

UNIVERSIDAD PONTIFICIA COMILLAS
ESCUELA TÉCNICA SUPERIOR DE INGENIERIA (ICAI)
Instituto de Investigación Tecnológica (IIT)

Train eco-driving optimisation based on simulation models

Tesis para la obtención del grado de Doctor

Directores: Prof. Dr. D. Antonio Fernández Cardador

Prof. Dra. Dña. Asunción Paloma Cucala García

Autor: Ing. D. Adrián Fernández Rodríguez



Madrid 2018

Extended Abstract

Eco-driving is considered a key measure to reduce the energy consumption of railway systems. Eco-driving consists in finding the speed profile that requires the minimum energy consumption without degrading commercial running times or passenger comfort.

The research presented in this thesis develops optimisation models for the calculation of train eco-driving based on detailed and realistic simulation of the train movement. The models proposed investigate Nature Inspired Computational Intelligence techniques because of their suitability to use realistic train simulation results. The cases of metropolitan and long-distance lines are differentiated.

Metropolitan trains are typically equipped with Automatic Train Operation system (ATO), which drives the train automatically according to a speed profile defined by several driving commands. During the operation, a traffic regulation system selects the speed profiles that the trains must perform from a pre-programmed set. Therefore, the problem is to find the combinations of driving commands that produce the optimal speed profiles and to select from them the pre-programmed set of speed profiles.

In this thesis, MOPSO (Multi-Objective Particle Swarm Optimisation) and NSGA-II (Non-dominated Sorting Genetic Algorithm II) algorithms for the optimal design of the ATO speed profiles are applied and compared based on the accurate simulation of the ATO

and train motion. The problem is stated as a multi-objective optimisation problem where the objectives are the minimisation of energy consumption and running time. Therefore, the result is the set of non-dominated speed profiles, i.e. the Pareto front. The assessment of the results obtained with both algorithms has been carried out using several metrics that compare them in terms of number of solutions provided, diversity of solutions and distance to the real optimum. The results show that MOPSO outperforms NSGA-II in all the metrics.

Later, the uncertainties in the train operation are studied and a method to design robust and efficient ATO speed profiles is proposed. The first stage of the method is the calculation of the optimal Pareto front for ATO speed profiles that are robust to changes in train load. A technique based on the conservation of the shape of the speed profiles (pattern robustness) is proposed and compared with robust optimisation technique. Both procedures are included in MOPSO. The results have showed that the pattern-robustness is more restrictive and meaningful than the robust optimisation technique. Then, the set of speed profiles to be programmed in the ATO equipment are selected from the robust Pareto front by means of an optimisation model whose aim is the energy consumption minimisation. This model takes into account the statistical information about delays in the line and is solved using a PSO algorithm. Using this model, additional energy savings between 3% and 14% can be obtained.

The following part of the thesis is dedicated to long-distance railways, in particular, to high-speed lines. Compared to metropolitan trains, high-speed trains are typically manually driven and the journeys between stations are long-distance travels. In most of the works of the literature, eco-driving has been applied offline in the design of commercial services. However, the benefits of the efficient driving can also be applied on-line in the regulation stage, for instance, to recover train delays.

The train regulation problem is stated in this thesis as a dynamic multi-objective optimisation model to take advantage in real time of accurate results provided by detailed train simulation. The aim of the optimisation model is to find the Pareto front of the possible speed profiles and update it during the train travel. It continuously calculates a set of optimal speed profiles and, when necessary, one of them is used to substitute the nominal driving. The new speed profile is energy efficient under the changing conditions of the problem. DNSGA-II (Dynamic NSGA-II) and DMOPSO (Dynamic MOPSO) are applied to solve this problem. The performance of the dynamic algorithms has been analysed in a case study and the results show that dynamic algorithms are faster tracking the Pareto front changes than their static versions. Furthermore, DMOPSO presents better convergence results than DNSGA-II. In addition, the chosen algorithms have been compared with the typical delay recovery strategy of drivers showing that DMOPSO provides 7.8% of energy savings.

The uncertainties related to high-speed trains operation are later included in the previous model. These uncertainties are associated with the manual execution of the driving parameters and with the possible future traffic disturbances that could lead to new delays. Thus, a new algorithm is proposed, including the uncertainty in manual driving by means of fuzzy numbers. Furthermore, a newly defined objective, the risk of delay in arrival, is introduced in the optimisation model as a third objective. The risk of

delay in arrival measures the sensitivity of speed profiles to arrive delayed to the next station because of traffic disturbances. The use of this algorithm provides energy savings and, in addition, it permits railway operators to balance energy consumption and risk of delays in arrival. This way, the energy performance of the system is improved without degrading the quality of the service.

The final part of the thesis studies the eco-driving problem taking into account the ATO over ERTMS features. ATO over ERTMS is an interoperable system that aims to bring the ATO benefits observed in metropolitan lines to long-distance services equipped with ERTMS. ERTMS is a standardized signalling system developed to ensure interoperability of European trains and to improve safety, capacity. According to the new specification of the ATO over ERTMS, the on-board equipment is in charge of generating the speed profile to fulfil the timetable. Timetable information is provided to the on-board equipment by means of timing points. Timing points define positions in the track and the target departure/arriving/passing time for these points. Compared with the typical eco-driving problem, the algorithms needed by the on-board ATO system must be capable of generating speed profiles that not only meet a target running time, but also meet intermediate timing points minimising the energy consumption. This introduces new constraints to the speed profile optimisation problem.

DE (Differential Evolution) algorithm is selected and a fitness function has been defined to handle the new constraints in the problem. The performance of the DE algorithm has been compared with the well-known GA. The results have shown that the GA does not find feasible solutions in this difficult-to-solve problem. Contrary, the DE algorithm proposed has demonstrated its capacity to find speed profiles that meet all the target times. Apart from finding feasible solutions, the algorithm is capable of finding which one has the lowest energy consumption.

1. INTRODUCTION	1
1.1. ECO-DRIVING.....	3
1.1.1. Eco-driving for urban railways	5
1.1.2. Eco-driving for long-distance railways.....	7
1.1.3. Future of Eco-Driving in Long distance railways: ATO over ERTMS	9
1.2. NATURE INSPIRED OPTIMISATION.....	11
1.3. THESIS OBJECTIVES.....	13
1.3.1. Main objective	13
1.3.2. Specific objectives.....	14
1.4. STRUCTURE OF THE DOCUMENT.....	16
2. OPTIMISATION ALGORITHMS FOR THE DESIGN OF ENERGY EFFICIENT ATO SPEED PROFILES IN METROPOLITAN LINES	17
2.1. INTRODUCTION	17
2.2. TRAIN MOTION SIMULATION MODEL.....	20
2.3. MULTI-OBJECTIVE OPTIMISATION FOR THE DESIGN OF ATO SPEED PROFILES.....	23
2.3.1. Mopso algorithm for the design of ATO speed profiles	26
2.3.2. NSGA-II algorithm for the design of ATO speed profiles	28
2.4. CASE STUDY AND RESULTS	30
2.4.1. Comparison of MOPSO and NSGA-II.....	31
2.5. CONCLUSIONS AND CONTRIBUTIONS.....	34
3. ROBUST SPEED PROFILES FOR THE AUTOMATIC TRAFFIC REGULATION SYSTEM IN METROPOLITAN LINES	37
3.1. INTRODUCTION	37
3.2. DESIGN OF EFFICIENT AND ROBUST ATO SPEED PROFILES.....	39
3.2.1. Generation of robust Pareto front of optimal speed profiles using a robust optimisation technique.....	39
3.2.2. Generation of robust Pareto front using an alternative method based on driving patterns.....	42
3.3. OPTIMAL SELECTION OF THE ATO SPEED PROFILE SET	44
3.4. CASE STUDY.....	47
3.4.1. Generation of the robust Pareto front	47
3.4.2. Generation of the robust Pareto front using the alternative method based on driving pattern	49
3.4.3. Optimal selection of the pre-programmed set of ATO speed profiles.....	51
3.5. CONCLUSIONS AND CONTRIBUTIONS.....	54
4. REAL TIME ECO-DRIVING OF HIGH-SPEED TRAINS.....	57
4.1. INTRODUCTION	57
4.2. TRAIN SIMULATION MODEL.....	59
4.2.1. Train dynamics simulation model.....	60

4.2.2. Train energy consumption simulation model	61
4.2.3. Manual driving simulation model	62
4.2.4. Delay response model	66
4.3. DYNAMIC MULTI-OBJECTIVE OPTIMISATION MODEL BASED ON SIMULATION FOR THE ON-LINE DRIVING REGULATION PROBLEM	67
4.3.1. DNSGA-II algorithm for the dynamic eco-driving calculation	69
4.3.2. DMOPSO algorithm for the dynamic eco-driving calculation	71
4.3.3. On-line speed profile selection	74
4.4. CASE STUDY	75
4.4.1. Algorithms convergence analysis	76
4.4.2. Energy consumption analysis	79
4.5. CONCLUSIONS AND CONTRIBUTIONS	81
5. BALANCING ENERGY CONSUMPTION AND RISK OF DELAY IN HIGH-SPEED TRAINS	83
5.1. INTRODUCTION	83
5.2. FUZZY MANUAL DRIVING MODEL	85
5.3. RISK OF DELAY IN ARRIVAL	86
5.3.1. Relationship of the risk of delay in arrival with the running time and the energy consumption	88
5.4. DYNAMIC NSGA-III WITH FUZZY PARAMETERS (DNSGA-III-F) FOR THE ON-LINE DRIVING REGULATION PROBLEM	90
5.4.1. DNSGA-III-F flowchart	93
5.4.2. Fuzzy dominance	95
5.4.3. Selection operator	96
5.4.4. DNSGA-III-F resolution	99
5.5. ALGORITHM TO UPDATE DRIVER'S COMMANDS	101
5.6. CASE STUDY	102
5.6.1. DNSGA-III-F convergence analysis	103
5.6.2. Delay response analysis facing one temporary speed limitation	106
5.6.3. Delay response analysis facing two temporary speed limitation	109
5.7. CONCLUSIONS AND CONTRIBUTIONS	113
6. ECO-DRIVING IN ATO OVER ERTMS	115
6.1. INTRODUCTION	115
6.2. PROBLEM DESCRIPTION	118
6.3. DIFFERENTIAL EVOLUTION ALGORITHM FOR THE ATO OVER ERTMS ECO-DRIVING PROBLEM.	120
6.3.1. Differential Evolution flowchart	120
6.3.2. Constraint-handling process	122
6.4. CASE STUDY	123
6.4.1. One intermediate timing point	124

6.4.2. Two intermediate timing points	127
6.4.3. Effect of the target times on the energy consumption	131
6.5. CONCLUSIONS AND CONTRIBUTIONS.....	133
7. CONCLUSIONS AND CONTRIBUTIONS.....	135
7.1. CONCLUSIONS AND CONTRIBUTIONS	135
7.2. PUBLICATIONS	145
7.3. FUTURE WORK	146
8. REFERENCES	149

List of symbols

Symbol	Description
ATO	Automatic Train Operation
CBTC	Communications-Based Train Control
ETCS	European Train Control System
ERTMS	European Railway Traffic Management System
GA	Genetic Algorithm
ACO	Ant Colony Optimization
EA	Evolutionary Algorithm
PSO	Particle Swarm Optimization
MOOP	Multi-Objective Optimization Problem
NSGA	Non-dominated Sorting Genetic Algorithm
SPEA	Strength Pareto Evolutionary Algorithm
PAES	Pareto Archived Evolution Strategy
MOPSO	Multi-objective Particle Swarm Optimisation
NSPSO	Non-dominated Sorting Particle Swarm Optimisation
RT	Running time
EC	Energy consumption
b	Deceleration rate
v_h	Holding speed command
v_c	Coasting speed command
v_r	Remotoring speed command
M_{eq}	Equivalent train mass
F_m	Train motors force
F_r	Running resistance force
F_g	Gradient resistance force
A	Running resistance coefficient
B	Running resistance coefficient
C	Running resistance coefficient
v	Train speed
m	Train mass
p	Average equivalent grade affecting the length of the train

Symbol	Description
I	Current consumed by the train
U	Line voltage
p_{max}	Maximum slope permitted in a point of coast start
F_n^{prev}	Traction effort at the previous step
v_{min}	Minimum permitted speed
$n_{remotoring}$	Number of re-motoring cycles executed along an interstation
n_{max}	Maximum number of re-motoring cycles permitted
$\vec{x}p_j(it)$	Position of the particle j^{th} at iteration it in MOPSO
$\vec{v}p_j(it)$	Velocity of the particle j^{th} at iteration it in MOPSO
$\vec{p}_j(it)$	Best previous position of the j^{th} particle at iteration it
c_1, c_2	Social factors
r_1, r_2	random numbers between 0 and 1
w	inertia weight
w_1, w_2	Extreme values of inertia weight
it_{max}	Maximum number of iterations
n_{swarm}	Size of the swarm of MOPSO
CD	Crowding distance
$\vec{f}(\vec{x}p)$	Objective function of the optimisation process
γ	Generational distance
Δ	Spread
ER	Error ratio
SP	Spacing
DE	Differential Evolution algorithm
$\vec{f}^{eff}(\vec{x}p)$	Mean effective objective function
S	Solution space
$ B_\delta $	Hypervolume of a chosen vicinity δ
$\vec{f}^P(\vec{x}p)$	perturbed objective vector
η	Robustness coefficient
$\vec{x}p^*$	robust solution
FB	Final braking driving mode
pt	Driving pattern of a speed profile
NS	Number of scenarios of traffic regulation system demand
p_s	Probability assigned to an scenario of traffic regulation system demand
RT_s	Running time assigned to an scenario of traffic regulation system demand
D	Number of speed profiles in the pre-programmed set
J	Pareto front
\vec{st}	Pre-programmed set of speed profiles
RT_d	Running time associated with speed profile d of a pre-programmed set
EC_d	Energy consumption associated with speed profile d of a pre-programmed set
EC_s	Energy consumption of an scenario of traffic regulation system demand
i	Position of a particle in time-sorted Pareto front
NP	Size of the swarm of PSO
HSR	High-speed railway
MPGA	Multi-population genetic algorithm
IBEA	Indicator Based Evolutionary Algorithm
DMOOP	Dynamic multi-objective optimisation problems
DNSGA-II	Dynamic Non-dominated Sorting Genetic Algorithm II

Symbol	Description
DMOPSO	Dynamic Multi-Objective Particle Swarm Optimisation algorithm
$s(t)$	Train position
$v(t)$	Train speed
$a(t)$	Train acceleration
$F_b(t)$	Effort of pneumatic brakes
ρ	Rotating mass factor
s_0	Initial train position
s_{end}	Arrival train position
v_0	Initial train speed
F_{NZ}	Braking effort applied by the train when travelling through neutral zones
$F_{max}(v)$	Maximum electrical traction effort
$F_{min}(v)$	Maximum electrical braking effort that depend on the speed of the train
snz_{start}^l	Initial position of the neutral zone l
snz_{end}^l	Final position of the neutral zone l
Z	Number of neutral zones
$P_{mec}(t)$	Train mechanical power
$P_{pantograph}(t)$	Train electrical power measured at pantograph
$P_{ss}(t)$	Train electrical power measured at substation
P_{aux}	Power consumed by train auxiliary systems
$\mu_T(v, f)$	Train electrical chain traction efficiency
$\mu_B(v, f)$	Train electrical chain braking efficiency
fr	ratio motor force
$\cos \varphi$	Power factor
$r(s)$	Line resistance
$EC_{pantograph}(t)$	Train energy consumed measured at pantograph
$EC_{ss}(t)$	Train energy consumed measured at substation
C_m	Command matrix
n_s	Number of sections in command matrix
sc_k	Position where the section k ends
vc_k	Values of holding speed without braking command of section k
$U_{reg}(t)$	Train control output
$U_{prop}(t)$	Proportional contribution to $U_{reg}(t)$
$U_{int}(t)$	Integral contribution to $U_{reg}(t)$
K_p, K_I	Proportional and the integral constants of train control
$e(t)$	Regulation error
$v_{obj}(t)$	Objective train speed
ts	Simulation time step
$v_{max}(t)$	Maximum train speed
$v_{ceiling}(s)$	Ceiling speed
$v_b(s)$	Speed of braking curve of the next ceiling speed reduction
$v_{fb}(s)$	Speed of braking curve at the next station arrival
$d_{service}$	Absolute value of the deceleration rate
s_{next}	Position of the next ceiling speed reduction
$F_{TB}(t)$	Total traction/braking effort demanded to the train
$F_{maxAcc}(s, v)$	Effort limit corresponding to the maximum acceleration rate
$F_{maxDec}(s, v)$	Effort limit corresponding to the effort for the maximum deceleration rate
$F_{maxJerk}(t)$	Maximum effort to limit the jerk
$F_{minJerk}(t)$	Minimum effort to limit the jerk

Symbol	Description
a_{max}	Maximum acceleration rate
d_{max}	Maximum deceleration rate
j_{max}	maximum jerk
ex	Execution of dynamic algorithms
n_{pop}	Size of the parent population of NSGA-II, DNSGA-II y DNSGA-III-F algorithms
P_{it}^{ex}	Parent population of NSGA-II, DNSGA-II y DNSGA-III-F algorithms
Q_{it}^{ex}	Offspring population of NSGA-II, DNSGA-II y DNSGA-III-F algorithms
R_{it}^{ex}	Result population of NSGA-II, DNSGA-II y DNSGA-III-F algorithms
$X_j(it)$	Position of the particle j^{th} of DMOPSO at iteration it
$VP_j(it)$	Velocity of the particle j^{th} of DMOPSO at iteration it
P_j	Best previous position of the j^{th} particle of DMOPSO at iteration it
P_g	Best position found by the whole swarm of DMOPSO at iteration it
sc_{min}	Minimum position from where the train can start the final coasting command
vc_{min}	Lower bound for the driving commands
\tilde{c}_k	Fuzzy position where the section k ends
$\tilde{v}c_k$	Fuzzy value of holding speed without braking command of section k
$\tilde{t}_{a/d}$	Fuzzy anticipation/delay time on new driving command application
$Margin(s)$	Slack time at position s
AH_{com}	Commercial arrival time
$H_{current}$	Current time
$AT_{flat}(s, v)$	Running time of the flat-out driving from the current train position and speed
RD	Risk of delay in arrival
$Margin_{tot}$	Total margin at the beginning of the journey
$\tilde{f}(X)$	Fuzzy objective function
$\tilde{RT}(X)$	Fuzzy running time
$\tilde{EC}(X)$	Fuzzy energy consumption
$\tilde{RD}(X)$	Fuzzy risk of delay in arrival
SP_{it}^{ex}	Surviving population at iteration it and execution ex
N_d	Required level of necessity for fuzzy dominance comparisons
nc_r	Niche count of reference point r used in DNSGA-III-F
n_{off}	Size of offspring population
N_p	Necessity of punctuality
w_{ec}, w_{rd}	Fitness weighting factors of energy consumption and risk of delay in arrival
OT_l	Objective time for the point l
$t_l(X)$	Resulting passing time of solution X in the timing point l
s_l	Location of timing point l
ε_l	Time window allowed to the passing point l
L	Number of timing points including the arrival point at the station
DV_j	Donor vector of the individual j
X_{best}	Best individual of the DE population
F	Scaling factor parameter of DE
TV_j^h	Component h of the trial vector associated with the individual j of DE
K	Randomly generated number between 1 and the population size in DE process
C_r	Crossover rate of the flat-out driving
EC_{flat}	Energy consumption of the flat-out driving

CHAPTER 1

INTRODUCTION

Transport plays an important role in the development and growth of the countries. It serves the function of integrating and communicating the productive, social and territorial sectors. The development of the steam engine during the Industrial Revolution was a turning point in the mobility of people and goods. Its application to build the first locomotives in the beginning of 19th century was the birth of the railway industry.

The appearance of the first trains improved the speed and capacity of human transport allowing longer and faster travels. Since then, railways have continued being an important transport mode despite the appearance of other transport methods such as airplanes or automobiles.

From 19th century until now, the necessities of society have changed and the demands for transport industry have also evolved. Increasing speeds, capacity and safety are traditional objectives common for every transport mode. However, the concern for climate change and energy prizes arises nowadays as one of the main challenges.

Transport sector is one of the major energy consumers in the world. Transport consumes 28.8% of global final energy consumption and it is responsible for 24.7% of CO₂ emissions. This implies an important responsibility in climate change and, for this reason, the transport sector and research community are making efforts to reduce its

energy consumption and its contribution to greenhouse gas emissions (IEA and UIC, 2017).

Rail transport is regarded as a fast, inexpensive and safe transport mode. Furthermore, this transport mode is energy efficient and its contribution to global warming is less severe compared with others. As an evidence of its efficiency, in 2015, passenger railways consumed 1% of final energy demanded by passenger transport while accounted for 9% of passengers transport activity (IEA and UIC, 2017). Despite this advantage, railway industry does not escape from the responsibility to contribute to the sustainability of transport.

Railway systems are expanding day by day throughout the world increasing not only their activity, but also their energy consumption. For instance, between 2013 and 2015, the world railroads grew by 25000 km and the number of vehicles increased by 28000 units (UNIFE/Roland Berger, 2016). It is expected that the growing trend of railway systems will continue in the next years as well as the increasing energy demand. For this reason, it is necessary to continue working on the improvement of the efficiency of rail transport to mitigate its contribution to global warming. The compromise of energy reduction of railways was reflected in the Paris agreement of 2015 (United Nations, 2015) where certain goals were established to reduce energy consumption per traffic unit (-50% by 2030 and -60% by 2050) and specific average CO₂ emissions from train operations (-50% by 2030 and -75% by 2050) (IEA and UIC, 2016).

It is important to take into account that the reduction of energy demand must be achieved without losing sight on increasing the quality of the service. By this way, the sustainability of transport will be improved not only by reducing the energy consumption of trains but also by reducing the activity of more polluting transport modes.

Several energy efficiency measures can be applied in railways systems. These measures can be divided into three groups: measures that involve infrastructure, rolling stock or those related to traffic operation (Bae, 2009; Douglas et al., 2015; X. Yang et al., 2016).

Among the measures that affect the rolling stock it can be found in the literature works related to the reduction of energy demanded by auxiliary systems (Beusen et al., 2013), the improvement of the drive chain efficiency (Kondo et al., 2014; Matsuoka and Kondo, 2014), the reduction of trains' resistance to motion (Bombardier, 2010) and the implementation of on-board energy storage system (Arboleya et al., 2014; Ciccarelli et al., 2012). On the other hand, measures related to infrastructure study the track topology (Lv et al., 2013; Xin et al., 2014) and the improvement of the electrical infrastructure by means of reversible substations (Cornic, 2010; Ibaiondo and Romo, 2010; López-López et al., 2014; Roch-Dupré et al., 2018) or wayside energy storage systems (Barrero et al., 2008; Lee et al., 2013; Roch-Dupré et al., 2017) to maximise the regenerative energy saving. All the previous energy efficient methods are long-term actions that require high investments and, in some cases, they can only be applied during the system design.

Measures related to traffic operation, on the contrary, require low investments and are short-midterm actions that allow to improve the performance of, not only new railway

systems, but also systems that are already in operation. The main energy efficiency actions that can be taken in the field of traffic operation are focused on: traffic design (Abril et al., 2008; Feng et al., 2013a; Hassannayebi and Zegordi, 2017; Peña-Alcaraz et al., 2011; Wang et al., 2012), on-line traffic regulation (Fay, 2000; Jia and Zhang, 1994; Lin and Sheu, 2011a; Sheu and Lin, 2012, 2011; Yin et al., 2016), maximization of regenerative energy transference among trains (Domínguez et al., 2012; Falvo et al., 2011; Lu et al., 2014) and eco-driving.

Eco-driving is considered a key measure to reduce the energy consumption of railway systems. Eco-driving is also named in the literature speed profile optimisation or train trajectory optimisation. It consists in obtaining the way to drive a train in a journey to fulfil a target running time with the minimum energy consumption. Eco-driving can be applied either by human drivers or by automatic train operation systems (ATO). The improvement in energy consumption obtained by eco-driving strongly depends on the timetable and the target running time. However, the literature consistently shows that important energy savings can be obtained in the short-midterm without high investment (Douglas et al., 2015).

The piece of research presented in this thesis document is focused on eco-driving. This work is intended to contribute in the techniques that can be applied to obtain energy efficient train driving. In this work, different optimisation methods will be studied considering different kinds of railways and making use of detailed simulation methods. The rest of the present chapter explains some important concepts used in this work and defines the motivations, the objectives and the general outline of the thesis.

1.1. ECO-DRIVING

Train eco-driving consists in solving an optimisation problem where the objective is to minimise the energy consumption in a journey, fulfilling timetable constraints, operational constraints and comfort constraints.

The first study on eco-driving was developed by Ichikawa (Ichikawa, 1968). He applied the Pontryagin's Maximum Principle to a very simplified train dynamics model where the running resistance was considered linear and the track was considered flat. He obtained, for the first time, the optimal regimes of the train control, which are: maximum acceleration, cruising, coasting and maximum braking (Ichikawa, 1968).

Strobel et al. continued this research line in 1974. In their work, a similar train model was applied but the running resistance was modelled by means of a quadratic function (Strobel and Horn, 1974). Milroy reformulated the problem in 1980 and obtained the model that has been used as the base for many optimal train control studies (Milroy, 1980). Using this model, in 1988, Howlett demonstrated that the solution of the optimal train control problem exists and it is unique (Howlett, 1988) and, in 1990, he used the Pontryagin's Maximum Principle to obtain the optimal driving regimes and the optimal solution of the problem (Howlett, 1990). Later, the eco-driving problem was transformed into seeking the switching points between optimal driving regimes (Howlett, 1996). This work showed that, as in (Ichikawa, 1968), the optimal solution to

drive a train in a flat track is as simple as the following sequence: a period of maximum acceleration, a period of cruising at certain speed value, a coasting phase and a final braking at the station. Figure 1-1 depicts the shape of the optimal speed profile in a flat track.

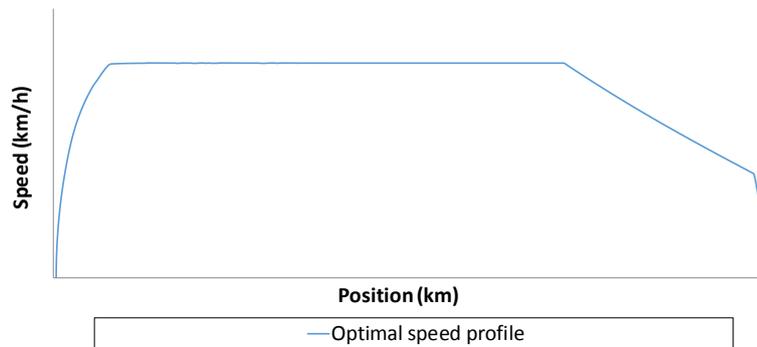


Figure 1-1. Optimal speed profile in a flat track

Since then, numerous works have followed this research line improving the train model considering variable motor efficiency (Franke et al., 2000) or considering regenerated energy (Khmelnitsky, 2000). The details about the track have been included in later research as variant grades and speed limits (Liu and Golovitcher, 2003). The optimal way to negotiate steep grades was obtained in (Howlett et al., 2009). As details were incorporated in the model, the complexity of the optimal solution increased compared with the optimal solution of Figure 1-1.

The number of techniques applied to the eco-driving problem since the Ichikawa's first study is remarkable. These techniques can be classified in analytical methods and numerical methods (X. Yang et al., 2016). Explicitly or implicitly, both groups of techniques make use of the optimal driving regimes obtained from the Pontryagin's Maximum Principle.

Most of analytical methods make use of the Pontryagin's Maximum Principle to obtain the optimality necessary conditions and, using these results, apply different algorithms to solve it. Among these algorithms it can be found: constructive algorithms (A. R. Albrecht et al., 2013; Howlett et al., 2009; Khmelnitsky, 2000; Liu and Golovitcher, 2003; Su et al., 2013; J. Yang et al., 2016), Dynamic Programming (T. Albrecht et al., 2013; Lu et al., 2013; Miyatake and Ko, 2010; Miyatake and Matsuda, 2009), Sequential Quadratic Programming (Gu et al., 2014; Miyatake and Ko, 2010) and Lagrange multiplier method over the discretised problem (Rodrigo et al., 2013). Other analytical methods are based on transforming the optimal control problem into a non-linear problem and solving it directly (Wang et al., 2014, 2013). Analytical methods can produce the optimal solution of the problem and, in most cases, using low computational times. However, the complexity of the problem and the requirements for obtaining the analytical solution lead to simplifications in the train model, line description and in the operational restrictions that the solution must comply.

On the contrary, numerical methods do not require simplifications in the train model, and any constraint related to passengers' comfort or the driving commands can be

included. Therefore, the solutions can be obtained from models as detailed as necessary and fulfilling any operational restriction to be applied in real life. Numerical methods received less attention at the beginning of the optimal train control because they are computationally expensive. However, in recent years there is an increasing number of numerical methods applied to solve the eco-driving problem due to the improvement of computers performance.

Different numerical methods can be found in the literature: direct search algorithms (De Cuadra et al., 1996; Wong and Ho, 2004a), Brute Force (Zhao et al., 2017), Monte Carlo Simulation (Tian et al., 2017), Artificial Neural Networks (Acikbas and Soylemez, 2008; Chuang et al., 2009) and algorithms of the Nature Inspired Computational Intelligence branch (Bocharnikov et al., 2010; Carvajal-Carreño et al., 2014; Lu et al., 2013; Sicre et al., 2012). Among these techniques, Nature Inspired algorithms are one of the most common methods applied to solve speed profile optimisation problem. The reason is that these techniques can be straightforward implemented and are independent of the specificities of the problem. It gives an enormous flexibility to solve eco-driving problems by means of different train models. They can be used in combination with complex models that can be easily substituted by other model when the features of railway system studied change.

Among the Nature Inspired Computational Intelligence techniques used to solve the train eco-driving problem it can be found in the literature: Genetic Algorithm (GA) (Bocharnikov et al., 2010; Chang and Sim, 1997; Lechelle and Mouneimne, 2010; Li and Lo, 2014; Lu et al., 2013; Sicre et al., 2012; Wong and Ho, 2004b, 2003; Yang et al., 2012), multi-population genetic algorithm (MPGA) (Huang et al., 2015; Wei et al., 2009), GA combined with fuzzy logic (Bocharnikov et al., 2007; Cucala et al., 2012b; Hwang, 1998; Sicre et al., 2014), Differential Evolution (Kim et al., 2013), Ant Colony Optimisation (Ke et al., 2012; Lu et al., 2013; Yan et al., 2016), Simulated Annealing (Keskin and Karamancioglu, 2017; Xie et al., 2013), Indicator Based Evolutionary Algorithm (IBEA) (Chevrier et al., 2013), Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Carvajal-Carreño et al., 2014; Domínguez et al., 2014) and Multi-Objective Particle Swarm Optimisation (MOPSO) (Domínguez et al., 2014).

Apart from the specific technique used, the eco-driving problem can be classified taking into account the kind of railway (urban or long-distance) and the stage (planning or regulation) when it is going to be applied.

1.1.1. ECO-DRIVING FOR URBAN RAILWAYS

Eco-driving application during the planning stage is similar for urban and long-distance trains. Before the operation, a nominal timetable is defined in an offline manner. From this timetable, the running time between stations that a train must comply is extracted. After that, for each journey between stations, it is obtained the driving that the train must perform to fulfil the timetable, minimising the energy consumption.

Most of the speed profile optimisation algorithms can be applied both to the case of urban or long distance railways. The main difference when applying eco-driving in an urban railway or in a long-distance railway are the driving commands to execute to

perform the required running times and fulfil the comfort restrictions. Driving commands are the way in which the optimal driving regimes of the train control are applied.

Urban railways are characterised by short stretches between stations. Besides, these systems are usually automatised where the trains are driven by an ATO system. Because of this, the driving commands applied in these lines are: holding speed to ensure the comfortability of the speed profile (Feng et al., 2012; Howlett et al., 1994; Jong and Chang, 2005; Liu and Golovitcher, 2003) and coasting-re-motoring cycles to perform efficient speed profiles taking advantage of high motor efficiency when using maximum traction and coasting periods (Bocharnikov et al., 2007; Chang and Sim, 1997; Coleman et al., 2010; Domínguez et al., 2011a; Howlett, 1996; Howlett et al., 2009; Ichikawa, 1968).

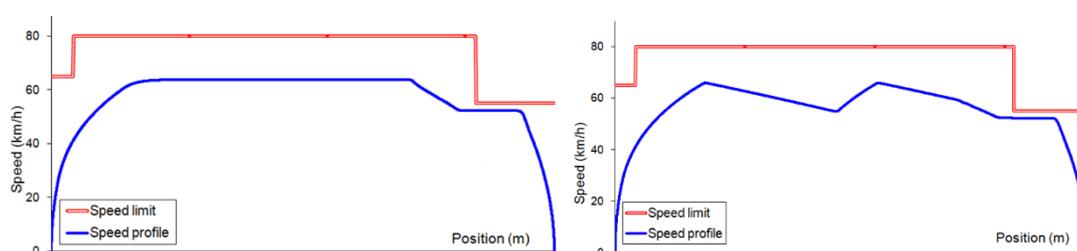


Figure 1-2. Examples of speed profiles with speed-holding (left) and coasting/re-motoring cycles (right)

The efficient speed profiles obtained for urban trains must be defined by the driving parameters used by the ATO. Therefore, it is important that the solutions provided by eco-driving algorithms can be easily translated to the driving parameters used by the specific ATO where it is going to be implemented. These driving parameters will be pre-programmed in the ATO equipment to be applied during the operation.

It is very important to apply detailed simulation models when designing speed profiles to give confidence to the Railway Operator that energy savings will be obtained without degrading the commercial running times or passengers' comfort. Focused on urban railways, it is crucial to represent in detail the characteristics and behaviour of the specific ATO used in the study line because it conditions the driving possibilities. The logic that decides when to start to brake in front of a speed limit reduction or to stop at the station along with the comfort restrictions applied determines the running times that are observed in reality. Only few works can be found in the literature that take into account real ATO equipments in the train model (Bocharnikov et al., 2007; Chang and Xu, 2000; Domínguez et al., 2011a). However, in these works the study of most modern signalling system has not been taken into account. New urban signalling systems, such as the CBTC (Communications based train control) use new communications channels to send driving commands to the train. The increased communication bandwidth improves exponentially the possibilities to drive the train. New algorithms are needed to explore the numerous speed profiles that a CBTC train can apply and find the most efficient ones. Apart from this, optimal speed profiles are usually obtained using a fix value of train mass. However, this mass value varies a lot depending on the time of the day. Therefore, it is important that new eco-driving techniques model this uncertainty to find solutions that can be applicable with different values of train mass.

