the following element: track circuit or switch; in fact the token can move itself from an element to another because there are specific Link-sets between objects that can connect both single external input-output places and two Port-sets objects composed by different external places.

Step 5: Route interlocking

The element locking procedure is similar to the freedom check but moves in opposite direction. In fact, after the verification that the last track circuit is free, the first track circuit is locked too. Backwardly each previous element will be locked until the first track circuit of the route. This sequence describes the procedure, nevertheless some signaling and control systems lock instantly all the elements: this event is easily reachable because it is sufficient not to impose a firing delay to the transitions that simulate the locking.

The event is reported with a line on the output file, with the signal that the train has locked the element and with the green color applied to the object until it is released.

When a MA token with an ID_TRAIN locks the element, it cannot be used anymore by different tokens and all operations are cancelled, except the running of a TRAIN token with the same ID_TRAIN.

Step 6: Train running

While the whole route is locked, the running procedure can start and the TRAIN token can enter in the principal subnet of the classes Track circuit and Switch managing the train running. In this sub-net the TRAIN token receives the information about its position in the railway line (length of block section where it runs and line kilometric progressive) and is addressed to a specific subnet managing the train kinematics, where it receives the information on the average speed, different for each train and each block section. In the model this speed is realized through a firing delay of a transition that simulates the running time of the train over the block section protected by that track circuit or over the switch.

The element running is completed when the TRAIN token goes out from element and enters in the next track circuit or switch; this procedure continues up to the last track circuit of the route.

During the running, reported on the output file with train speed information, the object takes the fuchsia color.

nella Sotto-rete di circuito di binario e di deviatoio che gestisce la percorrenza. In questa la marca TRENO riceve l'informazione sulla sua posizione nella linea, ovvero la lunghezza della sezione di blocco che sta attraversando e la progressiva a cui si trova, e viene indirizzato in alcune specifiche sottoreti che gestiscono la cinematica del treno, ovvero riceve l'informazione della velocità media che deve tenere in quella sezione di blocco. Nel modello tale velocità si traduce in un ritardo di scatto di una transizione che simula così il tempo di percorrenza del treno sulla sezione di blocco protetta da quel circuito di binario o su un deviatoio.

La percorrenza sull'ente si conclude con l'uscita della marca TRENO che entra nel circuito di binario o deviatoio successivo, fino all'ultimo circuito di binario dell'itinerario.

Durante tutta la percorrenza, segnalata nel file di output con l'indicazione della velocità che il treno ha sulla sezione di blocco, l'oggetto assume il colore fucsia.

Passo 7: Chiamata dell'itinerario successivo

Quando la marca TRENO ha percorso uno degli ultimi circuiti di binario, variabile da tratta a tratta, dall'oggetto TC corrispondente parte un collegamento alla Sotto-rete del TOP LEVEL in cui è disegnato l'itinerario successivo. Nell'esempio che abbiamo fatto per il treno 9373, l'itinerario successivo al primo è rappresentato da una Pagina simile a quella della figura 19 ma con l'oggetto TOR19, ultimo circuito di binario nel primo itinerario, che ora svolge le funzioni dell'oggetto ORARIO.

Passo 8: Liberazione

Finita la percorrenza, dopo un prefissato tempo transitorio, oppure una volta annullata la procedura di bloccamento, l'ente viene sbloccato e può essere utilizzato anche dalle altre marche.

I passi sopra esposti, ripetuti su tutta la linea, portano il treno a destinazione.

4.3. Modalità con il sistema automatico di precedenza

Il modello può funzionare anche attivando l'oggetto DCM che gestisce il sistema automatico di precedenza. Come abbiamo visto, in tale oggetto viene confrontato ciascun treno con quello che lo precede e con quello che lo segue per verificare se una precedenza deve essere inserita in linea.

Il sistema funziona nel modo di seguito descritto. I circuiti di binario rappresentanti le banchine delle stazioni sono collegati al DCM in maniera tale che quando un treno esce da tale oggetto TC, oltre ad entrare nel successivo entra anche in una Sotto-rete del DCM che smista la marca TRE-

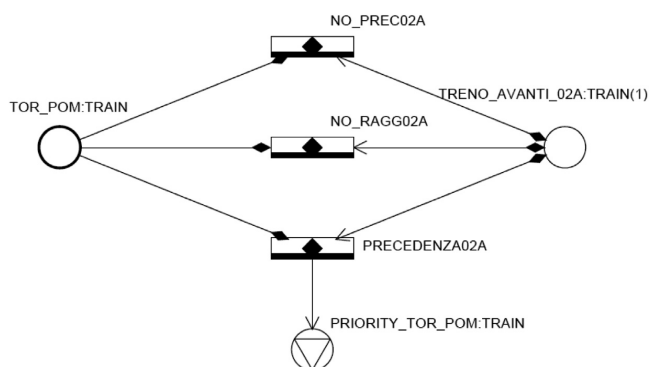


Fig. 12 – Porzione della Sotto-rete TOR_POM della classe DCM. Part of Subnet TOR_POM of the class DCM.

Step 7: Next route call

As soon as the TRAIN token runs to the last track circuit, from this object a link connects to the Top Level subnet in which is drawn the next route. In the example for the train 9373, the second route is represented from a sub-net similar to that in figure 5, but with the object TOR19, last track circuit of the first route, that now executes the function of ORARIO object.

Step 8: Locking release

After the train running on an element, after a fixed time, or if the locking procedure is cancelled, the element is un-locked and can be used by other tokens.

The steps above, replied along the whole railway line, carries the train to the final destination.

4.3. Automatic overtaking mode

The model can function also activating the DCM, that manages the automatic system of overtaking. In this object each train is compared with the trains running immediately before and after, in order to verify if and when an overtaking is necessary.

The track circuits corresponding to platforms of stations are linked to Movement Inspector, in such way that, as soon as the TRAIN tokens exit from this track circuit and enter in the next one, they are intercepted and move in a sub-net of Movement Inspector, that sorts the different TRAIN tokens in other sub-nets depending on direction and a kilometric progressive of the line. These subnets, similar to that shown in figure 12, perform the comparison between trains.

In particular the net in figure 12 manages all the

NO in un'altra sottorete dipendente dalla direzione e dalla progressiva a cui il treno si trova. Quest'altra Sotto-rete, del tipo di quella mostrata in fig. 12, effettua il confronto fra i treni.

In particolare, quella in fig. 12 gestisce tutti i passaggi dei treni fra le stazioni di Torricola e Pomezia in direzione Formia. La marca TRENO intercettata in linea entrerà nella Pagina raffigurata nel posto comune TOR_POM. Nel posto TRENO_AVANTI02A è presente, perché entrata in precedenza, la marca corrispondente al primo treno che lo precede. A questo punto le transizioni in figura effettuano il confronto secondo i seguenti criteri:

- la transizione NO_PREC02A scatta se il treno che segue non ha diritto di precedenza; questo può accadere se il treno che precede è di categoria superiore o uguale, oppure se il treno che precede è più veloce di quello che segue, oppure è previsto che il treno che precede percorra nella prima stazione successiva un itinerario che lo porta ad un binario diverso da quello principale della linea;
- la transizione NO_RAGG02A scatta se, pur avendo diritto di precedenza, il treno che segue non può raggiungere il treno che precede prima della stazione successiva;
- la transizione PRECEDENZA02A scatta se il treno che segue ha diritto di precedenza e ha una velocità relativa rispetto a quello che precede tale da poterla raggiungere prima della stazione successiva.

La condizione di raggiungimento del treno che precede da parte del treno che segue è così espressa:

$$t_s < t_p + d \times \left(\frac{1}{v_p} - \frac{1}{v_s} \right)$$

con t_s istante di arrivo del treno che segue nel punto di conflitto, t_p istante di arrivo del treno che precede nel punto di conflitto, d distanza del punto di conflitto dalla stazione successiva, v_p velocità del treno che precede, v_s velocità del treno che segue.

Lo scatto di una delle tre transizioni provoca la sostituzione nel posto TRENO_AVANTI02A della marca TRENO che vi si trovava prima dello scatto con la marca TRENO che prima si trovava in TOR_POM; in questa maniera quest'ultimo sarà pronto per il confronto con il treno successivo.

Lo scatto della transizione PRECEDENZA02 genera anche una marca TRENO nel posto esterno di output PRIORITY_TOR_POM. Tale marca rappresenta il treno che precede e ha inserito nella variabile PR un ordine di dare precedenza. Il posto esterno di output è collegato sul TOP LEVEL ad un circuito di binario in entrata alla prima stazione successiva, in questo caso quella di Pomezia. Nel circuito di binario la nuova marca TRENO sostituisce quella che deriva dalla percorrenza in linea con l'ordine di cambiare itinerario d'ingresso e di uscita a Pomezia e attendere il passaggio del treno che lo segue. In questa maniera la precedenza automatica sarà stata completata.

train running between two stations (Torricola and Pomezia) in a direction (towards Formia). The TRAIN token intercepted in line arrives in the common place TOR_POM. In the TRENO_AVANTI02A place is the TRAIN token corresponding to the train ahead. At this point the three transitions are compared according to the following criteria:

- the transition NO_PREC02A fires if the train behind has not right of way; this event happens if the train ahead is of higher category, if it runs with a higher average speed or if it is going to run in the next station a secondary route, different from the main line;
- the transition NO_RAGG02A fires if, though it has right of way, the train behind does not reach the train ahead before the next station;
- the transition PRECEDENZA02A fires if the train ahead has a right of way and is fast enough to reach it before the next station.

The reaching condition of the train ahead by train behind is the following:

$$t_s < t_p + d \times \left(\frac{1}{v_p} - \frac{1}{v_s} \right)$$

with t_s transit time of the train behind in the interception section, t_p transit time of the train ahead, d distance from interception section to the next section, v_p average speed of the train forward, v_s average speed of the train behind.

The firing of one of the three transitions causes in the TRENO_AVANTI02A place the substitution of the TRAIN token with that coming from TOR_POM place; this last token is now ready for the comparison with the next train.

The transition firing of PRECEDENZA02, like the figure 7 shows, creates also one TRAIN token in the output place of the class Movement Inspector named PRIORITY_TOR_POM. This token represents the train behind and has a value in the PR variable that means "give overtaking". The output place is linked on the Top Level to the track circuit located immediately before the next station (in this case Pomezia). In the track circuit the new TRAIN token substitutes that arriving from the line running with the order of changing the routes inside the Pomezia station and waiting for the running of the train behind. According to this procedure the automatic overtaking is performed.

This control mechanism is active on all the stations, in both directions and on two-way operated tracks also.

A last fundamental consideration is that the overtaking criteria above are ever modifiable; this means creating more scenarios with different automatic overtaking systems, and it is possible to choose the better criteria,

È importante far notare come questo meccanismo di controllo è attivo su tutte le stazioni, in entrambi i sensi di marcia e può funzionare anche su binari banalizzati.

Un'ultima fondamentale considerazione è che i criteri di precedenza prima descritti sono in ogni momento modificabili; ciò vuol dire che si possono creare più scenari con diversi sistemi automatici di precedenza e verificare quale fra questi minimizzi il ritardo medio o raggiunga un altro obiettivo desiderato.

5. Applicazione del modello nelle analisi di regolarità

5.1. Il caso di studio

In questo capitolo verranno descritte le simulazioni effettuate per la calibrazione del modello.

Gli esempi e le figure mostrate da adesso in poi riguardano il caso di studio affrontato, la linea Roma-Formia, comprendente 10 stazioni, centinaia di itinerari, quasi 700 enti fra circuiti di binario e deviatoi e con 63 treni "costruiti" per rappresentare l'orario di esercizio dalle 7 alle 10. La linea si è ben prestata alla calibrazione del modello sia per le criticità nella regolarità dell'esercizio sia perché presenta alcune complessità come la presenza di più linee, in quanto vi sono due diramazioni verso Terracina e verso Nettuno, o come l'utilizzo di alcuni binari in entrambi i sensi di marcia.

Nel modello sono stati rappresentati tutti i Posti di blocco presenti nel fascicolo di linea di RFI n.117 con altrettanti oggetti di classe TC. Sono inoltre stati rappresentati con la stessa classe i circuiti di binario presenti nei pressi delle stazioni, mentre per ogni deviatoio sono stati utilizzati due oggetti della classe SW.

Gli itinerari rappresentati non sono tutti quelli possibili sulla base dei piani schematici delle stazioni ma solo quelli effettivamente utilizzati sulla linea con l'aggiunta di quelli necessari a garantire la possibilità di precedenza, operabile in tutte le stazioni.

I parametri considerati da inserire come input agli oggetti e alle marche sono anch'essi stati rilevati dal fascicolo di linea per quel che riguarda lunghezze, ranghi di velocità e gradi di frenatura delle singole tratte. Per le caratteristiche dei veicoli, quali peso frenato, accelerazione, decelerazione, si sono utilizzati valori medi per classi di treno.

I treni da simulare sono stati individuati fra quelli circolanti fra le 7 e le 10 di mattina, incrociando i dati dei moduli M53 e M42 messi gentilmente a disposizione da RFI.

È importante sottolineare come il modello è stato dotato di Sotto-reti basate su principi di funzionamento della circolazione e indipendenti dalla tecnologia degli apparati rappresentati. Per questo, con piccole accortezze relative alla gestione dei tempi fra liberazione elastica e liberazione

which minimizes the average delay of the trains or that reaches other goals.

5. Model application for regularity analysis

5.1. Case study

In this chapter the simulations operated on the case study for the model validation are described.

The examples and the figures concern the Roma-Formia line, including 10 stations, hundreds of routes, almost 700 elements (switches and track circuits) and 63 trains representing the peak period (7.00-10.00) timetable. The line was a well suited validation test, both for the critical traffic conditions and some complexity arising from the operated branch lines to Terracina and Nettuno and the possible bidirectional use of tracks.

In the model all line block sections and station track circuits are represented, as well as a couple of objects of the SW class correspond to one switch.

The represented routes are just those regularly operated in the stations or anyway required for possible overtaking manoeuvres.

Objects and tokens inputs (lengths, speeds, slopes of single sections) have been extracted by official RFI documents, as well as mean trains parameters (braking weights, accelerations, decelerations) have been accounted, with reference to trains running in the peak period resulting from official M53 and M42 RFI modules.

The model was organised in sub-nets based on operational principles independently upon the technology of the represented lay-out. For this reason, with minor adjustments concerning the managements of elastic or rigid release times, it is possible to use the same objects (e.g. a TC object) within the representation of a train running with various station interlocking and line block systems.

The line model requires the management of 654 objects, 250 pages, 2796 link-sets, 966 external places, 459 simple places and 1514 transitions with 3736 links.

5.2. Simulation of planned timetable in the peak period

Simulation period

The first validation set of simulations reproduced the planned timetable for each intermediate station from Torricola to Itri.

In order to test the model in the heaviest traffic conditions, the first analysis was devoted to identify the peak period. The analysis highlighted (figure 13) a typical daily

rigida, è possibile ad esempio, con uno stesso oggetto di classe TC, gestire la circolazione di un treno sottoposto a controllo da diverse tipologie di apparato centrale o in linea dal blocco automatico.

Per dare un'idea della mole di dati gestiti nel programma, si consideri che per rappresentare l'intera linea sono stati necessari 654 oggetti, 250 pagine, 2796 linkset, 966 posti esterni, 459 posti semplici e 1514 transizioni, questi ultimi collegati con 3736 archi.

5.2. Simulazione dell'orario teorico nelle ore di punta

Periodo di simulazione

La prima serie di simulazioni effettuate per calibrare il modello è stata quella di riprodurre l'orario teorico, stazione per stazione, ricomponendo i dieci M53 delle stazioni da Torricola fino ad Itri.

Per poter testare il modello nelle condizioni più gravose, la prima analisi svolta ha riguardato l'individuazione delle ore con maggiore densità di traffico. L'analisi ha rivelato (fig. 13) la tipica distribuzione del numero medio di treni in linea lungo l'arco di una giornata.

La figura mostra valori superiori a 14 treni/ora negli intervalli 5-6, 7-8, 20-21 e 21-22, proponendo densità di traffico simili fra le ore di punta diurne e quelle serali. Facendo invece riferimento al solo traffico passeggeri, le ore di punta diurne hanno valori più elevati, con oltre 13 treni/ora fra le 7 e le 9.

Per la determinazione del periodo a maggiore traffico si sono prese in considerazione le tre ore con il maggiore numero medio di treni passeggeri (fig. 14), trascurando in questo modo il traffico merci. Infatti nel M53 sono presenti tutte le tracce teoriche dei treni, comprese quelle di quei treni merci che effettuano il servizio solo quando la traccia viene effettivamente acquistata dalle imprese ferroviarie. Peraltro le varie tracce vengono acquistate in maniera indipendente l'una dall'altra e ciò rende altamente improbabile il verificarsi della densità di traffico merci esposta in fig. 19.

La figura mostra come, per il solo traffico passeggeri, l'intervallo orario

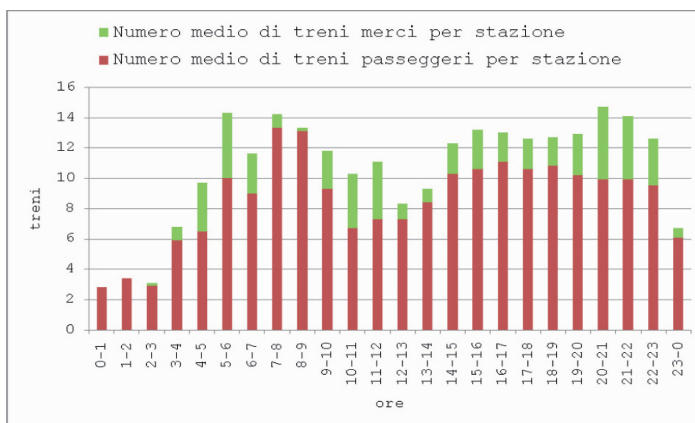


Fig. 13 - Numero medio di treni sulla tratta Torricola-Formia. *Mean number of trains on Torricola-Formia sections.*

(treni = trains - ore = hours - Numero medio di treni merci per stazione = Mean number of freight trains per station - Numero medio di treni passeggeri per stazione = Mean number of passengers trains per station).

distribution of traffic.

More than 14 trains/hour are planned in the periods 5.00-6.00, 7.00-8.00, 20.00-21.00 and 21.00-22.00, that means similar traffic density in morning and evening peak periods. For the passengers trains only the traffic is higher in the morning period 7.00-9.00.

For this study the highest flows have been considered in the 3 hours with the maximum mean number of passengers trains (figure 14). In fact the planned freight trains traces are often not operated by the railway under-

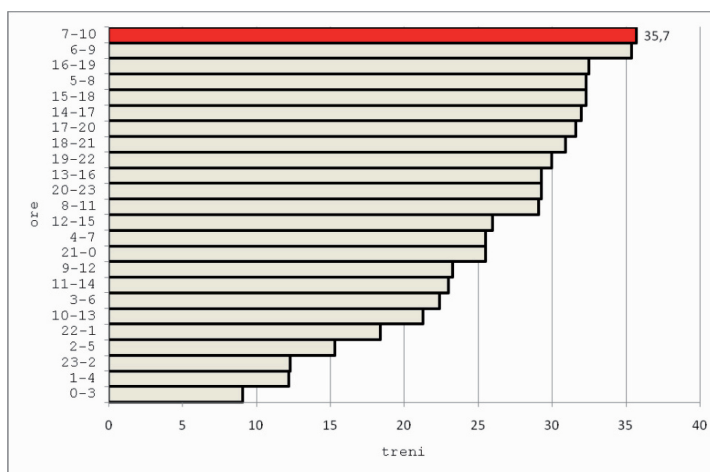


Fig. 14 - Numero medio di treni passeggeri sulla tratta Torricola-Formia. *Mean number of passengers trains on Roma-Formia line.*

(ore = hours - treni = trains).

mattutino con maggiore densità di traffico è quello 7-10 con 35,7 treni, mentre per le ore pomeridiane il primo intervallo è 16-19 con 3,2 treni in meno. Per tale ragione, tutte le simulazioni sono state effettuate con riferimento all'orario di esercizio della Roma-Formia fra le 7 e le 10 del mattino, ovvero sono stati "costruiti" nel modello tutti i treni che in tale intervallo di tempo hanno percorso la linea fra Torricola e Formia.

Esiti della simulazione

La simulazione è stata effettuata considerando da un lato il M53 e dall'altro i treni che effettivamente hanno fatto servizio, da M42, il 10-1-2007, il 14-1-2007, il 16-7-2006 e il 19-7-2006, rappresentativi rispettivamente di una giornata ferialle invernale, una festiva invernale, una ferialle estiva ed infine una festiva estiva. Seppure vi siano fra i dati di traffico di queste giornate differenze minime, piccole fra ferialle/festiva e insignificanti fra stagioni, la giornata con la maggiore densità di traffico fra le 7 e le 10 è risultata la prima. Pertanto i risultati delle simulazioni effettuate si riferiscono ad essa.

Per ognuno dei treni è stato dunque impostato l'orario d'ingresso sulla tratta, desunto dagli orari di passaggio nella stazione d'ingresso e calcolato in base alla velocità media fra la prima e la seconda stazione. Le stazioni d'ingresso considerate sono state Torricola (in direzione Formia), Campoleone, Priverno-Fossanova e Itri (in direzione Roma).

Per ciascun treno è stato registrato sul file di output l'entrata negli oggetti rappresentati e gli eventuali annullamenti di formazione itinerario dovuti a richieste non andate a buon fine.

Il confronto fra orario teorico da M53 e passaggi dei tre-

takings, therefore the freight traffic in figure 13 should be considered as a maximum.

The figure shows that for the passenger traffic the peak period is in the morning: 35,7 trains between 7.00 and 10.00 against 32,5 trains between 16.00 and 19.00. For this reason all the simulations have been carried out with reference to the morning period, by building in the model all the trains running between Torricola and Formia in this period.

Simulation results

The simulation has been carried out on the basis of the planned timetable (RFI module M53) and of the actual operated trains (RFI module M42) in the days 10.01.2007, 14.01.2007, 16.07.2006 and 19.07.2006, representing working days and holidays both in winter and summer periods. Though there are minimal differences, the first reference day experienced the highest traffic and the shown results are concerning it.

For each train was established the starting time at the entering station, the running time along the line and the mean speed. The considered entering stations were Torricola (direction Formia), Campoleone, Priverno-Fossanova and Itri (direction Roma).

On the output file the entering times at the represented objects and the possible cancelations of requested not assigned routes.

The comparison between M53 theoretical timetable and calculated schedules at mean speed was useful to check sub-nets correctness and operation, as well as links among various objects. From the point of view of the critical situations in the reference peak period (7.00-10.00) the simulation shown that about 10% of trains acquire more than 30 seconds delays (see figure 15) in unperturbed traffic conditions too. This value was chosen as upper bound of the admitted deviation between simulated and actual timetable, anyway determined with 30 seconds approximation.

The values in figure were calculated as absolute deviations between theoretical and simulated timetables. They are always negative with a mean value of -5,80 seconds, a standard deviation of 14,01 seconds and, therefore, a confidence interval $\mu \pm \sigma$ lower than 30 seconds.

The comparison between planned and simulated timetable was performed by the comparison

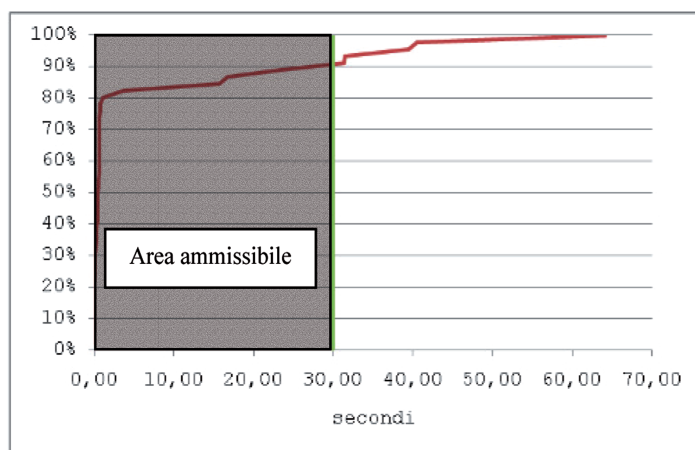


Fig. 15 – Percentuale di treni al variare dello scostamento assoluto fra orario simulato e orario teorico a destinazione. *Train percentage deviating from simulated and teorica timetable.*

ni calcolati dal modello di simulazione con un approccio a velocità media è stato utile per verificare la correttezza del funzionamento delle Sotto-reti e dei collegamenti fra i vari oggetti. Dal punto di vista delle criticità rilevate nella applicabilità dell'orario teorico fra le 7 e le 10, la simulazione ha mostrato come quasi il 10% dei treni, pur partendo tutti "in orario" e pur in condizioni di traffico imperturbato, giungono a destinazione con oltre 30 secondi di differenza rispetto all'orario teorico (fig. 15), valore che è stato scelto come limite superiore della differenza ammissibile fra orario simulato e orario reale, comunque determinato con un'approssimazione di ± 30 secondi.

La figura è calcolata considerando i valori assoluti della differenza fra orario teorico e simulato, anche se per i valori superiori al secondo si tratta sempre di ritardi del "treno simulato" rispetto a quello "teorico". La media dei valori di differenza (M53-simulazione) è pari a -5,80 secondi, con deviazione standard 14,01, quindi con intervalli di confidenza $\mu \pm \sigma$ entro il limite dei 30 secondi.

A conclusione di questo paragrafo è bene sottolineare come il confronto fra orario programmato e orario simulato è stato realizzato attraverso il confronto con i diversi tempi di percorrenza sull'intera tratta esaminata, quindi nelle condizioni più severe. Questa considerazione è però valida solo nell'ipotesi di velocità costante fra le stazioni e assenza di precedenze in linea. Come si vedrà più avanti, quando queste due ipotesi vengono rilasciate l'analisi delle differenze fra simulato e reale deve essere condotta oggetto per oggetto.

5.3. Simulazioni in regime perturbato nelle ore di punta

Analisi dei ritardi di linea

L'analisi ha avuto come obiettivo la determinazione delle criticità rilevabili nella circolazione dei treni durante il servizio effettivo e il confronto di queste con i risultati della simulazione del modello sulla base di un orario effettivo (da M42). Come accennato nel paragrafo precedente, sono state analizzate quattro giornate e, poiché le criticità maggiori per la regolarità sono state rilevate durante la giornata del 10-1-2007, si riportano in questo paragrafo i risultati riferiti a quest'ultima.

La fig. 16 mostra l'andamento percentuale dei ritardi dei treni passeggeri ad ogni stazione della Roma-Formia fra le 7 e le 10 del mattino. Con riferimento alla direzione verso Formia, si può notare come entro i 10 minuti di ritardo transita circa l'84% dei treni, entro i 5 minuti il 58%, entro i 30 secondi il 20%. I valori riferiti alla direzione verso Roma indicano entro i 10 minuti di ritardo il 70% dei treni, entro i 5 minuti il 42% ed infine entro i 30 secondi l'11%. Dunque, fra i treni circolanti fra le 7 e le 10 di mattina sulla Roma-Formia, in direzione Formia 1 ogni 5 arriva alle varie stazioni entro l'orario programmato nel M53, mentre in direzione Roma 1 ogni 9.

Per quanto riguarda le categorie dei treni che sono state utilizzate nelle varie simulazioni (ad eccezione di quelli

with various running times on the simulated section, therefore in the most severe conditions. This consideration is anyway valid just under the hypothesis of constant running speed and not allowed overtaking. As soon as these hypothesis are released the analysis of differences between simulated and real times should be performed object by object.

5.3. Simulation under perturbed conditions during peak periods

Analysis of line delays

The analysis was aimed to determine the actual critical operational situations and their comparisons with the results of models simulation of a real (M42 derived) timetable. As was outlined in the previous paragraph four days were analysed and are reported the results concerning the most perturbed day (10.01.2007).

Figure 16 shows the percentage of passengers trains delays at various station between 7.00 and 10.00. In the Roma-Formia direction 84% of trains is less than 10 minutes delayed, 58% of trains is less than 5 minutes delayed and 20% of trains is less than 30 seconds delayed. The values in Formia-Roma directions are 70% of delayed trains under 10 minutes, 42% of delayed trains under 5 minutes and 11% under 30 seconds. Therefore during the period the unperturbed trains are 1/5 towards Formia and 1/9 towards Roma.

According to the train categories (with exception of the trains running from/to Terracina and Nettuno branch lines) the experienced mean delay of trains running towards Rome is 4.74 minutes for the first category (Eurostar + Euronight trains), 5.38 minutes for the second category (Intercity + Express trains), 5.42 minutes for the third category (Regional trains), as well as a completely different situation is for freight trains with 95.90 minutes of mean delay. For all categories just a limited recover of delays (anyway under 60 seconds) is observed, that could be evaluated as a certain difficulty to regularise the perturbations generated by Rome node along the line, as well as in the opposite direction too.

For trains running towards Formia the absolute mean values of delays are higher: 6.70 minutes for first category, 15.08 minutes for the second category and 4.50 for the third one. The regional trains are normally increasing their delays along the line, but in this case the effect is moderated by the exception of trains directed towards the branch lines.

Performed simulations

The results of the perturbation analysis have been translated into inputs for delays simulations.

da/verso Terracina e da/verso Nettuno), in direzione di Formia il ritardo medio della prima categoria di treni (Eurostar + Euronight) è di 4,74 minuti, quello della seconda categoria (Intercity + Espresso) è di 5,38 minuti, quello della terza è di 5,42 minuti, mentre il solito discorso a parte meritano i treni merci, con un ritardo medio di 95,90 minuti. In tutte le categorie c'è un lieve recupero del ritardo (sotto il minuto, merci a parte), evento che potrebbe essere interpretato come una difficoltà della linea a stabilizzare i ritardi che provengono dal nodo di Roma, anche nelle ore in cui il massimo traffico è in entrata al nodo e non in uscita.

Nell'altro verso i valori assoluti dei ritardi, come visto, sono maggiori: 6,70 minuti per la prima categoria, 15,08 minuti per la seconda e 4,5 per la terza. In questo caso il relativamente basso valore della terza è probabilmente da imputare alla mancanza dei treni da Nettuno e da Terracina. Infatti i treni della terza categoria sono gli unici ad incrementare, seppur di poco, il loro ritardo medio lungo la linea, mentre il valore elevato del recupero di quelli della seconda categoria va confrontato con l'elevato valore assoluto del ritardo stesso. Anche in questo caso la tratta Formia-Torricola non riesce sostanzialmente a migliorare la situazione di traffico perturbato che proviene da sud.

Simulazioni effettuate

I risultati dell'analisi appena illustrata sono stati utilizzati per inserire come input delle simulazioni i ritardi dei vari treni.

Per studiare il comportamento del modello durante l'esercizio di con un orario effettivo sono state realizzate le seguenti simulazioni:

- simulazione dell'orario M42 di ciascun treno indipendente dal resto della circolazione;
- simulazione di traffico perturbato con imposizione di un ritardo medio per treno ricavato dagli M42 e approccio a velocità costante fra le stazioni;
- simulazione di traffico perturbato con imposizione di un ritardo iniziale per treno ricavato dagli M42 e approccio a velocità variabile fra le stazioni;
- simulazione di traffico perturbato con imposizione di un ritardo iniziale per treno ricavato dagli M42 e approccio a velocità variabile fra le stazioni con attivazione di un sistema di precedenza automatiche.

La prima simulazione è stata condotta ricostruendo l'orario di ciascun treno con i ritardi medi ricavati dai ritardi

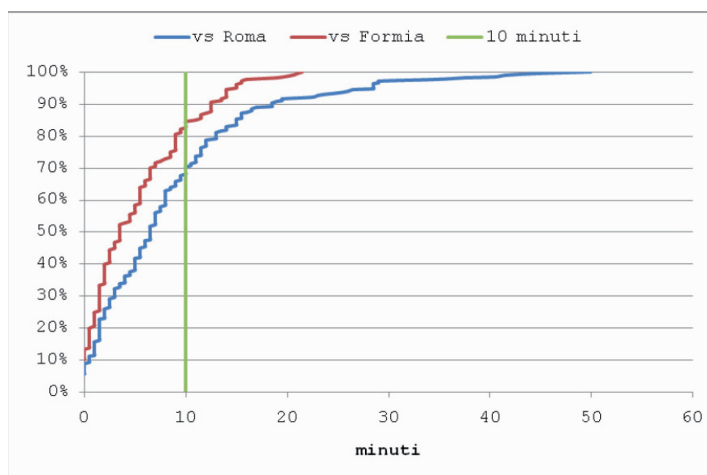


Fig. 16 – Andamento percentuale dei ritardi dei treni passeggeri sulla Roma-Formia (ore 7-10). Percentage of delays of passengers trains on the Roma-Formia between 7.00 and 10.00.