Delays arise during the traffic operation in real-time because of the accumulation of passengers at the stations or because of incidences in the line. At this stage, regulation plays a fundamental role. The objective of traffic regulation is to ensure the quality of the service bringing the traffic to the scheduled timetable. That means that, when a delay arises, the regulation system is in charge of making that the train make up the delay using the slack time included in the timetable. Contrary to the planning situation, the way eco-driving is in the regulation stage is different in urban railways and long-distance railways.

Urban railways usually present a carrousel operation. In other words, trains circulate in a loop where, once the terminal station is reached, they turn around and circulate in the opposite direction in the other track. Furthermore, the number of trains circulating in these systems is typically close to the maximum capacity. Therefore, it is frequent that the delay of a train is transmitted to the following trains in the line. For these reasons, traffic regulation in urban systems is performed considering all the trains jointly.

The Traffic Management Centre (TMS) is in charge of regulating the timetable deviations. It observes not only the scheduled timetable but also the scheduled interval between trains. Based on this information, TMS decides the dwell time that trains must spend on stations and the speed profile to be performed by each train (Fernandez et al., 2006). To carry out this task, the regulator algorithm online calculates the delay of each train and the interval between trains. With this information, the regulator chooses the most adequate speed profile among a set of pre-programmed speed profiles for each stretch (Fernandez et al., 2006). Typically, the list of the pre-programmed speed profiles is made up by four speed profiles per interstation:

- The flat-out driving. This speed profile performs the minimum running time. It is used to recover delays.
- The nominal driving: this speed profile performs typically the nominal running time. It is used to track the timetable if there are no delays.
- The slow driving: this speed profile performs a higher running time than the nominal one. It is used to reduce the speed of the train when the precedent train suffers a delay with the objective of increasing the interval between those trains.
- The extra-slow driving: this speed profile performs a higher running time than the slow one. Its mission is similar than the previous one.

Commonly, the pre-programmed speed profiles are designed off-line and selected manually to have the associated running time separated a constant time value between consecutive profiles (Domínguez et al., 2011a). However, this could not be the optimal solution. Extra energy savings could be obtained selecting the pre-programmed speed profiles by means of optimisation algorithms taking into account the usual delay distribution at each station.

1.1.2. ECO-DRIVING FOR LONG-DISTANCE RAILWAYS

Long-distance trains are characterised by larger stretches between stations where the trains are manually driven. Coasting-remotoring cycles, which are commonly applied by

urban trains, cannot be applied for long-distance trains. The continuous acceleration changes are perceived as unpleasant by passengers in a long distance travel. Moreover, it is very demanding for human drivers to implement frequent changes between traction and coasting in this type of travels. In this case, the efficient driving commands are holding speed (Ji et al., 2016) and its efficient version, holding speed without braking. The latter consists in applying traction as long as it is necessary to maintain the cruising speed but no braking is applied unless it is needed to observe a speed limitation or to brake before the arrival station. Instead of applying braking, coasting is performed so the train increases its speed (Hwang, 1998; Sicre et al., 2014, 2012). Figure 1-3 shows an example of a holding speed without braking command.

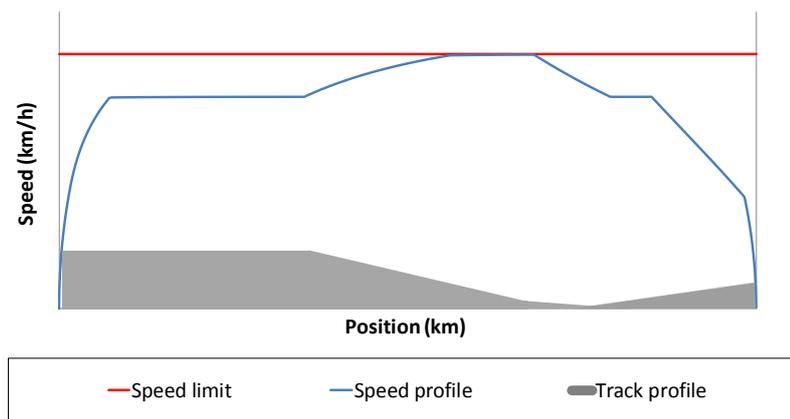


Figure 1-3. Example of speed profile performing a holding speed without braking command

A typical way of implementing eco-driving algorithms for long distance trains is by means of Driver Advisory Systems (DAS). DAS have been developed not only for energy saving reasons but also to increase the transport capacity and to reduce timetable deviations. Several commercial DAS can be found in the literature such as “MetroMiser” (Howlett et al., 1995), “CATO” (Yang et al., 2013) or “OptiDrive” (Lechelle and Mouneimne, 2010). Most of DAS systems calculate the speed profile before the departure of the train based on a timetable. Then, during the journey, it advises to the driver how to drive the train according to the optimal speed profile obtained. It can be done showing the driver the speed profile to follow or showing driving commands that are time, position or event dependant (Panou et al., 2013).

In a long distance system, different kinds of services can run on the same line, which could have a complex topology with crossing points and different routes. In this case, the regulation of traffic affects each train individually and the objective is the fulfilment of the planned timetable. In other words, each train is responsible of recovering its own delay.

When a train is delayed, the eco-driving obtained during the planning stage is not valid anymore because a faster speed profile is needed. Furthermore, the modification of the speed profile cannot wait to the next station arrival because of the long distance between stations. In these situations, drivers usually increase the speed of the train to recover the delay as fast as possible. However, this is not the most efficient way to meet the timetable.

Most of standalone DAS monitor the tracking of the optimal speed profile. By this way, they are capable of detecting delays more precisely and react accordingly. Once a deviation is detected, a new speed profile is obtained to allow the train to arrive on time at the next station. Thus, a new eco-driving optimisation is performed in an online manner to recover the delay energy efficiently. One of the many challenges of applying online eco-driving optimisation in long distance trains is the reduced calculation time available.

Some solutions have been proposed in the literature to solve the on-line eco-driving design. Several works make use of mathematical models taking advantage of their low computation time (Coleman et al., 2010; Howlett et al., 1994; Khmel'nitsky, 2000; Liu and Golovitcher, 2003). Other works make use of Nature Inspired algorithms (Chang and Sim, 1997; Sicre et al., 2014; Wong and Ho, 2004b). However, there is still room to improve the existing algorithms, on one hand by using detailed simulation models and incorporating uncertainty to these models, and in the other hand improving the execution times of simulation-based algorithms.

1.1.3. FUTURE OF ECO-DRIVING IN LONG DISTANCE RAILWAYS: ATO OVER ERTMS

Twenty-three different signalling systems coexist in Europe along fifteen countries. In the past, this has been a difficulty for cross-border trains. A train needed to be equipped with several signalling equipments to be capable of operating in different countries. This involved an extra cost because of the variety of equipments needed and waiting times when switching from one system to another (Dhahbi et al., 2011).

The European Commission undertook a major project called "European Railway Traffic Management System" (ERTMS) to solve this problem and to develop the specification of a standardised signalling system (Council Directive, 1996). The aim of this system is to ensure interoperability of European trains and to improve safety, transport capacity and economic effectiveness. This system has been successful and has been implemented not only in many lines in Europe but also in railways of other continents.

The ERTMS is composed of two main subsystems: ETCS (European Train Control System) y GSM-R (Global System for Mobile Communications - Railways). The ETCS is in charge of the control and safety of the traffic whereas GSM-R is in charge of communications.

ETCS provides an Automatic Train Protection system (ATP) that ensures the safe movement. There are three main different operational levels of ETCS:

- ETCS level 1: It is based on blocking stretches between signals and sends movement authority (MA) to trains using balises. There is a punctual communication from the track-side equipment to the train and lateral signals are needed.
- ETCS level 2: It is based on track circuits so the MA could reach the end of the closest track circuit to the precedent train that is not occupied. There is a continuous communication between the track-side equipment and the train by means of radio, and lateral signals are not needed.

- ETCS level 3: It is a moving-block signalling systems so the MA could reach the tail of the precedent train. There is a continuous communication between the track-side equipment and the train by means of radio, and lateral signals are not needed.

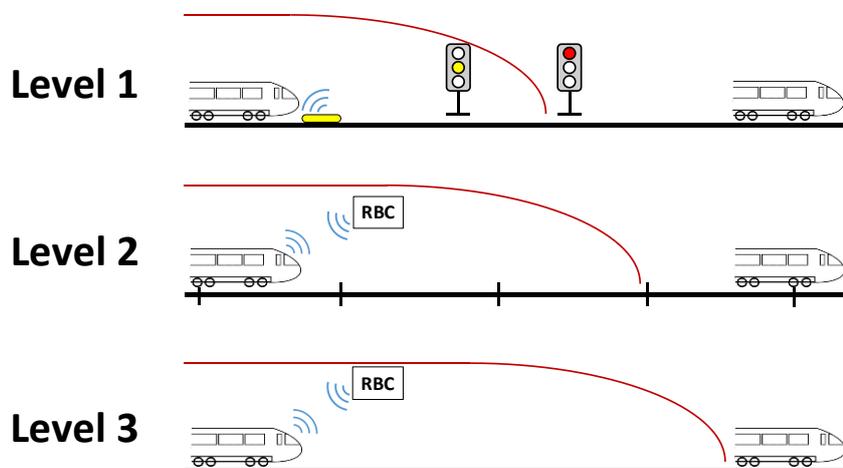


Figure 1-4. Operational levels of ETCS

The next feature of ERTMS is to control the train driving by means of an interoperable ATO system. ATO systems have been installed in urban railways for many years and their advantages has been proven. ATO leads to more deterministic running times, which permit increasing the transport capacity. Furthermore, eco-driving can be executed easily by automatic systems, increasing the energy efficiency of train operation (Emery, 2017). With the objective of bringing ATO benefits to long distance trains, a TEN-T project (TEN-T - ATO project, 2016) has been developed to include ATO in the ERTMS specification. The new ATO over ERTM standard will include the requirements that this system must comply to drive the train automatically and to be interoperable.

Several projects are currently under development to implement ATO over ERTMS in real lines as the Mexico-Toluca project developed by CAF Signalling (Villalba, 2016) and the Thameslink project in London developed by Siemens (Burton, 2009). The commissioning of these projects is expected from 2019.

The requirements of the ATO over ERTMS establish that wayside equipment must supply the train the information of the assigned timetable. This could be the nominal timetable, or a timetable generated on-line by complex algorithms with traffic regulation purposes. Timetable information is provided by means of timing points. These points are defined as a set of the following information:

- Type of point: Departure, passing or arrival.
- Position of the point.
- Time assigned.

Along with the timing information, the wayside equipment also sends the information of the topology and operational conditions of the track. Thus, the on-board ATO equipment generates the speed profile that the train must perform to fulfil the

timetable (TEN-T - ATO project, 2016). By this way, the interoperability of train is ensured. Each train is responsible of generating its own speed profile following its own rules or driving commands to meet the target times.

One of the objectives pursued in the ATO over ERTMS specification is the energy efficiency of the driving. Eco-driving algorithms have an important role to play in the development of these systems. These algorithms must be capable of generating speed profiles that, not only meet a target running time minimising energy consumption as usual, but also meet intermediate passing times at timing points. Furthermore, these calculations have to be executed in real-time so the algorithms must be fast enough.

These characteristics make necessary the development of new eco-driving techniques capable of fulfilling the requirements of the ATO over ERTMS specification maximising the energy savings of the train driving.

1.2. NATURE INSPIRED OPTIMISATION

Optimisation processes are present in many actions of nature. For instance, the evolution of species, the search for the best source of food or metallurgic processes are cases of optimisation found in nature. For several decades, researchers have taken inspiration from this kind of processes to solve complex problems proposing numerous optimisation techniques. Some of these methods are part of Nature Inspired Computational Intelligence research field. These techniques are also known in the literature as metaheuristic methods.

In words of Glover and Kochenberger, it includes “any procedures that employ strategies for overcoming the trap of local optimality in complex solution spaces, especially those procedures that utilise one or more neighbourhood structures as a means of defining admissible moves to transition from one solution to another, or to build or destroy solutions in constructive and destructive processes” (Glover and Kochenberger, 2003). These methods usually make use of stochastic components, they are not gradient based and are problem independent. Moreover, they can describe and resolve complex relationships with practically no knowledge of the search space.

Frequently, real-world problems are difficult to solve because of their complexity. When traditional optimisation method fails, Nature Inspired optimisation can help to find solutions to these problems. The use of Nature Inspired Computational Intelligence does not guarantee the optimal solution of the problem in a finite time because it exhibits an asymptotic convergence behaviour (Cerf, 1998; Chiang and Chow, 1994). However, it can find near optimal solutions that, for most of the engineering problems, are good enough solutions.

The hardware improvements have led to reduce calculation times. Because of this, Nature Inspired methods have been remarkably developed during last thirty years. These methods have become very popular to solve complex optimisation problems, most of them, using detailed simulation models. These techniques have been applied in a variety of engineering fields such as: computer networks, power systems, security,

robotics, production engineering, biomedical engineering, control systems, data mining, etc... (Binitha and Sathya, 2012). The reasons for that popularity can be found in the flexibility and efficiency of these algorithms that, furthermore, tend to be easy to implement.

Among Nature Inspired algorithms, it can be found trajectory-based algorithms and population-based algorithms (Yang, 2014). The most popular trajectory-based algorithms is Simulated Annealing (SA) (Kirkpatrick et al., 1983). This method takes inspiration from the metallurgic process of annealing to solve optimisation problems. This algorithm makes use of a solution that moves in the search space. The random movements of the particle are always accepted if they find a better location. However, a move that do not improve the location of the solution is accepted with a certain probability that is constantly reduced during iterations.

On the other hand, population based algorithms make use of a set of solutions that evolves, or move, through iterations seeking the best solution. Evolutionary algorithms are those population-based algorithms inspired in the Darwinian principles of the evolution of species. These methods apply crossover and mutation operators to generate new solutions and, at the end of an iteration, only the best solutions survive using selection operators. Among evolutionary algorithms it can be found Evolutionary Programming (EP) (Fogel et al., 1966), Evolution strategies (ES) (Holland, 1992), Genetic Algorithm (GA) (Goldberg, 1989), Genetic Programming (GP) (Koza, 1994) and Differential Evolution (DE) (Storn and Price, 1997).

Swarm Intelligence methods are also population based algorithms, which take inspiration from the social behaviour of insects or other animal societies. These algorithms use a set of solutions that moves through the search space seeking individually the best position and cooperating with other population members by means of information sharing about the best solutions found. Examples of Swarm Intelligence algorithms are Particle Swarm Optimisation (PSO) (Kennedy and Eberhart, 1995), Ant Colony Optimisation (ACO) (Dorigo et al., 2006), Artificial Bee Colony (ABC) (Karaboga and Basturk, 2007), Firefly Algorithms (FA) (Yang, 2009) or Cuckoo Search (Yang and Deb, 2009).

All these algorithms have in common the use of exploration and exploitation mechanisms and the trade-off between them (Yang, 2014). Exploration or diversification consists in generating diverse solutions to explore the maximum portion of the search space. On the other hand, exploitation or intensification consists in generating solutions focused on the most promising region of search space. The appropriate combination of these two components makes possible to find the optimal, or at least a near optimal solution, and avoid local optimum.

The first steps of Nature Inspired optimisation was focused on single-objective problems. However, metaheuristic algorithms have been demonstrated highly efficient solving multi-objective optimisation problems (MOOPs) (Deb, 2010).

Multi-objective optimisation refers to problems which objective is to optimise simultaneously two or more conflicting objectives. Therefore, the result obtained is a

set of non-dominated solutions instead of a single solution. Non-dominated solutions are those that cannot be improved at the same time in all the objectives. The set of non-dominated solutions is usually called Pareto front. In some cases, the information provided by the Pareto front is useful to make a decision of selecting one of the solutions because it reflects clearly the trade-off among the objectives of the problem.

Several population-based algorithms have been proposed to solve multi-objective optimisation problems. Many of them are extensions of previously mentioned algorithms. Examples can be found in the literature such as Strength Pareto Evolutionary Algorithm (SPEA) (Zitzler, 1999), Non-dominated Sorting Genetic Algorithm (NSGA) (Srinivas and Deb, 1994), Pareto Archived Evolution Strategy (PAES) (Knowles and Corne, 1999), Multi-Objective Particle Swarm Optimisation (MOPSO) (Coello et al., 2004).

Many real-life optimisation problems are changing constantly. Changes cause modifications in some part of problem definition, the objective function or constraints. Therefore, the optimal solution varies with time as the problem environment changes dynamically. In the literature, this kind of problems is called dynamic optimisation problems (DOPs) or time-dependant problems. DOPs are challenging problems because the objective is not only to find the optimal solution but also to track its changes with time.

In the last 20 years, Nature Inspired Computational Intelligence has developed mechanisms to provide algorithms that solve DOPs. These mechanisms are: change detection mechanisms, the introduction diversity in the population, the use of memory from past iterations, the use of prediction or multiple populations (Nguyen et al., 2012). These mechanisms combined with previously mentioned algorithms have yielded extensions of static algorithms to solve DOPs not only from the single-objective point of view (Cruz et al., 2011) but also from the multi-objective point of view (Helbig and Engelbrecht, 2014).

1.3. THESIS OBJECTIVES

1.3.1. MAIN OBJECTIVE

The main objective of this thesis is the development of optimisation models for the calculation of train eco-driving based on detailed and realistic simulation of the train movement. The models proposed will investigate Nature Inspired Computational Intelligence techniques because of their suitability to be combined with train simulation models.

Furthermore, these models will take into account the most important restrictions and sources of uncertainty related to the real train operation. In particular, the restrictions related to passengers' comfort and driving systems, the uncertainty associated with the traffic and the variability of the load will be evaluated.

The studies carried out will differentiate between the eco-driving application on the planning stage and the online application during the regulation phase. The particularities of the specific railway systems will be taken into account considering urban railways and long-distance railways. The case of long-distance trains will be focused on high-speed railways because of its growing importance not only in Spain but also in the world.

1.3.2. SPECIFIC OBJECTIVES

The main objective can be divided into 5 specific objectives that will constitute the contributions of this thesis. These objectives will cover the different scenarios in terms of type of railway system and stage of eco-driving application:

1. Comparison of multi-objective optimisation models for the design of speed profiles in urban railways equipped with CBTC signalling system.

The existing signalling systems before CBTC work with a reduced bandwidth to communicate driving commands to the train. Consequently, the solution space is small and it can be explored by brute force methods. Last generation of signalling systems, as CBTC, provides a better bandwidth for communications increasing the number of driving possibilities. Thus, the challenge of designing efficient speed profiles in urban railways equipped with CBTC is to assess an enormous search space. This search space cannot be exhaustively explored and, for that reason, it is necessary to apply algorithms that find the most efficient solution for each possible running time. A Pareto curve of efficient speed profiles can be provided to the designer and, in view of the trade-offs, the speed profiles to be programmed for their execution during the train journey can be chosen. Different algorithms will be tested to check their performance in this task.

2. Development of an optimisation model for the automatic design of speed profiles in urban railways equipped with CBTC signalling system, taking into account the variation of the train load and the frequency of use of the pre-programmed speed profiles.

Solutions provided by the Pareto front must be useful in every condition of the real train operation. During a period of operation, the train load presents high variability. If this variation is not taken into account when obtaining the Pareto front, the eco-driving solutions obtained for a specific value of train mass could be non-efficient using other mass values. Furthermore, modifications in the shape of the speed profile could happen if the train mass is increased or decreased, making comfortable solutions to be non-comfortable. An optimisation model will be proposed to find the Pareto front of solutions that are robust to mass changes providing speed profiles that are efficient under different conditions.

On the other hand, to maximise the energy savings, it is useful to improve the selection of the speed profiles to be programmed in the ATO equipment. The application of a pre-programmed set of speed profiles in an interstation depends on the exiting delays in the line. Therefore, this source of uncertainty has to be taken into account. Using this

information, pre-programmed speed profiles can be selected automatically from the Pareto front with the objective of minimising the expected energy consumption.

3. Development of a model to optimise in real-time the speed profile of a high-speed train, which is manually operated.

Some studies in the literature optimise the train speed profile by means of a Genetic Algorithm combined with a detailed train simulator (Sicre et al., 2014). These studies have demonstrated that efficient speed profiles can be obtained in real time using Nature Inspired algorithms. However, the results obtained can be improved modelling the eco-driving problem as a dynamic optimisation problem. Dynamic problems are those in which the objectives, constraints or even the problem itself vary with time. For solving the dynamic eco-driving problem, algorithms from Nature Inspired Computation specialised in solving dynamic problems (Helbig and Engelbrecht, 2014) will be tested.

Besides, if the problem is stated as a dynamic multi-objective optimisation problem, the result will be a Pareto front that is adapted during the train movement. This way, the train will have available a set of different speed profiles ready to be used when necessary.

4. Incorporation of uncertainty in the dynamic multi-objective optimisation model to calculate real-time eco-driving solutions for high-speed trains.

To ensure that a train using an eco-driving solution will arrive on time to the next station, besides using an accurate model, it is necessary to model the uncertainty associated to real operation. The main uncertainty in high-speed trains regulation is associated with the contingencies that may occur in the line. Contingencies are usual situations that produce delays and are related to temporary speed limitations and traffic perturbations. The commercial running time between two stations is designed using a time margin in timetables to deal with contingencies. If necessary, this margin is available to make up delays and, if not, it can be consumed during the train travel to perform efficient speed profiles. If the margin is quickly consumed during the journey, the train could perform an efficient speed profile but it might not have reaction capacity to unexpected delays. Contrary, the speed profiles that retain the time margin until the end of the journey are highly energy consuming but more robust to contingencies in the line. Measuring the time margin consumption rate could be useful to measure the sensitivity of the solutions to contingencies.

In addition, it is also important to model the uncertainty associated with manual driving, considering that there are always small deviations in the application of driving commands. The uncertainty in these parameters is usually better represented using fuzzy knowledge modelling (Bellman and Zadeh, 1970).

A new dynamic optimisation algorithm is proposed including the margin consumption rate (called risk of delay in arrival) as well as a fuzzy model of the manual driving.

5. Development of an eco-driving model for a high-speed train fulfilling the ATO over ERTMS specifications.

The new standard of ATO over ERTMS (nowadays a draft) establishes that the speed profiles will have to fulfil a final running time and several intermediate passing times at certain positions. This includes in the eco-driving optimisation problem new restrictions that constrain the search space. A Nature Inspired algorithm will be proposed to deal with this constrained optimisation problem.

1.4. STRUCTURE OF THE DOCUMENT

This thesis is structured into seven chapters:

- Chapter 1 has presented an introduction to the thesis besides the motivation and objectives pursued.
- In Chapter 2, the multi-objective optimisation eco-driving problem for urban railways is presented and solved by two different optimisation algorithms that are compared.
- In Chapter 3, the previous model is extended to find robust solutions to train load variations. Furthermore, an automatic design of pre-programmed speed profiles procedure is proposed based on the stochastic delay distribution.
- Chapter 4 addresses the online regulation problem of a high-speed train by means of a dynamic multi-objective optimisation model. Two different algorithms are presented and compared.
- In Chapter 5, uncertainty is included in the model of the previous chapter. A measure for quantify the risk of a speed profile to future delays is proposed and added to the model as a third objective. Furthermore, uncertainty of manual driving is introduced by means of fuzzy parameters. This way, a new optimisation algorithm is proposed, the DNSGA-III-F.
- Chapter 6 presents the particularities of the ATO over ERTMS system and how affects the train driving. Moreover, a new eco-driving optimisation algorithm is developed for this system.
- Chapter 7 collects the main conclusions obtained from the work presented in the thesis. The most relevant contributions and future works are also detailed in this chapter.

OPTIMISATION ALGORITHMS FOR THE DESIGN OF ENERGY EFFICIENT ATO SPEED PROFILES IN METROPOLITAN LINES

2.1. INTRODUCTION

The objective in efficient driving (eco-driving) design is to find the speed profile that requires the minimum energy consumption without degrading commercial running times. With this aim, various mathematical models have been applied. However, the difficulties involved in the analytical resolution of the problem lead to a number of simplifications in the approaches (Albrecht et al., 2011; Franke et al., 2000; Khmel'nitsky, 2000; Ko et al., 2004; Miyatake and Ko, 2010; Su et al., 2013) which makes their application to real cases impractical. The approaches based on simulation offer more promising alternatives. They do not require simplifications in the models and enable an accurate calculation of the running times and the energy consumption. A number of optimisation techniques have been used in combination with simulation - Genetic Algorithm (Bocharnikov et al., 2010, 2007; Cucala et al., 2012b; Lu et al., 2013; Sicre et al., 2012; Wong and Ho, 2004b, 2003; Yang et al., 2012), Artificial Neural Networks (ANN) (Chuang et al., 2009, 2008), a combination of both the techniques (Acikbas and Soylemez, 2008) and direct searching methods (De Cuadra et al., 1996; Wong and Ho, 2004a). ANN have also been used to optimise the traffic regulation by learning traffic data (Lin and Sheu, 2011b).

However, additional difficulties are involved when the trains are equipped with Automatic Train Operation systems (ATO) as seen in many metropolitan trains today. The function of this equipment on board the train is to drive the train automatically according to a pre-programmed speed profile (Bocharnikov et al., 2007; Chang and Xu, 2000). The ATO equipment provides a control reference to the train's traction equipment for observing the maximum and safe speed limits and for stopping the train at stations. The intervention of human drivers is limited just to the opening and closing of the doors and to the starting up of the train after each stop. As a result, the running times and energy consumptions are quite stable when the signalling systems do not affect the circulation of trains.

A set of pre-programmed ATO speed profiles are available per interstation with different running times and energy consumption. Depending on the running time required, the centralised traffic regulation system on-line selects the ATO speed profile to be executed by each train between two stations (Fernandez et al., 2006). Each speed profile is programmed as a combination of the ATO commands that are transmitted to the train from the track via encoded balises or antennas.

The traffic regulation systems performance and the total energy consumption strongly depend on the off-line design of these ATO speed profiles. Studies on the driving efficiency in metropolitan trains with ATO can be found in (Bocharnikov et al., 2007; Chang and Xu, 2000; Khmelnitsky, 2000; Wong and Ho, 2004a; Zhou et al., 2011). However, they do not satisfy realistic constraints and control capabilities of any particular ATO system or the ATO model is simplified (De Cuadra et al., 1996), which makes difficult their implementation in real ATO equipment. The particular features of the ATO systems, the short inter-stations in metropolitan lines and the differences of a few seconds between the ATO speed profiles to be designed, make it necessary to develop accurate models that optimise the combination of the ATO commands of each speed profile to be pre-programmed in the equipment.

Most of the metropolitan railways use four alternative speed profiles per interstation (Fernandez et al., 2006). The first one (number 0) is characterised by the minimum running time (flat-out), that is to say, applying maximum available acceleration, speed and deceleration. The remaining profiles are slower and correspond to lower energy consumption. The maximum time gap between the fastest and the slowest is bounded as an operational criterion (Cucala et al., 2012a; Domínguez et al., 2011b).

Typically, the configuration variables of the ATO systems consist of four commands: coasting speed, re-motoring speed, speed holding value and braking deceleration rate. These commands are transmitted to the antenna located under the train via encoded balises on the track, which permit a limited amount of data bits to be transmitted due to the channel capacity (Gill and Goodman, 1992). Thus, only a few and discrete values of the commands can be sent providing each interstation with a relatively small solution space of speed profiles consisting of the combinations of these discrete values. This way, the optimal solution can be found by exhaustive search, simulating all the possible combinations as described in (Domínguez et al., 2011b).

Other studies have also looked for saving energy under the framework of the current fixed-block signalling system (Ke et al., 2009). However, the new state-of-the-art of signalling technologies such as CBTC permit a better communication capacity (bandwidth) with high-resolution train location determination, and bidirectional train-wayside data communications (“IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements,” 1999). More command values can be sent since the increment is smaller, thus resulting in an exponential solution space.

Therefore, this calls for a new method for the optimal design of the ATO speed profiles without an exhaustive simulation of all the combinations. Some studies are already trying to find optimal driving in CBTC (Communications-Based Train Control) systems (Gu et al., 2011). Some studies have proposed algorithms based on GA (Genetic Algorithm) (Bocharnikov et al., 2007; Lu et al., 2013; Wong and Ho, 2004a), or ACO (Ant Colony Optimisation) (Lu et al., 2013) for the single objective optimisation, but they do not consider either realistic characteristics of the ATO equipment.

On the other hand, other works state the eco-driving problem as an optimisation problem with two objectives. Usually, the two conflicting objectives are energy consumption and running time, and the result is the Pareto front of solutions in these two dimensions. In view of Pareto front, the decision maker can select the most appropriate speed profile taking into account the trade-off between energy consumption and running time. For solving this task, population-based algorithms from the Nature Inspired Computational Intelligence are typically used. Population-based algorithms seem to be especially suited to MOOP due to their abilities to search simultaneously for multiple Pareto optimal solutions and to perform better global search of the search space (Mitchell, 1996). Among population-based algorithms, it can be distinguished Evolutionary Algorithms and Swarm Intelligence techniques.

Many Evolutionary Algorithms have been developed for solving MOOP. Examples are: NSGA-II (Deb et al., 2002), which is a variant of NSGA (Non-dominated Sorting Genetic Algorithm); SPEA2 (Zitzler et al., 2002), which is an improved version of SPEA (Strength Pareto Evolutionary Algorithm) and PAES (Pareto Archived Evolution Strategy) (Knowles and Corne, 2000). These EAs are population-based algorithms that possess an in-built mechanism to explore the different parts of the Pareto front simultaneously.

Among Swarm Intelligence techniques, Particle Swarm Optimisation (PSO) (Kennedy and Eberhart, 1995, 2001, 1997) is one of the most popular methods. PSO imitates the social behaviour of insects, birds or fish swarming together to hunt for food. The PSO is also extended to solve MOOP. Among those algorithms that extend PSO to solve multi-objective optimisation problems are Multi-objective Particle Swarm Optimisation (MOPSO) (Coello et al., 2004), Non-dominated Sorting Particle Swarm Optimisation (NSPSO) (Li, 2003) and the aggregating function for PSO (Parsopoulos and Vrahatis, 2002).

Several studies have implemented this kind of techniques in an eco-driving MOOP. In (Chevrier et al., 2013) the Indicator Based Evolutionary Algorithm (IBEA) is applied to obtain optimal speed profiles but they do not take into account a realistic ATO

simulation model. NSGA-II is applied in (Dullinger et al., 2017) to optimise a train traction system taking into account the efficiency in driving but ATO is not modelled.

In (Sicre, 2013) and (Carvajal-Carreño et al., 2014) a realistic CBTC ATO equipment was modelled to obtain energy efficient speed profiles. Those works stated the eco-driving problem as a multi-objective optimisation problem (MOOP) and solve it using Nature Inspired Computational Intelligence techniques. While MOPSO was proposed in (Sicre, 2013), NSGA-II was applied in (Carvajal-Carreño et al., 2014). Both optimisation algorithms exhibit good performance solving eco-driving MOOP.

Some studies have compared evolutionary algorithms solving theoretical benchmark MOOPs. The results indicated that NSGA-II has shown better performance than PAES (Deb et al., 2002) and similar results to SPEA-II working with lower dimensional objective spaces (Zitzler et al., 2001). In (Coello et al., 2004) a similar comparison can be found where the performance of different multi-objective algorithms has been tested. MOPSO and NSGA-II are part of these algorithms. The results of this study indicate that MOPSO is the best in covering the full Pareto front of all the benchmark functions used. In addition, it is found to converge with a low computational time.

Despite these theoretical studies, the algorithm performance depends on the problem characteristics. MOPSO and NSGA-II have been proven successful solving eco-driving problem but they have not been compared to determine which one presents a better performance.

Therefore, this chapter of the thesis develops a multi-objective optimisation model to obtain the Pareto front with the optimal ATO speed profiles of a real metropolitan line. The optimisation model is based on simulation to calculate in detail the realistic characteristics of the ATO driving and the type of commands that can be sent to the train. The results of MOPSO and NSGA-II finding the Pareto front of optimal ATO speed profiles will be compared. Several performance metrics will be applied to find the best optimisation algorithm.

In section 2.2, the simulation model of a train equipped with an ATO system is introduced. Then, the MOOP proposed for the design of ATO speed profiles is detailed in section 2.3. Furthermore, MOPSO algorithm and NSGA-II algorithm for solving the MOOP are described in this section. Section 2.4 describes a case study of a real line and the proposed MOPSO algorithm is compared with NSGA-II to determine which presents the better performance. The results of some interstations are shown and, in addition, the comfort and operational constraints are introduced in the design of the case study in section 2.4. Finally, conclusions are presented in section 2.5.

2.2. TRAIN MOTION SIMULATION MODEL

When it comes to design an energy efficient driving profile, the decision variables are running time (RT) and energy consumption (EC), whilst the comfort criteria must be met. Simulation results of these variables must be precise in order to make a proper decision. The validity of the simulation results depends on the accurate simulation of

the complete journey of a train between two stations. The simulation model applied in this thesis for the accurate simulation of the train movement with an ATO system was developed in (Domínguez et al., 2011b; Sicre, 2013). The simulator can obtain every possible speed profile for that ATO system and check the fulfilment of the operational and comfort constraints specified.

The simulation model is composed of three modules: train, line, and ATO modules, considering all the main variables that affect the train dynamics. All the constraints that can affect the train motion (speed limitations, maximum traction effort, etc.) are included in the simulation model.

The model of the train takes into account the length and mass of the train, running resistance, traction and braking maximum curves, variations of the motor efficiency with respect to speed and effort ratio, auxiliary equipment consumption and rotary inertia. The line characteristics included in the model are speed limits, tunnels, grades, grade transitions (and the effect along the train) and bends. The ATO module interprets and executes the commands that can be pre-programmed in real ATO equipment and always observes the speed limitations of the line and the stop points at stations applying brake effort when necessary. The inputs of the ATO module are the same commands that the train receives from balises (or radio) at each interstation: a braking deceleration rate (b), a speed holding value (v_h), or a coasting speed (v_c) and a remotoring speed (v_r). Figure 2-1 shows speed profiles based on speed-holding regulation (left) and speed profiles based on coasting-remotoring cycles (right). The ATO sets the train traction to zero (coasting) when train speed reaches the coasting speed value v_c , and the ATO sets train traction to maximum force when the train speed falls below $v_c - v_r$. When the train is controlled by a holding speed parameter v_h , the ATO regulates the train traction to maintain the train speed to this value.