In order to study the model behaviour during the actual timetable operation the following simulations have been performed:

- M42 timetable simulation for each train independently;
- perturbed traffic simulation with mean delay per train derived by M42 timetable and constant speed between stations;
- perturbed traffic simulation with mean delay per train derived by M42 timetable and variable speed between stations;
- perturbed traffic simulation with initial delay per train derived by M42 timetable, variable speed between stations and automatic overtaking.

The first simulation was carried out by producing the trains timetable including the mean delays derived by M42 in each station. The whole timetable resulted by the aggregation of single trains timetables. This simulation is required both for checking the compatibility among the delays resulting from M42 and to provide hypothetical delays of trains to/from Nettuno and Terracina. This simulation was able to reproduce the single train delays with less than 0,5 seconds errors, nevertheless showed a partial incompatibility among single M42 timetable. This implied the need of minor corrections in the simulated timetable.

The second simulation was performed by imposing to trains the mean delay derived by M42 modules, leaving the model working to estimate the corrections required to grant traffic compatibility. In this case a constant speed approach was adopted, therefore the trains

alle stazioni segnalati nel modulo M42. Ciascun treno è stato simulato in maniera indipendente dagli altri e l'orario complessivo è stato ricreato dalla somma degli orari dei singoli treni. Questa simulazione si è resa necessaria da un lato per verificare la compatibilità fra i vari ritardi registrati sugli M42 e dall'altro per ipotizzare dei possibili ritardi per i treni da/verso Nettuno e da/verso Terracina, dei quali come detto non si possiede l'orario effettivo di passaggio alle stazioni. Questa simulazione, nonostante abbia ricostruito con un errore inferiore al mezzo secondo la percorrenza di ciascun treno, ha mostrato una parziale incompatibilità del traffico segnalato negli M42. Ciò ha implicato l'apportare di alcune correzioni nell'orario effettivo da simulare.

La seconda simulazione è stata quindi realizzata imponendo ai treni il ritardo medio stimato dagli M42 e si è lasciato stimare al modello le correzioni da apportare all'orario per rendere compatibili le circolazioni. C'è da tenere presente che, poiché in questo caso si è adottato un approccio a velocità costante fra le stazioni, con questa velocità pre-impostata, i treni che hanno accumulato ritardo perché hanno trovato la via impedita da altri non hanno avuto la possibilità nelle tracce successive di recuperare parte del loro ritardo, portando quindi questo surplus fino a destinazione.

Volendo considerare gli eventuali recuperi di tempo dei treni che viaggiano in ritardo è necessario adottare un approccio a velocità variabile lungo la linea e dunque attivare nel modello le Sotto-reti cinematiche accennate nel capitolo 3. Anche in questa terza simulazione, come nella precedente, a ciascun treno è stato imposto il ritardo d'ingresso in linea derivato dall'analisi di regolarità ed è stato lasciato al modello il compito di risolvere le situazioni di conflitto che potevano presentarsi.

Per tener conto di eventuali sorpassi in linea non previsti dall'orario M53, ma opportuni per privilegiare la marcia di alcuni treni, occorre attivare l'oggetto DCM. Nel modello c'è sia la possibilità di determinare un ordine di priorità fra i treni nella maniera già illustrata nel capitolo 3 a proposito del funzionamento della classe MA, sia adottare un sistema di priorità treno per treno. Quello adottato nella quarta simulazione è riportato nella tabella 2 e segue in ogni caso le quattro categorie di treni esposte nel paragrafo precedente. Detto X il treno presente sulla prima colonna e Y quello sulla prima riga, la freccia verso l'alto indica un diritto di precedenza del treno X su Y, quella verso il basso il contrario, quella verso destra nessun diritto di precedenza per entrambi.

In questo caso la simulazione è stata effettuata con approccio a velocità variabile fra le stazioni, imponendo ai treni un ritardo d'ingresso in linea pari a quello emerso dall'analisi degli M42.

5.4. Considerazioni sulle simulazioni svolte

Come già detto, le simulazioni mostrate in questo capitolo hanno avuto per scopo la calibrazione delle Sotto-reti

delayed by red signals were not able to recover their delays before reaching their destination.

To include the possibility of delays recovering a variable speed approach was adopted by activating in the model the cinematic sub-nets (see chapter 3). In this case the delays resulting by M42 timetables were partially recovered by the model working for conflicts resolution.

To take into account possible overtaking not foreseen in M53 timetable, but useful to solve arising conflicts the DCM object was activated. The model allows both to fix a priority order among trains (see MA function in chapter 3) or to adopt a train by train priority criteria. The criteria adopted in this simulation is represented in table 2 and includes the four trains categories exposed in the previous paragraph: let *X* be the train in the first column and *Y* that in the first row, the upwards arrow means a priority of *X* over *Y*, the downwards arrow represents the opposite and the rightwards arrow means no priority for both them.

In this case the simulation was performed with a variable speed approach with initial delays resulting by M42 timetable.

5.4. Comments about simulations undertaken

As it was anticipated, the simulations shown in this chapter are aimed to calibrate the sub-nets of various classes more than the systematic reproduction of timetable. Nevertheless it is interesting to check the coherence of the results obtained by re-elaborating the M42 derived delays.

In particular figure 17 shows the difference between mean speed (*eff med*), variable speed (*eff var*) and with DCM object activation approaches. Delays and running times are expressed in relative comparison with the simulation reproducing M53 module.

It is possible to note that the introduction of variable cinematic sub-nets, by allowing the delays recovering, reduce the running times; this reduction increases more and more if the DCM is activated, because fast trains are able to overtake slow trains.

Figure 18 shows into details variations of mean running times by train in comparison with theoretical M53 timetable towards Roma and Formia. Southwards the variable cinematic and the DCM operation give a negative value of running time in comparison with planned times (M53 timetable), as well as it differently happens Northwards, where this difference is positive. This apparent contradiction is due to the line approaching in reverse order (in comparison with M53) of some trains without overtaking possibility in the second simulation. This situation is just a little reduced by DCM operation due to the amount of trains with same priority.

This evidence showed that the priority criteria by train category only is not enough to minimise the running time.

delle varie classi e non la ricostruzione fedele degli orari della linea. Ciononostante è interessante verificare la coerenza dei risultati ottenuti confrontando fra loro le simulazioni che hanno ripreso i dati dei ritardi dei moduli M42 e li hanno elaborati nei modi descritti.

In particolare la fig. 17 mostra la differenza fra gli approcci a velocità media costante fra le stazioni (eff med), a velocità variabile (eff var) e con attivazione dell'oggetto DCM. I valori dei ritardi e della percorrenza sono espressi in termini relativi rispetto alla simulazione che ha riprodotto il modulo M53.

Come si può notare l'introduzione delle Sotto-Reti con cinematica variabile, permettendo il recupero del tempo eventualmente perso, riducono i tempi di percorrenza; tale riduzione aumenta ancora di più se viene attivato il DCM, poiché treni di categoria superiore, generalmente più veloci, riescono a superare altri treni più lenti.

La fig. 18 mostra il dettaglio delle variazioni dei tempi di percorrenza medi per treno rispetto all'orario teorico M53 verso Formia e verso Roma. Mentre verso Formia l'introduzione di una cinematica variabile e ancor di più del DCM rende negativo il saldo rispetto alla percorrenza M53, ciò non accade verso Roma. Addirittura nel passaggio da approccio a velocità costante a quello a velocità variabile la percorrenza aumenta. Questa apparente contraddizione si spiega col fatto che nella seconda simulazione alcuni treni sono entrati in linea in ordine inverso e possedevano velocità tali che il treno che si trovava dietro veniva rallentato senza peraltro poter superare quello che lo precedeva (infatti nell'M53 lo seguiva). Tale situazione è solo un po' attenuata con il DCM in quanto alcuni di questi treni sono di pari categoria e dunque, per il criterio scelto, non poteva verificarsi alcun sorpasso in linea.

Tale evidenza ha mostrato come il solo criterio di precedenza per categorie non sia sufficiente a minimizzare i tempi di percorrenza sulla tratta.

Un'analisi ulteriore svolta a tal fine è stata quella di perturbare un treno alla volta e verificarne il grado di

TABELLA 2 – TABLE 2

PRIORITY BETWEEN SIMULATED TRAINS
PRIORITY FRA I TRENI SIMULATI

	2381	53013	809	9371	9381	2383	375	51019	12189	771	675	1911	9351	719	2385	12191	287	581	1673	2387	55271	1671	51067	12193	51191	51015	57923	57913	51005	9373	2389
2381	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
53013	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
809	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9371	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9381	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2383	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
375	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
51019	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12189	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
771	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
675	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1911	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9351	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
719	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2385	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12191	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
287	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
581	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1673	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2387	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
55271	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1671	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
51067	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12193	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
51191	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
51015	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
57923	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
57913	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
51005	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9373	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2389	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

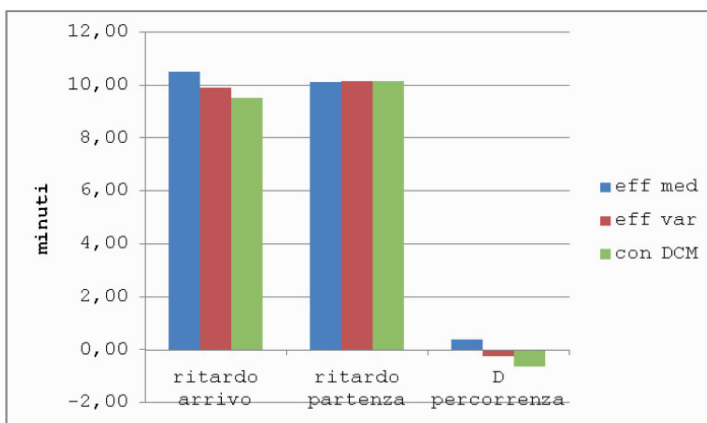


Fig. 17 – Confronto fra diversi approcci della simulazione (valori medi per treno).
Comparison among various simulation approaches: mean values by train.
(minuti = minutes - ritardo arrivo = arrival delay - ritardo partenza = departure delay - percorrenza = running time - eff med = mean speed - eff var = variable speed - con DCM = with DCM).

criticità nel sistema. I risultati dell'analisi così condotta hanno portato a delineare un nuovo sviluppo applicativo del modello, volto alla risoluzione dei conflitti ai nodi, che, pur non essendo l'obiettivo iniziale della ricerca, è accennato nel prossimo paragrafo come esempio di possibile ulteriore applicazione del modello descritto.

5.5. Ulteriori possibili applicazioni

Potenzialità del modello

Durante la realizzazione del modello, man mano che le potenzialità divenivano più esplicite, si è indagata la possibilità di applicarlo anche al raggiungimento di obiettivi più ampi di quelli inizialmente ipotizzati. Nei paragrafi seguenti verranno mostrate tre possibili applicazioni del modello.

La prima riguarda la possibilità di utilizzare il modello non solo in fase di pianificazione ma anche in fase di gestione della circolazione.

La seconda riguarda l'utilizzo del modello nelle analisi di capacità.

La terza, infine, riguarda la possibilità di utilizzare la versatilità del file di output per analisi di problemi collegati alla gestione ottimale dell'infrastruttura.

Risoluzione dei conflitti ai nodi

Dopo aver testato il funzionamento dell'oggetto DCM nelle simulazioni precedentemente descritte, si è tentato di verificare un suo possibile utilizzo nella risoluzione dei conflitti ai nodi. Per far ciò, sono state eseguite varie simulazioni a nodi semplici per poi estendere i risultati alle stazioni della linea. Essendo la sperimentazione ancora in corso, in questo paragrafo verrà esposto solo lo schema che permette l'utilizzo del modello per questa funzione, mostrando un esempio delle simulazioni svolte.

Un treno A si presenta alla stazione di Priverno con un ritardo tale da creare un conflitto con il treno B proveniente da Terracina. La situazione è esemplificata nello schema in fig. 19. Abbiamo visto che il modello può risolvere questo conflitto con l'attivazione dell'oggetto DCM e sulla base di un preciso ordine di priorità fra i treni. Ora il nuovo obiettivo è verificare come poter utilizzare il modello per poter minimizzare il ritardo complessivo di arrivo alla stazione di tutti i treni previsti dall'orario di esercizio. Poiché il modello può risolvere automaticamente conflitti "locali", fra più treni che contemporaneamente si contendono una risorsa, ma non conflitti "globali", cioè verificare automaticamente tutte le possibili soluzio-

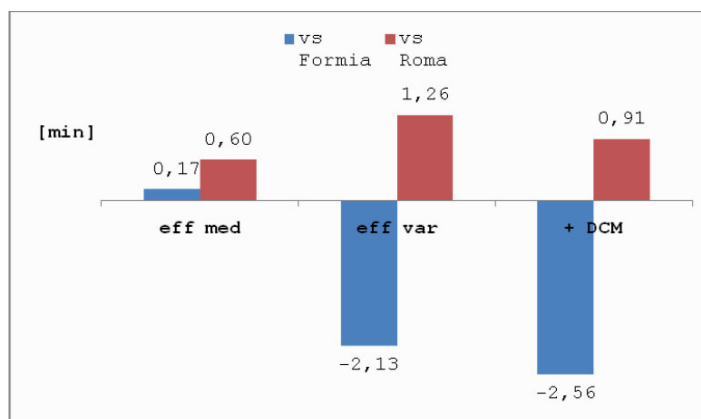


Fig. 18 – Variazione nella percorrenza media per treno nella fra diversi approcci della simulazione. *Variation of mean running time with different simulation approaches.*

5.5. Further possible applications

Model capabilities

During the model development, as soon as its capabilities were clarified, the possibility to apply it for broader goals were highlighted. In the next paragraphs three possible model applications will be shown.

The first one concerns the possibility to use the model not only for planning but also for traffic management.

The second one concerns its use for capacity analysis.

The third one, finally, concerns the possibility to use the versatility of output file to analyse the infrastructure optimal management problems.

Conflicts resolution

After having tested the DCM object operation in the previous simulations, it was attempted to check its possible use for the resolution of conflicts at nodes. For this scope various simulations of simple nodes have been carried out in view of its extension to the stations along the line. On the basis of the ongoing experimentation it is here described the scheme allowing to use the model for this specific aim and exemplificative simulations.

Let a train A approaching Priverno station with enough delay to create a conflict with train B arriving from Terracina. The situation is represented in figure 19. The model is able to solve this conflict by the DCM object activation on the basis of a precise trains priority order. The goal is now to check how to use the model to minimise the global delay of planned trains. The model is able to solve automatically local conflicts among trains requiring the same resource, but not global conflicts by automatically checking all possible conflicts and compare them by looking for the minimisation of the global line de-

ni e confrontarle fra loro fino a raggiungere un obiettivo globale di linea, come è appunto la minimizzazione del ritardo complessivo di linea, l'unico modo per arrivare ad una soluzione del problema è adottare una via semi-automatica: individuare a priori le possibili soluzioni a livello qualitativo e poi verificare quantitativamente nel modello quali fra le soluzioni soddisfa il criterio della minimizzazione del ritardo complessivo.

Nel conflitto ad un nodo semplice, come quello di fig. 19, è possibile individuare le otto soluzioni qualitative mostrate in fig. 20.

Nel dettaglio le soluzioni sono le seguenti:

- 1) il treno A e il treno B utilizzano lo stesso binario 1, continuando a ritardare il treno A, senza ritardare altri treni (ciò significa che se nel frattempo arriva un altro treno C che richiede la risorsa binario 1, A attende anche il transito di quest'ultimo e così via in maniera che nessun treno venga penalizzato da A);
- 2) il treno A e il treno B utilizzano lo stesso binario 1, continuando a ritardare il treno A, che passa appena dopo B anche se ciò comporta il ritardo di altri treni;
- 3) il treno A e il treno B utilizzano lo stesso binario 1, ritardando il treno B, evento che può eventualmente ritardare altri treni;
- 4) il treno A e il treno B utilizzano lo stesso binario 1, ritardando il treno B, senza ritardare altri treni;
- 5) il treno A viene dirottato sul binario 2, senza che ciò comporti ritardo per altri treni (eccetto B);
- 6) il treno A viene dirottato sul binario 2, potendo ciò comportare ritardo per altri treni;
- 7) il treno B viene dirottato sul binario 2, potendo ciò comportare ritardo per altri treni;
- 8) il treno A viene dirottato sul binario 2, senza che ciò comporti ritardo per altri treni.

Si noti che ciascuna di queste soluzioni non è quantificabile in maniera univoca, in quanto il calcolo del ritardo complessivo (cioè del ritardo dei treni A, B e di tutti quelli che seguono nell'orario di esercizio) dipende da numerosi fattori quali la scelta dei binari alternativi candidati al piazzamento dei treni (in questo caso per semplicità c'era un solo binario alternativo), l'entità del ritardo del treno A, la situazione di perturbazione complessiva del sistema.

Nella simulazione di cui si mostrano

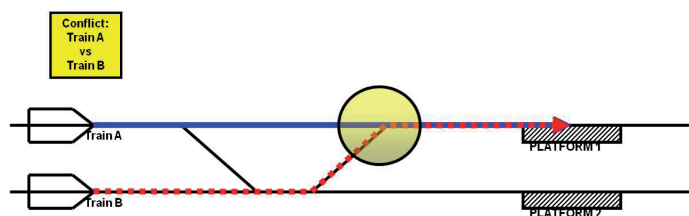


Fig. 19 – Conflitto di circolazione fra due treni. *Conflict between two trains.*

lay. The solution is to adopt a semi-automatic procedure: to identify preliminarily the possible solutions and then check quantitatively by the model which solution are compatible with the global delay minimisation.

In the simple node conflict (figure 19) it is possible to identify the eight solutions shown in figure 20.

The solutions are the following:

- 1) train A is delayed by train B using the same track 1, without delaying other trains: should another train C ask for track 1, train A has to wait for that too, so that no trains are penalised by A;
- 2) train A is delayed by train B using the same track 1, A will run as soon as B release track 1 also if other trains are delayed;
- 3) train B is delayed by train A using the same track 1, B will run as soon as A release track 1 also if other trains are delayed;
- 4) train B is delayed by train A using the same track 1, B without delaying other trains;
- 5) train A is deviated on track 2 without delaying other trains (except train B);

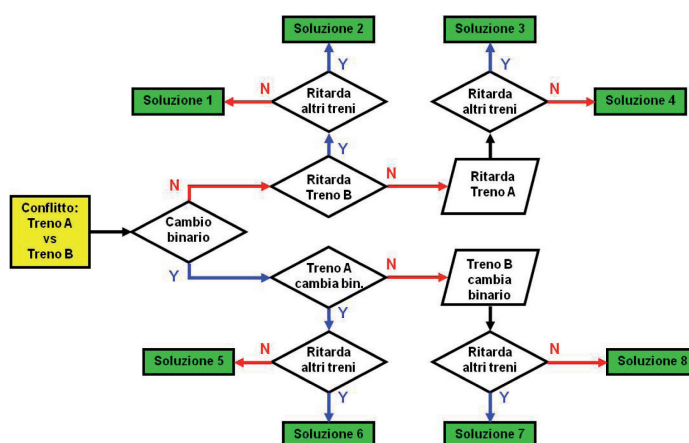


Fig. 20 – Albero delle soluzioni del conflitto di circolazione fra due treni. *Conflicts solutions tree.*

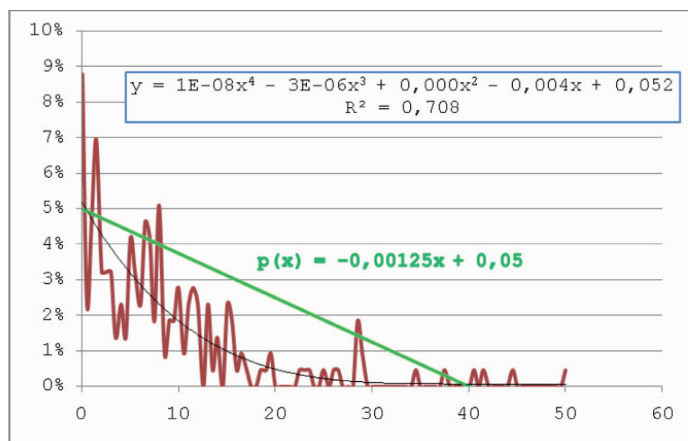


Fig. 21 – Densità di probabilità triangolare dei ritardi sulla Roma-Formia. *Triangular probability density of delays [minutes] on Roma-Formia line.*

i risultati si è imposto per il primo aspetto il vincolo di cambio di binario consentito solo fra binari adiacenti (con uso banalizzato).

Per tener conto degli altri due aspetti è stata condotta un'analisi con traffico perturbato e si è analizzata la sensibilità delle soluzioni trovate al variare del ritardo del treno A. Dunque, come primo passo si sono utilizzati i risultati dell'analisi di regolarità vista in precedenza per trovare una distribuzione che potesse essere utilizzata come "generatrice" di ritardo per i treni. La fig. 21 mostra la curva d'interpolazione dei ritardi dei treni al passaggio alle stazioni e la densità di probabilità triangolare prescelta per simulare i ritardi dei vari treni.

Nota la distribuzione di probabilità $p(x)$ come in figura, la probabilità di arrivo di un treno con un ritardo x è data da:

$$P(x) = \int_0^x p(x) dx = \int_0^x (-0,00125x + 0,05) dx = -0,000625x^2 + 0,05x$$

Facendo variare $P(x)$ fra 0 e 1 in maniera casuale, ad esempio estraendo i numeri p_i casualmente da una distribuzione uniforme, è possibile associare il seguente ritardo ad ogni p_i estratto:

$$x = \frac{-0,05 \pm \sqrt{0,05^2 + 0,00125p_i}}{0,000625}$$

In pratica con questa operazione si sono estratti i treni fra quelli successivi al treno A in maniera del tutto casuale e poi si sono applicati a questi ritardi distribuiti secondo la distribuzione triangolare prima mostrata.

Le simulazioni sono state svolte facendo funzionare il modello con un numero di iterazioni tali da soddisfare un livello di fiducia pari al 95%. L'errore ammesso (r) sul ritardo totale è stato di 5 minuti e il numero di iterazioni (n) è stato

6) train A is deviated on track 2 with possible delays of other trains;

7) train B is deviated on track 2 with possible delays of other trains;

8) train A is deviated on track 2 without delaying other trains.

Each solution cannot be univocally quantified because the calculation of global delay (acquired by trains A, B and the following ones) depend upon several factors, including the choice of alternative tracks (just one in the example), the original delay of train A, the global system perturbation level.

In the shown simulation the exchange was allowed between adjacent tracks only, though with bidirectional operation.

In order to take into account the additional two aspects an analysis with perturbed traffic and the sensibility of solutions by train A delay variation were developed. As a first step the results of the regularity survey have been used to build a distribution of trains delays suitable to be used as delays generator. Figure 21 shows the interpolation of delays curve and the triangular distribution used to simulate the trains delays.

On the basis of the probability distribution $p(x)$ according to figure 21, the probability of train arrivals with a delay x is the following:

$$P(x) = \int_0^x p(x) dx = \int_0^x (-0,00125x + 0,05) dx = -0,000625x^2 + 0,05x$$

Let $P(x)$ vary randomly between 0 and 1 (e.g. by randomly extracting p_i numbers) it is possible to link to each extracted p_i the following delay:

$$x = \frac{-0,05 \pm \sqrt{0,05^2 + 0,00125p_i}}{0,000625}$$

According to this procedure have been selected and randomly delayed on the basis of the derived triangular distribution the trains following train A.

The simulations have been performed with an amount of iterations enough to satisfy a confidence level of 95%. The allowed error (r) on the total delay was 5 minutes and amount of needed iterations was calculated on the basis of the limit values assumed by the 95 percentile of the Gauss distribution with stopping conditions (one queue test):

$$\frac{\sigma_n Z_{95\%}}{\sqrt{n}} \leq r$$

Practically for n iterations are obtained n values of mean global delay. Let R be the mean values, if is

calcolato in base al valore limite assunto dalla Gaussiana nel percentile 95%, con condizione di arresto (test a 1 coda):

$$\frac{\sigma_n Z_{95\%}}{\sqrt{n}} \leq r$$

In pratica per n iterazioni si ottengono n valori del ritardo medio complessivo. Detto R la media, se $(R \pm r)$ contiene

$$(R \pm \frac{\sigma_n Z_{95\%}}{\sqrt{n}}),$$

allora le simulazioni non vengono più iterate e R è il ritardo medio complessivo considerato per uno specifico ritardo del treno A.

Facendo variare il valore di ritardo di A da 0 a 1 ora, si ottiene la fig. 22, da leggere considerando che la soluzione con ritardo complessivo minore è quella che più si avvicina al bordo esterno della ragnatela.

Da sottolineare è il fatto che gli altri treni sono stati confrontati sulla base delle priorità fra categorie, ma nulla osta l'adottare di criteri differenti, come ad esempio il ritardo accumulato, o l'adottare di più criteri con diverse gerarchie.

Il procedimento descritto può essere esteso a qualsiasi altro nodo semplice di una stazione. Dal risultato delle simulazioni è poi possibile costruire un albero decisionale tale da essere utilizzato in fase di gestione del traffico per la risoluzione dei conflitti che di volta in volta si presentano.

E' chiaro che all'aumentare del numero di treni e, soprattutto, del numero di nodi, come ad esempio in stazioni complesse, il numero di simulazioni cresce in maniera notevole. Verificare l'efficienza del modello anche in nodi complessi rappresenta il primo degli sviluppi futuri previsti.

Valutazione della capacità ai nodi

Il modello di simulazione qui presentato è stato testato ed applicato nell'analisi di capacità della stazione Roma Ostiense, dotata di ACC. Il test ha avuto come scopo il confronto fra i principali metodi per la verifica della capacità degli impianti ferroviari e la simulazione con le Reti di PETRI. Per i risultati forniti dalle simulazioni si rimanda a [8].

Programmazione della manutenzione dell'infrastruttura

Non è stata realizzata una specifica simulazione per illustrare questa opportunità, tuttavia si vuole qui sottolineare ancora una volta una proprietà importante del modello di simulazione dell'esercizio finora descritto: la possibilità di inserire nelle transizioni un comando di scrittura su file di output ogni qual volta la transizione viene attivata. Ciò significa che, ad esempio, è possibile verificare quante volte un cir-

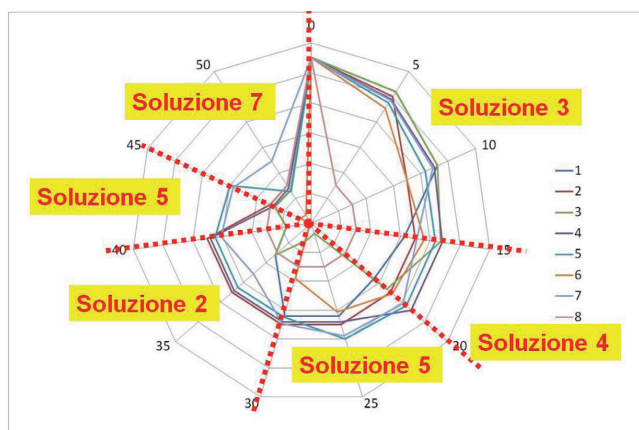


Fig. 22 – Soluzioni dei conflitti al variare del ritardo dl treno ritardante. Conflicts solutions by varying delaying train delays.

$$(R \pm \frac{\sigma_n Z_{95\%}}{\sqrt{n}}),$$

the simulations are no more iterated and R is the mean global delay corresponding to the specific delay of train A.

By varying the delay of train A between 0 and 1 minute, is obtained figure 22, to be read considering the solution with the minimum global delay corresponding to the shortest distance from the external border of the spider-net diagram.

The remaining trains have been compared on the basis of priorities among categories, but it is possible to adopt different criteria (e.g. minimum additional delay) or different hierarchies.

The described procedure may be extended to any node in a station. From the simulations' results it is possible to build a decision tree useful in traffic management for conflicts resolution.

By increasing the amount of involved trains and, above all, the amount of nodes (e.g. in complex stations) the amount of simulations strongly increases. In fact to check the model effectiveness also in complex nodes represent the first planned development.

Node capacity evaluation

The simulation model was tested and applied also in the capacity analysis of Roma Ostiense station equipped with an electronic interlocking system. The test was aimed to compare the main methods for railway capacity assessment with Petri Net. The results are summarised in [8].

cuito di binario è interessato da verifiche di libertà, quante volte un deviatore cambia di posizione durante la giornata, sostanzialmente quante volte un qualsiasi ente viene sollecitato. E' evidente che con questi dati, se opportunamente confrontati con indicatori tipo MTBF, è possibile ricostruire un programma di manutenzione dei componenti della linea che tenga conto non solo delle ore di durata media degli enti che compongono l'infrastruttura, ma anche di quelle di funzionamento teorico ed effettivo.

6. Generalizzazione del modello ad una linea qualsiasi

6.1. Potenzialità di generalizzazione

Il modello proposto può essere applicato negli studi di regolarità del traffico ferroviario su una linea, ad esempio confrontando i dati dei ritardi effettivi misurati con quelli simulati dal programma in analoghe condizioni di traffico ed individuando le cause del ripetersi di determinate criticità.

La flessibilità del file di output consente di spingere l'analisi fino al dettaglio del singolo ente, così da avere un'idea esaustiva delle criticità di circolazione correlate con l'uso dell'infrastruttura.

Inoltre il modello può essere usato efficacemente come modello di supporto alle decisioni sia nella pianificazione dell'esercizio su linee nuove, sia nell'adeguamento di linee esistenti con mutate condizioni di traffico.

Infine la naturale applicazione del modello con il sistema automatico di precedenza è il supporto alle decisioni da prendere in fase di gestione del traffico per minimizzare i ritardi. Infatti, dal confronto fra ritardi gestiti in maniera manuale o dal sistema automatico di precedenza, applicato con criteri diversificati, consente di definire un processo decisionale, nel caso si verificassero ritardi in linea, tale da contenere il più possibile le perturbazioni sul traffico complessivo.

Per poter applicare efficacemente il modello, in questo capitolo vengono brevemente descritte le fasi principali che portano alla costruzione e alla simulazione dello stesso su una generica linea.

6.2. Definizione degli obiettivi della simulazione

Prima di iniziare la costruzione del modello è bene definire fin da subito quelli che sono gli obiettivi del modello, in maniera da creare un numero adeguato di Sotto-Reti. Si pensi ad esempio al diverso numero di funzioni itinerario da creare nel caso di analisi di capacità di un nodo e nel caso si voglia individuare le criticità relative ad un determinato orario di esercizio.

I principali campi di applicazione del modello di simulazione sono riassumibili nei seguenti:

- stima dei tempi caratteristici dell'esercizio;

Infrastructure maintenance programming

A specific simulation to check this potential application opportunity was not performed, nevertheless it may be underlined once a time an important model feature: the possibility to include in the transitions an output file writing command as soon as the transition is activated. It means that it is possible to check how often an object is used (e.g. how often a track circuit is involved in freedom checks, how often a switch change its position during the day). With this data, if suitably compared with reliability indicators (e.g. MTBF), it is possible to build a maintenance plan of line components taking into account not only the mean duration of objects composing the infrastructure, but also theoretical and effective operation times.

6. Model generalisation

6.1. Power of generalisation

The proposed model may be applied for railway traffic regularity studies, e.g. by comparing the measured actual delays with those simulated in similar traffic conditions and by identifying the causes of critical situations.

The output file flexibility allows to push the analyses into the details of single objects, so that an exhaustive picture of traffic critical situations due to the infrastructure use may be defined.

Moreover, the model may be usefully employed as a decision support tool in operation planning both on new lines and in existing lines with upgraded traffic conditions.

The natural application of the model with the automatic overtaking function is finally the operation management decision support for delays minimisation. In fact, the comparison between the manual and automatic traffic management, applied with various criteria, allows to build a decision path to reduce as much as possible the global perturbation status.