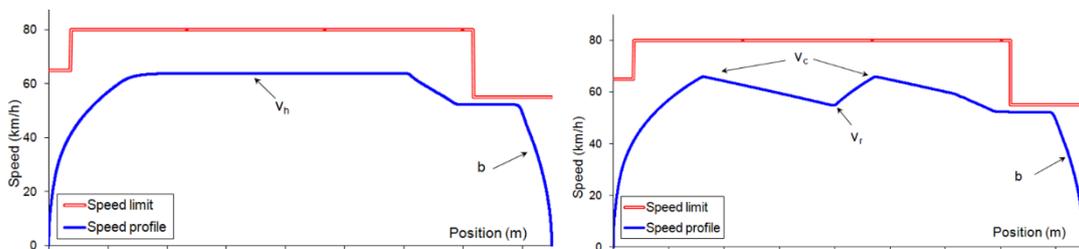


Figure 2-1. Speed profile with speed-holding (left) and coasting/re-motoring (right)

The train motion and the ATO modules and algorithms and the validation of them can be found in (Domínguez et al., 2011b). The parameter-setting and validation of the simulator to fit the models and adjust the results is an important aim to get reliable designs since the final goal is the implementation of the designed speed profiles and the regulation on the traffic operations allow only a few seconds of difference between profiles. With this purpose, all the real data of configuration of the ATO system, the train and the line such as the gradient with slope transition curves and a list of minimum speed limits along the curves are considered, as well as the distributed mass of the train (instead of a point mass train).

The time step-based simulation model calculates at each step the acceleration of the train from the balance of forces expressed in Eq. (2.1).

$$M_{eq} \cdot a = F_m - (F_r + F_g) \quad (2.1)$$

where the equivalent mass M_{eq} is the mass of the train plus the rotational inertial effect, a is the acceleration of the train, F_m is the motor force, F_r is the running resistance force and F_g is the force due to the track grades. Bends are modelled as equivalent grades. The equation of the running resistance is a function of the speed (v) of the train as shown in Eq. (2.2) (Davis formula), and the force due to the grades along the track is calculated as shown in Eq. (2.3)

$$F_r = A + B \cdot v + C \cdot v^2 \quad (2.2)$$

$$F_g = g \cdot m \cdot p \quad (2.3)$$

where m is the mass of the train, and p is the average grade of the track affecting the length of the train at each simulation step.

Once the acceleration rate is calculated, the new position of the train at each simulation step can be calculated from the uniformly accelerated motion equation. When the simulation from the departure station to the next one is finished, the running time RT is obtained.

The energy consumption of motors is calculated at each simulation step as the electrical power consumed multiplied by the time step, as shown in Eq. (2.4), when motoring. Since the ATO speed profiles design is carried out off-line and executed on different traffic scenarios, the voltage is assumed to be the nominal one. The current is obtained applying the ratio between the required force calculated and the maximum traction force, affected by the efficiency coefficient of the motor. See (Domínguez et al., 2011b) for a detailed description of the calculations.

$$EC = I \cdot U \cdot \Delta t \quad (2.4)$$

To calculate the energy consumption EC associated with the speed profile at that interstation, the power is integrated for all the simulation steps.

A comparison between the complete simulations and measured data has been carried out (Domínguez et al., 2011b). An average difference of 4.2% in traction energy and 1.0% in running times is obtained.

The simulated ATO speed profiles observe all the constraints associated both with the train motion (such as speed limitations, the maximum traction effort, etc.) and with the ATO driving. Likewise, there is an operational restriction of minimum speed limits along curves that must be observed in order to avoid an excessive wear of the wheels and track.

2.3. MULTI-OBJECTIVE OPTIMISATION FOR THE DESIGN OF ATO SPEED PROFILES

In (Domínguez et al., 2011b) a design method based on the accurate simulation of all the possible combinations of the ATO speed commands was presented. The commands are listed in Table 2-1. Combining the commands between the minimum and maximum with the corresponding increase and with a minimum difference of 5 km/h between the coasting and re-motoring speed, all the possible speed profiles per interstation (156) were obtained (the squared points in Figure 2-2). The Pareto front, which represents the minimum consumption for each running time, was also obtained for each interstation. It is a multi-criteria optimisation problem where the aim is to find the optimal trade-off between the energy consumption and running times. Decision theory techniques were used to solve it according to the shape of the Pareto front. The method has already been applied to real services such as the Madrid (Domínguez et al., 2011b), Barcelona and Bilbao (Cucala et al., 2012a) undergrounds in Spain achieving savings up to 15%.

	Deceleration rate (b) (m/s^2)	Speed holding (v_h) (km/h)	Coasting speed (v_c) (km/h)	Re-motoring speed (v_r) (km/h)
Minimum	0.6	30	30	10
Maximum	0.75	75	75	30
Increase	0.05	0.5	5	10

Table 2-1. Real values of the ATO speed commands

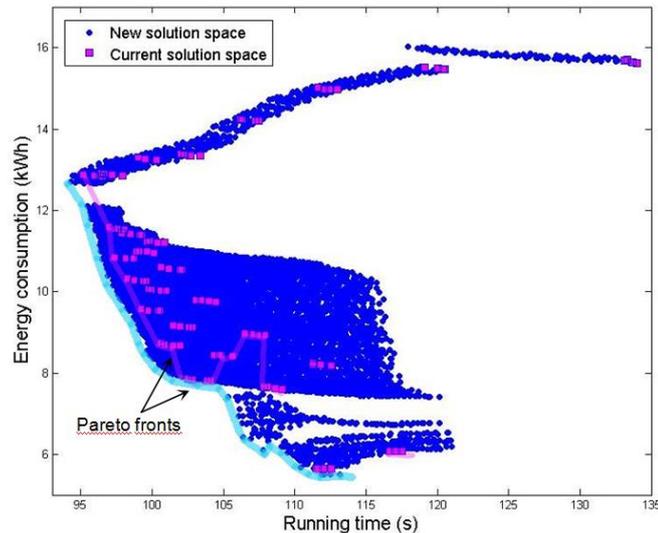


Figure 2-2. Solution space of one interstation

However, the new state-of-the-art of signalling technologies such as CBTC permit a better communication capacity (bandwidth) with high-resolution train location determination, and bidirectional train-wayside data communications (“IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional

Requirements,” 1999). More command values can be sent since the increment is smaller (Table 2-2), thus resulting in an exponential solution space. For instance, given the commands of Table 2-2, 20860 combinations are possible. In other words, there would be 20860 different speed profiles between two stations. Figure 2-2 shows the solution space of one interstation of the Madrid Underground as an example. The computation time is more than 1 hour. The figure also shows the current solution space using the commands of Table 2-1 of the same interstation. The computation time associated with Table 2-1 is less than 1 minute in this case, which allows the exhaustive search of the Pareto front. However, the large number of unavailable solutions and the benefit of using more combinations are evident. For instance, a speed profile with associated running time between 109 s and 111 s would never be selected. Likewise, a speed profile with running time 107 s would consume 7.6 kWh in the current solution space associated with Table 2-1 and 6.1 kWh in the new solution space associated with Table 2-2. Thus, 20% of savings is achievable thanks to the additional solutions.

	Deceleration rate (b) (m/s ²)	Speed holding (v_h) (km/h)	Coasting speed (v_c) (km/h)	Re-motoring speed (v_r) (km/h)
Minimum	0.6	30	30	5
Maximum	0.8	80	80	50
Increase	0.05	0.25	0.5	1

Table 2-2. ATO speed commands considered

Thus, a multi-objective optimisation model is proposed in the following for the design of ATO speed profiles in this framework with low computational time. As explained above, the objective of the design of ATO speed profiles is not to find a unique optimum point but a set of profiles with different running times, which is a MOOP. Thus, the objectives involved in the minimization function are now both the running time RT and the energy consumption EC (Eq. (2.5)).

$$\text{Min } \vec{f}(\vec{x}_p) = [EC(\vec{x}_p), RT(\vec{x}_p)] \quad (2.5)$$

where RT and EC are calculated by means of the simulation model previously described. The solution $\vec{x}_p = (b, v_h, v_c, v_r)$ is a 4-dimensional vector since there are four decision variables (deceleration rate, speed-holding, coasting speed and re-motoring speed).

The result of the optimisation is the set of non-dominated solutions. Non-dominated solutions are those that cannot be improved at the same time in all the objectives. Thus, in a non-dominated set, it is possible to find a speed profile with lower running time than another solution, but it will cost energy. In the same way, it is possible to find a solution with less energy consumption but it will require higher running time.

All the solutions provided by the model must be feasible with respect to driving constraints. Constraints associated with the train motion (such as speed limitations along the track, the maximum traction effort and braking curves of the train, etc.) are taken into account in the simulation model of the train and the line. In addition, constraints associated with the ATO driving are included in the ATO model.

Bearing in mind that the objective is to carry out a design for a possible implementation in a real line, it is important to take into account the comfort to ensure the successful implementation of suitable speed profiles from the passengers' point of view. For example, to apply coasting in a steep uphill results in an unpleasant feeling of falling. It highlights the importance of accurate and realistic simulations. An error of a few meters in a coasting point could make the train coast in an inappropriate place.

Therefore, a speed profile is considered unfeasible if it has a coasting point in a slope greater than 25 mm/m (Eq. (2.6)).

$$p < p_{max} \text{ if } F_m = 0 \text{ and } F_m^{prev} > 0 \quad (2.6)$$

where p is the slope, p_{max} is the maximum slope permitted in a point of coast start, F_m is the traction effort at this simulation step and F_m^{prev} is the traction effort at the previous step.

Some other comfort restrictions have been modelled and parameterised according to the metro operation experts' criteria: minimum speed throughout the journey (20 km/h) (Eq. (2.7)) and maximum number of re-motoring cycles per interstation (Eq. (2.8)).

$$v > v_{min} \quad (2.7)$$

where v is the speed of the train and v_{min} is the minimum permitted speed.

$$n_{remotoring} < n_{max} \quad (2.8)$$

where $n_{remotoring}$ is the number of re-motoring cycles executed along an interstation and n_{max} is the maximum number of re-motoring cycles permitted.

The comfort and operational constraints (Eqs. (2.6), (2.7) and (2.8)) are included in the model by means of a filtering process as will be described later.

The MOOP could be transformed into a single objective one, for instance through a weighted sum formulation or by choosing manually a solution, in the case the objective of the algorithm were to design a single driving for the interstation. However, as the purpose is to obtain the full Pareto front for all the possible running times, the proposed model will be more efficient than just repeating a single objective optimisation model repeatedly for different objective running times, to obtain one solution every time it is executed. Instead, the following algorithms are executed once, storing at each iteration the set of non-dominated solutions for different running times.

In the following, two algorithms are presented: the MOPSO (proposed in (Sicre, 2013)) and the NSGA-II and they are then compared to analyse their performance solving the eco-driving problem

2.3.1. MOPSO ALGORITHM FOR THE DESIGN OF ATO SPEED PROFILES

The population-based MOPSO algorithm conducts a search using a population of individuals. The individual in the population is called the particle and the population is called the swarm. The performance of each particle is measured according to domination criteria. Particles are assumed to “fly” over the search space in order to find promising regions of the landscape. In the minimization case, such regions possess lower functional values than other regions visited previously. Each particle is treated as a point in a space with a number of dimensions equal to no . They adjust its own “flying” according to its flying experience as well as the flying experience of the other companion particles. By making adjustments to the flying based on the personal best ($pbest$) guide and the global best ($gbest$) guide found so far, the swarm as a whole converges to the optimum point, or at least to a near-optimal point, in the search space.

The notations used in MOPSO and particularised to our problem are as follows: The j^{th} particle of the swarm is a specific combination of the ATO commands and in consequence, a specific speed profile. It is represented in iteration it by a 4-dimensional vector since there are four decision variables (deceleration rate, speed-holding, coasting speed and re-motoring speed), $\vec{x}p_j(it) = (b_j, vvh_j, vvc_j, vvr_j)$. Each particle or speed profile also has a position change known as velocity, which for the j^{th} particle in iteration it is $\vec{v}p_j(it) = (vb_j, vvvh_j, vvvc_j, vvvr_j)$. The best previous position (the position with the best fitness value) of the j^{th} particle is $\vec{p}_j(it) = (pb_j, pvvh_j, pvvc_j, pvvr_j)$. The best combination of commands in the swarm, is denoted by the index g . In a given iteration it , particle’s new velocity, based on its previous velocity and the distances from its current position to its $pbest$ and to the $gbest$ positions is updated using Eq. (2.9). The new velocity is then used to compute the particle’s new position Eq. (2.10).

$$\vec{v}p_j(it) = w \cdot \vec{v}p_j(it - 1) + c_1 \cdot r_1 \cdot (\vec{p}_j - x_i(it - 1)) + c_2 \cdot r_2 \cdot (\vec{p}_g - x_i(it - 1)) \quad (2.9)$$

$$\vec{x}p_j(it) = \vec{x}p_j(it - 1) + \vec{v}p_j(it) \quad (2.10)$$

where, $i = 1, 2, \dots, n_{swarm}$; $it = 1, 2, \dots, it_{max}$. The parameter n_{swarm} is the size of the swarm, and it_{max} is the iteration limit; c_1 and c_2 are positive constants (called “social factors”), and r_1 and r_2 are random numbers between 0 and 1; w is inertia weight that controls the impact of the previous history of the velocities on the current velocity, influencing the trade-off between the global and local experiences. It decreases during a run between w_1 and w_2 (Table 2-3) since a large inertia weight facilitates global exploration (searching new areas), while a small one tends to facilitate local exploration (fine-tuning the current search area).

The MOPSO algorithm used is similar to the ones described in the works of (Coello et al., 2004; Coello Coello and Lechuga, 2002) and (Alvarez-Benitez et al., 2005). It maintains an external archive A , containing the non-dominated speed profiles found by the algorithm so far that will constitute the Pareto curve. The MOPSO algorithm has been further refined by including the crowding distance mechanism (CD) found in (Raquel and Naval, Jr., 2005) and also used in (Zou et al., 2013) with the objective of achieving a more even distribution of the speed profiles on the Pareto front.

The crowding distance of a speed profile is the average distance of its two neighbouring solutions when all the speed profiles are sorted in ascending objective function values (the sum of both objectives). The speed profiles at the two extreme edges are given infinite crowding distance so that they are always selected to belong to the archive. In the MOPSO, the global best of the speed profile is selected with the Top select probability (Table 2-3) from among those non-dominated speed profiles with the highest crowding distance values. Selecting different guides for each combination of commands in a specified top part (Top select Table 2-3) of the sorted archive based on a decreasing crowding distance, allows the population to move towards those non-dominated speed profiles in the external archive that are in the least crowded area in the objective space.

The algorithm begins with the initialization of an empty external archive A . A population of n_{swarm} combinations of commands is randomly generated. The particle positions ($\vec{x}p_j$) and velocities ($\vec{v}p_j$) are initialised randomly and discretised. These are the values of the commands within their appropriate bounds, tabulated in Table 2-2. At each iteration it the velocities and the positions of the particles are updated using equations (2.9) and (2.10), respectively. The particle positions are checked if they are constrained to the bounded region after each update. Similarly, the velocities of particles are forced into the bounds, if they have crossed them. The objective functions (energy consumption and running time) $\vec{f}_j(\vec{x}p_j) = [EC(\vec{x}p_j), RT(\vec{x}p_j)]$ are then evaluated for each of the combination of commands, after performing simulation of the train drive.

The crucial parts of the MOPSO algorithm are selecting the personal and the global guides. There are no clear concepts of $pbest$ and $gbest$ that can be identified when dealing with a set of multiple objective functions. However, the following strategy can be applied to maintain the diversity in the swarm when selecting $pbest$ and $gbest$ positions.

If the current position of $\vec{x}p_j$ weakly dominates \vec{p}_j , that is, if the value of $\vec{x}p_j$ is lower or equal in both objectives than its current $pbest$, or if $\vec{x}p_j$ and \vec{p}_j are mutually non-dominating, then \vec{p}_j is set to the current position. However, if only one of these two objectives ($EC(\vec{x}p_j), RT(\vec{x}p_j)$) is improved, then \vec{p}_j is set to either the previous $pbest$ or the current position with a 50% probability.

Members of A are mutually non-dominating and no member of the archive is dominated by any $\vec{x}p_j$. All the members of the archive are, therefore, candidates for the global guide. However, if the members of the archive are selected as global guides based on uniform probability, the result is a Pareto front with substantial gaps in between. To fill up these gaps, the crowding distance MOPSO selects the global best guide from a specified top portion (Top select Table 2-3) of the archive A sorted in descending values of the crowding distance.

The MOPSO algorithm described is represented in Figure 2-3.

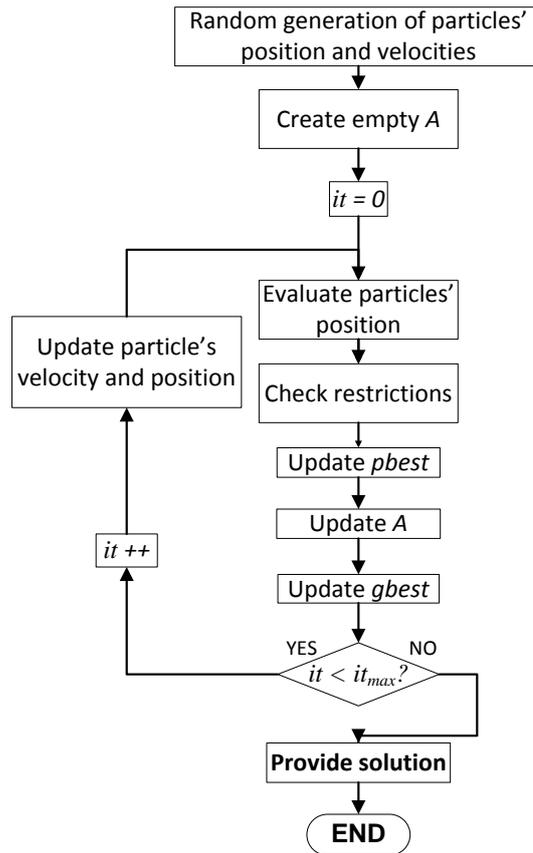


Figure 2-3. MOPSO Algorithm

2.3.2. NSGA-II ALGORITHM FOR THE DESIGN OF ATO SPEED PROFILES

NSGA-II uses a population of individuals to perform the search of Pareto front as well as MOPSO. NSGA-II is a class of genetic algorithm for solving MOOP. Therefore, it takes inspiration from Darwinian principles of the evolution of species. The population of solutions evolves through iterations where only the best solutions survive. The best solutions produce new solutions by means of crossover and mutation operators. These new solutions are called offspring. The fitness of individuals of this population are calculated by means of the domination level metric.

The domination level of an individual is calculated as the number of solutions that dominate that specific individual. Notice that the Pareto front found by the algorithm will be the set of solutions with dominance level equal to zero. Therefore, solutions with lower level of domination level are preferable because they are closer to Pareto front.

Each solution j^{th} of the population is a specific combination of the ATO commands and has associated a specific speed profile. As in MOPSO, it is represented in iteration it by a 4-dimensional vector $\vec{x}p_j(it) = (b_j, vh_j, vc_j, vr_j)$. Solutions are generated randomly at the beginning of the optimisation process or by means of crossover and mutation operators.

Crossover generates a solution from two individuals from solutions called parents. This operator is usually related to exploitation. The parents are randomly selected from a

portion of the surviving population with certain probability (Table 2-4). The characteristics of the two parents are combined to form a new solution. Thus, one parent is selected randomly as the basis of the new solution and a driving parameter is substituted taking the value from the same parameter of the other parent. If the solutions are not compatible, for example because coasting-remotoring parent is mixed with a regulation parent, the offspring is discarded and the process is repeated.

On the other hand, mutation operator generates solutions from a single surviving solution. This operator is usually related to exploration. The parent is randomly selected from a portion of the surviving population with certain probability (Table 2-4). The driving parameters of the parent is taking as the basis of the offspring solution. Then, one parameter is randomly chosen to be mutated. The value of this parameter is then substituted by a random value. Again, if the offspring solution is not feasible, for example because it presents coasting-remotoring parameters and a holding speed parameter, the offspring is discarded and the process is repeated.

The optimisation process is the same proposed in (Deb et al., 2002). NSGA-II starts creating a parent population P_0 of n_{pop} random solutions. Mutation and crossover operators are used to generate the offspring population of n_{pop} individuals (Q_0). After that, P_0 and Q_0 are joined to generate the result population R_0 of size $2n_{pop}$. Then, domination level is used to sort R_0 population and solutions that do not fulfil comfort and operational restrictions are given the worst possible punctuation (Eqs. (2.6), (2.7) and (2.8)).

Once the R_0 population is ranked, the best n_{pop} solutions survive and the rest are eliminated. The parent population for the next iteration P_1 is generated obtaining the n_{pop} solutions from R_0 with the lowest domination level. This process is performed adding solutions in sets of individuals that share the same domination level. If a set of solutions cannot be included because it exceeds the size of the parent population, the same crowding distance operator (CD) used for MOPSO (Raquel and Naval, Jr., 2005) is applied to that set of solutions to select the ones that will be part of P_1 . CD operator calculates the distance, in terms of the optimisation objectives, of a solution to the individuals that surround it. Therefore, the solutions of the last set of individuals are included in P_1 in decreasing order of CD until the parent population is filled. This way, the zones with low density of solutions are prioritised.

Once P_1 is created, the next iteration of the algorithm starts generating the offspring population Q_1 using crossover and mutation operators and, after that, a new result population R_1 is created. Again, each solution $\vec{x}p_j$ of the population is evaluated by means of the simulation model presented in 2.2. This process is repeated until a number of iterations it_{max} is reached ($it = it_{max}$). The final result is the set of solutions with 0 domination level of the last result population $R_{it_{max}}$.

In Figure 2-4 the NSGA-II flowchart is depicted.

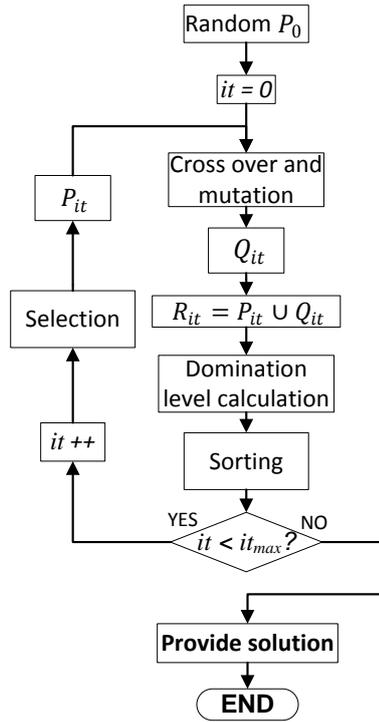


Figure 2-4. NSGA-II Algorithm

2.4. CASE STUDY AND RESULTS

The experiments presented have been carried out using real data from a train and a line of an urban railway system. In particular, Metro de Madrid Line 3 data have been used as case study. Before the test, both algorithms have been tuned searching the best parameter configuration.

Table 2-3 shows the values selected for the MOPSO parameters.

c_1	c_2	w_1	w_2	Top select (%)	Top select probability (%)
1	1	0.9	0.2	6	98

Table 2-3. Tuned parameters of the MOPSO algorithm

NSGA-II has been programmed in the same platform as MOPSO to ensure that the computation time is comparable. Furthermore the population size in the NSGA-II programmed has the same value than the number of particles in MOPSO and the maximum number of iterations is equal in both cases. This way, the number of simulations made by both algorithms in a run is equal. Simulation are computational expensive and, therefore, practically the calculation time required by the algorithms depends on the number of evaluations carried out. The configuration parameters that have been used in the NSGA-II are shown in Table 2-4.

Population size	Number of mutations	Number of crossovers	Top select (%)	Top select probability (%)	Maximum number of iterations
80	40	40	25	85	50

Table 2-4. Tuned parameters of the NSGA-II algorithm

2.4.1. COMPARISON OF MOPSO AND NSGA-II

The assessment of the results obtained with both algorithms has been carried out using the metrics described in (Coello et al., 2004) and (Jiménez et al., 2013) that compare the results of the algorithms with the points of the real Pareto front. Such real Pareto front has been obtained through exhaustive simulation of the full space of possible solutions.

a) Normalised Hypervolume.

The normalised Hypervolume metric represents the fraction of the objective space that is not dominated by any of the solutions obtained. This indicator provides information about the distance of the solutions to the real Pareto front and about the spread. This metric is calculated as shown in Eq. (2.11).

$$HV = 1 - \frac{\sum_{j=1}^{NI} \left[(f_{no}^{max} - f_{no}^j) \prod_{q=1}^{no-1} (f_q^{sup_j} - f_q^j) \right]}{\prod_{j=1}^{no} (f_q^{max} - f_q^{min})} \quad (2.11)$$

where f_q^j is the value of the objective q for the individual j , no is the number of objectives and NI the number of individuals of the efficient front obtained with the algorithm. f_q^{max} and f_q^{min} are the maximum and the minimum values for the q^{th} objective if the objective space is bounded. If the objective space is not bounded these are the values that satisfy $f_q^{max} > f_q^j$ and $f_q^{min} < f_q^j$ for each individual j . $f_j^{sup_q}$ is the value of the objective q^{th} for the individual higher adjacent in the q^{th} objective to individual j .

a) Generational distance

The generational distance calculates the proximity of the solutions obtained by the algorithm to the population of the real Pareto front. This indicator is calculated as shown in Eq. (2.12).

$$\gamma = \frac{\sqrt{\sum_{j=1}^{NI} d_j^2}}{NI} \quad (2.12)$$

where d_j is the Euclidean distance between the solution j and the nearest solution in the real Pareto front.

b) Spread.

The spread indicator estimates the diversity of the set of solutions obtained. This metric is calculated as shown in Eq. (2.13).

$$\Delta = \frac{\sum_{q=1}^{no} d_q^e + \sum_{j=1}^{NI} |d_j - \bar{d}|}{\sum_{q=1}^{no} d_q^e + NI \cdot \bar{d}} \quad (2.13)$$

where d_q^e is the Euclidean distance between the extreme solutions in the front obtained with the algorithm and the extreme solutions in the real Pareto front. d_j is the Euclidean distance between adjacent solutions and \bar{d} is the mean value of such measurements.

c) Error ratio

The error ratio indicator calculates the percentage of solutions provided by the algorithm that are not members of the real Pareto front. This indicator is calculated as shown in Eq. (2.14).

$$ER = \frac{\sum_{j=1}^{NI} e_j}{NI} \cdot 100 \quad (2.14)$$

where $e_j = 0$ if the particle j is a member of the real Pareto front and $e_j = 1$, otherwise.

d) Spacing

Spacing indicator measures the distance variance of neighbouring individuals in the front obtained with the algorithm. The value obtained assesses how well the solutions in such front are distributed. This indicator is calculated as shown in Eq. (2.15).

$$SP \triangleq \sqrt{\frac{1}{no-1} \sum_{j=1}^{no} (\bar{d}' - d_j')^2} \quad (2.15)$$

where d_j' is the distance of neighboring individuals in the front obtained with the algorithm and \bar{d}' is the mean value for these distances.

A real case based on AL1 (Almendrales, track 1) interstation from Metro Madrid has been used to test the performance of both algorithms. The obtained results are measured by the foregoing indicators and are presented in Table 2-5 and Table 2-6. Figure 2-5 shows graphically the comparison of the results.

It is shown in Figure 2-5 that the MOPSO algorithm front is closer to the real Pareto front and it has a better distribution of the individuals in the objective space. Solutions provided by the NSGA-II algorithm can consume up to 10% more than the solutions provided by MOPSO. Furthermore, for higher running times the NSGA-II has difficulties to find non dominated solutions.

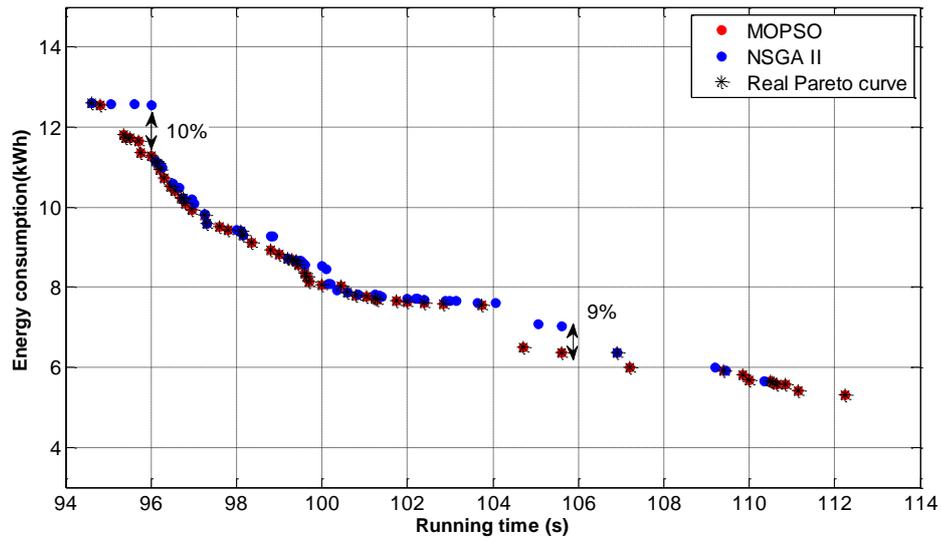


Figure 2-5. Front obtained by MOPSO and NSGA-II

It has to be noted that the real Pareto front, calculated by exhaustive simulation of all the possible combinations of ATO commands, is a non continuous curve and the discrete solutions are not uniformly distributed, as just the solutions that can be executed by the real ATO equipment belong to such real front. Consequently, it is important to calculate the diversity measures for the real Pareto front (Hypervolume, spread and spacing) in order to later compare the measures of MOPSO and NSGA-II algorithms with them.

Table 2-5 summarises the previously defined measures associated with the proximity to the real Pareto front and the diversity of solutions. For a computational time practically equal in both algorithms, the measure of the proximity to real Pareto front shows that the MOPSO provides better solutions (for example, the error ratio of MOPSO is 16.95% much less than 77.55% of the NSGA-II)

	Normalised Hypervolume	Generational distance	Spread	Error ratio	Spacing	Computational time (minutes)	Number of solutions
MOPSO	0.3526	0.0007936	0.5611	16.95%	0.4286	11.96	59
NSGA-II	0.3873	0.0044525	0.5738	77.55%	0.5161	11.86	49
Real Pareto front	0.3500	N.A.	0.4394	N.A.	0.3513	59.79	73

Table 2-5. MOPSO and NSGA-II performance indicators

For the sake of clarity, Table 2-6 shows the distance of the diversity measures in percentage for both algorithms to the measures of the real Pareto front. It should be noted that the Hypervolume of the solution obtained by the MOPSO is practically the Hypervolume of the optimal solution (0.74% difference) while for the NSGA-II is 10%

higher. The spread, spacing and number of solutions metrics of MOPSO have closer values to the real Pareto front than NSGA-II metrics.

	Normalised Hypervolume (difference in %)	Spread (difference in %)	Spacing (difference in %)	Number of solutions (difference in %)
Difference MOPSO-real Pareto front	0.74%	27.69%	22.00%	-19.18%
Difference NSGA-II - real Pareto front	10.66%	30.58%	46.91%	-32.88%

Table 2-6. Distance between metric of MOPSO and NSGA-II to the real Pareto front

Figure 2-6 shows the evolution of the calculations for both algorithms through the normalised Hypervolume indicator. It shows that the MOPSO is improving the result of the found solution until the last iteration while NSGA-II finds a stable solution in 24 iterations but with a higher error.

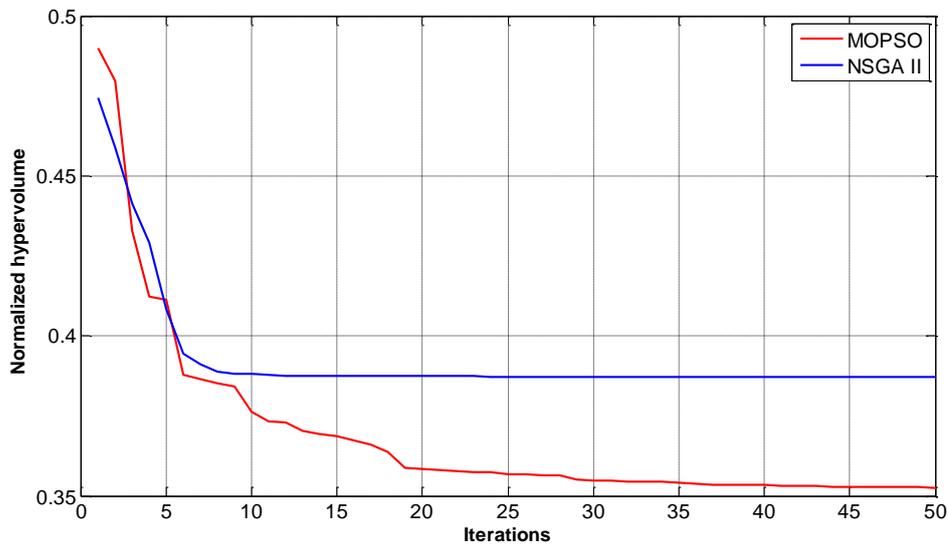


Figure 2-6. Normalised Hypervolume value during the evolution of MOPSO and NSGA-II

Thus, it can be concluded that the MOPSO algorithm has better performance than the NSGA-II algorithm, for diversity of solutions and proximity to the real Pareto front.

2.5. CONCLUSIONS AND CONTRIBUTIONS

Previous works have already highlighted the importance of the optimal design of ATO speed profiles, taking into account the technical requirements, the quality of service and additionally the minimisation of energy consumption. However, the design procedures are based on the exhaustive simulation of all the possible combinations of the speed commands, and the solutions obtained provide an average energy savings of 15%.

The new signalling technologies such as CBTC permit a better communication capacity and thus, the possible values of the ATO parameters that can be sent to the train (that is, the associated number of different possible speed profiles) is drastically higher. It has been shown that some of these new possible speed profiles are more efficient from the energy point of view (up to 20%) and are located precisely in some of the Pareto front gaps. Consequently, it is important to propose a new design algorithm that can find efficiently the Pareto front in this new solution space.

In this chapter of the thesis, two algorithm for the optimal design of the ATO speed profiles has been applied and compared based on the accurate simulation of the ATO and train motion.

MOPSO algorithm is efficient in computational time and has better performance solving the eco-driving MOOP proposed than the NSGA-II algorithm, for metrics of diversity and proximity to the real Pareto front.