In order to apply effectively the model, in this chapter are shortly describe the main model building and application phases.

6.2. Definition of simulation goals

Before starting to build the model must be initially well defined the model goals, so that a coherent number of sub-nets are created: e.g. the different number of route functions to analyse the node capacity or to identify the critical situations of a defined timetable.

The main model application fields are the following:

- operational times estimation;

- definizione dell'orario di servizio;
- valutazione della robustezza dell'orario di servizio;
- gestione del traffico;
- valutazione della capacità;
- gestione dei programmi di manutenzione della via.

Per ognuno di questi è possibile definire più obiettivi specifici. Ad esempio la definizione dell'orario di servizio può essere finalizzato alla verifica dell'effettiva fattibilità di orari forniti con modelli di ottimizzazione, oppure alla creazione dell'orario ex-novo, attraverso un procedimento euristico. Fondamentale è inoltre la definizione della scala dell'intervento: si potrà infatti procedere con un modello meno dettagliato e di macro-scala ovvero definire in dettaglio l'infrastruttura per analisi specifiche locali.

Allo stesso modo, per analisi di robustezza o stabilità, dell'orario di servizio l'obiettivo sarà strettamente dipendente dalla finestra temporale in cui si vuole considerare le circolazioni. Così come nella gestione del traffico ferroviario il criterio di risoluzione di conflitti, locale e globale, è essenziale nella definizione degli obiettivi.

Infine, per interventi sull'infrastruttura, le analisi di capacità e di utilizzo della stessa sono vincolate al tipo di indicatore che si intende ricavare dal modello: coefficiente di utilizzazione globale di un nodo complesso, analisi ente per ente, analisi sui nodi semplici, ecc.

6.3. Scelta degli output e definizione degli input

Definiti gli obiettivi, nel modello proposto è essenziale la scelta e la conseguente programmazione degli output che si intende creare. Infatti, come esposto a proposito del file di output, quest'ultimo può essere personalizzato. In linea generale è possibile far scrivere sul file i valori assunti durante la simulazione dalle variabili strutturate rappresentate dalle marche, dai parametri caratteristici degli oggetti, dalle operazioni imposte con tali valori e, non ultimo, il tempo di simulazione in cui una particolare marca attiva una transizione.

Ad esempio, in un'analisi di regolarità dell'esercizio sarà possibile verificare il passaggio alle stazioni di ciascun treno e gli eventuali annullamenti delle chiamate di formazione itinerario per verificare le tratte più critiche. In una analisi di capacità sarà necessario tener conto del numero di volte in cui è impegnato un itinerario con apposito contatore. Nella risoluzione dei conflitti ai nodi occorrerà prevedere un contatore che sommi i ritardi generati dalle circolazioni ostacolate da treni ritardanti.

6.4. Definizione degli input e raccolta dati

Una volta chiariti gli output desiderati sarà possibile inserire gli input necessari al modello fra quelli corrisponden-

- operational timetable definition;
- estimation of operational timetable robustness;
- traffic management;
- capacity assessment;
- infrastructure maintenance planning management.

For each field it is possible to define more specific goals. The timetable definition may be finalised to the actual feasibility check of timetables produced by optimisation models or heuristic processes. It is moreover very relevant the definition of the problem scale: in fact it will be possible to proceed with a macroscopic low detailed model or to define into details the infrastructure for specific local analysis.

Similarly, for robustness analysis or timetable stability the goal is strictly depending upon the considered time window. Also for this case the conflicts resolution criteria, local and global, is essential for goals definition.

Finally, for modifications to the infrastructure, the capacity analysis and its utilisation are depending upon the typology of indicators to be produced by the model: global utilisation rate of a complex station, object by object analysis, analysis on simple nodes, etc.

6.3. Output choice and input definition

Once the goals have been defined, it is essential in the model the choice and the corresponding outputs programming. In fact, the output file may be personalised. In general it is possible to write on it the value assumed during the simulation by structured variables represented by the tokens, by the objects typical parameters, by the actions imposed by these values and, last but not least, the simulation time when a particular token activates a transition.

In a regularity analysis will be e.g. possible to check the run of each train into the stations and the possible cancellations of the routes building calls for the selection of the most critical sections. In a capacity analysis it will be necessary to take into account the frequency of use of each route with a suited counter. In conflicts resolution at nodes it will be necessary to dispose a counter summing the delays generated by perturbing and perturbed trains.

6.4. Inputs definition and data collection

Once the desired outputs have been clarified, it will be possible to include the required model inputs among those corresponding to tokens variables, objects parameters and sub-nets of the management module.

In general the inputs necessary for model running are:

- line topologic and cinematic features;

ti alle variabili delle marche, ai parametri degli oggetti e alle Sotto-Reti del modulo di Gestione.

Gli input necessari al funzionamento del modello sono in generale:

- caratteristiche topografiche e "cinematiche" del tracciato;
- posizione degli enti sul tracciato;
- orari di esercizio reali o fittizi;
- caratteristiche cinematiche dei veicoli.

Le fonti da cui attingere tali dati sono:

- profili di linea del gestore dell'infrastruttura;
- piani schematici delle stazioni;
- fascicoli di linea;
- prefazione generale all'orario di servizio (in Italia);
- schede tecniche dei treni;
- orari di stazione programmati e reali (moduli M53 e M42 in Italia);
- sopralluoghi e rilevazioni integrative.

6.5. Scelta ed eventuale adeguamento delle classi del modulo Comune

Come spiegato per le classi TC e SW, le classi del modulo Comune sono dipendenti essenzialmente dal tipo di sistema di controllo presente in linea ma non dal programma esercito, con la parziale eccezione delle Sotto-reti cinematiche. Ciò significa che su una qualsiasi linea con blocco automatico è possibile riportare tal quale le classi già realizzate. Inoltre, poiché le Sotto-Reti simulano il principio di funzionamento degli enti e non la loro tecnologia, con piccoli adeguamenti è possibile passare da un oggetto di classe TC in linea ad uno in stazione con apparato ACE, ACEI o ACC. Gli adeguamenti riguardano sostanzialmente il diverso modo di gestire il bloccamento e la liberazione, rigida o elastica, degli itinerari.

Nel caso si vogliano simulare altri sistemi occorrerà prevedere l'adozione di altre classi, come ad esempio il pedale conta-assi nel blocco conta-assi, il Radio Block Center e il Nucleo Vitale Periferico nell'ERTMS di II livello, le barriere nel caso di passaggio a livello; tutte classi, queste, già analizzate nell'ambito di ricerche svolte presso il DITS, con la creazione di Portset opportune, che consentano agli oggetti istanziati dalle diverse classi di comunicare fra loro.

6.6. Ricostruzione del Layout

A questo punto, realizzate le classi del modulo Comune con le relative Portset, è possibile ricostruire la Sotto-Rete LAYOUT, ovvero la rappresentazione dell'intera linea fisica, attraverso la connessione dei vari oggetti.

Si tratta in sostanza di clonare per tutta l'infrastruttura la figura base mostrata in figura 12 e, in un secondo momento, definire su un file di testo, che viene letto come azio-

- objects positions along the line;
- actual and fictitious timetables;
- vehicle cinematic performances;

The sources to collect this data are:

- line profiles by the infrastructure managers;
- station lay-outs;
- line timetable books;
- general timetable preface (in Italy);
- trains technical data;
- planned and actual stations timetables (M53 and M42 modules in Italy);
- visits and integrative surveys.

6.5. Choice and possible adjustment of Common Module classes

As already explained for TC and SW classes, the Common Module classes are essentially not depending upon the command and control system, but depending upon the timetable, with the partial exception of cinematic sub-nets. It means that on any automatic block line it is possible to reproduce the same classes. Moreover, the sub-nets simulate the objects operational principle but not their technologies, with minor adjustments it is possible to paste a TC class line object into ACE, ACEI or ACC managed station TC class object. The adjustments mainly concern the different way to manage the locking and the rigid or elastic routes release.

In order to simulate other systems, it will be necessary to adopt other classes, e.g. the axle counter for simple block systems, the Radio Block Centre and the peripheral vital nucleus in Level 2 ERTMS, the level crossings barriers, etc.; all classes developed in various researches at DITS, with the creation of suitable Portsets, allowing the each other communication of various classes instanced objects.

6.6. Lay-out building

Once the Common Module classes and the concerned port-sets have been built, it is possible to build the LAYOUT sub-net, i.e. the representation of the whole line, by linking the various objects.

It means to clone for the whole infrastructure the basic morphology shown in figure 12 and then define on a text file, to be read as simulation initial action, the values of typical parameters of various classes. The action may be speeded up if the link-set number between objects is limited, that means if the objects interfaces are effectively built up.

ne iniziale delle simulazioni, i valori dei parametri caratteristici delle varie classi e tutti gli oggetti da esse istanziate. L'operazione può essere velocizzata se il numero di Linkset fra gli oggetti è limitato, cioè se le interfacce fra gli oggetti sono costruite in modo efficiente.

6.7. Ricostruzione delle Funzioni Itinerario

Il Top Level viene completato con la realizzazione delle Funzioni Itinerario presenti in linea e nelle stazioni. E' dunque necessario conoscere la matrice degli itinerari di ogni stazione per poterla ricostruire con le Reti di PETRI.

Per le funzioni in stazione sarà inoltre necessario costruire una Pagina per ogni stazione in cui poter inserire le informazioni relative alla posizione che i deviatoli devono tenere durante la formazione dei diversi itinerari. Per tutte, invece, occorrerà prevedere una Pagina di annullamento della formazione dell'itinerario.

6.8. Implementazione dell'orario di esercizio e inizializzazione dei treni

Il modulo di Gestione, a differenza del modulo Comune, dipende ovviamente dal programma che si intende esercire. L'implementazione dell'orario avviene nella classe TT, dove non solo vengono creati i treni, ma vengono inizializzate anche le variabili corrispondenti alle marche TRENO e AM dei treni da simulare.

Inoltre, in questa fase è anche necessario impostare le Sotto-reti cinematiche a seconda che si voglia adottare un approccio a velocità costante fra le stazioni o un approccio a velocità variabile. Nel primo caso occorrerà impostare le velocità direttamente sugli oggetti, in maniera che questi, una volta riconosciuta la marca TRENO che si trova in percorrenza su di essi, possano modificare la velocità attuale del treno a seconda della progressiva a cui si trova. Nel secondo caso occorre invece inizializzare il treno e gli oggetti con le proprie caratteristiche cinematiche: velocità massima del treno, accelerazione media, decelerazione media, rango di velocità, pendenza massima in discesa e velocità ammissibile nella sezione occupata e in quella successiva.

6.9. Definizione delle funzioni del Dirigente Centrale

Per completare il modulo di Gestione occorre costruire le Sotto-reti che dirimono i conflitti fra treni nell'oggetto DCM.

Nel modello sono state create tante Pagine quanti sono i binari di stazionamento sulla tratta. Questo approccio è da considerarsi solo uno dei criteri con cui può essere impostato l'oggetto DCM. Si pensi ad esempio all'introduzione di diversi livelli di priorità, con un tipico ordinamento a bolle, in cui se un conflitto non viene risolto dalla prima regola imposta, si passa alla successiva. Così è possibile impostare un ordinamento basato sull'accumulo di ritardo di ciascun treno, sulla differenza fra ritardo programmato ammissibile e ritardo attuale, FIFO, LIFO, ecc.

6.7. Routes functions building

The Top Level is completed by the building of lines and stations routes functions. It is therefore necessary to the routes matrix of each station and to be able to build it by PETRI Nets.

For the station functions it will be necessary to build one page per station, where to include the switches position information for routes building. Moreover, for all functions a route building cancelation page will be foreseen.

6.8. Trains timetable implementation and initialisation

The Management Module is, instead, depending upon the timetable to be run. The timetable implementation happens in TT class, where not only the trains are created but the variables corresponding to TRAIN and AM tokens of trains to be simulated are built up.

Moreover, in this phase it is also necessary to build the cinematic sub-nets based on a constant or variable speed approach between the stations. In the first case it will be planned directly the speed on the objects, in such a way that, once they recognize the TRAIN token running on them, are able to modify the present train speed according to its position. In the second case it is, instead, necessary to initialize the train and the objects with their cinematic performances: maximum train speed, mean acceleration, mean deceleration, speed classes, maximum descending slope and allowed speeds in occupied and following sections.

6.9. Definition of Dispatcher functions

In order to complete the Management Module it is necessary to build the sub-nets solving the conflicts among trains in DCM object.

In the model as many pages as the station tracks have been built. This approach is just one of the possible criteria to implement the DCM object. In fact, with different priority levels, in a typical bubbles classification, should a conflict not be solved by the first rule, the following one is applied. Therefore it is possible to implement a queuing based on the delay acquired by each train, on the difference between tolerated delay threshold and actual delay, on FIFO or LIFO criteria, etc.

Should the global line delay minimisation be among the goals, once the decision trees resulting by § 5.5.2 analysis and the several simulations bringing sub-optimal system answers are established (figure 28), it is also possible to fix the trains priority order, which may vary by the delays acquired by them. Obviously this approach is complex for complex nodes and relevant traffic.

Se fra gli obiettivi c'è quello di minimizzare il ritardo complessivo in linea, una volta noti gli alberi decisionali derivanti dall'analisi mostrata al paragrafo 5.5.2 e derivanti da numerose simulazioni che portano alla definizione di risposte sub-ottimali del sistema, come mostrate in figura 28, è anche possibile impostare l'ordine di priorità fra treni, ordine che può variare al con i ritardi acquisiti da ciascuno di essi. Ovviamente tale impostazione risulta essere molto onerosa se i nodi sono complessi e il traffico è elevato.

6.10. Pianificazione ed esecuzione delle simulazioni

Finita la costruzione del modello, l'analista può decidere il livello di confidenza richiesto per i risultati in base al numero di iterazioni realizzate.

Inoltre, le simulazioni per verificare la robustezza di un orario, possono essere condotte sia in regime regolare, ritardando un treno alla volta e verificando negli output le conseguenze sulla circolazione di tutti i treni nei tre diversi approcci, sia in regime perturbato, nel quale l'imposizione di un ritardo ad un treno per volta è accompagnata dal ritardo comunque già presente nel resto della circolazione e distribuito secondo densità di probabilità derivate dall'analisi della regolarità della linea (usualmente triangolare o lognormale). In quest'ultimo caso occorrerà prevedere una serie di iterazioni con diversi regimi di perturbazione e per ciascuno verificare il ritardo medio complessivo e il livello di confidenza raggiunto.

Durante l'esecuzione delle simulazioni è possibile sia lanciare il programma e attendere il file di output, sia verificare passo passo i comportamenti delle marche in ogni Pagina, compreso verificare quali valori assumono le variabili in ogni istante della simulazione. Ciò rappresenta un rilevante strumento di analisi quando si voglia effettuare analisi di micro-scala.

7. Conclusioni

7.1. Risultati e valore aggiunto della ricerca

I principali risultati della ricerca sono riassumibili nei seguenti punti:

- 1) definizione di una procedura per la realizzazione di un modello di simulazione dell'esercizio ferroviario dipendente dai vincoli imposti dai sistemi di segnalamento e controllo;
- 2) costruzione di uno strumento in grado di:
 - a. progettare un orario ex novo;
 - b. valutare l'influenza dell'inserimento di uno o più treni nel programma di esercizio e dunque aggiornare un orario esistente;
 - c. prevedere gli effetti di una perturbazione, con o senza risposta di un sistema automatico di prece-
 - d. creare, sulla base di criteri modificabili, un albero

6.10. Simulations planning and execution

Once the model building is completed, the analyst may decide the reliability level required for the results, on the basis of the number of performed iterations.

Moreover, the simulations taken to check the timetable robustness may be performed both in a regular regime, by considering the single trains delays and checking in the outputs the feedback on the other trains in the 3 possible approaches, and in perturbed regime, by summing the new acquired delays of the single trains to the basic traffic perturbation distributed according to probability densities derived by regularity analysis (usually triangular or log-normal). In this last case it will be necessary a series of iterations with various perturbations regimes by checking their mean global delay and the reached reliability level.

During the simulations execution, it is possible both to let the program running and to wait for the output file or checking step by step the behaviours of the tokens in each page, including the single values time by time acquired by the variables during the simulation. It represents a powerful analysis tool when a micro-scale analysis is required.