The main contributions of this chapter are:

- The application of the NSGA-II algorithm to the eco-driving problem defined by real ATO speed commands.
- The use of several metrics to assess the performance of optimisation algorithms when solving the eco-driving problem.
- The comparison between MOPSO algorithm and NSGA-II solving the eco-driving problem.

CHAPTER 3

ROBUST SPEED PROFILES FOR THE AUTOMATIC TRAFFIC REGULATION SYSTEM IN METROPOLITAN LINES

3.1. INTRODUCTION

In the previous chapter, a multi-objective optimisation model was proposed for obtaining the Pareto front of efficient speed profiles and two algorithms were tested to determine the best one. That study does not take into account the uncertainty of the solutions obtained. The main sources of uncertainty are the train load and the train delays.

The high precision of the ATO equipment in the execution of the pre-programmed driving parameters is practically just affected by the uncertainty in the mass of the train associated with the passengers load (Lin and Sheu, 2008). The value of the train mass has two components. The empty vehicle mass and the passenger load. The empty vehicle mass is a fixed value whereas the passenger load varies depending not only on the departure station but also on the moment of the observation. It is obvious that variations of passenger load produce different results on energy consumption. However, it also causes variations in the running time produced by a set of driving parameters because the train resistance is dependant of the train mass. For the same reason, the load variations could cause variations in the shape of the speed profile making an expected comfortable solution to become non-comfortable.

The CBTC system permits pre-programming different sets of speed profiles for different operation periods, as peak-hours and off-peak-hours. However, this is not enough. At each period, the passengers load takes a different mean value, but it is not constant throughout the period. Passengers flow and the trains headway are constantly varying leading to important oscillations of load carried by different trains for the same stretch.

It can be observed in the literature that most of eco-driving studies do not take into account the uncertainty caused by mass variation. Usually, as in the previous chapter of this thesis, the train mass is considered a fixed value. This value corresponds to the mean passenger load expected. Therefore, these studies do not consider mass variations obtaining a result that could be non-optimal for the real operation.

Few works can be found that address the eco-driving problem with variable train mass. The passenger load is represented by means of fuzzy numbers in (Carvajal-Carreño et al., 2014). In this work, NSGA-II with fuzzy parameters is applied to calculate the Pareto front of efficient speed profiles whose objectives are running time and energy consumption. The result obtained by this method is a Pareto front where the solutions are non-dominated according to a necessity value. This approach is useful to maximise the energy savings taking into account different mass conditions. However, the robustness of the solutions, i.e. the sensitivity of solutions to vary its energy consumption and running time with mass variation, is not considered and the implementation of the solutions could be difficult. A set of driving parameters could produce a different driving shape with a different mass value. This could lead to a drastic change in the results of running time and energy consumption. Drastic changes in running time and energy results are negative for traffic regulation because the behaviour of the train is not predicted.

A similar conclusion could be derived by the work in (Xin Yang et al., 2016). This paper proposed a bi-level stochastic model to solve the timetable optimisation jointly with the speed profile optimisation. In this model, the mass variation is represented by means of a stochastic distribution. Therefore, the result is the best energy performance for the conditions of the study but the robustness of solutions is not considered.

The other main source of uncertainty associated with the traffic operation is the occurrence of delays that must be corrected by the traffic regulator (Lin and Sheu, 2008). The statistical distribution of the delays determines the frequency with which the controller demands each pre-programmed ATO speed profile at each station. Typically, the four selected speed profiles of the pre-programmed set are equidistant in time, and the frequency of use is not taken into account (Domínguez et al., 2011b).

In this chapter, a new method to design efficient and robust ATO speed profiles in CBTC lines is proposed, considering the previously described sources of uncertainty.

Two steps compose this method. In the first step, the Pareto curve of robust ATO speed profiles, taking into account the uncertainty in the mass, is generated. Two procedures are applied to generate the robust Pareto curve. The first one is a robust optimisation technique in time and energy consumption. An alternative procedure, based on the analysis of the relationship between the robustness and the conservation of the shape

of the speed profiles (pattern robustness), is proposed. This procedure makes use of the MOPSO algorithm described in (Domínguez et al., 2014) adding a driving pattern identifier. In the second step, the set of ATO speed profiles to be programmed in the regulation system is selected from the robust Pareto front by means of an optimisation model. This model is based on the Particle Swarm Optimisation algorithm (PSO) (Kennedy and Eberhart, 2001) to minimise the energy consumption throughout an operation period, taking into account the statistical distribution of running times demanded by the regulation system.

In section 3.2 the two proposed methods to obtain a robust Pareto front of the speed profiles are presented. Section 3.3 describes the optimisation model to select the pre-programmed speed profiles. The methodology proposed is applied to a case study and the results and analysis obtained from the design of ATO speed profiles are discussed in section 3.4. In addition, in section 3.4 the energy savings associated with the increase of the number of ATO speed profiles to be programmed in the regulation system are analysed. Finally, conclusions are presented in section 3.5.

3.2. DESIGN OF EFFICIENT AND ROBUST ATO SPEED PROFILES

In this section, the Pareto curve of optimal ATO speed profiles is calculated, where the objectives are the running time and the energy consumption. For each running time the optimisation algorithm has to find the ATO speed profile with the minimum consumption. For example, in Figure 3-1, two speed profiles with the same running time are shown. These optimal speed profiles are later used by the regulation system in an operation period (peak or non-peak hour). The speed profiles must be robust against variations of the train mass in that period. The decision variables are the driving parameters of the speed profiles, that is, the typical configuration variables of the ATO equipment: coasting speed (v_c), re-motoring speed (v_r), holding speed (v_h) and braking rate (b). The ATO sets the train traction to zero (coasting) when train speed reaches the coasting speed value v_c , and the ATO sets train traction to maximum force when the train speed falls below $v_c - v_r$. When the train is controlled by a holding speed parameter v_h , the ATO regulates the train traction to maintain the train speed to this value. Each ATO speed profile (a possible solution) is defined as a configuration vector of driving parameters $\vec{x}_p = (b, v_h, v_c, v_r)$. In addition, the ATO always observes the speed limitations and the stopping point at the station by applying braking force when necessary (see (Domínguez et al., 2011b) for more details).

In the following subsections, the robust optimisation method is described and also the alternative procedure based on conservation of the driving pattern.

3.2.1. GENERATION OF ROBUST PARETO FRONT OF OPTIMAL SPEED PROFILES USING A ROBUST OPTIMISATION TECHNIQUE

A robust solution is the less sensitive to perturbations in the decision variables in its vicinity (Deb and Gupta, 2005). The multi-objective optimisation approach deals with the search of solutions which are non-dominated by any other feasible solutions. Thus,

it is necessary to check the sensitivity of each solution \vec{x}^p in the *no* objectives to changes in all the decision variables to provide a robust Pareto front.

In (Deb and Gupta, 2005), two approaches are proposed to obtain the robust optimal front. In order to obtain a multi-objective robust solution Type-I a mean effective objective function ($\vec{f}^{eff}(\vec{x}^p)$) is used for the optimisation instead of the original objective function ($\vec{f}(\vec{x}^p)$). Therefore, a solution \vec{x}^p^* is a robust solution of Type-I if it belongs to the Pareto-optimal solution of the following problem (3.1).

$$\begin{aligned} \text{Minimize } \vec{f}^{eff}(\vec{x}^p) &= (f_1^{eff}(\vec{x}^p), f_2^{eff}(\vec{x}^p), \dots, f_{no}^{eff}(\vec{x}^p)) \\ \text{subject to } \vec{x}^p &\in S \end{aligned} \quad (3.1)$$

where S is the solution space and $f_j^{eff}(\vec{x}^p)$ is the mean effective objective function of the j objective for the neighbourhood δ . The mean effective objective function is defined in (3.2).

$$f_j^{eff}(\vec{x}^p) = \frac{1}{|B_\delta|} \int_{\vec{y} \in \vec{x}^p + B_\delta} f_j(\vec{y}) dy \quad (3.2)$$

where $|B_\delta|$ is the Hypervolume (Deb and Gupta, 2005) of the chosen vicinity δ .

The second approach proposed by Deb and Gupta adds a restriction to the original problem so that the user can adjust the maximum value of the solutions sensitivity. Therefore, a solution \vec{x}^p^* is a robust solution of Type-II if it belongs to the Pareto-optimal solution of the problem (3.3):

$$\begin{aligned} \text{Minimize } (f_1(\vec{x}^p), f_2(\vec{x}^p), \dots, f_{no}(\vec{x}^p)) \\ \text{subject to } \frac{\|\vec{f}^P(\vec{x}^p) - \vec{f}(\vec{x}^p)\|}{\|\vec{f}(\vec{x}^p)\|} \leq \eta \end{aligned} \quad (3.3)$$

$$\vec{x}^p \in S$$

where $\vec{f}^P(\vec{x}^p)$ is the perturbed objective vector and η is the maximum value of sensitivity required. The perturbed objective vector can be chosen either as the worst case or as the mean effective of the neighbourhood.

Type-II was chosen to obtain the optimal speed profiles because it is more practical than Type-I as it is highlighted in (Deb and Gupta, 2005). Furthermore, in the present ATO design problem, the worst cases, that is, the maximum variations of running time and energy consumption occur in the extreme values of train mass for the considered operation period. As previously mentioned, the high precision of the ATO equipment in the execution of the driving parameters is affected, in practice, only by the uncertainty in the mass of the train associated with the passenger load.

Therefore, the Type-II robust optimisation model has been selected because the robustness can be controlled by a maximum sensitivity value. The sensitivity is

controlled comparing the objective function for the mean value of the train mass in the operation period with the worst case (maximum and minimum mass). The optimisation problem is formulated as (3.4).

$$\begin{aligned} \text{Minimize } \vec{f}(\vec{xp}) &= (f_1(\vec{xp}), f_2(\vec{xp}), \dots, f_{no}(\vec{xp})) \\ \text{subject to } \frac{\|\vec{f}_{max}^P(\vec{xp}) - \vec{f}(\vec{xp})\|}{\|\vec{f}(\vec{xp})\|} &\leq \eta \\ \frac{\|\vec{f}_{min}^P(\vec{xp}) - \vec{f}(\vec{xp})\|}{\|\vec{f}(\vec{xp})\|} &\leq \eta \end{aligned} \quad (3.4)$$

$\vec{x} \in S$

where $\vec{f}_{max}^P(\vec{xp})$ and $\vec{f}_{min}^P(\vec{xp})$ are the objective vector for the maximum and the minimum mass, for the considered period, and $\vec{f}(\vec{xp})$ is the objective vector for the mean value of the mass in that period.

A multi Objective Particle Swarm Optimisation algorithm with crowding distance mechanism (MOPSO-CD) (Domínguez et al., 2014) is applied to generate the Pareto curve of the problem defined in (3.4).

In the MOPSO algorithm, each particle (possible solution) is treated as a point in an no -dimensional space. The particle “flies” over the search space to find promising regions of the landscape. The movement of a particle is directed by the best position found by itself ($pbest$) and the best position ($gbest$) found by the whole swarm (population). So the position and the velocity of the particle j are updated at each iteration using (3.5) and (3.6).

$$\vec{vp}_j(it) = w \cdot \vec{vp}_j(it-1) + c_1 \cdot r_1 (\vec{p}_j - \vec{xp}_j(it-1)) + c_2 \cdot r_2 (\vec{p}_g - \vec{xp}_j(it-1)) \quad (3.5)$$

$$\vec{xp}_j(it) = \vec{xp}_j(it-1) + \vec{vp}_j(it) \quad (3.6)$$

where $\vec{xp}_j(it)$ and $\vec{vp}_j(it)$ are the position and the speed of particle j at iteration it . The positions \vec{p}_j and \vec{p}_g are respectively the $pbest$ and the $gbest$. c_1 and c_2 are positive constants called (“social factors”), and r_1 and r_2 are random numbers between 0 and 1. w is an inertia weight that controls the effect of the previous history of velocities.

The MOPSO algorithm deals with a multi-objective optimisation problem where the objective is to find the Pareto curve. Pareto curve is the set of the non-dominated solutions. A solution is dominated if there is another feasible solution that performs a lower value in one objective (energy consumption or running time) and, it has a lower or equal value in the other objective. The objective function is shown in (3.7).

$$\text{Minimise } \vec{f}(\vec{xp}) = (RT(\vec{xp}), EC(\vec{xp})) \quad (3.7)$$

where $RT(\vec{xp})$ is the running time and $EC(\vec{xp})$ is the energy consumption.

$RT(\vec{x})$ and $EC(\vec{x})$ are calculated by means of a detailed simulator of the train movement and the ATO equipment (Domínguez et al., 2011b).

The algorithm maintains an external archive A that stores the non-dominated solutions found throughout the iterations. The $gbest$ is randomly selected at each iteration from the solutions in the archive A . However, if the solutions of A were selected based on uniform probability distribution, the resulting Pareto curve would contain substantial gaps. To avoid this problem, the crowding distance MOPSO (Raquel and Naval, Jr., 2005) selects the $gbest$ from a top portion of the archive A sorted in decreasing values of crowding distance (CD). The CD is calculated as the sum of distances in all the objectives of its two neighbouring solutions

On the other hand, the $pbest$ of a particle is selected as the current position if it dominates the previous $pbest$. If the previous $pbest$ dominates the current position, the $pbest$ remains. However if the current position of the particle and its $pbest$ are mutually non-dominated, the $pbest$ is selected randomly between them.

3.2.2. GENERATION OF ROBUST PARETO FRONT USING AN ALTERNATIVE METHOD BASED ON DRIVING PATTERNS

The driving pattern determines the shape of the speed profile. A pattern is characterised by the sequence of the different working modes that the train executes during its journey. These working modes are traction (T), braking (B), coasting (C) and final braking (FB). Alternative solutions \vec{x} are obtained exploring the variable decision space (values for v_c, v_r, v_h, b). They are simulated to calculate their associated pattern, running time and energy consumption.

In Figure 3-1, two examples of driving patterns are presented. The ATO command is the ratio between the force demanded by the automatic driver and the maximum force that the motor can provide. Positive values of the ATO command produce traction force and negative values of the ATO command produce braking force. In coasting-remotoring stages the ATO command present typically two values (0 and 1 in Figure 3-1). Zero value of the ATO command corresponds to a coasting phase (see (Domínguez et al., 2011b) for more details about ATO model).

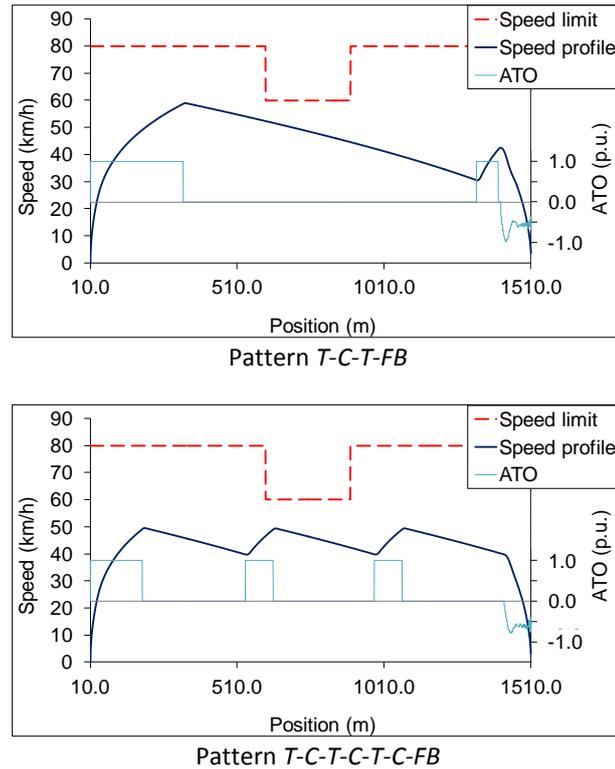


Figure 3-1. Different driving patterns with the same running time

The driving pattern of a speed profile provides information to the designer about the passengers' comfort. Furthermore, information about the robustness of the speed profile can be obtained from the driving pattern. Frequently, the speed profiles which change their driving pattern with the train load have associated high variations in running time and energy consumption. A solution $\bar{x}\bar{p}$ is said to be pattern robust if its associated pattern does not change with train mass variations.

Therefore, the following alternative method is proposed to obtain a robust Pareto front based on the detection of changes in the pattern:

- Step 1: Obtain the Pareto front considering the mean value of the train mass in the operation period using the MOPSO-CD algorithm without any robustness restriction
- Step 2: Identify the different driving patterns of the solutions obtained in step 1.
- Step 3: Eliminate the solutions with uncomfortable patterns according to the designer's criteria.
- Step 4: For each pattern p (not eliminated in step 3), obtain a different pattern-robust Pareto front using the MOPSO-CD adding the following restriction:
 - The solution $\bar{x}\bar{p}$ must perform a pattern pt using the mean value of the train mass (if not, $\bar{x}\bar{p}$ is discarded).
 - The solution $\bar{x}\bar{p}$ must be pattern-robust for the worst-case train mass variations, that is, it must conserve the pattern pt simulating with the maximum and the minimum values of train mass in the operation period.
- Step 5: The global robust Pareto front is made by the non-dominated solutions combining all the previous Pareto fronts of each pt in step 4.

With this alternative method, the designer can filter or give priority to certain driving patterns over other patterns based on comfort criteria defined by the operator (typically the maximum number of coasting-remotoring cycles, and the minimum duration of the last traction before braking) (Domínguez et al., 2014). In the case study explained in Section IV the solutions obtained using this method and using the method explained in Section 3.2.1 will be compared.

3.3. OPTIMAL SELECTION OF THE ATO SPEED PROFILE SET

Once the robust Pareto front has been obtained, the objective of the optimisation method described in this section is to select a set of ATO speed profiles to be programmed in the ATO equipment. These speed profiles are optimised for a specific operation period (i.e. peak-hour or non-peak-hour) to be handled by the traffic regulation system during that period.

The candidates to be programmed are the speed profiles from the robust Pareto front obtained applying the method described in the previous section. Therefore, the candidates are the most efficient speed profiles for each possible running time.

The problem is stated as a stochastic optimisation model. The statistical distribution of running times demanded by the traffic regulation system between two stations in the considered operation period is an input data for the proposed model. Figure 3-2 shows an example of a running time distribution, where p_s is the statistical probability of demanding a running time. This figure shows a typical case where the highest probability is concentrated around the times close to the nominal running time (125 seconds). The scenarios where the regulation system demands higher running times than the nominal one occur when the train is slowed down by the regulation system to reduce its time interval with the following train (Fernandez et al., 2006). On the other hand, scenarios that require lower running times than the nominal one occur when the train is delayed.

The statistical distribution of running times is discretised into NS scenarios with regular time intervals assigning a probability p_s to each scenario of running time RT_s as shown in Figure 3-2.

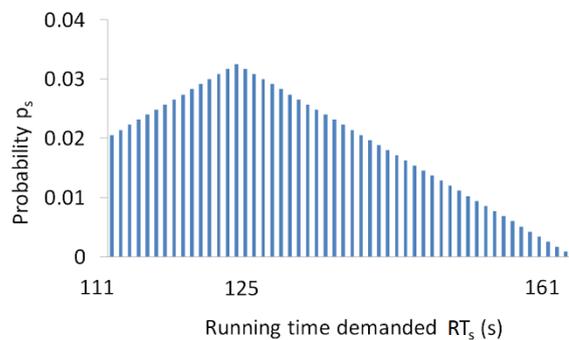


Figure 3-2. Example of the discrete probability distribution of the running time demanded by the regulator

The objective is to select a set of D number of speed profiles from the Pareto front (3.8).

$$st^d \in J \quad d = (1, 2..D) \quad (3.8)$$

where st^d is the d speed profile of the pre-programmed set \vec{st} with an associated running time RT_d and energy consumption EC_d . J is the Pareto front.

The energy consumption associated with the scenario s is the energy consumption of the speed profile selected from the pre-programmed set by the traffic regulation system. In order to obtain the energy consumption of each scenario it is necessary to apply the selection logic of the regulation system (3.9).

$$EC_s = \{min(EC_d) \mid RT_d \leq RT_s\} \quad (3.9)$$

The regulation system on-line selects the speed profile that has the lowest energy consumption from those that perform a running time lower than or equal to the demanded running time (3.9). The running time demanded by the traffic regulation system is constrained by the shortest running time associated with the interstation. Furthermore, the flat out speed profile will always be selected as an element of the pre-programmed set to ensure that there is a solution that fulfils (3.9) when the regulator demands the shortest time (see Figure 12 in the case study).

The aim of the problem is the selection of a set of speed profiles with the minimum expected energy consumption. Therefore, the objective function is expressed in (3.10):

$$min \sum_s^{NS} (p_s \cdot EC_s) \quad (3.10)$$

Particle Swarm Optimisation algorithm (PSO) (Kennedy and Eberhart, 1995, 2001, 1997) is used to solve this problem because its ability to work with discrete exploration spaces, its simplicity in concept and coding implementation and its less sensitivity to the nature of the objective function. The algorithm makes use of a swarm of NP particles whose position (\vec{st}_j) and velocity (\vec{vp}_j) is randomly initialised. The position \vec{st}_i of the particle j is defined by the speed profiles of the Pareto front J selected to form the pre-programmed set \vec{st}_j . The information of the Pareto front is introduced in the algorithm assigning to each speed profile, besides their running time and energy consumption, an integer number i which represents their position in the Pareto front respect to the other speed profiles when the Pareto set is time sorted. Therefore, the pre-programmed set \vec{st}_j is a vector of D dimensions where each dimension st_j^d has a value i assigned which represents the Pareto position of a speed profile selected in the pre-programmed set \vec{st}_j . The dimensions of each \vec{st}_j vector are sorted as a function of the position of the Pareto position i in order to increase the computational efficiency of the algorithm. The movement of the particles varies depending on their velocity, which is updated at each iteration it . The value of the velocity depends on the best position found by the particle (\vec{p}_i) and the best position found by the whole swarm (\vec{p}_g) following (3.11) and (3.12).

$$\vec{vp}_j(it) = w \cdot \vec{vp}_j(it - 1) + c_1 \cdot r_1 (\vec{p}_j - \vec{st}_j(it - 1)) + c_2 \cdot r_2 (\vec{p}_g - \vec{st}_j(it - 1)) \quad (3.11)$$

$$\vec{st}_j(it) = \vec{st}_j(it - 1) + \vec{vp}_j(it) \quad (3.12)$$

where c_1 and c_2 are the “social factors” constants, r_1 and r_2 are random numbers between 0 and 1 and w is an inertia weight that controls the effect of the previous history of velocities.

The $pbest$ (\vec{p}_i) and $gbest$ (\vec{p}_g) are obtained assigning at each solution a fitness value. The fitness value is the expected value of the energy consumption of the pre-programmed set. Thus, a lower value of fitness implies a better solution.

PSO algorithm is a single-objective algorithm. Therefore, it does not require an external archive A because the final solution is the $gbest$ in the last iteration (Figure 3-3 and Figure 3-4).

```

For i = 1:NP                               NP particles (population size)
    { $\widehat{st}_i, \widehat{v}_i$ }                           Initialize positions & velocities
    Sort( $\widehat{st}_i$ )                               Sort the dimensions of the vector according to position
                                                in Pareto front
     $y_i = f(\widehat{st}_i)$                            Evaluate objective functions
     $\widehat{p}_i = \widehat{st}_i$                              Initialize  $pbest$ 
End For

 $\widehat{p}_g = \min(f(\widehat{p}_i))$                          Initialize  $gbest$ 
For n = 1:N                                 N iterations
    For i = 1:NP                             NP particles
        For d = 1:D                           D dimensions of vectors  $\widehat{st}_i$  and  $\widehat{v}_i$ 
             $v_i^d = wv_i^d + c_1r_1(p_i^d - st_i^d) + c_2r_2(p_g^d - st_i^d)$    Update velocity
             $st_i^d = st_i^d + v_i^d$            Update position
        End For

        Sort( $\widehat{st}_i$ )                           Sort the dimensions of the vector according to
                                                position in Pareto front
         $y_i = f(\widehat{st}_i)$                        Evaluate objective function

        If  $y_i < f(\widehat{p}_i)$  Then  $\widehat{p}_i = \widehat{st}_i$    Update  $pbest$ 
    End For

     $\widehat{p}_g = \min(\widehat{p}_i)$                          Update  $gbest$ 
End For

```

Figure 3-3. Pseudocode of PSO algorithm to obtain the energy efficient pre-programmed set

```

//f( $\widehat{st}_i$ ) function//                         consumption of  $\widehat{st}_i$ 
totalCons=0                                  Initialization
For s = 1:S                                  S scenarios
    For d = 1:D                                 D dimensions
        If  $RT_d \leq RT_s$  Then                Search the closer speed profile with lower time
             $EC_s = EC_{i,d}$                        than the time required in scenario s
        Else
            Exit For
        End If
    End For

    totalCons = totalCons +  $p_s \cdot C_s$        Total consumption is a weighted sum of each scenario
                                                consumption
End For

f( $\widehat{st}_i$ ) = totalCons

```

Figure 3-4. Pseudocode of the algorithm used to calculate the expected value of the energy consumption of a solution

3.4. CASE STUDY

The methods previously described have been applied to a case study considering a typical journey between stations in Metro de Madrid of 1500 meters, with different speed limits along the journey (see Figure 3-6). A class-3000 train of Metro de Madrid has been modelled. The empty mass of the train is 160 tons, the maximum passenger's load is 78 tons and the length of the train is 90 meters. The maximum power of the train is 1500 kW and the traction network voltage is 1500 V DC.

The speed profile design is carried out for an operation period characterised by an average passenger load of 50% of the maximum. Furthermore in the 90% of the situations the passenger load is between 30% and 70% of the maximum. These limits will be used to check the robustness conditions of the speed profiles to mass variations.

A detailed train simulator (average accuracy of 4.2% in traction energy and 1.0% in running times) and a simulator of real ATO equipment both described in (Domínguez et al., 2011b) are used to calculate the running time and the energy consumption of speed profiles.

3.4.1. GENERATION OF THE ROBUST PARETO FRONT

The robust MOPSO-CD algorithm (Type-II) has been applied to the case study and the results obtained with the MOPSO-CD algorithm (Domínguez et al., 2014) and the proposed robust MOPSO-CD are shown in Figure 3-5. The robustness coefficient η is set to 0.07 because this value has demonstrated an acceptable performance in the context of the problem.

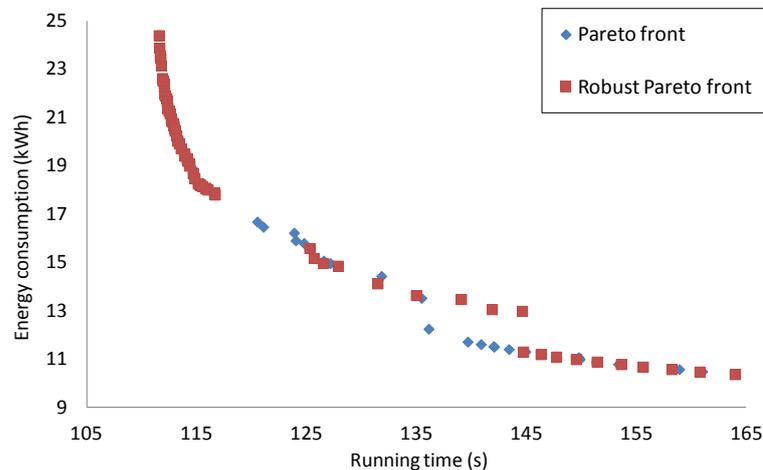


Figure 3-5. Comparison of the current Pareto front and robust Pareto front using the robust MOPSO-CD algorithm

Comparing both Pareto fronts, the robust one presents a wider time-gap around the running time 120 s, and higher energy consumption from 135 to 145 s. The steps in energy consumption at 135 s and at 145 s correspond to a change in the driving pattern.

Non-robust solutions, that suffer great variations in time and energy consumption with train load, were analysed. It was observed that most of these solutions change the

shape of their driving pattern for extreme values of train load with respect to the average value. For instance, Figure 3-6 shows two different speed profiles considering the same ATO parameters ($v_c = 58 \text{ km/h}$, $v_r = 21 \text{ km/h}$, $b = 0.65 \text{ m/s}^2$) but different passenger load (50% and 100%). The increase in the train mass causes a lower starting acceleration. For this reason, the train does not reach the coasting speed, and instead the control system activates the braking mode to fulfil the speed limitation of 60 km/h. This fact changes the shape of the speed profile and causes large differences in energy consumption and running time as shown in Table 3-1.

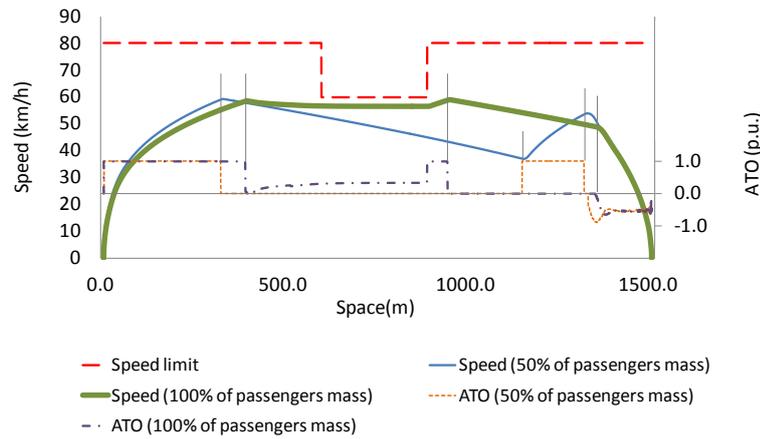


Figure 3-6. Simulation of ATO configuration: ($v_c = 58 \text{ km/h}$, $v_r = 21 \text{ km/h}$, $b = 0.65 \text{ m/s}^2$) with 50% and 100% passenger load

Passengers' load	Time (seconds)	Energy consumption (kWh)	Pattern
50%	130.30	17.11	T-C-T-FB
100%	121.25	20.73	T-B-T-C-FB

Table 3-1. Performance of ATO configuration: ($v_c = 58 \text{ km/h}$, $v_r = 21 \text{ km/h}$, $b = 0.65 \text{ m/s}^2$)

This effect has been studied during the execution of the robust MOPSO-CD algorithm. With this purpose, pattern variations has been checked, and the number of pattern changes are counted separately for robust a non robust solutions. In addition, different values of the η parameter have been considered in order to compare the results for higher and lower sensitivity (see Table 3-2).

η	Rejected solutions	Rejected non pattern- robust solutions	Accepted non pattern- robust solutions
0.15	78	78	585
0.1	108	108	559
0.07	132	132	482
0.04	1206	568	374
0.02	1275	322	115
0.01	3948	3402	0

Table 3-2. Performance of the robustness restriction in the robust MOPSO algorithm for different η values

For η higher than or equal to 0.07, all the solutions rejected by the robust restriction change their pattern, so they are also non pattern robust to train load changes. On the

other hand, there are many solutions that fulfil the robust restriction but change their pattern.

For η values lower than 0.07 the situation changes: the robust algorithm rejects many solutions that are pattern robust, and a considerable amount of non pattern-robust solutions continue being accepted by the robust restrictions.

In conclusion, for high enough η , solutions that preserve the driving pattern against load changes are also robust in energy and time. This robustness criterion based on patterns is more restrictive and can be useful for designers, because it guarantees a qualitative level of passenger's comfort associated with the pattern (see next section).

3.4.2. GENERATION OF THE ROBUST PARETO FRONT USING THE ALTERNATIVE METHOD BASED ON DRIVING PATTERN

The first step to obtain the robust Pareto front based on maintaining the driving patterns is the identification of patterns present in the original Pareto front. For this purpose, the MOPSO algorithm without any robustness restriction and average load is executed to obtain the optimal front as shown in Figure 3-7.

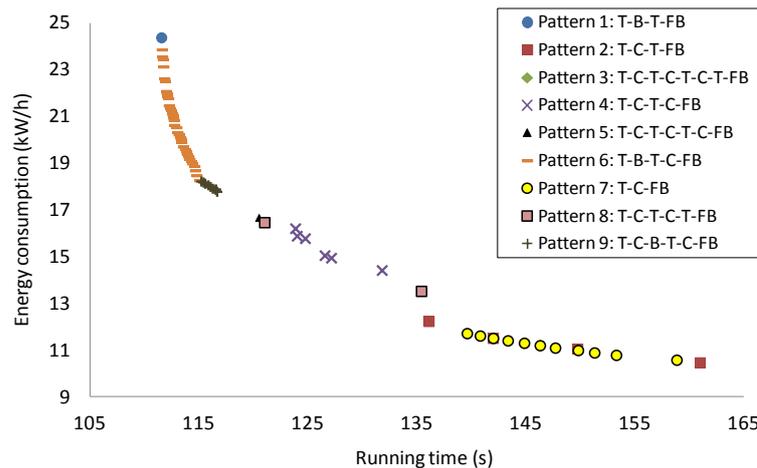


Figure 3-7. Optimal Pareto front of the possible speed profiles and driving patterns obtained

In this case study, there are 9 different driving patterns. The shape of some driving patterns is shown in Figure 3-8. Patterns 3 and 8 are rejected because they present a short traction period before starting the final braking. This situation can be perceived by the passengers as uncomfortable.