7. Conclusions

7.1. Results and research added value

The main research results may be summarised in the following items:

- 1) definition of a procedure to build up a railway operation simulation model depending upon the constraints imposed by signalling and control systems;
- 2) building up of a tool capable of:
 - a. ex novo timetable planning;
 - b. feasibility check for additional trains introduction in an existing timetable;
 - c. perturbations effects forecast, with and without the operation of an automatic overtaking system;
 - d. featuring off-line decision tree for conflicts management on the basis of a fixed timetable.

Moreover, additional positive secondary effects of the research are:

- 3) possibility to develop infrastructure use analysis (over-loaded and under-loaded objects) to be performed, e.g., in devices maintenance planning;
- 4) possibility to analyse nodes and lines capacity;
- 5) possibility to easily check capacity and regularity levels depending upon partial infrastructural im-

di decisioni off-line per la gestione dei conflitti sulla base di un dato programma d'esercizio.

Inoltre, come "esternalità positive" della ricerca si citano i seguenti risultati collaterali, che esulano dall'obiettivo principale della ricerca:

- 3) possibilità di sviluppare analisi d'uso degli impianti (enti sovra-utilizzati o sotto-utilizzati) da effettuare, ad esempio, nella realizzazione di un programma di manutenzione mirato per ciascun ente;
- 4) possibilità di analizzare la capacità nei nodi e in linea;
- 5) possibilità di verificare i nuovi livelli di capacità e regolarità dipendenti da un cambio parziale dell'infrastruttura (ad esempio, un nuovo binario) in maniera poco onerosa se già si possiede il Top Level precedente dell'infrastruttura.

Rispetto al panorama attuale dei modelli di simulazione, quello descritto si distingue per il fatto che la simulazione a macro-scala della percorrenza dei treni è accompagnata, o per meglio dire è generata, dalla simulazione a micro-scala degli stati interni di ciascun ente appartenente all'infrastruttura e ai sistemi logici che controllano la marcia del treno. Ciò consente un controllo simultaneo della simulazione sia a macro-scala (quale azione compie il treno?) sia a micro-scala (in che stato si trova il singolo ente ferroviario?)

Un altro valore aggiunto da non trascurare è infine il fatto che il modello di simulazione è stato realizzato ex-novo. Tale software peraltro traduce le Reti di PETRI estese realizzate in un "programma sorgente" scritto in C, riutilizzabile in altre applicazioni e modificabile, cosa che consente la sostanziale indipendenza dal sistema di segnalamento.

7.2. Criticità e sviluppi futuri

Durante lo sviluppo della ricerca, sono stati incontrati diversi punti critici; solo per alcuni la risoluzione è rimasta in una fase embrionale. Per ciascuno di questi è stato individuato un potenziale percorso di risoluzione, da sviluppare in ricerche successive.

- *Rappresentazione grafica del layout ferroviario poco dettagliata.* Il software Faber© rappresenta gli oggetti tutti allo stesso modo indipendentemente dalla classe, distinguendoli solo per la codifica del nome ("nome_oggetto:tipo_classe"); ciò significa avere una bassa potenza di rappresentazione che, con i piani schematici di nodi complessi in cui centinaia di enti sono presenti in un'unica stazione, può presentare qualche problema nel seguire visivamente la simulazione, la quale evolve accendendo e spegnendo gli oggetti secondo una precisa codifica di colori. Una soluzione potrebbe essere quella di utilizzare il codice C generato dal modello per creare uno strumento software indipendente anche da Faber con un'interfaccia-utente più "appetibile" e soprattutto più rispondente alla simbologia tradizionale del mondo ferroviario.
- *Onerosità nella costruzione iniziale del layout e nella inizializzazione dei treni dell'orario.* Se è vero che il modulo Comune può essere preso tal quale per riprodurre li-

provement (e.g. new tracks) on the basis of the existing infrastructural Top Level.

In comparison with the state-of-the-art, the proposed model introduces the macro-scale simulation of trains running accompanied/generated by the micro-scale simulation of single signalling and control systems devices (objects) occupation states. It allows a simultaneous control of both macro-scale (actions performed by the trains) and micro-scale (single devices status).

A further added value is the innovative version of the simulation model, therefore the software is able to translate the extended Petri Nets in a C language source program, suitable to be modified reused in different applications, that means a substantial independence upon the signalling system.

7.2. Critical aspects and future developments

During the research development various critical points have been encountered; for some of them only the resolution remained in an embryonic phase, for the others it was identified a potential resolution path to be further developed in future researches.

- *Low detail lay-out graphic representation.* The Faber© software represents all the objects in the same shape without distinction by class, but only by name coding; it means a low power of representation, with difficulties to look at the simulation development in complex nodes layouts including hundreds of devices switching rapidly from a colour to another one. A possible solution could be to use the C code generated by the model for building a software tool independent by Faber with a more user friendly interface based on traditional railway symbologies.
- *Complexity in the initial lay-out scheme building and trains initialisation in the timetable.* The Common Module may be used as such to reconstruct lines model managed by usual signalling and control systems, nevertheless Top Level and Management Modul are strictly depending upon the line lay-out and the timetable to be simulated. On this aspect it is not possible to speed up the simulation building procedure, due to the need to initialise the trains (by introducing their cinematic performances and assigned routes) for avoiding relevant simplifications (e.g.: constant speed by train classes). This complexity is compensated by the speed of the simulations, once the model is operating. In fact, should not be necessary to follow the simulation step by step, but just analyse the final output, the simulation speed depend upon the number of rows to be written in this file. These are once a time depending upon the amount of information (system transitions) to be recorded. E.g. a

nee con sistemi di segnalamento e controllo usuali, è altrettanto vero che la costruzione del Top level e del modulo di Gestione sono necessariamente dipendenti dalla linea e dal programma di esercizio che occorre simulare. Su questo punto, non si ritiene possibile velocizzare la fase di costruzione della simulazione che si intende espletare, nel senso che è impossibile prescindere dalla inizializzazione dei treni (cioè dall'introduzione dei parametri caratteristici della cinematica e del percorso che i treni dovranno seguire) a meno di incorrere in semplificazioni notevoli (ad esempio: velocità costanti per classi di treno). Tale onerosità iniziale è compensata dalla estrema velocità con cui si possono svolgere alcune simulazioni mirate, una volta costruito il modello. Infatti, nel caso in cui non si desidera seguire la simulazione passo per passo ma solo disporre dell'output finale, la velocità di simulazione dipende dal numero di righe che devono essere scritte in tale file. Queste dipendono a loro volta dal numero di informazioni che si desidera registrare, essendo ogni riga una transizione del sistema. Ad esempio, una simulazione di un periodo di traffico di circa 3 ore, con circa 60 treni lungo una linea di circa 100 km con 10 stazioni, può avere una durata dell'ordine del minuto se si registrano sul file di output solo i tempi di passaggio alla stazione per ciascun treno.

- *Gestione dei conflitti semi-automatica.* Con la dizione "semi-automatica" si intende sottolineare il fatto che, nel caso in cui si presenti un conflitto di circolazione fra due o più treni, il modello è strutturato per poter individuare una soluzione automatica, in base a diversi criteri (le simulazioni dipendono dalla categoria del treno, ma nulla osta l'imposizione di altri criteri) e poi valutarne gli effetti di propagazione dei ritardi nel sistema. Non può, allo stato attuale, effettuare la scelta automaticamente fra più soluzioni di un conflitto, calcolando in anticipo per tutte quale sarà l'esito della propagazione dei ritardi. Quest'ultimo processo può essere calcolato, una volta individuato il conflitto, solo imponendo le diverse soluzioni in simulazioni distinte, ad esempio cambiando di volta in volta il criterio di selezione, e verificando successivamente, "manualmente", la soluzione che meglio soddisfa il criterio di minimizzazione del ritardo complessivo del sistema. Inoltre, poiché ad ogni soluzione di conflitto potrebbe verificarsi un nuovo, per ogni soluzione proposta è comunque necessario un numero di iterazioni dipendente essenzialmente dalla complessità della linea e dalla densità del traffico. Ciò, in linee con nodi non eccessivamente complessi potrebbe rappresentare un problema di facile superamento attraverso la costruzione di un albero decisionale di confronto fra treni, da inserire nel modello in maniera che il sistema che gestisce le precedenze sappia in anticipo come comportarsi. Di contro, in nodi complessi la costruzione dell'albero decisionale richiederebbe un numero immenso di simulazioni, tale da rendere poco percorribile questo tipo di soluzione.

3 hours timetable simulation including 60 trains along a line of about 100 km with 10 stations is about 1 minute long when the transit time at stations of each train are only recorded.

- *Semi-automatic conflicts management.* Semi-automatic diction means that the system is structured to identify automatically solutions for trains conflicts on the basis of selected criteria (based on trains categories or others) and than to evaluate the delays propagation effects. Nevertheless at the present state it is not able to choose automatically among the possible solutions, by calculating in advance the delays propagation effects. This process may be calculated, for each conflict, just imposing the various solutions in different simulations and than checking manually the solution satisfying as much as possible the global delay minimisation criteria. Moreover, each solution may generate a new conflict, therefore for each solution is anyway necessary a number of iterations essentially depending upon the line complexity and the traffic density. On lines of limited complexity it might be a problem easy to be overtaken by building a decision tree comparing solutions and suggesting the most effective one. On the contrary in complex nodes the building of a decision tree would require such a large amount of simulations, to make this approach poorly effective.

The future research development should cover mainly the possible applications of the model to timetable stability analysis (how a timetable reacts to a system perturbation?) and the management of conflicts.

For the first problem would be interesting to assess the influence of a stochastic approach, with the possible substitution in Faber© of running times deterministic calculation with various probability distributions and, moreover, to compare this approach with the most recent published researches based on Max-plus algebra by GOVERDE [10], on stochastic only approach by YUAN [18], pseudo-optimal procedures based on PESP (Periodic Event Scheduling Problem) models [3] and other simulative approaches [14].

For the second aspect it would be strongly innovative a research aimed to the final building of a simulation model adding to the already existing functionalities described above the real time rescheduling function.

APPENDIX 1

Theory, Algebra and graphic formalism of PETRI Nets

The German mathematician C.A. PETRI developed in 1955 a theory for the building of formal tool for model-

Gli sviluppi futuri della ricerca dovrebbero riguardare soprattutto i temi dell'applicazione del modello alle analisi di stabilità degli orari ferroviari (come reagisce un orario ad una perturbazione del sistema?) e alla gestione dei conflitti di circolazione.

In particolare per il primo tema sarebbe interessante valutare l'influenza di un approccio stocastico alla questione, con la sostituzione, possibile in Faber[®], del calcolo deterministico dei tempi di percorrenza con diverse distribuzioni di probabilità ed inoltre, vista l'estrema attualità della tematica, effettuare una comparazione con le ultime ricerche pubblicate sull'argomento, quali le analisi con la Max-plus algebra di GOVERDE [10], gli approcci esclusivamente statistici come quello di YUAN [18], approcci con soluzioni pseudo-ot-time come quelle discendenti dai modelli PESP (*Periodic Event Scheduling Problem*) [3], altri approcci simulativi [14].

Per quel che riguarda il secondo aspetto, avrebbe un forte contenuto innovativo una ricerca che avesse come obiettivo finale la realizzazione di un modello di simulazione che aggiunga alle funzionalità presenti in quello descritto in questo articolo quella del *real time rescheduling*.

APPENDICE 1

Teoria, Algebra e formalismo grafico delle Reti di PETRI

Il matematico tedesco C. A. PETRI sviluppò nel 1955 una teoria per la costruzione di uno strumento formale di modellazione ed analisi di sistemi concorrenti complessi. La teoria, che verrà perfezionata in seguito con la collaborazione di A. W. HOLT e J. DENNIS, prenderà appunto il nome di Reti di PETRI [21].

Le Reti di PETRI sono un linguaggio formale costituito da una semplice rappresentazione mediante un grafo di stato dinamico che sottintende una formulazione analitico-matematica.

Gli elementi costitutivi di una Rete sono quattro:

- i *posti*, che rappresentano gli stati del sistema rappresentato e sono raffigurati con dei cerchi;
- le *transizioni*, che rappresentano lo svolgersi di una determinata azione all'interno del sistema e sono raffigurati mediante rettangoli;
- gli *archi*, che sono l'elemento di collegamento fra posti e transizioni;
- le *marche*, che sono l'elemento dinamico del sistema, indicano quando un posto è attivo e sono rappresentate da semplici punti inseriti all'interno dei posti.

Gli archi del grafo connettono sempre e solo i posti con le transizioni ed il loro orientamento definisce la relazione "input - output" che si genera fra le coppie posto - transizione.

Lo stato d'esercizio attuale è definito dalla posizione delle marche, il cui movimento tra i posti consente una simulazione visiva degli avvenimenti per gestire l'analisi del sistema.

ing analyzing complex concurring systems. The theory, which was later upgraded in cooperation with A.W. HOLT and J. DENNIS, just took the name of PETRI Nets [21].

The PETRI Nets are a formal language based on a simple representation by a dynamic state-graph, which includes an analytical matrix based formulation.

The network composing elements are four:

- *places*, corresponding to the system states and represented by circles;
- *transitions*, corresponding to the development of an action within the system and represented by triangles;
- *arcs*, linking elements between places and transitions;
- *tokens*, dynamic system elements, expressing the places activation and represented by simple points included within the places,

The graph arcs link always and only the transitions and their orientation defines the input-output relationship generated between places and transitions.

The operational state is defined by the position of tokens, whose movement among places allows a visual simulation of events for system analysis management.

This movement is due to a state shift (transition activation) whose completion in subject to a single constraint: all transition input places must be marked. In this case the transition is activated and able to fire. The firing activate the removal of tokens from input places and the following addition of a token in each output place, by determining a new system state, as shown in figure A1.

The links may be also multiple, when are represented by more than one arc. To each arc may be also associated a *weight* w , integer positive value representing the amount of tokens to be contained in the transition to occupy the following place and activate the following transition.

From an algebraic point of view, the PETRI Nets may be formalised as a graph:

$$G = (N, A),$$

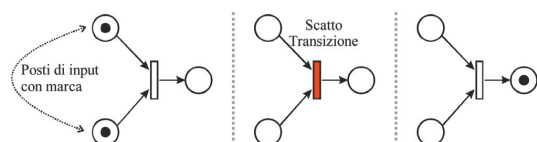


Fig. A1 – Elementi di una Rete di PETRI di basso livello. *Elements of a low level PETRI Net.*
(Posti di input con marca = *Input places with token* - Scatto transizione = *transition firing*).

Tale spostamento è frutto di un passaggio di stato (scatto di una transizione), il cui compimento è soggetto ad un unico vincolo: tutti i posti di input della transizione devono essere marcati. In questo caso la transizione è abilitata e può scattare. Lo scatto comporta la rimozione delle marche dai posti di input e la conseguente aggiunta di una marca in ogni posto di output, determinando così un nuovo stato del sistema, come mostrato in fig. A1.

Le connessioni possono essere anche multiple, nel caso in cui vengono rappresentate da più di un arco. Ad ogni arco può anche venire associato un *peso* w , ovvero un valore intero positivo, che rappresenta il numero di token che la transizione deve contenere affinché il posto successivo possa essere occupato e che il posto deve contenere affinché la transizione successiva sia abilitata.

Dal punto di vista algebrico, le reti di PETRI possono essere formalizzate come un grafo del tipo:

$$G = (N, A),$$

in cui $N = (n_1, n_2, \dots, n_s)$ è l'insieme dei nodi e $A = (a_1, a_2, \dots, a_r)$ è l'insieme degli archi orientati che li collegano.

L'insieme N dei nodi è costituito dai due sottoinsiemi dei posti $P = (p_1, p_2, \dots, p_n)$ e delle transizioni $T = (t_1, t_2, \dots, t_m)$, tali che $N = P \cup T$ e $P \cap T = \emptyset$.

Un posto p_i è detto *posto di input* di una transizione t_j se l'arco che li collega è definito da $(p_i, t_j) \in A$. L'insieme $I(t_j) = \{p_i : (p_i, t_j) \in A\}$ è detto *insieme dei posti di input* della transizione t_j .

Un posto p_o è detto *posto di output* di una transizione t_j se l'arco che li collega è definito da $(t_j, p_o) \in A$. L'insieme $O(t_j) = \{p_o : (t_j, p_o) \in A\}$ è detto *insieme dei posti di output* della transizione t_j .

Ovviamente un posto può essere contemporaneamente di input per una transizione e di output per un'altra.

Più posti possono contemporaneamente contenere delle marche, ovvero essere attivi. Una configurazione della rete con delle marche assegnate a dei posti è definita *marcatura della rete*. La marcatura è definita da un vettore $\mu = (\mu_1, \mu_2, \dots, \mu_n)$, che esprime per ogni posto p_i il numero di marche ad esso assegnato. La Rete di PETRI marcata è quindi caratterizzata come:

$$M = (G, \mu).$$

Per definire una Rete di PETRI è sempre necessario specificare la sua marcatura iniziale, ovvero il suo stato iniziale, poiché ogni marcatura successiva è strettamente dipendente dalla precedente e dunque, a parità di rete, due marcature iniziali diverse portano ad evoluzioni del sistema potenzialmente diverse.

Molti sono gli eventi riproducibili con uno strumento apparentemente così semplice. Nella fig. A2 sono mostrate le situazioni fondamentali rappresentabili con le Reti di PETRI di basso livello.

where $N = (n_1, n_2, \dots, n_s)$ is the node set and $A = (a_1, a_2, \dots, a_r)$ is the oriented arcs set linking them.

The nodes set N includes the sub-sets $P = (p_1, p_2, \dots, p_n)$ and the transitions $T = (t_1, t_2, \dots, t_m)$, so that is $N = P \cup T$ e $P \cap T = \emptyset$.

A place p_i is named *input place* of a transition t_j if their linking arc is defined by $(p_i, t_j) \in A$. The set $I(t_j) = \{p_i : (p_i, t_j) \in A\}$ is named t_j transition input places set.

A p_o place is named *output place* of a transition t_j if their linking arc is defined by $(t_j, p_o) \in A$. The $O(t_j) = \{p_o : (t_j, p_o) \in A\}$ set is named output places set.

Obviously a place may be simultaneously input of a transition and output for another one.

Several places may simultaneously be activated (containing tokens). A network configuration with tokens assigned to places is defined *network marking*. The marking is defined by a vector $\mu = (\mu_1, \mu_2, \dots, \mu_n)$, expressing for each place p_i the amount of tokens assigned to it. The marked PETRI Net is therefore represented by

$$M = (G, \mu)$$

In order to define a PETRI Net it is always necessary specify its initial marking, i.e. its initial state, because any following marking is strictly depending upon the previous one and therefore, by network parity, two initial different markings bring towards potentially different system evolutions.

Many events may be reproduced with such an apparently simple tool. In figure 2 the key situations, which may be represented by low level PETRI Nets, is shown.

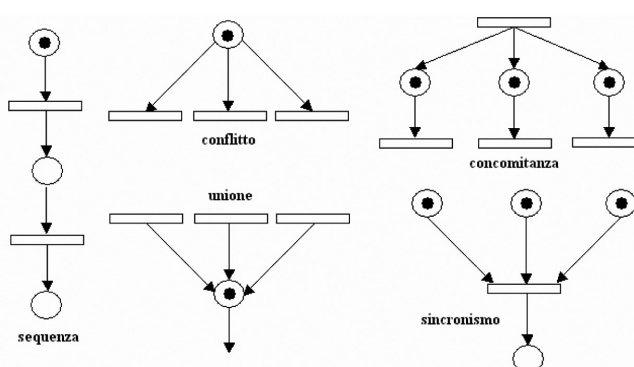


Fig. A2 – Situazioni fondamentali in una Rete di PETRI. Key situations in a PETRI Net. (conflitto = conflict - concomitanza = simultaneity - sequenza = sequence - unione = union - sincronismo = synchronism).

APPENDICE 2

Classi di oggetti e sotto-reti

Classe TC (Circuito di binario)

Il nome di tale classe potrebbe ingannare sul suo reale utilizzo nel modello sviluppato. Infatti essa è rappresentativa dell'ente circuito di binario per quel che concerne le funzioni logiche da svolgere, ma ad essa sono anche associati parametri fisici corrispondenti alla tratta di linea che lo stesso circuito deve proteggere, spesso coincidente con una sezione di blocco. Dunque ognuno degli oggetti istanziati dalla classe TC rappresenta un circuito di binario più la sezione di linea immediatamente a valle, con la sua lunghezza, la sua pendenza e le sue caratteristiche di rango di velocità ammissibili.

La classe TC è dunque dotata di suoi parametri che assumono valori diversi nei vari oggetti della classe; tali valori possono essere impostati su un file di testo richiamato al momento della simulazione. I parametri impostati nel file per ogni oggetto appartenente alla classe TC sono i seguenti:

- lunghezza della sezione di blocco;
- pendenza massima in discesa nella sezione;
- pendenza massima in discesa nella sezione successiva;
- velocità massima di linea sulla sezione per il rango x di velocità;
- velocità massima di linea sulla sezione successiva per il rango x di velocità.

All'aumentare del numero di ranghi, aumenta il numero di parametri; in particolare nel modello sono stati considerati i 4 ranghi di velocità presenti sulle linee italiane (A, B, C, P) e dunque i parametri sono 11 ($3 + 4 \times 2$).

È importante notare come le Sotto-reti, in questa come nelle altre classi, tengono conto solo degli aspetti che possono influenzare significativamente la regolarità del traffico. Ciò significa che le funzioni logiche riprodotte non hanno lo scopo di rappresentare passo dopo passo l'esatta corrispondenza del passaggio di informazioni all'interno dell'ente, quanto piuttosto mettere in luce gli effetti macroscopici di tale passaggio. Così, ad esempio, dell'ente circuito di binario sono rappresentate le condizioni logiche che permettono la verifica di libertà del suo tratto di linea, il bloccamento e il rilascio dello stesso, la percorrenza con diverse condizioni cinematiche, ma non le condizioni che hanno un'influenza soprattutto sulle modalità microscopiche con cui un ente funziona, come ad esempio la registrazione dell'ente in un itinerario, le modalità con cui può essere costantemente controllato da un ente centrale o il suo funzionamento in condizioni di inclusione/esclusione. Questi ultimi aspetti sono comunque tenuti in considerazione con l'attribuzione di un tempo aggiuntivo da inserire nella simulazione (ad esempio quello, peraltro esiguo, di registrazione di un itinerario prima della formazione dello stesso). In sintesi, si è rappresentato un ente attraverso "ciò che fa" e non attraverso "come lo fa".

APPENDIX 2

Objects classes and sub-nets

TC Class (Track Circuit)

The name of this class could let to misunderstand its function in the developed model. In fact, it represents not only the track circuit object but also the physical parameters corresponding to the line section protected by it, possibly corresponding to a single block section. Therefore each object instantiated by TC class represents a track circuit plus the following line section, with its length, slope and allowed speed class.

The TC class is anyway characterised by proper parameters assuming different values for the various objects of the class; these values may be initialised on a text file recalled during the simulation. The input parameters of any TC class object are the following:

- block section length;
- maximum descending slope in the section;
- maximum descending slope in the following section;
- maximum line speed for x speed class on the section;
- maximum line speed for x speed class on the following section;

The parameters increase with the class speeds; in particular in the model have been considered the 4 class speeds adopted by Italian railways (A, B, C, P) with a global amount of 11 ($3 + 4 \times 2$) parameters.

In this class and in the others the sub-nets take into account only the aspects which may significantly influence the traffic regularity. It means that the reproduced logical functions have not the scope to represent step by step the precise information flow, but to highlight macroscopic effects of this flow. On this basis with the track circuit object are represented: the logical conditions allowing the freedom check of its section, locking and release of the section itself and running with different cinematic conditions, but not the conditions influencing above all the microscopic object operation (e.g. route recording, permanent monitoring by a control centre, exclusion/inclusion into operation). These aspects are anyway considered by assigning an additional time into simulation (e.g. the short route registration time before its building). An object is finally represented by *what* it does instead *how* it does it.

SW class (Switch)

This class reproduces the logical conditions allowing the switch operation. Actually the operation of a single switch object is simulated by two SW objects jointly representing the entire switch structure.

Classe SW (Deviatoio)

Tale classe riproduce le condizioni logiche che permettono il funzionamento di un deviatoio. In realtà, il funzionamento di un singolo ente deviatoio è simulato mediante l'utilizzo di due oggetti della classe SW, rappresentanti insieme l'intera fisicità di un deviatoio.

La classe SW è composta da Sotto-reti totalmente indipendenti dalla linea in cui sono istanziati i suoi oggetti e dal programma di esercizio dei treni che la attraversano.

Anche la classe SW è dotata di suoi parametri che assumono valori diversi nei vari oggetti della classe; tali valori possono essere impostati su un file di testo richiamato al momento della simulazione. I parametri impostati nel file per ogni oggetto appartenente alla classe SW sono i seguenti:

- lunghezza della sezione di blocco;
- velocità massima sopra il deviatoio.

Quindi, a differenza della classe precedente, sopra un oggetto SW tutti i treni vanno alla stessa velocità, dipendente proprio dal tipo di deviatoio. Tale ipotesi viene modificata, variando la velocità ammessa sul deviatoio, quando si desidera effettuare una simulazione con approccio a velocità costante fra le stazioni; in questo caso la velocità inserita nel parametro del deviatoio sarà la stessa già in possesso dalla marca TRENO.

Altre classi utilizzabili nel modello

Le due classi del modulo Comune presentate nei paragrafi precedenti sono quelle che, da sole, consentono di effettuare le analisi sulla regolarità dell'esercizio su linee e stazioni dotate degli usuali sistemi di segnalamento e controllo (in particolare quelli presenti sulla Roma-Formia, tratta utilizzata per calibrare il modello).

Se si volesse costruire un modello estremamente versatile e che funzionasse su qualsiasi linea e soprattutto anche con finalità diverse dalle analisi di regolarità, altre classi di oggetto andrebbero previste. Presso il DITS, da diversi anni sono in atto ricerche che hanno visto finora la costruzione di diverse classi di oggetto nell'ambito di cinque diverse tesi di laurea [15] [16] [17] [19] [20] ed una tesi di dottorato [12], che, eccetto una in cui il modello qui presentato è stato applicato nelle analisi di capacità della stazione Roma Ostiense [8], hanno avuto lo scopo di esplicitare con le Reti di PERTI i principi di funzionamento di diversi sistemi di segnalamento e controllo.

Le classi finora rappresentate in queste ricerche, oltre le due prima citate, sono:

- Segnale di avviso;
- Segnale di blocco;
- Sezione di blocco;
- ERTMS Radio Block Center;
- ERTMS Nucleo Vitale Periferico;

The SW class is composed by sub-nets not depending upon the line where their objects are instantiated and the trains timetable running.

Also the SW parameters assume different values for the various class objects; these values may be included on a text file recalled during the simulation. The parameters of any SW object are the following:

- block section length;
- maximum speed over the switch.

Therefore, differently by the previous class, on a SW object all trains will run at the same speed, strictly depending just upon the switch typology. This hypothesis is modified, by varying the allowed running speed over the switch, when a simulation with constant speed between the stations is run; in this case the speed imposed by the switch parameter will be the same possessed by the TRAIN token.

Other classes to be used in the model

The two classes of Common Module presented in previous paragraphs are those allowing alone to perform operation regularity analysis on lines and stations equipped with usual signalling and control systems (e.g. those operating on Roma-Formia lines used for model tuning).

An extremely versatile model capable to simulate any line and with different goals other classes of objects should be foreseen. At DITS many years of research activities allowed, within five Degree Thesis [15] [16] [17] [19] [20] and a PhD Thesis [12], to develop many classes of objects, capable to represent the functional principles of various signalling and control systems operating on various lines sections and Roma Ostiense [8] station.

The classes of objects developed until today additionally to those above, are:

- advance signal;
- block signal;
- block section;
- ERTMS Radio Block Centre;
- ERTMS Vital Peripheral Nucleus;
- Eurobalise;
- axles counter;
- level crossing barrier.

Details on these representation may be extracted by studies listed in references. An integration into the model of these objects is possible by building suitable interfaces with TC and SW classes objects.

TT class (Timetable)

In the Timetable object of the Management Module are designed the sub-nets managing trains departures ac-

- Eurobalise;
- Pedale conta-assi;
- Barriera di passaggio a livello.

Per un dettaglio su tali rappresentazioni si rimanda agli studi citati in bibliografia, sottolineando come un'integrazione nel modello di tali oggetti sarebbe possibile previo costruzione di opportune interfacce con gli oggetti di classe TC e SW.

Classe TT (Orario)

Nell'oggetto Orario del modulo di Gestione sono disegnate le Sotto-reti che gestiscono le partenze dei treni secondo l'orario di esercizio voluto. Poiché tale modulo è gestibile all'interno di Fa-ber[®] con un file parametrico, è possibile di volta in volta modificare l'orario semplicemente agendo sui parametri del file. Tale opportunità si rivela molto utile nello studio di orario perturbato, dove ad esempio è possibile imporre un determinato ritardo ad un treno e verificare attraverso la simulazione cosa accade alla circolazione.

L'oggetto si compone di una Sotto-rete denominata GEN_TRENI e di una serie di Sotto-reti, ognuna del tipo a_XX_YYY, con le X indicanti il numero identificativo della stazione dalla quale il treno entra nel modello e le Y il nome di tale stazione.

La prima azione compiuta dal programma in fase di simulazione è la generazione istantanea di tutti i treni che verranno simulati. Questa azione è gestita dalla Sotto-rete mostrata in fig. A3, che ha appunto il compito di generare tutti e solo i treni dei quali si intende analizzare la circolazione. La Sotto-rete è inizialmente marcata nel posto START. La marca che attiva la transizione GEN_TRENI è di tipo TRENO ma è, in questo momento della simulazione, una struttura dati "vuota", nel senso che i treni vengono inizialmente generati anonimi; sarà la transizione stessa a porre un numero identificativo su ogni marca rilasciata e solo in un'altra Sotto-rete avverrà la "creazione" vera e propria del treno, mediante l'assegnazione dei suoi parametri caratteristici.

Le transizioni concorrenti ZZ e WW servono a stabilire quali identificativi di treno possono passare al posto comune Z per "viaggiare" in altre Sotto-reti e quali invece devono essere eliminati; entrambe puntano al posto START in maniera da permettere la generazione di nuovi treni fino ad una condizione di arresto posta in ZZ.

L'assegnazione delle proprietà delle varie marche TRENO rappresentative di altrettanti treni fisici, in pratica la loro inizializzazione, avviene in varie Sotto-reti simili a quella mostrata in fig. A4. Dal posto comune Z le varie transizioni si attivano nel momento in cui riconoscono l'identificativo del treno da inizializzare e assegnano i valori, permanenti o modificabili durante la simulazione, alle variabili semplici che costituiscono la marca TRENO. In particolare è in queste transizioni che vengono definite le proprietà intrinseche del treno e gli itinerari che dovrà seguire lungo la sua marcia.

cording to the planned timetable. This module may be managed within Faber[®] with a parametric file, therefore it is possible once a time to modify the timetable simply acting on file parameters. This possibility is very useful for studying perturbed timetables by assigning to trains fixed delays and check, by simulation, the traffic development. The objects is composed by a sub-net named GEN_TRENI and a set of sub-nets of XX_YYY type, with X meaning the Id station number and Y the name of this station.

The first action performed by the program during the simulation is to generate instantaneously all trains to be simulated. This action is managed by the sub-net shown in figure A3, which is actually devoted to generate just all the trains to be considered in traffic analysis. The sub-net is initially marked in START place. The token activating the transition GEN_TRENI is a TRAIN token, but, in this simulation phase, it is an empty structure, because the trains are initially generated anonymously; it will be the transition itself to assign an Id number to each active token and only in another sub-net the effective train creation will be performed by fixing its characteristic parameters.

The concurrent transitions ZZ and WW are required to fix which train Id may run on common place Z towards other sub-nets and which must be eliminated. They are both oriented to START to allow new trains generation until the stop condition in ZZ.

The assignment of features to various TRAIN token representing physical trains, their initialisation, happens in various sub-nets similar to that shown in figure A4. From common place Z the various transitions are activated as soon as they recognize the train Id to be initialised and assign the values, fixed or variable during the simulation, to the simple variable representing the TRAIN token. It is particularly in these transitions that the intrinsic train characteristics and the routes to be followed during its running are fixed.