Robust speed profiles for the automatic traffic regulation system in metropolitan lines

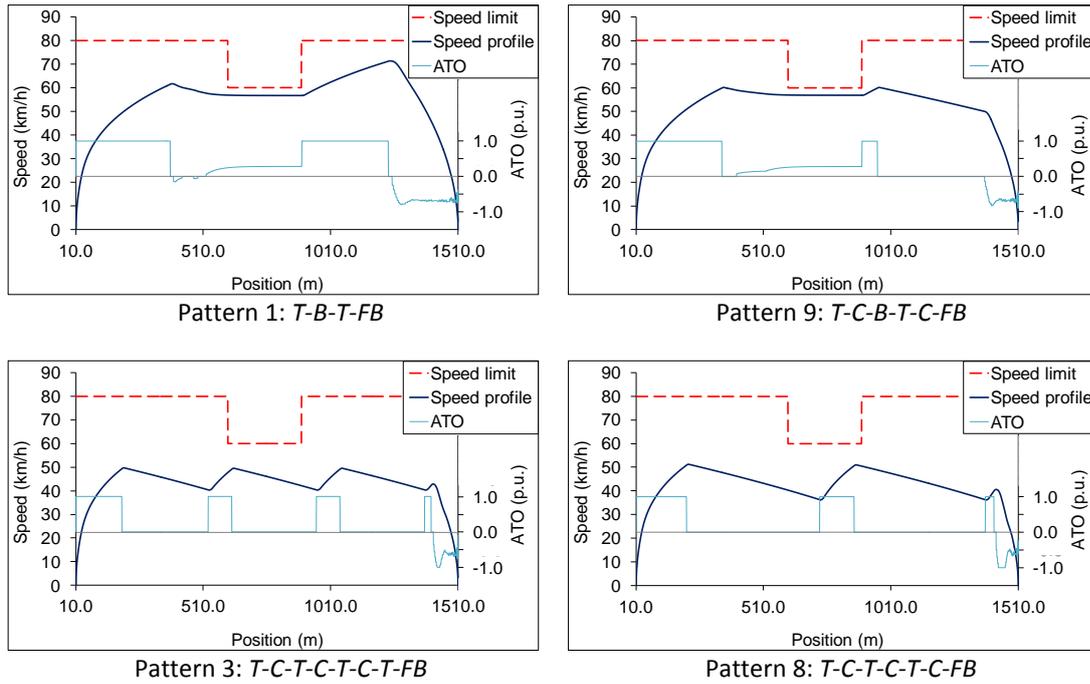


Figure 3-8. Example of the speed profile of different patterns

For each non-filtered driving pattern, a Pareto front of the solutions that performs each pattern is obtained. A pattern restriction is added to the MOPSO algorithm to select solutions of the specific pattern. Then, the pattern-robustness of each solution is checked. All the solutions are simulated using the maximum value of the train load (70%), and using the minimum one (30%). Solutions that change the pattern (not pattern-robust) are eliminated and the result is a set of robust Pareto fronts for each driving pattern (Figure 3-9).

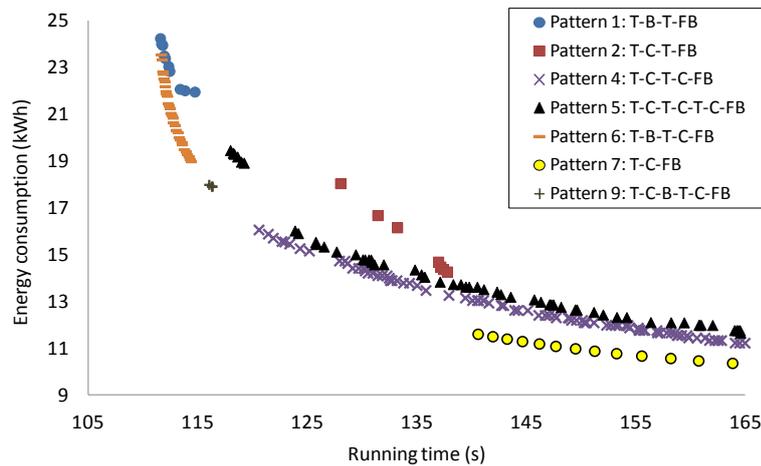


Figure 3-9. Pattern-robust Pareto fronts for each driving pattern

Dominated solutions are eliminated to obtain a single robust Pareto front (Figure 3-10).

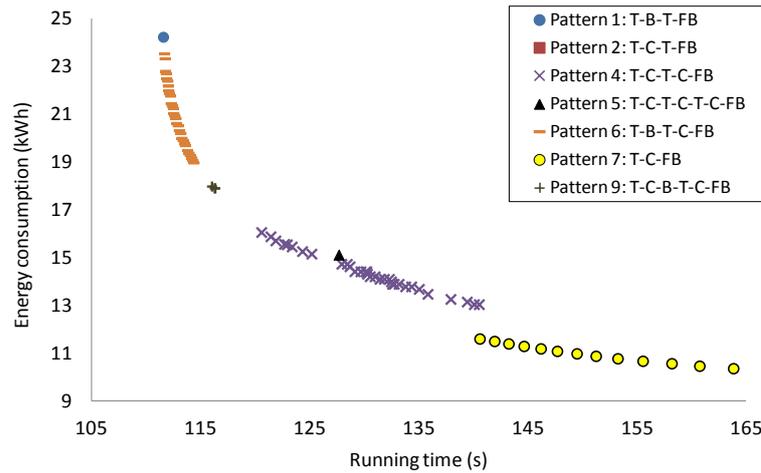


Figure 3-10. Robust Pareto front obtained using the method based on driving pattern robustness

It has been necessary to simulate an average of 5000 journeys to obtain each robust Pareto front.

In Figure 3-11, the comparison of this solution with the previously obtained by using the robust MOPSO-CD algorithm for $\eta = 0.07$, shows that the number of solutions is higher, and the running-time gaps have been reduced. The designer can select a speed profile taking into account not only its energy consumption but also the preferred driving patterns, considering the passenger comfort or other operation criteria.

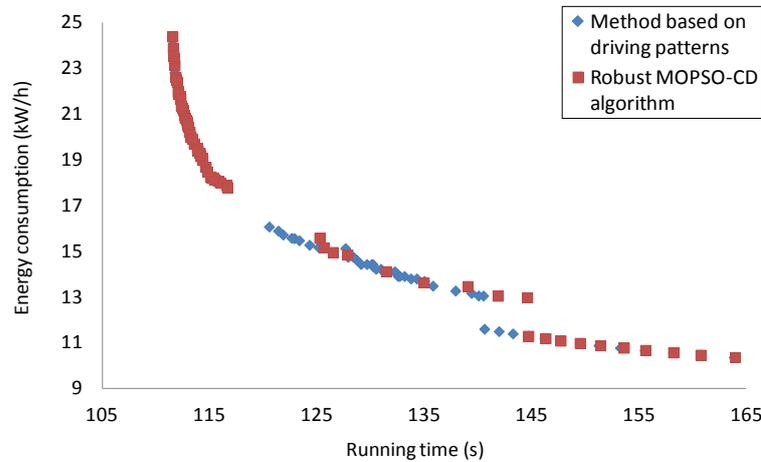


Figure 3-11. Comparison between robust MOPSO-CD algorithm and the pattern robust algorithm

3.4.3. OPTIMAL SELECTION OF THE PRE-PROGRAMMED SET OF ATO SPEED PROFILES

Traffic regulation systems typically handle a set of 4 pre-programmed alternative speed profiles between stations (Fernandez et al., 2006). The first speed profile (number 0) performs the lowest running time (flat-out), and it is used to recover delays. The second speed profile (number 1) is usually chosen to provide the nominal running time. The third (number 2) and the fourth (number 3) speed profiles have higher associated running times and lower energy consumption, and are used to reduce the time interval with the following train. Typically, the speed profiles have been selected equidistant, that is uniformly distributed with the same time separation (Domínguez et al., 2011b).

The optimisation method explained in Section 3.3 has been applied to our case study to select the 4 speed profiles. The inputs of the proposed PSO algorithm are the robust Pareto front obtained in Section 3.4.2, the size of the pre-programmed set (4 speed profiles in this case study) and the discrete statistical distribution of running times demanded by the traffic regulation system at the control centre. Table 3-3 shows the tuned parameters of the PSO algorithm.

c_1	c_2	w	NP	Iterations
1	1	0.6	300	400

Table 3-3. Tuned parameters of the PSO algorithm

Three different shapes of probability distributions of demanded running times are considered in this study: decreasing, increasing and uniform distributions. The first one represents stations where delays are frequent, or the nominal running time is close to the minimum one. The second distribution represents stations where trains are regulated frequently to reduce their time interval with the following train, or the nominal running time is close to the maximum one. The third distribution corresponds to an intermediate situation where running times are demanded with the same probability.

Considering the previous distributions of demanded running times, the four optimal speed profiles are obtained by the proposed PSO algorithm (see Figure 3-12).

Figure 3-12 shows that the selected optimal solutions are not uniformly distributed in time (as the typical design criterion does), when the objective function is the minimisation of the energy consumption along an operation period.

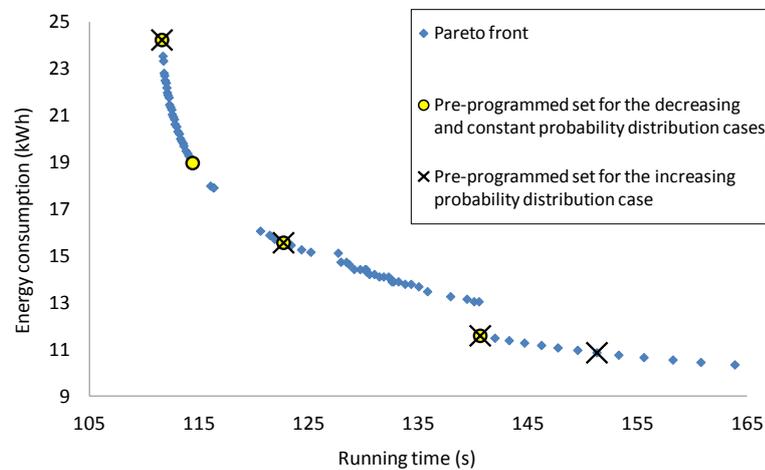


Figure 3-12. Optimal speed profiles selected by the PSO algorithm

The PSO algorithm selects the speed profiles where the energy benefit is higher taking into account the frequency of use of the speed profiles. Figure 3-12 shows that the optimal selection is the same for the decreasing and the uniform distributions, and the optimal selection is different in the increasing probability distribution case.

Table 3-4 shows the numerical results in terms of energy consumption and savings.

	Decreasing probability (kWh)	Rising probability (kWh)	Constant probability (kWh)
Optimal	16.64	12.90	14.78
Equidistant	17.25	15.02	16.11
Savings	3.5%	14.1%	8.3%

Table 3-4. Energy consumption and savings obtained comparing the optimisation results with the equidistantly pre-programmed set

The speed profiles selected by the uniform distribution of running times are not the most efficient. The optimal selection of speed profiles generated by the PSO algorithm provides energy savings between 3.5% and 14% compared with the typical design criterion. These are additional savings to those obtained from the optimal design of the ATO speed profiles (that are around 20%) described in (Domínguez et al., 2011b).

Optimal solutions provide average advance time of 4,8 s (decreasing and constant probability distribution case) and 6,5 s (increasing probability distribution case). These advance times are compensated increasing the dwell time, and thus, there is no schedule advance at departures.

In the previous optimisation model, the number of speed profiles is a fixed parameter. However, the new signalling system based on radio communication, CBTC, allows the programming of more speed profiles in the equipment. In order to exploit this advantage, the energy efficiency that can be obtained increasing the number of speed profiles to be pre-programmed has been analysed. In addition, when the number of speed profiles is increased, the schedule advance at arrivals is reduced.

The PSO algorithm has been executed with different sizes (from 2 to 20 speed profiles) of the pre-programmed set, taking into account the 3 previous probability distributions. The curves plotted in Figure 3-13 represent the energy savings obtained for each size compared with size = 1.

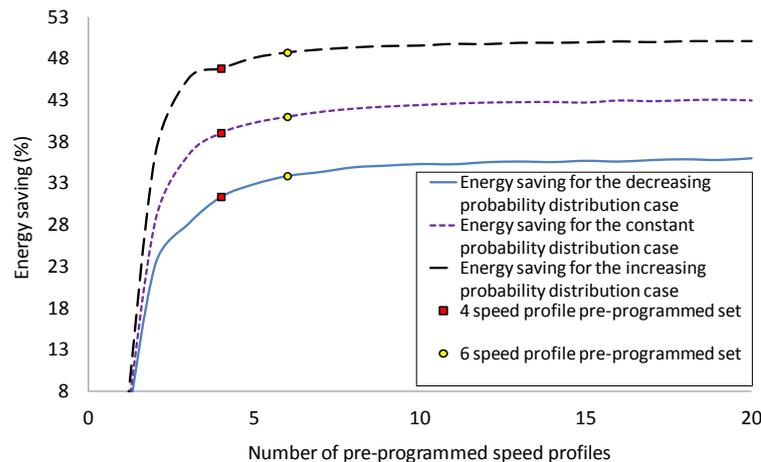


Figure 3-13. Expected value of energy savings obtained for different pre-programmed set size compared with the pre-programmed of 1 speed profile

In view of the results, it can be concluded that the lower importance have the delays in the line the larger the achievable energy saving. The usual 4 speed profile pre-programmed set is in general an acceptable solution. Nevertheless, significant energy

savings can be reached with 1 or 2 extra-speed profiles. Table 3-5 compares the optimal sets of 4 and 6 speed profiles.

	Decreasing prob. (kWh)	Increasing prob. (kWh)	Constant prob. (kWh)
6 speed profiles	16.04	12.43	14.31
4 speed profiles	16.64	12.90	14.78
Savings	3.64%	3.60%	3.19%

Table 3-5. Comparison of the optimal pre-programmed set of 4 and 6 speed profiles

3.5. CONCLUSIONS AND CONTRIBUTIONS

A new procedure to design energy efficient speed profiles to be programmed in the signalling equipment of a metropolitan system has been proposed. The procedure takes into account the main uncertainties in the traffic operation: train load and delays in the line.

The proposed model is based on the calculation of the Pareto curve of the possible speed profiles that are robust against passenger load variations. Then, the set of speed profiles to be programmed in the signalling system is taken from the robust Pareto front by means of a PSO optimisation algorithm, considering energy efficiency and delay distribution in the line.

Two algorithms for obtaining the robust Pareto front have been proposed and compared using a case study. The first model is a robust multi-objective optimisation algorithm that makes use of a robust definition as a restriction. The second one is an alternative method based on the robustness of the solution to changes in their driving pattern. It has been shown that pattern-robustness requirement is more restrictive than definition of robustness type-II. Moreover, the pattern-robustness requirement is more useful because it guarantees the comfort of the speed profile. Besides that, the alternative procedure has found more solutions than the standard robust optimisation algorithm. For this reason, the pattern recognition gives to the designer more possible solutions to choose and, some of them, have lower energy consumption for a given running time.

The proposed selection model including train delays information has been compared with the traditional selection method that distributes speed profiles uniformly in time. The results show important energy savings, around 3 - 14%. This model has also been used to study the energy benefits obtained from increasing the number of speed profiles in the pre-programmed set. The typical size of this set is 4 and the energy consumption can be reduced 3.5% by the inclusion of two extra speed profiles.

The main contributions of this chapter are:

- The application of the robust optimisation method proposed in (Deb and Gupta, 2005) combined with MOPSO algorithm to be applied in the ATO eco-driving problem.
- A process to design robust and efficient ATO speed profiles based on the proposed pattern of the driving model.

- The comparison between the method proposed in (Deb and Gupta, 2005) and the pattern-based method proposed in this thesis.
- A model to select the optimal speed profiles from a Pareto front to be programmed in the ATO equipment based on the delay distribution in the line.
- The application of PSO algorithm to obtain the optimal pre-programmed set of speed profiles of the traffic regulation system.
- The assessment of the energy saving that can be obtained selecting optimally the pre-programmed set, using delay information and the energy savings that can be obtained increasing the number of speed profiles in the pre-programmed set.

REAL TIME ECO-DRIVING OF HIGH-SPEED TRAINS

4.1. INTRODUCTION

Previous developments of the thesis were focused on urban railways. The next chapters will be dedicated to long-distance lines, particularly, to high-speed railways. High-speed railways (HSR) are expanding throughout the world becoming an important energy consumer. It is considered an energy efficient transport mode (Givoni, 2007), however, it can be improved and many studies are being carried out to reduce the energy consumption of HSR (Hasegawa et al., 2016). These works have the objective of reducing both the economic costs for railway operators and the environmental impact of railways (Feng et al., 2014, 2013b). As the rest of the thesis, the work developed is focused on eco-driving framework.

There are important differences when comparing urban and high-speed railways that affect the eco-driving application. Metropolitan railways are highly automated systems where trains are typically driven by automatic train operation (ATO) equipment. The driving strategies applied in ATO equipped trains are basically: speed regulation (Feng et al., 2012; Liu and Golovitcher, 2003) and coasting-remotoring defined by coasting points (Coleman et al., 2010) or by upper and lower speed limits (Bocharnikov et al., 2007; Domínguez et al., 2014, 2011b; Howlett, 1996). Typically, high-speed trains are driven manually (Yang et al., 2015) and the journeys between stations are long-distance travels. The driving strategies applied in HSR are speed regulation (Ji et al., 2016) and its efficient version, holding speed without braking (Hwang, 1998; Sicre et al., 2014,

2012). It consists in maintaining a constant speed as long as traction effort is needed. If braking is needed to maintain the speed command, the train will coast increasing its velocity in order to save energy. This command has been tested in Spanish high-speed lines and proved to be energy efficient and easily executed by train drivers (Sicre et al., 2012).

Most of the eco-driving work in the literature is related to the offline planning of railway's efficient driving using analytical methods (A. R. Albrecht et al., 2013; T. Albrecht et al., 2013; Gu et al., 2014; Howlett et al., 2009; Khmel'nitsky, 2000; Liu and Golovitcher, 2003; Lu et al., 2013; Miyatake and Ko, 2010; Su et al., 2013; Wang et al., 2014; J. Yang et al., 2016) or Nature Inspired Computational Intelligence techniques (Bocharnikov et al., 2010; Chang and Sim, 1997; Cucala et al., 2012b; Huang et al., 2015; Ke et al., 2012; Keskin and Karamancioglu, 2017; Kim et al., 2013; Lechelle and Mouneimne, 2010; Li and Lo, 2014; Lu et al., 2013; Sicre et al., 2012; Wei et al., 2009; Wong and Ho, 2004b, 2003; Yang et al., 2012).

However, optimal speed profiles can also be obtained in the regulation stage as an on-line calculation. If the train is delayed, the offline eco-driving design is not valid anymore and a new on-line eco-driving calculus is required to recover the delay in an energy efficient way. The challenge of the on-line eco-driving application is the low computation time available to carry out the optimisation and the changing situations that may occur along the trip.

Some solutions have been proposed in the literature to solve the on-line eco-driving design. Several works make use of analytical models (Coleman et al., 2010; Howlett et al., 1994; Khmel'nitsky, 2000; Liu and Golovitcher, 2003). A pseudospectral method was applied by Wang et. al. to transform the optimal train control into a Mixed Integer Linear Programming problem (MILP) (Wang et al., 2014, 2013). Pseudospectral method approximates the differential equations of the problem by orthogonal polynomials. Using this method, Wang et. al. obtained solutions with very low computational time. However, the train model lies in simplifications, it does not take into account comfort restrictions and, besides, pseudospectral method introduces additional inaccuracies because of the approximations made.

Other works combine GA with simulation (Chang and Sim, 1997; Wong and Ho, 2004b) but solutions provided are not suitable for high-speed lines because they demand many coasting-remotoring cycles. In (Sicre et al., 2014), Sicre et al. present a model to calculate in real time a new set of holding speed without braking commands when a delay occurs in a high-speed line. This model is stated as a single-objective optimisation problem and makes use of a GA with fuzzy parameters.

Previous models can be improved with the application of Population-based algorithms for dynamic multi-objective optimisation problems (DMOOPs) (Helbig and Engelbrecht, 2014). The use of a dynamic optimisation algorithm would reduce the calculation time required for the optimisation process, allowing frequent updates of the Pareto front.

The main contribution of this chapter of the thesis is the on-line calculation of the eco-driving of a high-speed train by means of a dynamic multi-objective optimisation

algorithm. The proposed algorithm can be executed in real-time and improves the energy savings provided by static simulation-based eco-driving algorithms, fulfilling the punctuality requirements and taking into account the passenger's comfort. The DMOOP regulation algorithm provides a set of updated and non-dominated solutions that can be used to change the current speed profile as soon as a delay arises or the traffic situation changes. These solutions make use of the holding speed without braking strategy that has been found as the most appropriate manual driving strategy to HSR. Two algorithms are proposed and compared to calculate eco-driving: Dynamic Non-dominated Sorting Genetic Algorithm II (DNSGA-II) (Deb et al., 2007) and Dynamic Multi-Objective Particle Swarm Optimisation algorithm (DMOPSO) (Salazar Lechuga, 2009). Dynamic algorithms are faster tracking the Pareto front changes than their static versions, providing better energy savings. In addition, the proposed algorithms take into account the passengers' comfort constraints including a lower bound for the speed commands, maximum acceleration and deceleration rates, maximum jerk and a limited number of different speed commands in the journey

The chapter is organised as follows: Section 4.2 describes the simulation model of a high-speed train, the manual driving model and introduces the delay response mechanisms used. The dynamic optimisation model of the high-speed train regulation problem and the algorithms proposed to solve it are presented in Section 4.3. In Section 4.4, the results of the application of the dynamic algorithms to a case study are analysed. Finally, Section 4.5 details the main conclusions obtained in this piece of research.

4.2. TRAIN SIMULATION MODEL

The aim of the time-step simulation model is the accurate calculation of the running time and the energy consumption of a high-speed train journey (Goodman et al., 1998). The simulation model is divided into three modules: train, line and manual driving (Sicre et al., 2012).

The train module takes into account the characteristics of the specific high-speed train used in the simulation. These characteristics are length, mass, running resistance and rotary inertia. The features of the motor are also included in the train module by means of the maximum traction and braking effort curves as a function of the train speed and the efficiency as a function of the effort ratio and train speed.

The line module includes the physical information of the track: grades, grade transition curves (and the effect along the train), bends, bend transition curves and tunnels. In addition, the line module includes the operational characteristics of the track such as permanent and temporary speed limits. The electrical network is also taken into account including the location of neutral sections and electrical substations in order to compute the energy losses in transmission lines.

The manual driving module determines the traction and braking force (F_m) that is demanded to the motors as a function of the driving commands, stopping point at stations and speed limits. The effort provided by the motor is represented by means of

a single variable that represents propulsion (positive values), coasting (zero value) and braking (negative values).

The simulator was validated against real measurements registered on-board high-speed trains on commercial services and nocturnal tests on the Madrid–Barcelona high-speed line. It was shown that the simulated running times differ on average 1.2% from measured times, and the difference between energy consumptions is 0.4%, which provides strong evidence for the validity of the simulator. This accurate simulator was used for the design of eco-drivings and tested in collaboration with Renfe and Adif, and important energy savings were measured (Sicre et al., 2012).

4.2.1. TRAIN DYNAMICS SIMULATION MODEL

The simulation model determines at each time step the position $s(t)$, the speed $v(t)$ and the acceleration $a(t)$ of the train using information from the different modules and the equations of motion (4.1), (4.2) and (4.3).

$$\frac{ds}{dt} = v(t) \quad (4.1)$$

$$\frac{dv}{dt} = a(t) \quad (4.2)$$

$$a(t) = \frac{F_m(t) + F_b(t) - (F_r(v) + F_g(s))}{\rho \cdot m} \quad (4.3)$$

where m is the train mass, ρ is the dimensionless rotating mass factor, $F_r(v)$ is the running resistance and $F_g(s)$ includes the gravitational force caused by grades and the resistance caused by bends (modelled as equivalent grades). The force $F_m(t)$ is the motors electrical tractive/braking effort and force $F_b(t)$ is the effort of pneumatic brakes. Both forces are determined at each simulation step by the manual driving module as a function of the position and speed of the train, the driving commands and the speed limits.

The initial and end conditions for $s(t)$ and $v(t)$ are expressed in (4.4) and (4.5).

$$s(0) = s_0, \quad v(0) = v_0 \quad (4.4)$$

$$s(RT) = s_{end}, \quad v(RT) = 0 \quad (4.5)$$

where s_0 and s_{end} are the train initial position and the position of the arrival station. The parameter v_0 represents the initial speed of the train which is equal to zero when s_0 is the position of the departure station. RT is the running time of the trip between s_0 and s_{end} .

The value of $F_m(t)$ is bounded by a maximum electrical traction effort curve and a maximum electrical braking effort curve which are dependent on the train speed as shown in Eq. (4.6) and Figure 4-1. Pneumatic brakes force $F_b(t)$ is set to 0 when the motors are tractioning or coasting (Eq. (4.7) and (4.8)).

When a train is passing through a neutral zone, auxiliary systems cannot be fed from catenary, thus motors apply at least a constant braking effort F_{NZ} to maintain the charge of the batteries that feed auxiliary systems along neutral zones (see Eq. (4.9)).

$$F_{min}(v) \leq F_m(t) \leq F_{max}(v) \quad (4.6)$$

$$F_b(t) \leq 0 \quad (4.7)$$

$$F_b(t) = 0 \quad \text{if } F_m(t) \geq 0 \quad (4.8)$$

$$F_m(t) \leq F_{NZ} \quad \text{if } snz_{start}^l \leq s(t) \leq snz_{end}^l \quad \text{for } l = 1, 2, \dots, Z \quad (4.9)$$

where $F_{max}(v)$ and $F_{min}(v)$ are respectively the value of maximum electrical traction effort and maximum electrical braking effort that depend on the speed of the train (v). The total braking effort is the electrical braking effort plus the pneumatic braking effort (blending). The pneumatic braking effort complements the electrical one if it is necessary to achieve the deceleration command.

The parameters snz_{start}^l and snz_{end}^l are the initial and the final position of the neutral zone l being Z the total number of neutral zones.

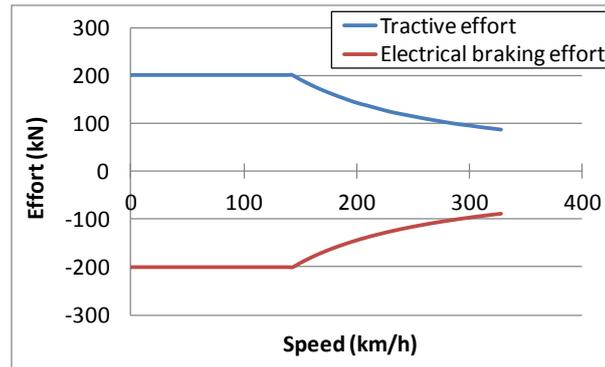


Figure 4-1. Maximum Electrical Traction/Braking curves of the train motor

Running resistance ($F_r(v)$) is determined by means of Davis formula as shown in Eq. (4.10) whereas the gravitational resistance ($F_g(s)$) is calculated using Eq. (4.11).

$$F_r(v) = A + B \cdot v + C \cdot v^2 \quad (4.10)$$

$$F_g(s) = g \cdot m \cdot p(s) \quad (4.11)$$

where A , B and C are positive coefficients, g is the gravity acceleration and $p(s)$ is the average equivalent grade. The value of the average equivalent grade is obtained as the average grade affecting the complete length of the train plus the equivalent grade associated to the resistance produced by bends.

4.2.2. TRAIN ENERGY CONSUMPTION SIMULATION MODEL

The simulation model also calculates the energy consumed by the train at each time step. The electrical power consumed by the train measured at the pantograph and the

estimation of the power consumed at the substation is obtained from the following expressions:

$$P_{mec}(t) = F_m(t) \cdot v(t) \quad (4.12)$$

$$\begin{aligned} P_{pantograph}(t) &= \frac{P_{mec}(t)}{\mu_T(v, fr)} + P_{aux} && \text{if } P_{mec} \geq 0 \text{ and not}(snz_{start}^l \leq s(t) \leq snz_{end}^l) \\ & && \text{for } l = 1, 2, \dots, Z \\ P_{pantograph}(t) &= P_{mec}(t) \cdot \mu_B(v, fr) + P_{aux} && \text{if } P_{mec} < 0 \text{ and not}(snz_{start}^l \leq s(t) \leq snz_{end}^l) \\ & && \text{for } l = 1, 2, \dots, Z \\ P_{pantograph}(t) &= 0 && \text{if } snz_{start}^l \leq s(t) \leq snz_{end}^l \\ & && \text{for } l = 1, 2, \dots, Z \end{aligned} \quad (4.13)$$

$$P_{ss}(t) = P_{pantograph}(t) + r(s) \cdot \left(\frac{P_{pantograph}(t)}{U \cdot \cos \varphi} \right)^2 \quad \text{if not}(snz_{start}^l \leq s(t) \leq snz_{end}^l) \quad \text{for } l = 1, 2, \dots, Z \quad (4.14)$$

$$P_{ss}(t) = 0 \quad \text{if } snz_{start}^l \leq s(t) \leq snz_{end}^l \quad \text{for } l = 1, 2, \dots, Z$$

where $P_{mec}(t)$ is the mechanical power, $P_{pantograph}(t)$ is the electrical power measured at the pantograph and $P_{ss}(t)$ is the electrical power measured at the substation. The power consumed by auxiliary systems is modelled by a constant value P_{aux} . The electrical chain efficiency is modelled for the traction case $\mu_T(v, fr)$ and the braking case $\mu_B(v, fr)$. Furthermore, the efficiency is modelled as function of the train speed (v) and the ratio fr of the motor force divided by the maximum motor force ($fr = F_m/F_{max}$). The line voltage U is modelled as a constant (the nominal line voltage) and the power factor ($\cos \varphi$) is also modelled as a constant. The electrical line resistance $r(s)$ is dependent on the position of the train, as it increases linearly with the distance to the substation.

The energy consumed by the train measured at the pantograph and the train energy consumption measured at the substation are calculated by means of equations (4.15) and (4.16).

$$EC_{pantograph}(t) = \int_0^t P_{pantograph}(t) \cdot dt \quad (4.15)$$

$$EC_{ss}(t) = \int_0^t P_{ss}(t) \cdot dt \quad (4.16)$$

where $EC_{pantograph}(t)$ and $EC_{ss}(t)$ are respectively the energy consumed measured at the pantograph and the substation.

4.2.3. MANUAL DRIVING SIMULATION MODEL

4.2.3.1. COMMAND MATRIX MODEL

The speed profiles to be executed by the driver must be defined by commands that can be easily applied. The speed profiles obtained as a solution in the optimisation process

are defined using the Command Matrix (C_m) proposed in (Sicre et al., 2012). C_m is a command matrix that divides the journey into n_s sections where the driver has to apply a certain driving command. The first $n_s - 1$ sections are defined by a value of holding speed without braking command. In these sections the driver applies traction when it is needed to reach and maintain a speed value. If braking is needed to maintain the speed command, the driver will apply coast (null traction). The last section (n_s) is defined by a coasting command. In this section, the driver has to coast up to the braking curve to stop at the station. The orders collected in C_m are high level driving commands that do not affect the train safety performance. That means that if the train needs to brake to observe a speed limitation and the driving command demands coast or traction, the driver will apply the service brake. An example of a train journey that makes use of a holding speed without braking command and a coasting command before the final braking is shown in Figure 4-2.

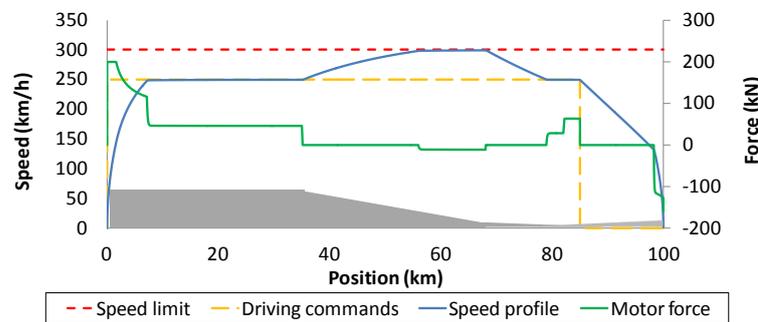


Figure 4-2. Train journey using a 250 km/h holding speed without braking command and a coasting command before the final braking

The command matrix C_m is as a matrix of $n_s - 1$ rows and 2 columns. The first column (sc_k values) contains the points of the track (position) where the sections end. The beginning of the first section is defined by the starting point of the journey (s_0). The beginning of the other sections is defined by the position of the end of the previous section sc_{k-1} . The second column represents the values of the holding speed without braking command (vc_k) that will be applied in that section. The last section (n_s) is not represented in the C_m because it is the final coasting command. In Figure 4-3 an example of a C_m of 4 sections is shown.

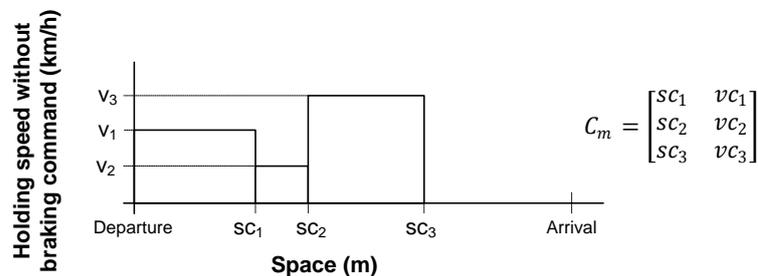


Figure 4-3. Command matrix of four sections

When selecting the value of n_s parameter, it is necessary to balance its effect on the energy consumption and its effect on passengers' comfort and drivers' performance. The number of possible solutions that the train can perform increases when n_s is incremented. Consequently, the increase of the solution space can lead to higher

energy savings. However, the value of n_s should be limited because the application of a high number of different driving commands may have a negative impact in passengers' comfort and drivers' execution.

It is difficult to implement a command matrix C_m with a high number of short sections in manual driving because it would demand frequent operations by drivers that could be stressful and non comfortable for them. Furthermore, in the transition from a section to another the driver will apply traction or coast to adapt the train speed to the new driving command depending on whether the new target speed is higher or lower than the previous one. If the number of sections (n_s) is high, traction and coasting will be frequent and that will be perceived as unpleasant by passengers in a long distance travel.

For these reasons the value of n_s must be established by the railway operator depending on the journey length, comfort criteria and drivers requirements.

4.2.3.2. DRIVING MODEL

To emulate the human control of the train, a closed-loop proportional-integral regulator is used. The tractive/braking effort demanded by the control $U_{reg}(t)$ is obtained at each simulation step using the following equations:

$$e(t) = v_{obj}(t) - v(t) \quad (4.17)$$

$$U_{prop}(t) = K_p \cdot e(t) \quad (4.18)$$

$$U_{int}(t) = \frac{K_p}{K_I} \cdot \frac{(e(t) + e(t - ts))}{2} + U_{int}(t - ts) \quad (4.19)$$

$$\begin{aligned} U_{reg}(t) &= U_{prop}(t) + U_{int}(t) && \text{if } v(t) \geq v_{max}(t) \text{ or } U_{prop}(t) + U_{int}(t) \geq 0 \\ U_{reg}(t) &= 0 && \text{if } v(t) < v_{max}(t) \text{ and } U_{prop}(t) + U_{int}(t) < 0 \end{aligned} \quad (4.20)$$

where $U_{reg}(t)$ is the total tractive/braking effort demanded by the control, $U_{prop}(t)$ is the proportional contribution to $U_{reg}(t)$ and $U_{int}(t)$ is the integral contribution to $U_{reg}(t)$. K_p and K_I are the proportional and the integral constants respectively. The error of regulation $e(t)$ is calculated as the difference between the objective speed $v_{obj}(t)$ and the current speed of the train $v(t)$. The constant ts is the simulation time-step and $e(t - ts)$ is the error of regulation in the previous simulation step. The minimum value for the total tractive/braking effort demanded is set to 0 when the train speed is below the maximum $v_{max}(t)$ to perform the holding speed without braking driving command (Eq. (4.20))

The objective speed $v_{obj}(t)$ is obtained from the driving commands and the maximum speed $v_{max}(t)$ as shown in Eq. (4.21). The maximum speed $v_{max}(t)$ is obtained as a function of the ceiling speed in the position of the train, the braking curves calculated for each speed limit reduction and the final braking at station, as shown in Eq. (4.22).

$$\begin{aligned} v_{obj}(t) &= \min(v_{max}(t), v_{c_k}) && \text{if } sc_{k-1} \leq s(t) \leq sc_k \\ v_{obj}(t) &= 0 && \text{if } sc_{ns-1} < s(t) \end{aligned} \quad (4.21)$$

$$v_{max}(t) = \min(v_{ceiling}(s), v_b(s), v_{fb}(s)) \quad (4.22)$$

where $v_{ceiling}(s)$ is the ceiling speed at the position of the train, $v_b(s)$ is the braking curve of the next ceiling speed reduction and $v_{fb}(s)$ is braking curve at the next station arrival. Braking curves are calculated as follows:

$$v_b(s) = \sqrt{v_{next}^2 + 2 \cdot d_{service} \cdot (s_{next} - s)} \quad (4.23)$$

$$v_{fb}(s) = \sqrt{2 \cdot d_{service} \cdot (s_{end} - s)} \quad (4.24)$$

where $d_{service}$ is the absolute value of the deceleration rate and s_{next} is the position of the next ceiling speed reduction corrected with the train length.

After that, the value of total traction/braking effort ($F_{TB}(t)$) demanded to the train is obtained from $U_{reg}(t)$ applying Eq. (4.25). The constraints related to the maximum traction force, maximum acceleration, maximum deceleration and force applied when traveling through neutral zones are applied to the output $U_{reg}(t)$.

$$\begin{aligned} F_{TB}(t) &= \min(U_{reg}(t), F_{max}(v), F_{maxAcc}(s, v), F_{maxJerk}(t)) \\ &\quad \text{if } U_{reg}(t) \geq 0 \text{ and not}(snz_{start}^l \leq s(t) \leq snz_{end}^l) \text{ for } l = 1, 2, \dots, Z \\ F_{TB}(t) &= \max(U_{reg}(t), F_{maxDec}(s, v), F_{minJerk}(t)) \\ &\quad \text{if } U_{reg}(t) < 0 \text{ and not}(snz_{start}^l \leq s(t) \leq snz_{end}^l) \text{ for } l = 1, 2, \dots, Z \\ F_{TB}(t) &= \max(\min(U_{reg}(t), F_{NZ}), F_{maxDec}(s, v), F_{minJerk}(t)) \\ &\quad \text{if } snz_{start}^l \leq s(t) \leq snz_{end}^l \text{ for } l = 1, 2, \dots, Z \end{aligned} \quad (4.25)$$

where $F_{maxAcc}(s, v)$ and $F_{maxDec}(s, v)$ are respectively the effort for the maximum acceleration rate and the effort for the maximum deceleration rate. The variables $F_{maxJerk}(t)$ and $F_{minJerk}(t)$ are the maximum and minimum effort allowed to motors and brakes to limit the jerk.

The limitations due to maximum acceleration and maximum deceleration ($F_{maxAcc}(s, v)$ and $F_{maxDec}(s, v)$) are taken into account to consider passengers' comfort and are calculated using Eq. (4.26) and Eq. (4.27). The constraints presented in Eq. (4.28) and (4.29) limit the train's jerk, which may affect the passengers' comfort (Chang et al., 1999; Chang and Sim, 1997; Chang and Xu, 2000; Wang et al., 2011).

$$F_{maxAcc}(s, v) = a_{max} \cdot \rho \cdot m + (F_r(v) + F_g(s)) \quad (4.26)$$

$$F_{maxDec}(s, v) = d_{max} \cdot \rho \cdot m + (F_r(v) + F_g(s)) \quad (4.27)$$

$$F_{maxJerk}(t) = F_m(t - ts) + F_b(t - ts) + (j_{max} \cdot \rho \cdot m) \cdot \Delta t \quad (4.28)$$

$$F_{minJerk}(t) = F_m(t - ts) + F_b(t - ts) - (j_{max} \cdot \rho \cdot m) \cdot \Delta t \quad (4.29)$$

where a_{max} is a positive constant that represents the maximum acceleration rate, d_{max} is a negative constant that represent the maximum deceleration rate and j_{max} is a constant that represents the maximum jerk allowed to the force applied by the train.

Finally, the values of $F_m(t)$ and $F_b(t)$ to be applied in the train model and energy consumption model presented in Sections 4.2.1 and 4.2.2 are obtained from Eq. (4.30) and Eq.(4.31), where $F_{TB}(t)$ is the total traction/braking effort previously calculated. The maximum electrical braking effort is considered to calculate the blending between electrical braking effort and pneumatic braking effort.

$$\begin{aligned} F_m(t) &= F_{TB}(t) & \text{if } F_{TB}(t) \geq 0 \\ F_m(t) &= \max(F_{TB}(t), F_{min}(v)) & \text{if } F_{TB}(t) < 0 \end{aligned} \quad (4.30)$$

$$\begin{aligned} F_b(t) &= 0 & \text{if } F_{TB}(t) \geq 0 \\ F_b(t) &= \min(F_{TB}(t) - F_{min}(v), 0) & \text{if } F_{TB}(t) < 0 \end{aligned} \quad (4.31)$$

4.2.4. DELAY RESPONSE MODEL

In urban railway lines, where headway between consecutive trains is small, traffic disturbances lead to frequent perturbed speed profiles of trains tracking the preceding one (Carvajal-Carreño et al., 2016). Instead, in low traffic density high-speed lines (like the Spanish ones) delays are less frequent and caused by temporary speed limitations (due to either civil or maintenance works, or to adverse weather conditions), or by changes in the target arrival time (due to on-line rescheduling).

When a train is affected by these situations, the nominal driving has to be updated to a faster speed profile to arrive on time at the next station.

The typical behaviour of the drivers when facing delays was explained in (Sicre et al., 2014) and it is called “immediate” delay recover strategy. It consists in driving the train as fast as possible when the delay is detected until the train is circulating on time. At this moment, the driver coasts to link with the nominal driving commands. This delay recovery strategy is not energy efficient because the driver does not make an optimal use of the time margins. The flow chart of the process used by the simulator to model the immediate delay recover strategy is shown in Figure 4-4.

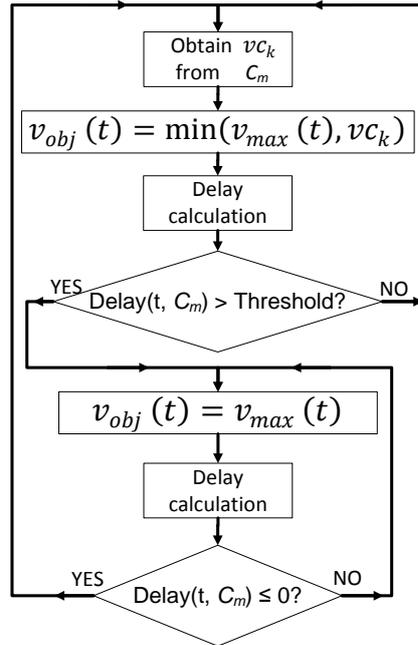


Figure 4-4. Immediate delay recover strategy

The DMOOP modelling explained in Section 4.3 can be used to recover delays in an energy efficient manner. When a delay is detected, the nominal driving is substituted by a speed profile from the Pareto front, taking into account the running time required to arrive on time and the energy consumption of the optimal solutions. This procedure is explained in detail in the following section.

4.3. DYNAMIC MULTI-OBJECTIVE OPTIMISATION MODEL BASED ON SIMULATION FOR THE ON-LINE DRIVING REGULATION PROBLEM

The on-line regulation problem of a high-speed train is modelled as a DMOOP and the objective function includes two objectives (see Eq. (4.32)):

$$\text{Min } f(X) = (RT(X), EC(X)) \quad (4.32)$$

where $RT(X)$ and $EC(X)$ are the simulated running time and the energy consumption of the solution X . A solution X is a set of driving commands gathered in a C_m . This solution is optimal and included in the resulting optimal Pareto front if X is not dominated by any other solution, that is, no other solution provides both lower RT and lower EC .

During each iteration of the dynamic problem, each solution X is simulated to calculate its running time $RT(X)$ and its energy consumption $EC(X)$ starting from the current position and speed of the train, up to the destination (see Figure 4-5). The simulation is performed applying equations (4.1) to (4.31).

The objective function is subject to the following constraint:

$$v_{c_k}(X) > v_{c_{min}} \quad \text{for } k = 1, 2, \dots, ns - 1 \quad (4.33)$$

The possible values of holding speed without braking commands (vc_k) are limited by a lower bound (vc_{min}) with the objective of not exploring solutions that can drive the train at low speeds that can be perceived as unpleasant by the passengers. Other restrictions are included via the train simulator model such as speed limits, maximum and minimum motor force.

The aim of the model is to find the Pareto front of the possible speed profiles that the train can perform in the journey and to track its variations during the train travel. The solutions that are in the optimal front are the non-dominated solutions. These solutions are those that cannot be improved at the same time in both energy consumption and running time.

The dynamic optimisation is applied to the on-line regulation problem because there is a continuous variation in the evaluation function due to the train movement. As the train travels, the results obtained using the simulator for the same solutions (running time up to the station and energy consumption) varies while the distance to the station decreases, and the number of solutions in the Pareto front decreases as well.

Figure 4-5 shows the variation of the Pareto front and 3 possible speed profiles at 4 different instants in the train movement. The optimisation uses as initial conditions the current position and speed of the train at each case represented (first instant: $s_0 = 0$ and $v_0 = 0$, second instant: $s_0 = 250$ and $v_0 = 240$, third instant: $s_0 = 270$ and $v_0 = 230$, fourth instant: $s_0 = 290$ and $v_0 = 190$). Running time and energy consumption of the Pareto front solutions only take into account the journey from the current train position to the arrival.

A filter is implemented to discard solutions with high running time in order to reduce the search space to the useful speed profiles for the train. This upper bound is given by the flat-out running time plus two times the remaining time margin at the optimisation instant. To filter the solutions with high running times, the model checks if the running time associated to each solution exceeds the upper bound. In this case, the algorithm evaluates that solution as a dominated solution and assigns the worst possible punctuation to its fitness value.

The optimisation is carried out repeatedly each time period t_c (calculation time), updating the Pareto curve of optimal solutions. Shorter calculation times provide a better tracking of the current situation of the train, but the number of iterations that the optimisation algorithm can use to find a solution is reduced. Thus, the parameter t_c must be adjusted carefully.

This DMOOP model is solved by means of a population-based optimisation algorithm. Two algorithms are chosen to compare their results solving the proposed dynamic model: DNSGA-II (Dynamic Non-dominated Sorting Genetic Algorithm II) and DMOPSO (Dynamic Multi-Objective Particle Swarm Optimisation algorithm). These algorithms are explained in the following subsections.

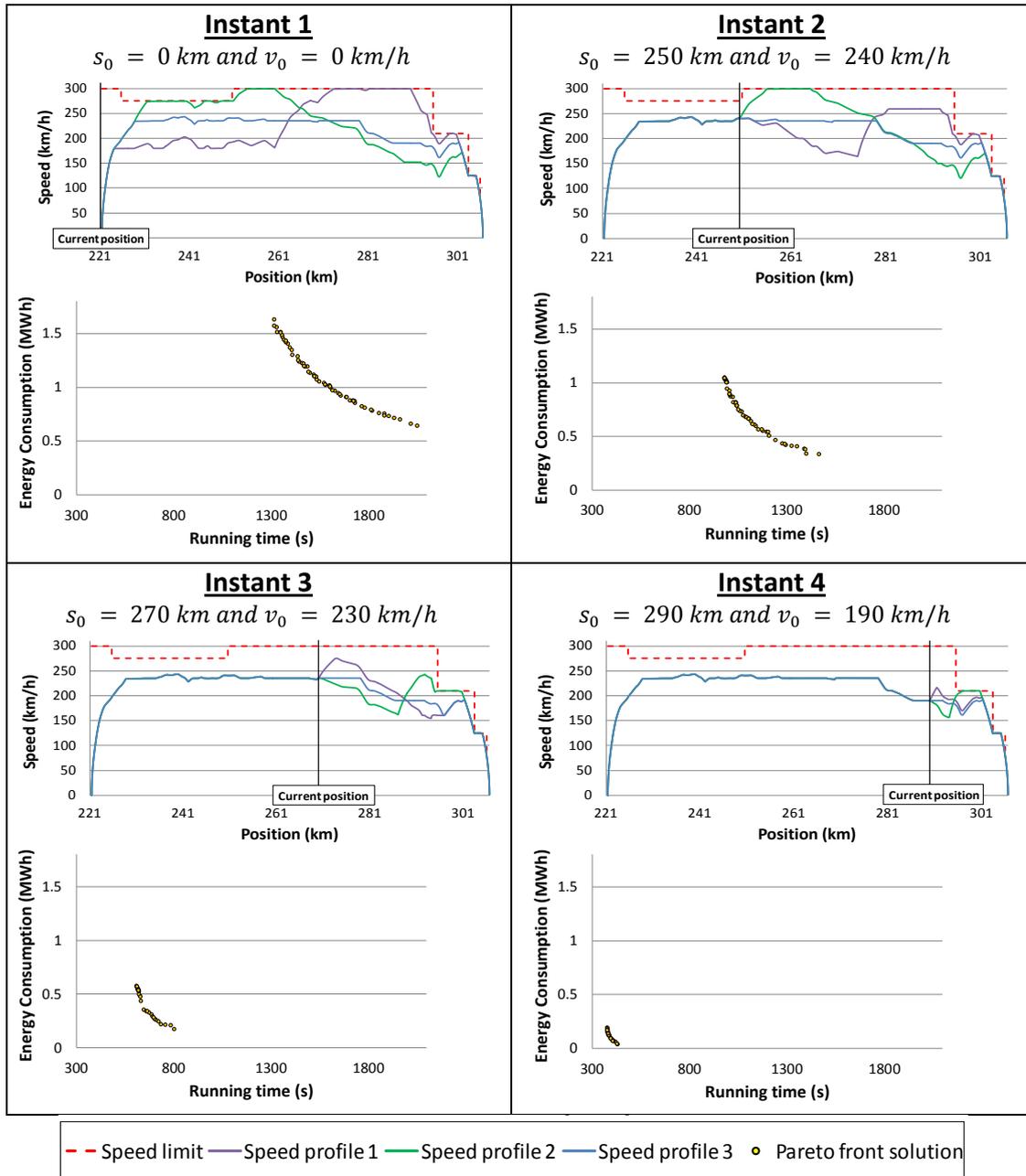


Figure 4-5. Variation of the Pareto front and some possible drivings during a train travel

4.3.1. DNSGA-II ALGORITHM FOR THE DYNAMIC ECO-DRIVING CALCULATION

DNSGA-II (Deb et al., 2007) algorithm is based on NSGA-II (Deb et al., 2002) but incorporates several mechanisms to take advantage of the knowledge obtained in the past. This knowledge allows the algorithm to improve its response working under dynamic environments. DNSGA-II is a multi-objective searching algorithm that imitates the natural selection and the natural genetic mechanisms. This algorithm evolves a population of individuals (solutions) through iterations using crossover and mutation operators. To share knowledge between different optimisation processes, DNSGA-II

uses as the initial population of every optimisation, part of the result population of the previous optimisation carried out.

The first execution of the algorithm ($ex = 1$) is launched before the departure of the train. This initial execution is computed in the same way as the static NSGA-II and it is used to obtain the initial set of optimal solutions and to decide the efficient driving that the train will perform during the journey fulfilling the schedule. The first execution starts creating a parent population P_0^1 of n_{pop} random solutions. Mutation and crossover operators are used to generate n_{pop} individuals of the offspring population Q_0^1 . The mutation and crossover operators applied in this algorithm were proposed in (Sicre, 2013). After that, P_0^1 and Q_0^1 are joined to generate the result population R_0^1 of size $2n_{pop}$. Each solution X of the population is a command matrix containing speed commands vc_k and it is simulated by means of equations (4.1) to (4.31), calculating the associated running time and energy consumption.

Then, domination level is used to sort R_0^1 population and unfeasible solutions are eliminated. Domination level of each solution is calculated as the number of solutions that dominate it. A zero value of domination level means that the individual is a non-dominated solution.

Once the R_0^1 population is ranked, the parent population for the next iteration P_1^1 is generated obtaining the n_{pop} solutions of R_0^1 with the lowest domination level. This process is performed adding solutions in sets of individuals that share the same domination level. If a set of solutions cannot be included because it exceeds the size of the parent population, the crowding distance operator (CD) (Raquel and Naval, Jr., 2005) is applied to that set of solutions to select the ones that will be part of P_1^1 . CD operator calculates the distance, in terms of the optimisation objectives, of a solution to the individuals that surround it. Therefore, the solutions of the last set of individuals are included in P_1^1 in decreasing order of CD until the parent population is filled. This way, the zones with low density of solutions are prioritised.

Once P_1^1 is created, the next iteration of the algorithm starts generating the offspring population Q_1^1 using crossover and mutation operators and, after that, a new result population R_1^1 is created. Again, each solution X of the population is simulated by means of equations (4.1) to (4.31). This process is repeated until a number of iterations it_{max} is reached ($it = it_{max}$). The final result of that execution $ex = 1$ of the algorithm is the set of solutions with 0 domination level of the last result population $R_{it_{max}}^1$.

The regulation system then selects among these optimal solutions, the one with the lowest energy consumption from those that have a running time lower than or equal to the scheduled running time (see section 3.3). The driving commands of the selected speed profile are shown to the driver to be executed and the train starts its travel.

During the train journey the algorithm will be executed at time intervals of t_c seconds using as initial conditions the current position and the speed of the train (s_{train} and v_{train}) at that moment (see Figure 4-5). Since these executions are used on-line to update the Pareto front, they will be limited by a time bound t_c and not by the number

of iterations. When the calculation time reaches t_c the algorithm is stopped, the Pareto front is updated and a new execution of the algorithm is started.

To accelerate the optimisation process and improve solutions, DNSGA-II takes advantage of the information of previous executions. Thereby, the starting parent population of an on-line execution of the algorithm P_0^{ex} is not generated randomly as in the first execution. Instead, it is generated using the result population of the previous execution ($ex - 1$), re-evaluating those solutions and applying a diversity mechanism (Deb et al., 2007). These mechanisms are applied to the re-evaluated result population R_{end}^{ex-1} in order to increase the diversity of the initial parent population of execution ex . There are two versions of DNSGA-II depending on the diversity mechanism applied:

- DNSGA-II-A: a percentage $\zeta\%$ of the result population is replaced by randomly created individuals.
- DNSGA-II-B: a percentage $\zeta\%$ of the result population is replaced by mutated solutions of randomly selected individuals of the population.

Figure 4-6 shows the flow chart of DNSGA-II.

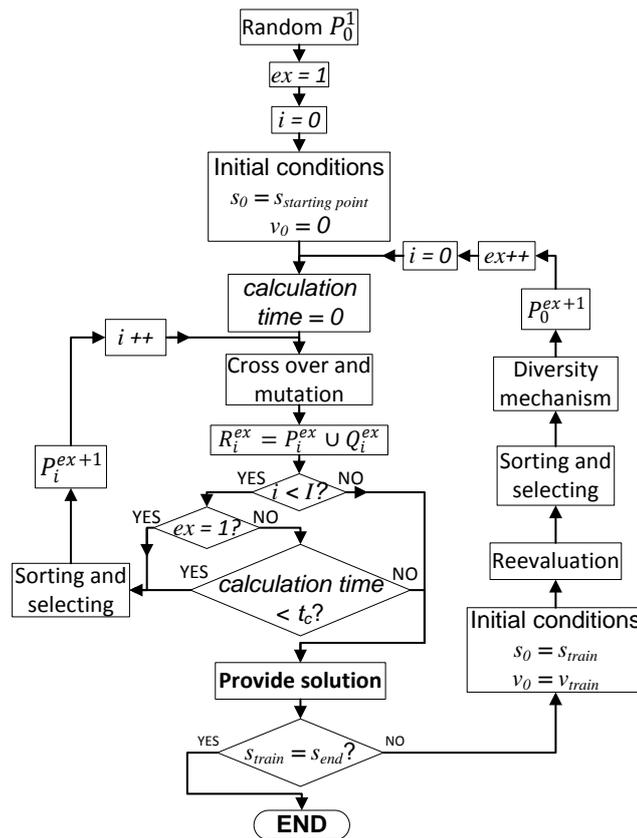


Figure 4-6. Flow chart of DNSGA-II

4.3.2. DMOPSO ALGORITHM FOR THE DYNAMIC ECO-DRIVING CALCULATION

The second algorithm selected to solve the DMOOP of the high-speed train regulation is the DMOPSO (Salazar Lechuga, 2009) that is the extension of MOPSO algorithm (Coello et al., 2004) to dynamic problems. As in the previous case, DMOPSO is based on

MOPSO but including some mechanisms to take advantage of the previous knowledge. DMOPSO imitates the behaviour of insects swarming together to hunt for food. In this algorithm, a set of particles moves through iterations within the search space. The particles position is updated using a velocity that depends on the best position found by each particle ($pbest$) and the best position found by the whole swarm ($gbest$). As in the previous case, past knowledge is shared between different optimisation processes. DMOPSO uses as the initial position of the particle the last position where the particles were in the previous optimisation carried out. Furthermore, the Pareto front obtained in the last optimisation process is shared with the following optimisation.

As in the DNSGA-II algorithm, the first execution of the DMOPSO is carried out before the departure of the train and this first execution ($ex = 1$) is exactly as the static MOPSO. A set of particles is generated randomly and their velocity vectors are also generated randomly. Then the positions of the particles are compared and the non-dominated solutions are stored in an external archive A . The solutions in A are sorted using CD operator in decreasing order, to give more priority to the zones of the Pareto front with low density of solutions. Then, the $pbest$ of each particle is updated as its current position and $gbest$ is randomly selected from the archive A giving more probability to the solutions that are in the top of A .

The position of the particles is updated for the next iteration it using Eq. (4.34) and Eq. (4.35).

$$VP_j(it) = w \cdot VP_j(it - 1) + c_1 \cdot r_1 (P_j - X_j(it - 1)) + c_2 \cdot r_2 (P_g - X_j(it - 1)) \quad (4.34)$$

$$X_j(it) = X_j(it - 1) + VP_j(i) \quad (4.35)$$

where $X_j(it)$ the position and $VP_j(it)$ is the speed of particle j at iteration it . Position P_j is the $pbest$ of particle j and position P_g is the $gbest$ for the whole swarm. Constants c_1 and c_2 are “social factors” and determine the weight of the distance to $pbest$ and $gbest$. r_1 and r_2 are random numbers between 0 and 1 and w is the “inertia” that determines the weight of the previous velocities.

Once the new position of the particles in iteration it is calculated, archive A is updated adding new non-dominated solutions and deleting the solutions that are dominated by the new ones. The $pbest$ of each solution is updated to the current position of the particle if it dominates the previous $pbest$. In other case, the previous $pbest$ remains. To select $gbest$, the solutions in A are sorted in decreasing order of CD . Then, the position $gbest$ is updated selecting randomly a solution from A giving higher probability to the solutions in the top of the archive.

At this first execution, the process explained is repeated updating the Pareto front stored in A until a number of it_{max} iterations is reached. When the optimisation finishes, the system selects from A the speed profile with the lowest energy consumption from those with a running time lower than or equal to the commercial running time. The solution is given to the driver and the train starts the journey.

The following executions ex of the algorithm will be started at intervals of t_c seconds using as initial conditions the position and the speed of the train at those moments.

These on-line executions will be limited by a calculation time of t_c and not by number of iterations. When the calculation time reaches t_c the algorithm is stopped and the solutions contained in A are provided to the system to update the Pareto front. Then, a new execution is started.

The executions carried out during the journey do not start with a random set of particles. Instead, the particles stored in A at the previous execution ($ex - 1$) are used and their positions are re-evaluated under the new conditions. This way, the past knowledge obtained in the previous executions is included in the dynamic algorithm.

To calculate the $pbest$ of each particle in the first iteration of an execution ex , two mechanisms were proposed in (Salazar Lechuga, 2009) that result in two versions of DMOPSO:

- DMOPSO-A uses $Response_a$ that sets $pbest$ of the particle to its current position if the current position dominates $pbest$. In other case, the previous $pbest$ remains.
- DMOPSO-B uses $Response_b$ that sets $pbest$ to current position of the particle.

Once $pbest$ is updated, $gbest$ is selected from A and, using this information, the position and the velocity of each particle is updated using Eq. (4.34) and Eq. (4.35).

Figure 4-7 shows the flow chart of DMOPSO.

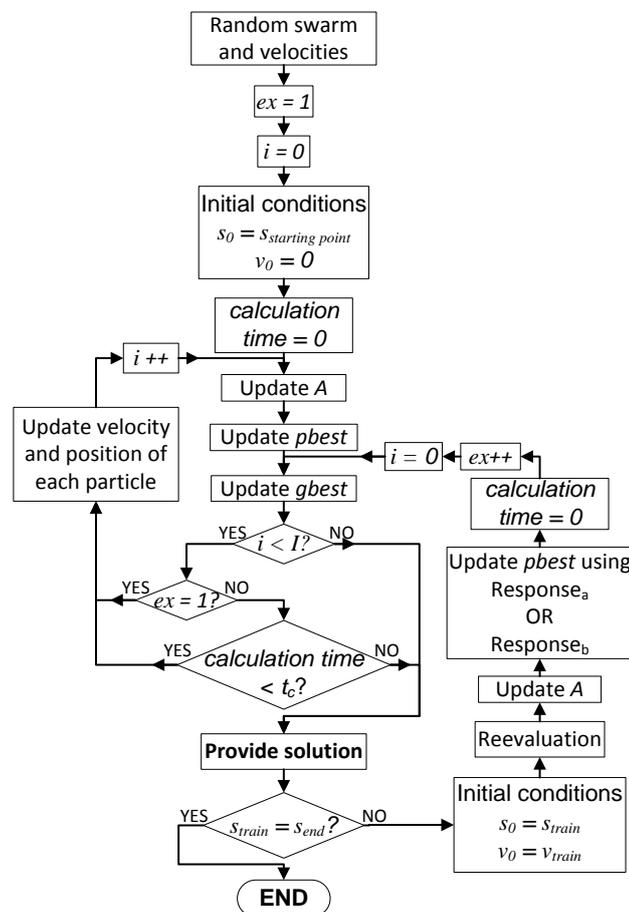


Figure 4-7. Flow chart of DMOPSO

4.3.3. ON-LINE SPEED PROFILE SELECTION

During the journey, the Pareto front is calculated by the optimisation algorithm and updated every t_c seconds. However, a new set of commands is not presented to the driver, unless a significant delay arises (over a threshold value). This way, the system does not change continuously the driving commands.

When the train delay exceeds the threshold, the system firstly warns the driver to apply the flat-out speed profile. Flat-out driving is used to check when it is possible to reduce the delay. When the delay starts to decrease, the system selects and displays the new driving commands from the updated Pareto front. The speed profile selected is the one that allows the train to arrive on time consuming the lowest amount of energy.

As was previously illustrated, the result of the optimisation algorithms is a set of non-dominated solutions. For this reason, an online decision maker is implemented to select the most appropriated driving commands to each situation. When a new speed profile is needed, the decision maker calculates the objective running time subtracting the current time to the nominal arrival time. With this information, the online decision maker selects the solution that fulfils at the same time the following conditions:

- The solution chosen must have an associated running time lower than or equal to the objective running time. This way, it is ensured that the train will arrive on time at the destination.
- The solution chosen must have an associated running time as close as possible to the objective running time. This way, it is ensured that the train will consume the lowest amount of energy for the objective of time imposed.

If the slack time is consumed there will not be a solution that complies with the first condition. In this case, the decision maker selects flat-out driving.

In Figure 4-8 the flow chart of the on-line speed profile selection process is depicted, as well as its relation with the dynamic optimisation algorithm.

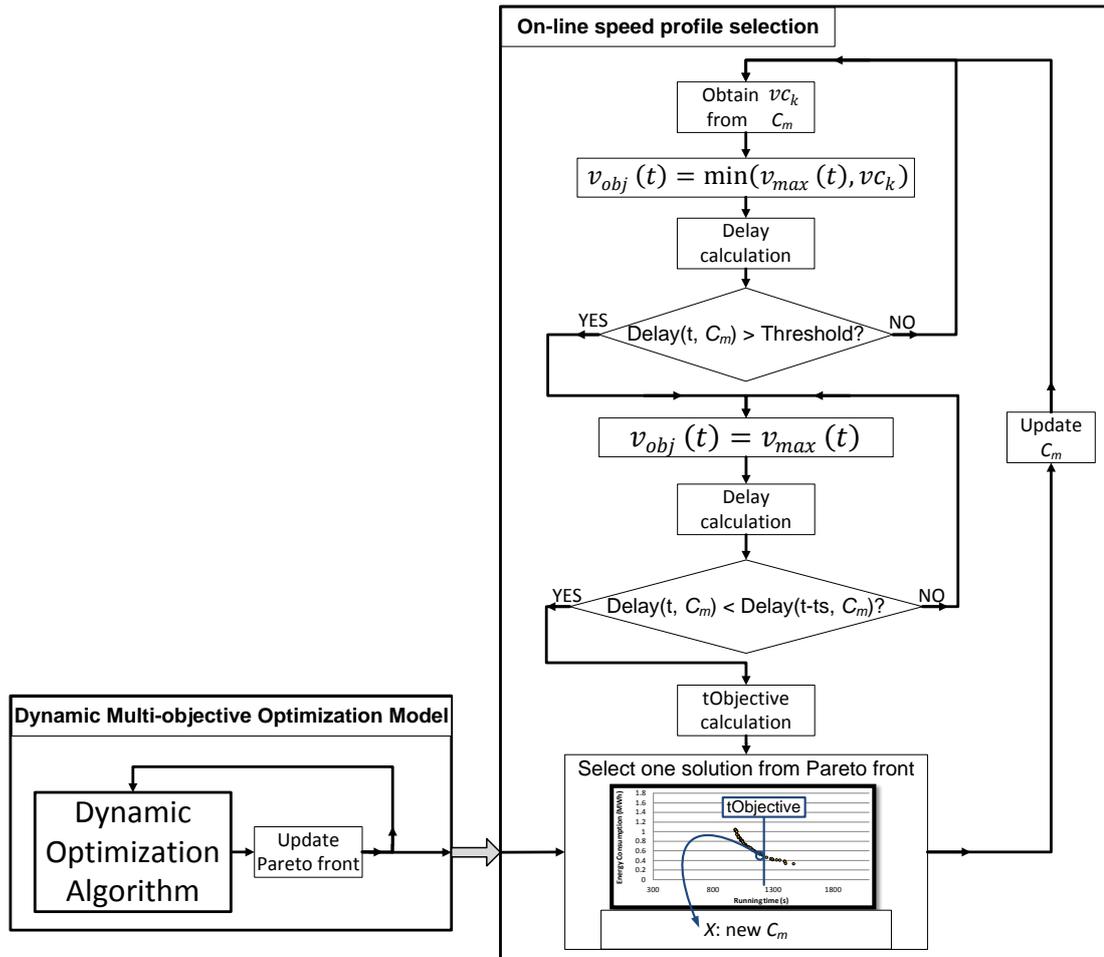


Figure 4-8. On-line speed profile selection

4.4. CASE STUDY

The DMOOP formulation of the driving regulation problem and the different versions of the DNSGA-II and DMOPSO have been applied to a case study using real data from a Spanish high-speed line. The stretch analysed runs from Calatayud to Zaragoza and its length is 85.4 km. The high-speed train is a Talgo-Bombardier class 102. This train has two motors of 8 MW and 200 kN of maximum traction effort. The train is 200 m long and its empty weight is 324 t. The operational restriction $v_{c_{min}}$ for the driving commands is set to 150 km/h to avoid low speed phases in the middle of the journey.

To check the performance of the methods presented, the train is simulated to suffer a disturbance to accumulate a delay. This disturbance is an unexpected temporary speed limitation of 90 km/h located between 229 km and 230 km. If the driver does not change the nominal speed profile calculated at the beginning of route, the train would arrive at the next station 1 minute and 20 seconds late. Figure 4-9 represents the nominal speed profile affected by the unexpected speed limitation.

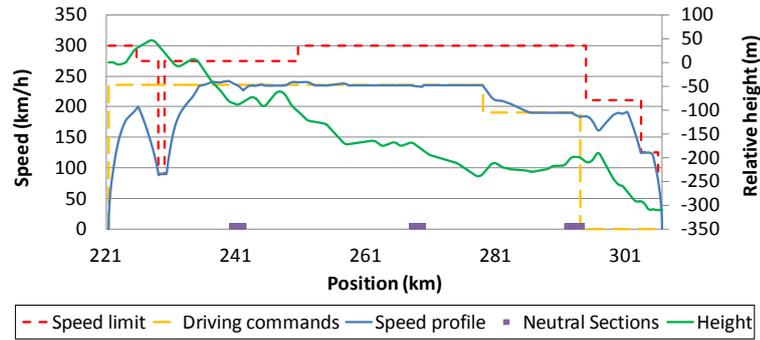


Figure 4-9. Nominal speed profile observing the temporary speed limitation

4.4.1. ALGORITHMS CONVERGENCE ANALYSIS

The DMOOP formulation of the regulation problem is applied in this case study to compensate the schedule deviation caused by the unexpected temporary speed limitation. DNSGA-II and DMOPSO are applied to solve the optimisation problem and their respective versions are tested in order to select the best configuration of each algorithm. Finally the performance of the versions selected of DNSGA-II and DMOPSO are compared.

The test that has been carried out to check the algorithms performance is based on two threads of simulation. The main thread simulates the whole journey of the train matching the simulation clock with the real-time clock. The secondary thread carries out the dynamic optimisation algorithm that makes several fast simulations to evaluate possible solutions. Both threads were executed using an Intel Core i7-2600 CPU@3.4 GHz and 16GB of RAM.