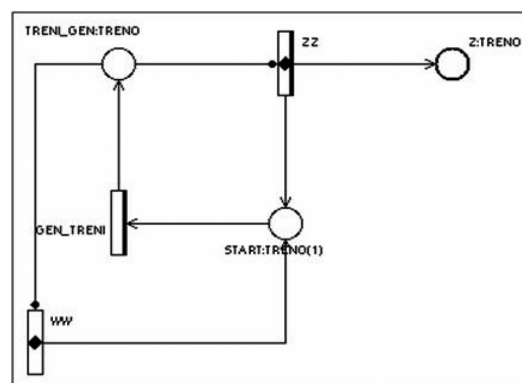


Fig. A3 – Sotto-rete GEN_TRENI della classe TT. GEN_TRENI sub-net of TT class.

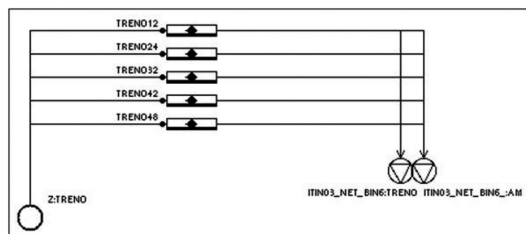


Fig. A4 – Sotto-rete a_03_Campoleone della classe TT per i treni entranti fra le 7 e le 10 sulla linea Roma-Formia dalla stazione di Campoleone. *a_03_Campoleone sub-net of TT class for trains approaching between 7.00 and 10.00 the Roma-Formia line from Campoleone station.*

Ciò avviene generando due diverse marche, una AM ed una TRENO, da depositare in altrettanti posti esterni di output. Da qui le due marche entreranno in una particolare Sotto-rete rappresentante la prima funzione itinerario che dovranno percorrere e daranno così vita all'inizio della marcia del treno.

È importante notare come la generazione dei treni sia istantanea, ovvero avvenga al “tempo zero” della simulazione, mentre lo scatto delle transizioni TRENOXX può avvenire solo dopo applicazione di un ritardo di scatto che simula di fatto l'orario di partenza dei treni. Tutti i ritardi di scatto sono peraltro gestiti per via parametrica, attraverso un'apposita funzione in Faber®, e possono essere modificati prima di ogni simulazione.

Dunque il modello non ha un vero e proprio orario di esercizio sul tipo dei moduli M53, nel senso che è presente nel modello un “orario di entrata in linea”, ove questa entrata può avvenire nelle stazioni di testa e coda della linea e nelle eventuali stazioni con presenti delle linee diramate. Il resto dell'orario di esercizio (compresi i vari M53 nel caso di riproduzione dell'orario teorico) viene ricavato dagli esiti della simulazione.

Classe MA (DCM - Dirigente Centrale di Movimento)

Il modello ha la possibilità di funzionare considerando un sistema di precedenza automatiche per diverse tipologie di treno. L'oggetto DCM del modulo di Gestione ha il compito di gestire tali precedenza, confrontando in punti prestabiliti della linea il treno che precede con il treno che segue e, obbedendo ad un pre-impostato ordine di priorità, comandando al treno che precede un cambio di itinerario con fermata alla prima stazione incontrata nel caso in cui il treno che segue abbia diritto di precedenza e sia impossibilitato a raggiungere la propria velocità d'esercizio proprio per la presenza del treno che lo precede.

È importante sottolineare come il modello può funzionare anche in modalità “normale”, cioè non attivando il Dirigente Centrale. In pratica, per configurare il modello come vero e proprio strumento di supporto alle

This happens by generating two different tokens, AM and TRAIN, to be positioned in output places. From here these tokens will enter a particular sub-net representing the first route function to be run for starting train running.

The train generation is instantaneous, starting the simulation at *zero time*, as TRENOXX transitions may happen after an initial delay application only, simulating train departure time. All delays are anyway managed parametrically, by a specific Faber® function and may be modified before each simulation.

Therefore the model has not a specific timetable on a M53 module, where a time for train entering in line from initial, final or branching stations is fixed. The remaining operational timetable (including various M53 modules for reproducing planned timetable) is extracted by simulation results.

MA Class (DCM – Central Dispatcher)

The model may operate with an automatic overtaking system for different train typologies. The DCM object in the Management Module is devoted to manage these overtaking, by comparing in specific line sections preceding and following train on the basis of a fixed priority order and ordering to the first train a stop in the first station as soon as the second cannot reach its desired operational speed due to the first train position.

The model may operate also in normal mode, without activating the DCM Module. Practically for the most effective model configuration as decision support tool the perturbations management is an open function, which may run manually or by an automatic system working by train classes, capable to calculate and compare actual and simulated delays.

The DCM object built up on Roma-Formia line includes the subnet PRECEDENZE, shown in figure A5, and many

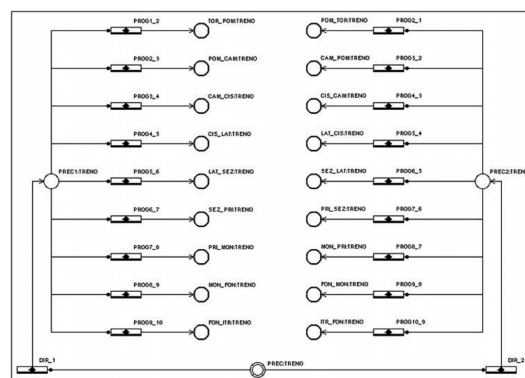


Fig. A5 – Sotto-rete PRECEDENZE della classe MA per la linea Roma-Formia. *PRECEDENZE sub-net of MA class for Roma-Formia line.*

decisioni, si è lasciata la possibilità di gestire le precedenza, in caso di perturbazioni dell'orario di esercizio, sia manualmente, come accade spesso in una Dirigenza Centrale di Movimento, sia con un sistema automatico per classi di treni, così da poter più efficacemente confrontare la gestione dei ritardi con orari reali o simulati.

L'oggetto DCM costruito sulla Roma-Formia è formato dalla Sotto-rete PRECEDENZE mostrata in fig. A5 e da molte altre Sotto-reti che gestiscono possibili conflitti, del tipo di quella in fig. A6.

Nella prima una marca TRENO entra dal posto esterno di input PREC e, in base alla direzione di marcia e alla progressiva alla quale il treno si trova, viene smistata verso una delle altre Sotto-reti attraverso i posti comuni presenti al centro della figura.

La Sotto-rete 02_TOR_POM in fig. A6 gestisce uno dei possibili conflitti che si presentano in circolazione: la presenza di due treni che si contendono una stessa risorsa (un binario), con uno che procede davanti all'altro.

Per la descrizione del funzionamento della Sotto-rete insieme al resto del modello si rimanda al capitolo 4. Qui si vuole notare come i due posti interni siano occupati dalle due marche TRENO, una rappresentante il treno che precede, l'altra quello che segue; lo scatto di una delle tre transizioni concorrenti, in particolare PRECEDENZA02A, determinerà l'assegnazione di una priorità al treno che segue, con conseguente comando di cambio itinerario al treno che precede per permettere il sorpasso.

Replicando questa semplice rete su qualsiasi binario è possibile effettuare molteplici confronti a due fra i vari treni in circolazione, tenendo presente che, poiché il modello è sincrono, uno stesso treno può essere sottoposto contemporaneamente a più confronti. E' ciò che necessariamente accade nelle stazioni complesse sottoposte ad un regime orario perturbato, dove i programmi di esercizio sono tali da rendere probabile l'arrivo contemporaneo di più treni

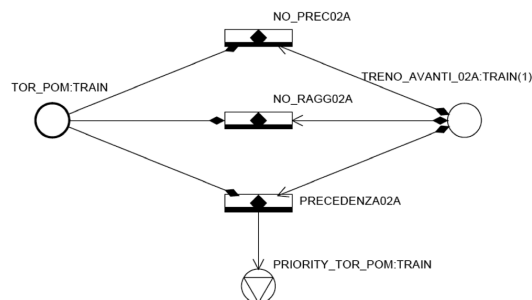


Fig.A6 – Parte della Sotto-rete 02_TOR_POM della classe MA sulla Roma-Formia. Part of 02_TOR_POM sub-net of MA class for Roma-Formia line.

other sub-nets managing possible conflicts (an example in figure A6).

In the first sub-net a TRAIN token enters from an external input place PREC and, on the basis of running direction and progressive, is redirected towards one of the other sub-nets by the common places located in the middle.

The 02_TOR_POM sub-net in figure A6 manages one of the possible traffic conflicts: the presence of two trains sharing the same resource (one after the other on the same track).

The two internal places are occupied by two TRAIN tokens: the first representing the previous and the following train; the activation of one of the concurring transitions (in particular PRECEDENZA02A) will determine the priority assignment to the following train and the activation of the overtaking procedure.

The replication of this simple network on any track allows to manage all possible dual comparisons between trains running, taking into account that the model synchronism lets it possible for a single train to be simultaneously involved in various comparisons. It obviously happens in the complex stations under perturbed conditions, where the probability of simultaneous trains movements is high.

BIBLIOGRAFIA - REFERENCES

- [1] A. TIERI, "Verifica e previsione della regolarità del traffico ferroviario con un modello di simulazione basato sulle Reti di PETRI", Tesi di Dottorato di ricerca, Università di Roma "La Sapienza", 2009.
- [2] A. TIERI, S. RICCI, "Previsione e risoluzione dei conflitti nell'esercizio ferroviario con un modello di simulazione basato sulle Reti di PETRI", atti del convegno SEF09, Roma, 2009.
- [3] I.A. HANSEN, J. PACHL, *Railway Timetable & Traffic*, ISBN 978-3-7771-0371-6, Hamburg, Germany, 2008.
- [4] "IEC 62551 Ed. 1.0: Analysis techniques for dependability – PETRI net modeling", Frankfurt am Main, Germany, bozza 6/5/2008.
- [5] S. RICCI, A. TIERI, "Railway traffic management: a decision support tool based on the PETRI Nets models", in Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems "Forms/Format 2008", Budapest, Hungary, 2008.

- gary, 2008.
- [6] S. RICCI, A. TIERI, "A PETRI Nets based decision support tool for railway traffic conflicts forecasting and resolution", in 11th International Conference on Computer System Design and Operations in the Railway and Other Transit Systems "CompRail 2008", Toledo, Spagna, 2008.
 - [7] A. TIERI, "Verifica e previsione della regolarità del traffico ferroviario con un modello di simulazione basato sulle Reti di PETRI", in SIDTgiovani2008 "Riunione Nazionale Dottorandi in Ingegneria dei Trasporti", Padova, 2008.
 - [8] A. DICEMBRE, "Ricerca degli ambiti ottimali di applicazione per la simulazione della circolazione ferroviaria con le Reti di PETRI: confronto con metodi sintetici ed analitici per la stazione di Roma Ostiense", Tesi di Laurea, Università di Roma "La Sapienza", 2007.
 - [9] S. RICCI, A. TIERI, "Railway traffic regularity monitoring and forecasting: the use of the PETRI nets models", in 6th International Symposium on Formal Methods for Automation and Safety in Railway and Automotive Systems "Forms/Format 2007", Braunschweig, Germany, 2007.
 - [10] R.M.P. GOVERDE, "Punctuality of Railway Operations and Timetable stability Analysis", Ph.D. Thesis. Delft University of technology, TRAIL Thesis Series, no. T2005/10, Delft, The Netherlands, 2006.
 - [11] S. IMPASTATO, S. RICCI, A. TIERI, "Monitoring and forecasting of railway traffic regularity by a PETRI nets model", in 4th International Scientific Conference "Challenges in Transport and Communication", Pardubice, 2006.
 - [12] C. SPELLUCCI, "Studio di sistemi ferroviari complessi mediante modello basato sulle reti di PETRI: applicazione sulla stazione di Roma Casilina", Tesi di Dottorato di ricerca, Università di Roma "La Sapienza", 2006.
 - [13] G. MALAVASI, S. RICCI, "La modellazione dell'esercizio ferroviario con le Reti di PETRI", Ingegneria Ferroviaria n. 03/2005.
 - [14] Th. SIEFER, A. RADTKE, "Railway Simulation, Key for Operation and Optimal use of Infrastructure", 1st International Seminar of Railway and Operations Modelling and Analysis, Delft, The Netherlands, 2005.
 - [15] A. BALDASSARRA, "Sistemi di segnalamento ferroviario con blocco conta-assi e controllo della marcia del treno. Analisi e simulazione con le Reti di PETRI", Tesi di Laurea, Università di Roma "La Sapienza", 2005.
 - [16] F. CIRCI, "Analisi della simulazione ferroviaria con il sistema ERTMS mediante l'utilizzo delle Reti di PETRI", Tesi di Laurea, Università di Roma "La Sapienza", 2003.
 - [17] N. CRISARÀ, "Analisi della transizione tra sistemi di segnalamento ERTMS e BACC sulla linea Roma-Napoli e simulazione con le Reti di PETRI", Tesi di Laurea, Università di Roma "La Sapienza", 2003.
 - [18] J. YUAN, R.M.P. GOVERDE, I.A. HANSEN, "Propagation of Train Delays in stations", COMPRAIL VIII, WIT Press, Southampton, pp.975-984, 2002.
 - [19] P. RAPONI, "Analisi e simulazione, mediante le reti di PETRI, degli apparati centrali statici di stazione e degli impianti di segnalamento di linea", Tesi di Laurea, Università di Roma "La Sapienza", 1999.
 - [20] L. GUERRUCCI, "Apparati di sicurezza degli impianti ferroviari: analisi e modello di simulazione tramite le reti di PETRI", Tesi di Laurea, Università di Roma "La Sapienza", 1998.
 - [21] Carl A. PETRI, "Kommunikation mit Automaten". Ph.D. Thesis. University of Bonn, 1962.
 - [22] Faber® on line: <http://www.fabersoftware.com/>
-

Sommaire

VERIFICATION ET PREVISION DE LA REGULARITE DE LA CIRCULATION FERROVIAIRE AVEC UN MODELE DE SIMULATION BASE SUR LES RESEAUX DE PETRI.

L'article montre les caractéristiques principales d'un modèle de simulation de l'activité ferroviaire. En utilisant le formalisme graphique des réseaux de PETRI, le modèle simule le fonctionnement des entités élémentaires qui constituent les systèmes de contrôle de l'activité ferroviaire à travers la reconstruction des passages logiques qui les caractérisent, à l'aide des Réseaux de PETRI, étendus, hiérarchiques et colorés.

Le modèle peut être appliqué dans les études de régularité de la circulation ferroviaire sur une ligne mais il peut être aussi employé efficacement comme support aux décisions dans la planification de nouvelles lignes et de leurs horaires d'utilisation, dans l'adaptation des lignes existantes aux changements des conditions de la circulation, mais aussi dans la phase de gestion de la circulation afin de minimiser les retards, étant le modèle fournis d'un système de gestion automatique des priorités.

Zusammenfassung

NACHPRÜFUNG UND BERECHNUNG DES EISENBAHN-BETRIEBSREGELMÄßIGKEIT DURCH PETRI-NETZEN GESTUTZTE SIMULATIONSMODELLEN

Es werden die Hauptmerkmalen eines Betriebssimulationsmodells vorgezogen. Durch die graphische Darstellung, typisch an PETRI-Netzen, simuliert das Modell den Funktionieren der Elementärstellen, die ein Beeinflussungssystem bilden; das geschieht durch die Nachahmung deren logischen Zusammenhänge, mittels breiten, hierarchischen und gefarbenen PETRI-Netzen.

Man kann damit die Betriebsregelmäßigkeit einer bestehenden Eisenbahnlinie erforschen und desgleichen die Ausbaumaßnahme, die veränderte Betriebszustände benötigen.

Das Modell funktioniert auch als Zugleiter weil es ein automatisches Konfliktlösungssystem besitzt.



ITALIAN BRAKES
E C O B R A K E®

ORGANIC BRAKE BLOCKS TYPE "LL"
100.000 Km
but I don't look it and without noise

ORGANIC BRAKE PADS
travel fast brake in silence



IB Italian Brakes S.p.A. • Via Ponte di Napoli • 80036 Palma Campania (NA) • info@italianbrakes.com • www.italianbrakes.com • Tel. 081.8277927