A test consists in the simulation of a train travel. Initially, the nominal speed profile is calculated before departure, the train starts the journey following nominal commands, and then it faces an unexpected speed limitation. When the delay is detected, a delay response strategy is applied to obtain a new speed profile, the train follows the new driving commands and finally arrives at the next station.

To analyse and compare the convergence of the algorithms, they are executed and the new commands are applied (displayed to the driver) when the train has left the temporary speed restriction. 30 iterations are allowed for the optimisation that provides the new speed profile with the objective of studying in detail the behaviour of the algorithms

Hypervolume metric (Jiménez et al., 2013) is used in these tests to measure and compare the quality of the Pareto fronts obtained. Hypervolume measures the piece of the search space that is not dominated by any solution of the Pareto front. It is a quality metric that assesses not only the proximity of the result to the real Pareto front, but also the spread of the solutions obtained. Hypervolume is calculated as shown in Eq. (4.36).

$$HV = 1 - \frac{\sum_{j=1}^{NI} \left[(f_{no}^{max} - f_{no}^j) \prod_{q=1}^{no-1} (f_q^{sup_j} - f_q^j) \right]}{\prod_{q=1}^{no} (f_q^{max} - f_q^{min})} \quad (4.36)$$

where f_q^{max} and f_q^{min} are the maximum and the minimum values of the q^{th} objective. NI represents the number of individuals that form the Pareto front obtained from the optimisation algorithm and no is the number of objectives in the problem. The variable f_q^j is the value of the objective q for the individual j and $f_q^{sup_j}$ is the value of the objective q^{th} for the solution higher adjacent to individual j in the q^{th} objective. The value of f_{no}^{max} is taken from the maximum value of the objective no . On the other hand, f_{no}^j is the value of the objective no for the individual j .

Due to the random search nature of the algorithms, different solutions can be obtained for similar situations. To obtain a robust analysis, the hyper volume results provided are the mean value of 10 tests in the same journey.

4.4.1.1. DNSGA-II-A AND DNSGA-II-B CONVERGENCE ANALYSIS

Versions A and B of DNSGA-II presented in Section 4.3.1 are analysed and compared with the application of the static NSGA-II at each interval t_c . It is necessary to balance between a better tracking of the current situation of the train and the number of iterations that the algorithms can perform. With this purpose, two values of the parameter t_c were used to make the comparisons: 60 s and 120 s. Note that the train simulator includes the behaviour of the onboard signalling system with a control cycle of 250 ms. The driving module of the simulator is in charge of generate the traction/braking effort taking into account the command matrix and the signalling system speed limits and braking curves. Therefore, safety requirements are fulfilled because the braking curves are calculated four times every second although the Pareto front of high level manual driving commands solutions is updated every 60/120 s.

Table 4-1 presents the tuned parameters for all the DNSGA-II versions and NSGA-II.

Initial population size (n_{pop})	Number of crossover	Number of mutations	Number of sections in C_m (n_s)	Diversity mechanism parameter (ζ)
40	13	26	4	15 %

Table 4-1. Tuned parameters of DNSGA-II algorithm

Figure 4-10 shows the evolution of the mean value of the Hypervolume metric through the iterations of the DNSGA-II and NSGA-II algorithms. The algorithms that make the calculation with $t_c = 60$ s carry out 5 iterations and the algorithms with $t_c = 120$ s carry out 9 iterations. Two vertical lines highlight the results for 5 and 9 iterations in the figure. In the case of the static NSGA-II there is no difference between the results for $t_c = 60$ s and $t_c = 120$ s because it does not use the previous history and always starts with a random population.

Figure 4-10 shows that DNSGA-II outperformed clearly the static NSGA-II because the use of previous knowledge allows it to start with a population closer to the real optima. This way the optimisation process is accelerated and less number of iterations are necessary to converge.

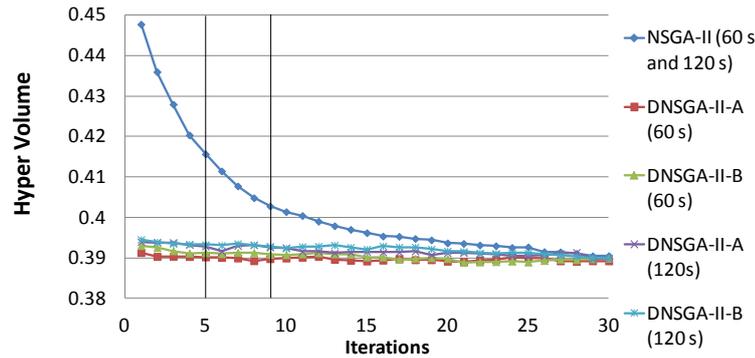


Figure 4-10. Mean value of normalised Hypervolume during the evolution of the DNSGA-II versions

It can be observed that the algorithms with $t_c = 120\text{ s}$ start from a population with a value of HV larger than the algorithms with $t_c = 60\text{ s}$. The final results obtained for $t_c = 60\text{ s}$ show a low value of HV although the algorithms with $t_c = 120\text{ s}$ carry out more iterations. Furthermore, using a low value of t_c has the advantage of updating the Pareto front with a higher frequency.

Regarding version A and B of DNSGA-II, the results demonstrate that DNSGA-II-A provides solutions with higher quality than DNSGA-II-B. The substitutions of individuals of the first population by random individuals instead of mutated individuals make that the algorithm starts from a point closer to the optimal solution, converging with a lower number of iterations.

4.4.1.2. DMOPSO-A AND DMOPSO-B CONVERGENCE ANALYSIS

As in the previous case, versions A and B of DMOPSO presented in Section 4.3.2 are applied to the high-speed train regulation problem and compared with static MOPSO. The values of t_c used in this study are 60 s and 120 s. Table 4-2 presents the tuned parameters of DMOPSO algorithms.

Swarm size (n_{swarm})	Social factor (c_1)	Social factor (c_2)	Inertia weight (w)	Top of A select	Top of A select probability
40	2	2	0.2	3 %	99 %

Table 4-2. Tuned parameters of DMOPSO algorithm

The evolution of the HV metric through the iterations is shown in Figure 4-11. As in the previous section, two vertical lines highlight in the figure the results for 5 and 9 iterations as these are the number of iterations that the algorithm is capable to carry out with $t_c = 60\text{ s}$ and $t_c = 120\text{ s}$ respectively.

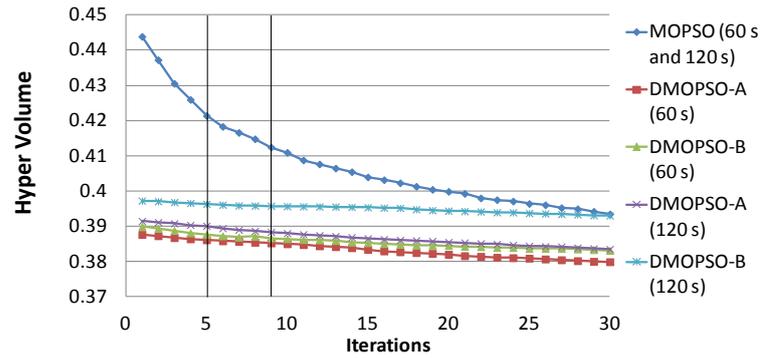


Figure 4-11. Mean value of normalised Hypervolume during the evolution of the DMOPSO versions

The conclusions obtained are similar to the ones obtained in the convergence analysis of DNSGA-II. DMOPSO outperformed MOPSO because the use of the previous knowledge accelerates the optimisation. DMOPSO with $t_c = 60 s$ has better results than DMOPSO with $t_c = 120 s$.

DMOPSO version A presents lower values of HV than version B. The use of the history contained in $pbest$ instead of substituting it for the current position of the particle makes that the algorithm find an initial better Pareto front accelerating the convergence.

4.4.1.3. DNSGA-II AND DMOPSO COMPARISON

The results of DNSGA-II-A and DMOPSO-A with $t_c = 60 s$ are compared in Figure 4-12.

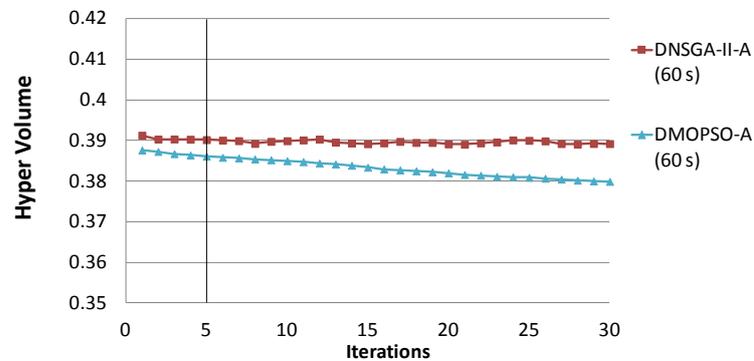


Figure 4-12. Mean value of normalised HV evolution of DNSGA-II-A and DMOPSO-A using $t_c = 60 s$

As can be seen, DMOPSO-A initial population has a value of HV lower than DNSGA-II-A, and it obtains finally a better Pareto front.

4.4.2. ENERGY CONSUMPTION ANALYSIS

In this section the energy savings using DMOOP formulation are compared with the “immediate” delay recover strategy. The train regulation problem is solved using DMOPSO-A and DNSGA-II-A with $t_c = 60 s$ because, as explained in the previous section, these are the algorithms that obtain better results. In addition, the static version of MOPSO is included in the comparison to show the benefits of the dynamic formulations in terms of energy savings.

Table 4-3 presents the mean value and the standard deviation of the energy consumption and train delay at the end of the journey for 10 tests. Regenerative energy is taken into account to obtain the net energy consumption, and it is calculated at the pantograph and at the substations considering the electrical network losses.

In view of the results, it can be concluded that the use of a static optimisation algorithm such as MOPSO can produce relevant energy savings (5.6%) compared with the immediate delay recover strategy. However, dynamic algorithms improve these results: DNSGA-II provides 6.9% of energy savings and DMOPSO provides 7.8%. Furthermore, the results of energy consumption obtained by DMOPSO are less spread than the results obtained by MOPSO and DNSGA-II since the standard deviation of the DMOPSO results has a lower value.

The quality of the obtained Pareto front affects energy consumption in two ways. On one hand, when solutions are close to the real optima the energy consumption is close to the minimum. On the other hand, the number of solutions in the Pareto front and their spread affects the energy consumption. As explained in Section 4.2.4 the speed profile selected to recover a delay will be the one that presents the lowest energy consumption from those in the Pareto front that have an associated running time lower than the objective. Therefore, the smaller the gap between solutions, the closer the running time to the objective running time and, consequently, the lower the energy consumption. This effect can be checked in the train delay at the arrival presented in Table 4-3.

		Pantograph net energy consumption (MWh)	Substation net energy consumption (MWh)	Delay at the arrival (s)
Immediate delay recover		1.043	1.170	-38.00
Mean value	MOPSO	0.985	1.104	-16.60
	DMOPSO	0.965	1.078	-7.80
	DNSGA-II	0.976	1.089	-16.20
Standard deviation	MOPSO	0.01776	0.02071	11.56
	DMOPSO	0.00344	0.00349	2.60
	DNSGA-II	0.00909	0.01047	6.03
Mean value of the variations compared with the immediate delay recover	MOPSO	5.51%	5.66%	
	DMOPSO	7.39%	7.82%	
	DNSGA-II	6.40%	6.95%	

Table 4-3. Mean value of energy savings and delays obtained by DMOOP for the complete journey

DNSGA-II produces 1.2% of increment in the result of energy saving compared with MOPSO although the delay at arrival results are similar (16 s ahead). This improvement on the energy consumption is because the solutions of DNSGA-II are closer to the real optima. DMOPSO improves energy savings in 2.1% and 0.9% compared with MOPSO and DNSGA-II respectively. This improvement is achieved not only because solutions are closer to the real optima, but also due to the reduction of gaps in the Pareto front. As a result, the time that the train arrives ahead is reduced with the DMOPSO algorithm (7.8 s).

The speed profiles obtained by the DMOPSO algorithm and the “immediate” delay recover strategy (the typical behaviour of drivers) are compared in Figure 4-13. In the “immediate” delay recover strategy, when the delay is detected and the train leaves the speed limitation, it has accumulated 1 minute and 20 seconds of delay and it starts flat out driving. The delay is made up at 267 km and then the train starts coasting to link with the nominal speed profile. As a result, the train arrives at the station 38 s ahead.

The speed profile obtained by DMOPSO is clearly different. When the train leaves the speed limitation (230 km) it accelerates and when delay starts to be recovered (236 km) the system selects a speed profile from the Pareto front and the driver executes the new set of driving commands. This solution is the result of the last optimisation carried out (with the initial condition $s_0 = 234 \text{ km}$). As can be seen in Figure 4-13, the new speed profile distributes the delay recovery along the rest of the distance to run. This way, the energy consumption is reduced using eco-driving.

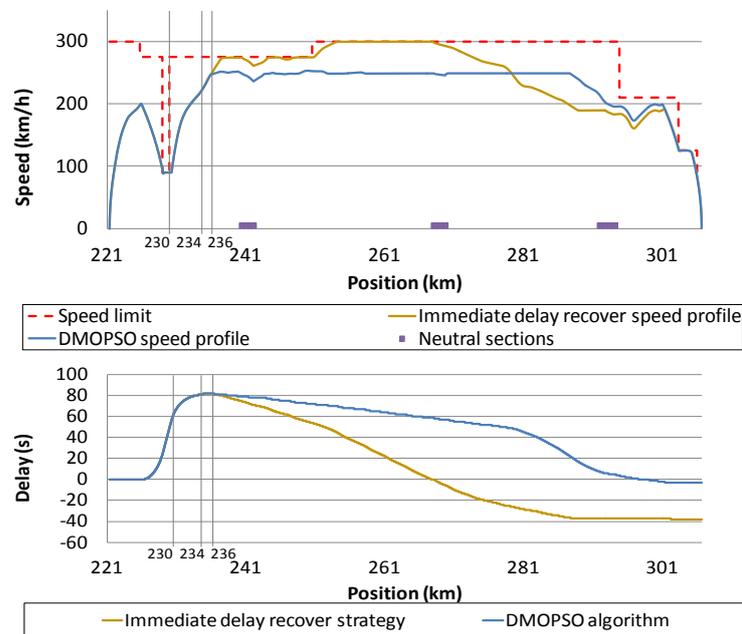


Figure 4-13. Speed profiles and delay evolution of “immediate” delay recover strategy and DMOPSO.

4.5. CONCLUSIONS AND CONTRIBUTIONS

This chapter of the thesis proposes a dynamic multi-objective optimisation model based on accurate simulation of the train motion to handle the real-time regulation of a high-speed train using eco-driving. The model calculates the Pareto front of the possible speed profiles before the departure of the train. After that, during the journey, this Pareto front is updated at time intervals by an optimisation algorithm that adapts it to the new conditions of the train. Using this model, a set of energy efficient speed profiles is always available to be executed when it is necessary to change the nominal driving commands to perform a new speed profile with a different running time associated.

Two algorithms have been tested to solve the dynamic model: DNSGA-II and DMOPSO. These algorithms are the dynamic extension of NSGA-II and MOPSO that have been

demonstrated suitable to eco-driving problems. Both algorithms take advantage of the solutions obtained in previous executions to accelerate the optimisation process.

Versions A and B of DMOPSO have been compared with the two versions of DNSGA-II and the static algorithms MOPSO and NSGA-II. The results demonstrate that the dynamic algorithms outperformed their static versions tracking the changes in the Pareto front. Furthermore, the Pareto front provided by DMOPSO-A has an associated better Hypervolume metric and convergence.

To analyse the energy benefits of the dynamic models, MOPSO, DMOPSO and DNSGA-II have been applied to obtain a new speed profile when an unexpected temporary speed limitation affects the train. These algorithms have been compared with the typical behaviour of drivers: the “immediate” delay recover strategy.

The solutions obtained applying a static algorithm such as MOPSO provide important energy savings (5.6% of the whole trip energy) compared with the “immediate” delay recover strategy.

However, these energy savings can be improved significantly using the dynamic version of the algorithms. The best performance is obtained by the DMOPSO, which has associated energy savings of 7.8%.

It has been shown that the simulation of the train motion can be applied in real-time to recover delays minimizing the energy consumption, with the associated advantages of model flexibility and accuracy.

The main contributions of this chapter are:

- A dynamic multi-objective optimisation model of the online eco-driving of high-speed trains.
- The assessment of the performance of two versions of DNSGA-II solving the online eco-driving problem.
- The assessment of the performance of two versions of DMOPSO solving the online eco-driving problem.
- The comparison of the results obtained by DNSGA-II and DMOPSO.
- The delay response mechanism based on a continuously updated Pareto front.
- The analysis of the energy savings that can be obtained by means of the proposed dynamic model compared with the static models and the typical delay response method applied by drivers.

CHAPTER 5

BALANCING ENERGY CONSUMPTION AND RISK OF DELAY IN HIGH-SPEED TRAINS

5.1. INTRODUCTION

In the previous chapter, a dynamic multi-objective optimisation model was proposed for the real-time regulation of a high-speed train. Furthermore, two algorithms with different configurations were tested and compared with the performance of static algorithms. The study did not take into account the uncertainty of the solutions obtained.

To take into account punctuality requirements, besides using an accurate model, it is necessary to model the uncertainty associated to HSR operation. For this reason, in the present chapter the model proposed will be extended to incorporate all the main sources of uncertainty that are present in the real operation of a high-speed train.

The main uncertainty in HSR regulation is associated with the contingencies that may occur in the line. Contingencies are usual situations that produce delays and are related to temporary speed limitations and traffic perturbations. The commercial running time between two stations is designed using a time margin in timetables to deal with contingencies. If necessary, this margin is available to make up delays and, if not, it is consumed during the train travel to perform efficient speed profiles. If the margin is quickly consumed during the journey, the train could perform an efficient speed profile but it might not be able to react to unexpected delays. On the other hand, the speed

profiles that retain the time margin until the end of the journey are highly energy consuming but more robust to contingencies in the line. In the literature, it was not found an eco-driving work that takes into account the robustness of solutions to contingencies in the line.

In addition, it is also important to model the uncertainty associated with manual driving, considering that there are always small deviations in the application of driving commands. The uncertainty in these parameters is usually better represented using fuzzy knowledge modelling (Bellman and Zadeh, 1970). The main advantages of using fuzzy modelling are its capacity to work with imprecise or incomplete data and its flexibility and simplicity to be implemented providing fast calculation times. The information of how drivers apply driving commands is usually incomplete or non-existent because each driver has its own driving style. Uncertainty model by means of fuzzy knowledge has been applied in several studies related to eco-driving. GA is used in (Bocharnikov et al., 2007) to obtain the optimal speed profile in urban DC railways where the running time and energy consumption of the solutions are represented by fuzzy sets. In (Cucala et al., 2012b), a model to design jointly timetable and speed profiles using a fuzzy model of delays and punctuality constraints is proposed to be applied in an offline manner. The algorithm proposed in (Carvajal-Carreño et al., 2014) obtains the Pareto curve of speed profiles for urban automatically driven railways where the uncertainty in mass value is modelled using fuzzy sets. In (Sicre et al., 2014), the uncertainty in the application of manual driving commands is modelled using fuzzy knowledge, considering the fluctuations in the holding speed command and the variability in the response time in the update of driving commands. However, the model proposed does not take into account the perturbations caused by contingencies.

In this chapter of the thesis a new real-time eco-driving algorithm is proposed to obtain the energy efficient speed profile that the train must perform during its journey to fulfil punctuality requirements. This algorithm uses a detailed train simulation model to obtain the optimal solutions. Moreover, the speed profiles obtained are defined by a set of holding speed without braking commands. One of the main contributions of this piece of research is the definition of the “risk of delay in arrival” (*RD*) and its inclusion in the optimisation problem as a third objective to take into account the uncertainty of line contingencies. *RD* depends on the time margin consumption rate of the speed profile. Considering this, the objectives of the optimisation algorithm are not only running time and energy consumption, as usual in the literature, but also a third objective that is the risk of delay in arrival. The uncertainty in the manual application of driving commands is modelled by means of fuzzy numbers.

To overcome the computational time limitations when optimising using detailed simulation models, the algorithm proposed states the problem as a dynamic multi-objective optimisation problem (DMOOP) (Helbig and Engelbrecht, 2014) as was proposed in Chapter 4. DMOOPs are multi-objective optimisation problems with at least one objective changing over time. The product of a multi-objective optimisation algorithm is the set of non-dominated solutions, also called Pareto front. In the case of the HSR regulation, the Pareto front is a set of speed profiles, each of them with different driving commands and different running time, energy consumption and *RD*.

When it is necessary to update the running time, for instance, when a delay arises, a new speed profile is selected from the Pareto front.

The running time, energy consumption and RD of solutions in the Pareto front are changing continuously during the journey because of the train movement. Thus, the algorithm has to update the Pareto front periodically under the new circumstances.

The new algorithm proposed is an hybrid optimisation technique where the Non-dominated Sorting Genetic Algorithm III (NSGA-III) (Deb and Jain, 2014) is combined with the fuzzy modelling of the driving commands and designed to solve DMOOPs. NSGA-III was selected as the base of the proposed algorithm because it has demonstrated good performance solving three-objective problems.

The chapter is organised as follows. Section 5.2 presents the fuzzy manual driving model. The new parameter risk of delay in arrival is defined in section 5.3. The proposed DNSGA-III-F algorithm is presented in Section 5.4. In Section 5.5, the algorithm to update the driver's commands in real-time is described. The results of the application of the algorithm to a case study of a real high-speed line are analysed in Section 5.6. Finally, section 5.7 details the main conclusions obtained in this piece of research.

5.2. FUZZY MANUAL DRIVING MODEL

The fuzzy driving model used in this chapter of the thesis was proposed by Sicre et al. in (Sicre et al., 2014). This model is based on the Command Matrix model (C_m) that makes use of the holding speed without braking driving described in Chapter 4. The model deals with the vagueness associated with manual driving by expressing the holding speed and changing points in C_m by means of fuzzy numbers.

Human drivers can hold the speed of the train with precision. However, it is expected certain fluctuations around the holding speed value during the journey. To model this variability each holding speed value in C_m will be represented using a symmetrical triangular fuzzy number \widehat{vc}_k whose core is the value of vc_k contained in the crisp C_m and whose support is a constant Δv as shown in Eq. (5.1).

$$\mu_{\widehat{vc}_k}(y) = \begin{cases} 0 & \text{if } y \leq vc_k - \frac{\Delta v}{2} \\ \frac{2}{\Delta v} \cdot y - \frac{2vc_k}{\Delta v} + 1 & \text{if } vc_k - \frac{\Delta v}{2} < y \leq vc_k \\ -\frac{2}{\Delta v} \cdot y + \frac{2vc_k}{\Delta v} + 1 & \text{if } vc_k < y \leq vc_k + \frac{\Delta v}{2} \\ 0 & \text{if } y > vc_k + \frac{\Delta v}{2} \end{cases} \quad (5.1)$$

On the other hand, human drivers do not change the driving commands exactly at the changing positions sc_k and the change is made with certain anticipation or delay. This deviation is modelled as a fuzzy anticipation/delay time in the application of a new

command ($\tilde{t}_{a/d}$). The fuzzy anticipation/delay time will lead to an advance/delay distance from the crisp changing position sc_k resulting in a fuzzy changing position (\tilde{sc}_k). The fuzzy number \tilde{sc}_k can be obtained as the sum of the crisp value sc_k and the product of the fuzzy numbers \tilde{vc}_k and $\tilde{t}_{a/d}$ (Gao et al., 2009) as shown in Eq. (5.2) (see Figure 5-1).

$$\tilde{sc}_k = sc_k + \tilde{vc}_k \cdot \tilde{t}_{a/d} \quad (5.2)$$

The shape of the membership function of $\tilde{t}_{a/d}$ is also modelled as a symmetrical triangular fuzzy number whose core is 0 and whose support is the constant Δt as shown in Eq. (5.3) (see Figure 5-1).

$$\mu_{\tilde{t}_{a/d}}(y) = \begin{cases} 0 & \text{if } y \leq -\frac{\Delta t}{2} \\ \frac{2}{\Delta t} \cdot y + 1 & \text{if } -\frac{\Delta t}{2} < y \leq 0 \\ -\frac{2}{\Delta t} \cdot y + 1 & \text{if } 0 < y \leq \frac{\Delta t}{2} \\ 0 & \text{if } y > \frac{\Delta t}{2} \end{cases} \quad (5.3)$$

Triangular shape is used for the membership functions of \tilde{vc}_k and $\tilde{t}_{a/d}$. However, the optimisation algorithm proposed is also valid for other membership functions (linear, exponential, hyperbolic, piece-wise linear (Xiao et al., 2012) and S-shaped (Chang, 2010; Vasant et al., 2012)).

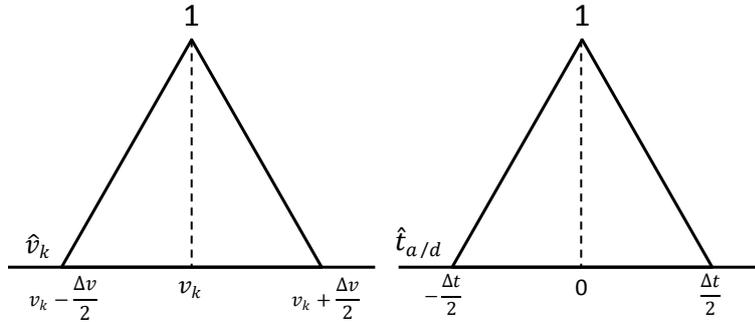


Figure 5-1. Fuzzy holding speed and fuzzy anticipation/delay time

5.3. RISK OF DELAY IN ARRIVAL

A measure is proposed to calculate, for each solution, its risk of delay in arrival. This measure is based on the evolution of the time margin of the train.

The commercial running time between two stations is designed using a margin in timetable, also called slack time, to deal with delays. Margin is calculated at each position using Eq. (5.4).

$$Margin(s) = (AH_{com} - H_{current}) - AT_{flat}(s, v) \quad (5.4)$$

where $Margin(s)$ is the margin of the train at position s , AH_{com} is the commercial arrival time which is a fixed value given by the timetable, $H_{current}$ is the current time and $AT_{flat}(s, v)$ is the running time of the flat-out driving, i.e. the fastest possible driving, from the current position of the train (s) and the current speed of the train (v).

In other words, margin is the maximum delay that the train can recover applying the fastest speed profile to arrive on-time at the next station. Eco-driving consists in using the margin to perform a slower speed profile than the flat-out driving saving energy (Cucala et al., 2012b).

The margin consumption rate of a speed profile determines the sensitivity of this driving to arrive late at the next station. If the margin is quickly consumed during the journey, the train could perform an efficient speed profile but it might not be able to react to unexpected delays. Contrary, the speed profiles that retain the margin up to the end of the journey might consume more energy but are more robust to delays during the train travel. Therefore, the margin consumption rate of a speed profile is related to the risk of delay in arrival.

The measure proposed is called RD and can be calculated using the following formula:

$$RD = 1 - \frac{\int_{s_0}^{s_{end}} Margin(s) \cdot ds}{Margin_{tot} \cdot (s_{end} - s_0)} \quad (5.5)$$

where $Margin_{tot}$ is the total margin at the beginning of the journey, s_0 is the position of the train at the beginning of the optimisation execution and s_{end} is the position of the train at the end of the journey.

The calculation of RD is based on the integral of the margin over the space. The result of the integral depends on the amount of margin retained by the train at each position, i.e. depends on the speed profile applied. The value of this integral is greater as lower is the margin consumption rate. Besides, the integral is normalised dividing it by the initial margin multiplied by the total running distance. Notice that the margin consumed by the train during the journey cannot be recovered, so the initial margin is the maximum one. Finally, the normalised value of the integral is subtracted from one to give higher values of RD to solutions with higher risk of delay in arrival. Thus, the RD will be equal to zero for the flat-out driving since it does not consume margin during the journey ($\int_{s_0}^{s_{end}} Margin(s) \cdot ds = Margin_{tot} \cdot (s_{end} - s_0)$). The other possible speed profiles will always perform higher running times and, as a consequence, the value of RD will be greater than zero.

The value of RD can be understood as the area contained over the margin curve and the straight line with margin value equal to the initial margin in a margin vs position graph. In an example of a graphical representation of RD is shown.

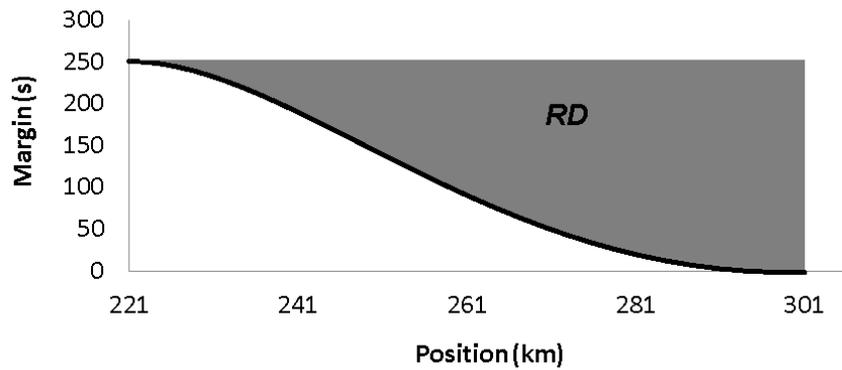


Figure 5-2. Graphical representation of RD

5.3.1. RELATIONSHIP OF THE RISK OF DELAY IN ARRIVAL WITH THE RUNNING TIME AND THE ENERGY CONSUMPTION

The optimisation process detailed in Section 5.5 has as objective the minimisation of running time, energy consumption and risk of delay. It is well-known that the energy consumption and running time are conflicting objectives. That means that improving one objective causes that the other objective worsens.

It is necessary to demonstrate the existence of a Pareto front to propose a three-objective optimisation with running time, energy consumption and risk of delay in arrival as objectives. In other words, it is necessary to demonstrate the existence of non-dominated solutions in the problem. Non-dominated solutions are those that cannot be improved in all the objectives at the same time.

To demonstrate the existence of non-dominated solutions in the three dimensions solution space proposed, the simplest case of eco-driving calculation is used. This case is the optimisation of a train driving on a flat track, with a constant speed limit of 300 km/h and neglecting the effect of the regenerated energy. In all the following cases, the commercial running time established in the timetable is AH_{com}^1 .

Optimal control theory establishes that the optimal driving, for an objective running time RT^1 , consists of a maximum acceleration phase, a holding speed phase at certain speed, a coasting phase and a maximum braking phase (Howlett, 1990). Assume that the speed profile 1 presented in Figure 5-3 is the solution that minimises the energy consumption for a running time $RT^1 = AH_{com}^1$ using a C_m with $n_s = 2$, i.e. one holding speed command followed by a coasting command. This speed profile has associated a running time $RT^1 = AH_{com}^1$, an energy consumption EC^1 and a risk of delay in arrival RD^1 .

It is obvious that the risk of delay in arrival of speed profile 1 can be improved by reducing the objective running time. Thus, the speed profile 2 is obtained, which is represented in Figure 5-3 and whose running time is $RT^2 < AH_{com}^1$. Solution 2 improves the risk of delay in arrival and the running time of solution 1. However, the energy consumption is increased and, for that reason, solutions 1 and 2 are non-dominated.

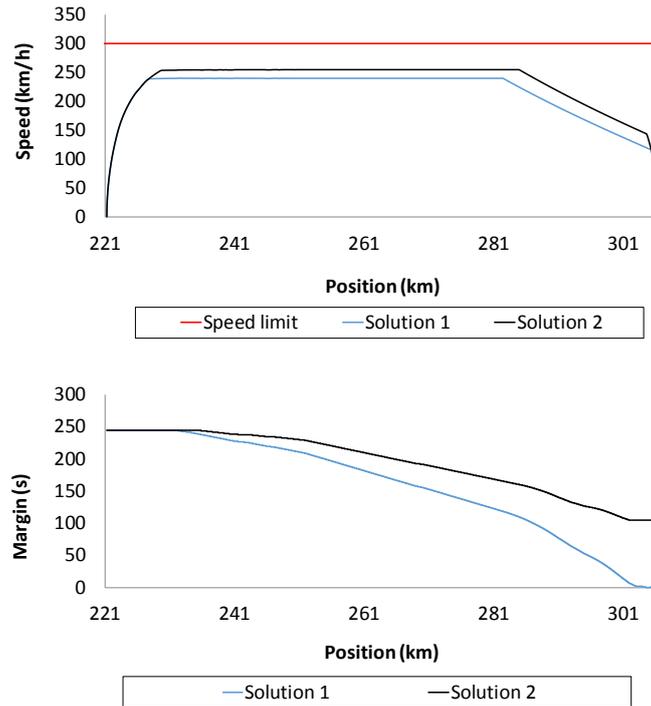


Figure 5-3. Comparison of two energy optimal speed profiles with different running time

In the previous example, the lower the running time, the lower the risk of delay in arrival. However solutions can be found with the same running time and different values of RD . For instance, it can be easily found a solution with the minimum risk of delay in arrival that meets the objective running time AH_{com}^1 . This solution consists in, using a C_m with $n_s = 2$, a holding speed command at maximum speed followed by a coasting command at the position needed for achieving a running time equal to AH_{com}^1 . Thus, speed profile 3 is obtained and represented in Figure 5-4. Comparing solutions 1 and 3 it can be found that both have the same running time by definition. Besides, the risk of delay in arrival of solution 3 is lower than result of solution 1 because solution 3 retains the initial margin up to the coasting phase, while solution 1 is consuming the margin along the journey reducing it constantly as can be seen in Figure 5-4. Looking at the energy consumption, it can be found that solution 3 demands more energy given that solution 1 is the energy optimal solution for the running time AH_{com}^1 . Therefore, it can be concluded that solutions 1 and 3 are non-dominated solutions because speed profile 3 improves the risk of delay in arrival of solution 1, maintaining the same running time but worsening the energy consumption. Moreover, it can be concluded that the risk of delay in arrival does not depend directly on the running time of the solution.

It is not possible to find a solution that dominates speed profile 3. To find a solution with RD lower than solution 3, it is necessary to build a solution 4 that has a larger holding speed phase at maximum speed. Solution 4 will present higher energy consumption compared with solution 3 in any case. Therefore, although running time and risk of delay are improved, the energy consumption objective worsens.

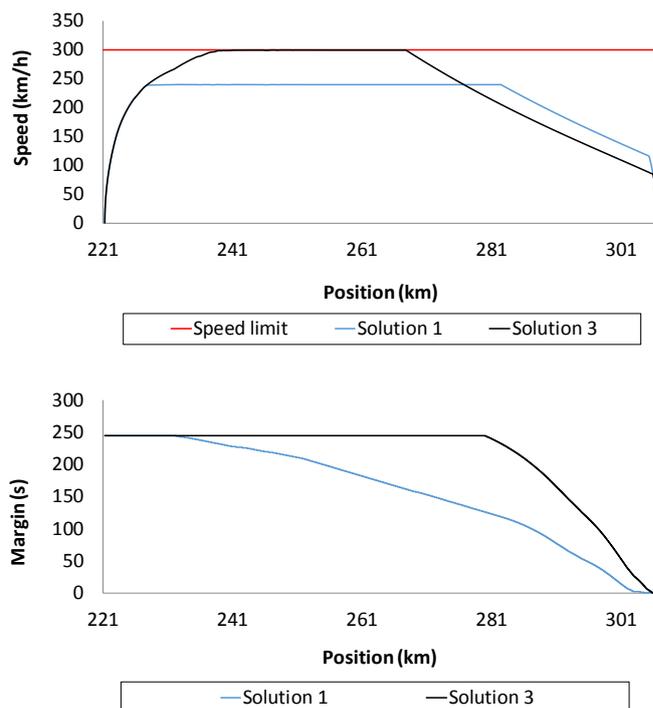


Figure 5-4. Comparison of two speed profiles with the same running time but different risk of delay in arrival

Varying the maximum holding speed and the start position of coasting a continuous solution space can be found with the same running time that solution 1 and with different values of RD .

Finally, a Pareto surface can be obtained repeating the previous procedure for other running times for Solution 1. This demonstrates the existence of non-dominated solutions in the three dimensions of the problem. These solutions are equally valid and the selection of one of them depends of the preferences of the decision maker.

5.4. DYNAMIC NSGA-III WITH FUZZY PARAMETERS (DNSGA-III-F) FOR THE ON-LINE DRIVING REGULATION PROBLEM

In this section, the train driving is optimised considering 3 objectives: running time, energy consumption and risk of delay in arrival. These objectives are conflicting, therefore the result of the algorithm will be a three-dimensional Pareto surface. The Pareto front is composed of a set of optimal solutions, where each solution is a command matrix. On the other hand, apart from obtaining the Pareto front, the proposed algorithm has to track its variations during the train movement.

The running time, energy consumption and risk of delay in arrival of a solution, i.e. of a set of driving commands, depend on the position and speed of the train. Figure 5-5 shows an example of the variation of the Pareto front during a train travel. This figure depicts the Pareto front obtained by means of an optimisation and three possible speed profiles that the train can perform at 3 different instants during a train journey. During this journey, the train is following a nominal speed profile and the Pareto surface is

recalculated as the train moves. The nominal speed profile is always represented by a blue line in the speed vs space graphs. Instant 1 is the departure, thus, the position of the train is the departure station and the speed of the train is 0. From this initial conditions, multiple speed profiles can be performed with different values of running time, energy consumption and *RD*. In Figure 5-5, the optimal speed profiles for this instant are represented by the Pareto surface of blue dots and three examples of these speed profiles are shown in the green, blue and brown lines in the speed vs position graph of Instant 1.

In Instant 2, the train is at 250 km position with a speed of 240 km/h. From these initial conditions, a variety of speed profiles can be applied that are calculated for a shorter section of the line (from the current position up to the arrival station). The Pareto solutions of Instant 2 are represented by the Pareto surface of orange dots and three examples of these speed profiles are shown in the green, blue and brown lines in the speed vs position graph of Instant 2. These solutions are different from the solutions obtained in Instant 1 because most of Instant 1 optimal speed profiles have a speed different from 240 km/h at 250 km position. Later, in Instant 3 the train is at 270 km position travelling at 230 km/h. From these conditions, the number of possible speed profiles that can be applied are reduced because the journey length is shorter. This is shown in the Pareto surface represented by yellow dots, which is smaller because the results in running time, energy consumption and risk of delay are lower and because the variety of possibilities is reduced.

As can be seen, the running time and the energy consumption of the Pareto solutions are lower at later time instants because the remaining distance to arrival decreases.

Furthermore, as the train moves the maximum difference in running time, energy consumption and risk to future delays are reduced. The reason is that as the running distance decreases, the number of different speed profiles that the train can perform is reduced.

Considering the above, the Pareto front of the possible solutions varies continuously during the train travel because the position and the speed of the train varies continuously. Thus, it is necessary to discretise the continuous changes of the problem to perform successive optimisations in order to track Pareto front variations. With this aim, the train travel is divided into periods of t_c seconds. During each period, the conditions of the problem are considered as constant and an execution of the algorithm will be carried out. Each execution of the algorithm will use the position and the speed of the train at the beginning of the period as initial conditions of the simulation model that evaluates the solutions. Thus, at the end of each period the Pareto front is updated to the changing conditions of the train.

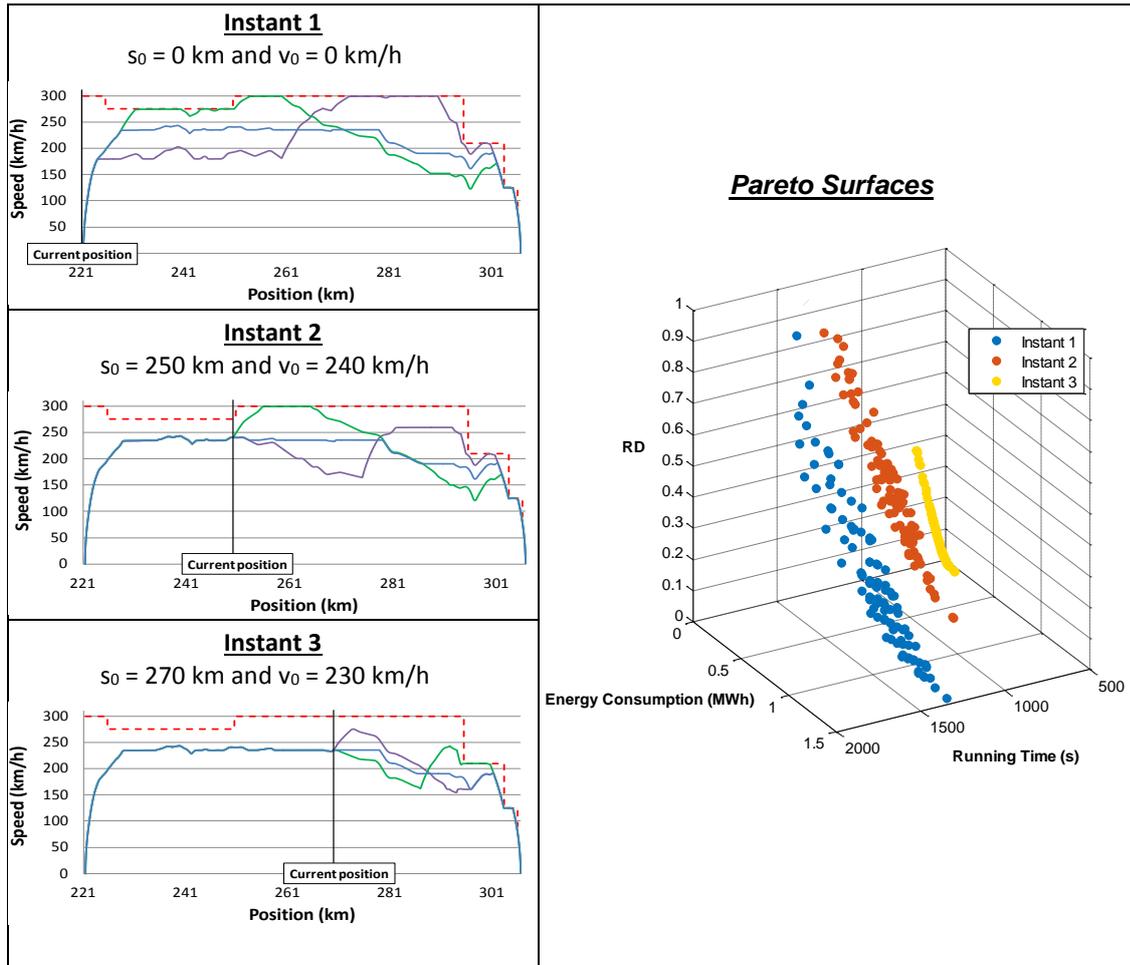


Figure 5-5. Variation of the Pareto front and some possible speed profiles during a train travel

To accelerate the optimisation process and improve the quality of the solutions obtained, a re-initialisation mechanism is included in the algorithm to take advantage of the information of previous executions. It is expected that the problem conditions will change smoothly during the train travel. As a consequence, the Pareto front obtained at the end of a period should be similar to the one obtained in the previous period. Therefore, it is useful to initialise each new execution of the optimisation with the optimal solutions obtained in the previous one. This way, the algorithm starts from a population close to the real optimal front reducing dramatically the calculation time needed to converge.

Dynamic Multi Objective Optimisation (DMOO) algorithms apply this strategy to the reduce de calculation time allowing a frequent update of the Pareto Front.

As described in section 3.2, the driving parameters of a solution X are modelled as fuzzy numbers, thus its associated running time, energy consumption and risk of delay are fuzzy numbers.

This way, the problem is modelled as a DMOO problem with fuzzy parameters, and the objective function includes three objectives (Eq.(5.6)):

$$\text{Min } \tilde{f}(X) = (\tilde{RT}(X), \tilde{EC}(X), \tilde{RD}(X)) \quad (5.6)$$

where $\widetilde{RT}(X)$, $\widetilde{EC}(X)$ and $\widetilde{RD}(X)$ are the fuzzy running time, energy consumption and risk of delay in arrival of the solution X . Minimizing these three fuzzy objectives demands the application of fuzzy dominance concepts which are explained in Section 5.4.2. In addition, a procedure based on α -cut arithmetic is proposed in Section 5.4.4 to reduce the calculation time of the algorithm.

A new dynamic NSGA-III algorithm with fuzzy parameters DNSGA-III-F to obtain the Pareto surface of the possible speed profiles during a high-speed train travel is proposed.

5.4.1. DNSGA-III-F FLOWCHART

The crisp NSGA-III algorithm (Deb and Jain, 2014) is a version of the well-known NSGA-II algorithm to solve many-objective problems. The NSGA-III algorithm has shown its suitability when solving problems of three or more objectives. The basic framework of NSGA-III is similar to the previous NSGA-II. A population of solutions evolves through iterations and the individuals of this population are classified into groups with the same domination level. The domination level of an individual is calculated as the number of solutions that dominate that specific individual. Notice that the Pareto front found by the algorithm will be the set of solutions with dominance level equal to zero. In NSGA-III, elitism is introduced in each group using a spread operator based on reference points (instead of the crowding distance operator used in NSGA-II). Solutions with the same domination level are sorted attending to its distance to a predefined reference point. This way, the diversity of solutions is promoted obtaining well-spread Pareto fronts.

A new dynamic algorithm with fuzzy parameters is proposed based on the static NSGA-III, using the Pareto front of an execution as the initial population of the next one, and including the uncertainty associated to the train driving.

The flowchart of the proposed DNSGA-III-F algorithm is shown in Figure 5-6.

The first execution ex is launched before the train departure and the initial parent population P_0^1 is generated randomly (size n_{pop}). The initial position of the train s_0 is the departure station s_{dep} , and the initial speed v_0 is zero. This parent population is evaluated simulating each solution. The offspring population Q_{it}^{ex} is generated at each iteration it applying cross-over and mutation operators to individuals in the parent population. The cross-over operator constructs a new solution from two individuals of the parent population. Thus, the C_m of the new solution will be formed taking part of the C_m of a parent solution and part of the C_m of the other parent solution. On the other hand, the mutation operator takes the C_m of a parent solution and varies randomly the value of a command matrix item. The mutation and crossover operators applied in this algorithm were proposed in (Sicre, 2013).

This offspring population is evaluated and a result population R_{it}^{ex} is generated joining P_{it}^{ex} and Q_{it}^{ex} . The fuzzy dominance is calculated to obtain the domination level of each individual in R_{it}^{ex} , and a selection criterion is applied to select the individuals that will survive SP_{it}^{ex} for the next generation to form the next parent population P_{it+1}^{ex} .

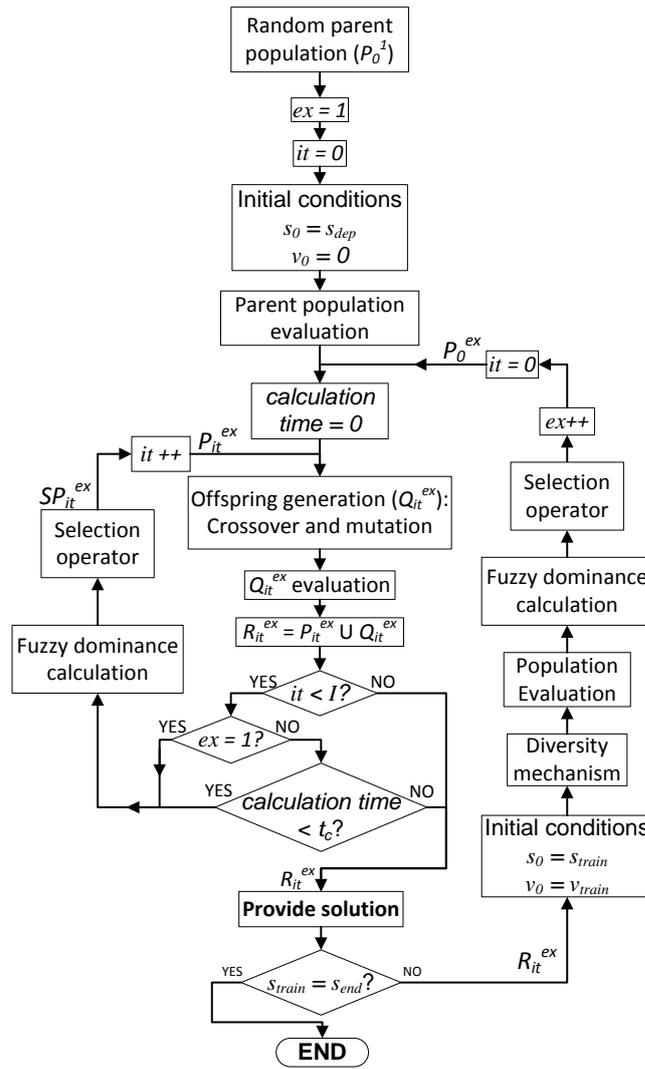


Figure 5-6. Flow chart of DNSGA-III-F

The stopping criterion of the first execution of the algorithm is given by a maximum number of iterations it_{max} , and it is executed while the train is stopped at the departure station. The stopping criterion of the following executions is the computation time t_c .

When a new execution ex of the algorithm starts, the previous result population R_{it}^{ex} at $ex - 1$ is used as the initial population, and a diversity mechanism is applied to increase the searching ability of the population. Two mechanisms are used in the DNSGA-III-F algorithm to be compared, based on those proposed by Deb et al. in (Deb et al., 2007):

- Random Mechanism: a percentage $\zeta\%$ of the result population is replaced by randomly created individuals.
- Mutation Mechanism: a percentage $\zeta\%$ of the result population is replaced by mutated solutions of randomly selected individuals of the population.

Then, this population is re-evaluated using the initial conditions of the current execution (current position strain and speed v_{train} of the train). Fuzzy dominance is calculated and the selection criterion is applied to obtain the next parent population.

The calculation time required for each iteration depends mostly on the calculation time of the simulator. Solutions are evaluated at each iteration by means of the simulator, thus, compared with the computational cost of the population evaluation, the calculation time of the rest of the modules can be practically neglected.

The value of the parameter t_c (maximum calculation time) must be adjusted carefully. The optimisation is performed repeatedly at each time period t_c . Therefore, shorter calculation times provide better tracking of the current situation of the train because the Pareto front is updated more often. However, the number of iterations that the optimisation algorithm can use to find a solution is lower. On the contrary, if the value of t_c is large, the algorithm will have many iterations to find a good Pareto front but it could be outdated because the conditions of the problem may change significantly.

When the computation time t_c is completed, the new Pareto surface obtained is provided to the Algorithm to update driver's commands which is in charge of deciding when a new set of driving commands has to replace the current ones.

In the following subsections, a more detailed description about the fuzzy dominance calculation and the selection criterion will be provided.

5.4.2. FUZZY DOMINANCE

The selection criterion of the individuals that will survive at the next generation is based on the domination level of each solution, as previously indicated. The domination level of a solution is calculated as the number of solutions that dominate that point.

As the solutions are defined by means of fuzzy driving commands, the resulting 3 objectives of each solution are fuzzy numbers as well, and the dominance concept has to be expressed in fuzzy terms. An individual A fuzzy dominates another B if it fuzzy dominates for each objective i:

$$\tilde{A} < \tilde{B} = \cap \tilde{A} <_j \tilde{B} \quad (5.7)$$

where $j = 1, 2, 3$ and the three objectives are the fuzzy running time, energy consumption and risk of delay in arrival.

Therefore, the fuzzy dominance of a solution A over a solution B can be defined as the intersection of the three objectives as $\tilde{RT}(B) > \tilde{RT}(A)$ and $\tilde{EC}(B) > \tilde{EC}(A)$ and $\tilde{RD}(B) > \tilde{RD}(A)$ (Carvajal-Carreño et al., 2014).

In order to calculate the fuzzy dominance, the min t-norm is used, and the fuzzy numbers $\tilde{RT}(A)$, $\tilde{RT}(B)$, $\tilde{EC}(A)$, $\tilde{EC}(B)$, $\tilde{RD}(A)$ and $\tilde{RD}(B)$ are compared in terms of the necessity measure (Dubois and Prade, 1983), to calculate the strong fuzzy dominance (Carvajal-Carreño et al., 2014). Thus, the solution A fuzzy dominates B if $N(\tilde{RT}(B) > \tilde{RT}(A)) \geq N_d$ and $N(\tilde{EC}(B) > \tilde{EC}(A)) \geq N_d$ and $N(\tilde{RD}(B) > \tilde{RD}(A)) \geq N_d$.

In Eq. (5.8), the fuzzy dominance of A over B with the min t-norm is represented.

$$\min \begin{pmatrix} N(\widetilde{RT}(B) > \widetilde{RT}(A)) \\ N(\widetilde{EC}(B) > \widetilde{EC}(A)) \\ N(\widetilde{RD}(B) > \widetilde{RD}(A)) \end{pmatrix} \geq N_d \quad (5.8)$$

where N_d is the required level of necessity for the fuzzy comparison.

Notice that the necessity measure of $\widetilde{RT}(B) > \widetilde{RT}(A)$ is equal to one minus the possibility of $\widetilde{RT}(B) \leq \widetilde{RT}(A)$ as shown in Eq. (5.9).

$$N(\widetilde{RT}(B) > \widetilde{RT}(A)) = 1 - \Pi(\widetilde{RT}(B) \leq \widetilde{RT}(A)) \quad (5.9)$$

Thus, the equation $N(\widetilde{RT}(B) > \widetilde{RT}(A)) > N_d$ can be written as:

$$\Pi(\widetilde{RT}(B) \leq \widetilde{RT}(A)) < 1 - N_d \quad (5.10)$$

A graphical representation of the previous relation is shown in Figure 5-7. This figure shows that A dominates B regarding the running time objective because $N(\widetilde{RT}(B) > \widetilde{RT}(A)) > N_d$

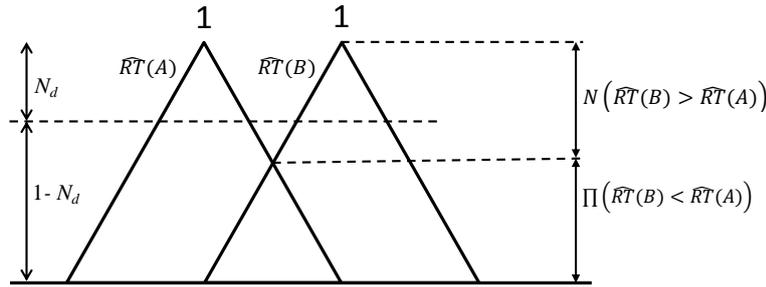


Figure 5-7. Representation of fuzzy dominance of A over B.

5.4.3. SELECTION OPERATOR

The DNSGA-III-F algorithm uses a set of reference points to promote the spread of solutions in the population. This reference set can be provided by the users based on their knowledge or can be generated in a structured manner. In this work Das and Dennis' systematic approach is applied (Das and Dennis, 1998). This method places the reference points on a normalised hyper-plane, which is equally inclined to all objective axes and has an intercept of one on each axis. The number of points (H) obtained depends on the number of objectives (no) and the number of divisions desired along each objective (nd):

$$H = \binom{no + nd - 1}{nd} \quad (5.11)$$

In Figure 5-8 an example with $nd = 4$ and 15 reference points is shown.

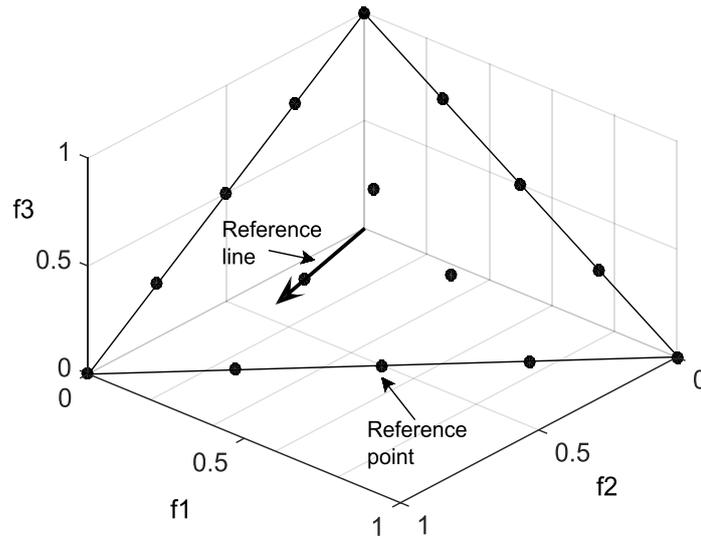


Figure 5-8. Fifteen reference points placed on a normalised hyper-plane using Das and Dennis' systematic approach

The niche preservation operator associates the population individuals with the reference points. Before doing that, individuals' objective values are normalised as shown in Eq. (5.12).

$$nf_q^j = \frac{f_q^j - z_q^{min}}{b_q} \quad q = 1, 2, 3 \quad (5.12)$$

where f_q^j is the original value in the objective q of the individual j and nf_q^j is its normalised result in the objective q . The value of z_q^{min} is the minimum value of the objective q found in the set of individuals where niche preservation operator is applied. The value of b_q is equal to the maximum value of $f_q^j - z_q^{min}$ found in the set of individuals.

Once the normalisation is completed, the individuals in SP_{it} are associated to a reference point. The association needs to generate the reference line of each reference point. Reference lines are built joining the reference point with the origin. Thereafter, for each individual in F_t , the perpendicular distance of the normalised objectives to the reference lines are calculated. An individual is associated to a reference point if its reference line is the closest to this normalised population member.

When the association process is finished, the niche count of each reference point r is calculated (nc_r) as the number of individuals of SP_{it} associated with the reference point r . Then, the selection operator works as follows:

The reference points with the minimum niche count are identified and, among then, a reference point r is randomly selected.

If there is one or more individuals in F_l associated to the reference point selected, the individual with the shortest distance to its reference line is included in the set of solutions that will survive. The niche count of the reference point selected is incremented by one.

If there is no individual in F_l associated to the reference point selected, it is not considered.

This process is repeated until the size of the set of surviving solution is equal to n_{pop} .

5.4.3.1. ADAPTIVE EVOLUTION OF REFERENCE POINTS

After the application of the niche preservation operation, the parent population for the following iteration is created and the niche count of each reference point is updated. As the number of reference points is similar to n_{pop} , it is expected that $nc_r = 1$ for all the reference points, i.e. all the reference points have associated at least one population member. However, it is possible that some reference points r have $nc_r \geq 2$. That means that other reference points do not have an associated member. If there are many useless reference points, a certain fraction of the population is pushed to be spread but the rest of the population will be randomly selected, which would affect to the diversity of the Pareto front.

To avoid this, the adaptive evolution of reference points proposed in (Jain and Deb, 2014) is applied in DNSGA-III-F. A simplex of 3 points is included for every reference point with $nc_r \geq 2$ using the Das and Dennis' method with $nd = 1$ and having a distance between points equal to the distance between two consecutive reference points in the original simplex as shown in Figure 5-9. The centroid of every simplex added is a reference point with $nc_r \geq 2$.

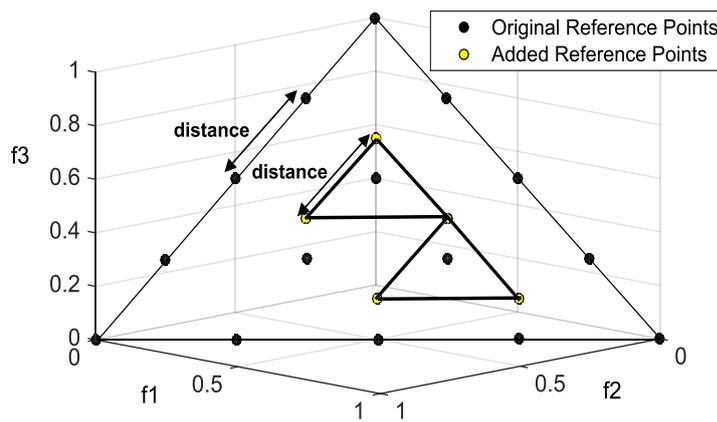


Figure 5-9. Addition of reference points

After that, the included reference points that lay outside the boundaries of the original reference set are eliminated. Furthermore, if there are coincident reference points, only one is retained and the rest are eliminated.

5.4.4. DNSGA-III-F RESOLUTION

To improve the computational time of the algorithm, it is possible to apply the α -cut arithmetic if the three objectives are monotonous functions of the command matrix parameters (Chanas et al., 1984).

To determine if a solution A dominates a solution B, the algorithm has to calculate the upper or the lower limits of the α -cuts of the fuzzy RT , EC and RD (see Figure 5-7). To obtain the α -cuts efficiently, it is necessary to determine whether RT , EC and RD are increasing or decreasing functions of the speed command vc_k and the anticipation/delay time $t_{a/d}$.

Regarding the holding speed command, it is obvious that the running time always decreases as the trains runs faster, thus, the running time is a decreasing function of the speed. Similarly, the faster the speed of the train, the lower the risk of delay (decreasing function). The energy consumption is an increasing function of the value of holding speed command because of the increase in the movement resistance.

Regarding the anticipation/delay time in the application of a new command speed $t_{a/d}$, two cases can be distinguished:

- Case A. The next holding speed command is higher than the previous one ($vc_k < vc_{k+1}$). Therefore, the higher the $t_{a/d}$, the later the driving command will be updated and the higher will be the time using a slower driving command. As a consequence, the higher $t_{a/d}$, the higher the running time and risk of delays and the lower the energy consumption will be obtained.
- Case B. The next holding speed command is lower than the previous one ($vc_k > vc_{k+1}$). Therefore, the higher $t_{a/d}$, the later the driving command will be updated and the higher will be the time of driving using a faster driving command. As a consequence, the higher $t_{a/d}$, the lower the running time and risk of delays and the higher the energy consumption will be obtained.

Taking into account the previous relationships, the upper limit of the α -cuts of the fuzzy running time $RT_{\bar{\alpha}}$ can be calculated simulating with the lower limit of the same α -cut of the fuzzy speed $vc_{k,\underline{\alpha}}$ (as RT is decreasing with v), and in Case A, with the upper limit of the anticipation/delay time $t_{a/d, k, \bar{\alpha}}$ (increasing function), while in case B with the lower limit $t_{a/d, k, \underline{\alpha}}$ (decreasing function):

$$RT_{\bar{\alpha}} = F(vc_{k,\underline{\alpha}}, t_{a/d, k, \bar{\alpha}} \text{ in case A, } t_{a/d, k, \underline{\alpha}} \text{ in case B}) \quad (5.13)$$

On the other hand, the lower limit of the α -cut of the fuzzy running time $RT_{\underline{\alpha}}$ can be calculated as:

$$RT_{\underline{\alpha}} = F(vc_{k,\bar{\alpha}}, t_{a/d, k, \underline{\alpha}} \text{ in case A, } t_{a/d, k, \bar{\alpha}} \text{ in case B}) \quad (5.14)$$

Similarly, the upper limits and the lower limits of the α -cuts of the fuzzy energy consumption and the fuzzy risk of delay can be calculated as indicated in equations (5.15), (5.16), (5.17) and (5.18):

$$EC_{\bar{\alpha}} = F(vc_{k,\bar{\alpha}}, ta_{/d, k,\bar{\alpha}} \text{ in case A, } ta_{/d, k,\bar{\alpha}} \text{ in case B}) \quad (5.15)$$

$$EC_{\underline{\alpha}} = F(vc_{k,\underline{\alpha}}, ta_{/d, k,\bar{\alpha}} \text{ in case A, } ta_{/d, k,\underline{\alpha}} \text{ in case B}) \quad (5.16)$$

$$RD_{\bar{\alpha}} = F(vc_{k,\underline{\alpha}}, ta_{/d, k,\bar{\alpha}} \text{ in case A, } ta_{/d, k,\underline{\alpha}} \text{ in case B}) \quad (5.17)$$

$$RD_{\underline{\alpha}} = F(vc_{k,\bar{\alpha}}, ta_{/d, k,\underline{\alpha}} \text{ in case A, } ta_{/d, k,\bar{\alpha}} \text{ in case B}) \quad (5.18)$$

The previous relationship can be seen graphically in Figure 5-10.

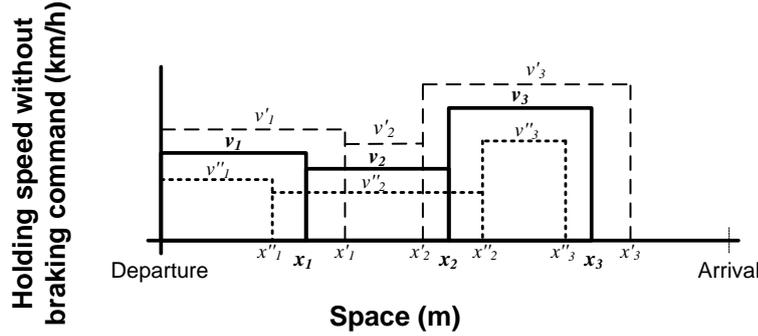


Figure 5-10. Fastest and slowest case for a given value of α

The resolution of DNSGA-III-F algorithm makes use of the α -cut arithmetic to improve the computational efficiency. It can be described following these steps:

1. Generation of the initial population of the algorithm first execution $ex = 1$, launched at the departure station. This initial parent population (P_0^1) is composed by n_{pop} individuals generated randomly, and the following constraints are applied to ensure the comfort of the resulting speed profiles. Eq. (5.19) establishes a lower bound (vc_{min}) for driving commands to avoid low speed during the train travel. On the other hand, Eq. (5.20) establishes the minimum position (sc_{min}) from where the train can start to coast before the final braking. The objective of these limitations is to avoid the train being driven at low speeds that can be perceived as unpleasant by passengers.

$$vc_k(X) > vc_{min} \quad (5.19)$$

$$sc_{n_s-1}(X) > sc_{min} \quad (5.20)$$

2. The initial conditions of the simulation model in the first execution of the algorithm are the position of the departure station and a zero value for the initial speed.
3. The parent population is evaluated using the simulator. For each individual, given a necessity level for the dominance requirement N_d , the upper and lower limits of α -cuts for $\alpha = 1 - N_d$ of the fuzzy running time, energy consumption and risk to delay in arrival are calculated by means of equations (5.13)-(5.18). The cores of these fuzzy numbers are calculated in a similar way.
4. Generation of the offspring population. At each iteration it, the offspring population (Q_{it}^{ex}) is composed of n_{off} individuals generated applying crossover and mutation operators to individuals in the parent population P_{it}^{ex} . Restrictions (5.19) and (5.20) are also applied in the offspring generation process.

5. The offspring population is evaluated using the simulator as described in step 3 with the parent population.
6. Generation of the result population. The result population of iteration it (R_{it}^{ex}) is generated joining P_{it}^{ex} and Q_{it}^{ex} .
7. Fuzzy dominance. To calculate domination level of each individual, the members of the population are compared in the following way.

Given two solutions A and B, A dominates B if $RT_{\bar{\alpha}}(A) < RT_{\bar{\alpha}}(B)$ and $EC_{\bar{\alpha}}(A) < EC_{\bar{\alpha}}(B)$ and $RD_{\bar{\alpha}}(A) < RD_{\bar{\alpha}}(B)$ comparing lower and upper limit with $\alpha = 1 - N_d$, where is the necessity level imposed for the dominance comparison. Besides, solutions with high running time are discarded during the algorithm execution to reduce the search space. To filter the solutions with higher running time, the algorithm checks if the core of the fuzzy running time associated to each solution exceeds a threshold. In this case, the algorithm evaluates that solution as a dominated solution and assigns the worst possible score to its domination level. The threshold is given by the flat-out running time plus two times the remaining time margin.

8. Generation of next parent population P_{it+1}^{ex} . The selection operator described in section 5.4.3, based on the niche count, is applied to the result population to obtain the members of the population that will survive at the next iteration.
9. Execution stopping criteria. Two stopping criteria are defined:
 - The first execution of the algorithm is performed before the departure of the train. Therefore, the stopping criterion applied is a maximum number of iterations it_{max} .
 - The rest of the executions of the algorithm are performed online during the train travel. Therefore, to track adequately the Pareto front changes, a maximum calculation time t_c is applied as the stopping criterion.
10. Initialisation of the next algorithm execution. Once the execution of the algorithm is finished, the Pareto surface is updated and the initial conditions for the simulation model are set to the current position and speed of the train. A diversity mechanism, Random Mechanism or Mutation Mechanism, is applied in the last iteration result population R_{it}^{ex} . The population obtained is simulated to obtain the results of the individuals under the new initial conditions. Thereafter, fuzzy dominance and selection operator are applied to generate the initial parent population P_0^{ex} of the new algorithm execution.
11. The process is repeated from step 4 up to the arrival station.

5.5. ALGORITHM TO UPDATE DRIVER'S COMMANDS

In low traffic density high-speed lines (like the Spanish ones) delays are less frequent than in urban lines. They are caused typically by temporary speed limitations, due to civil or maintenance works, or to adverse weather conditions.

When a train is affected by these situations, the nominal driving has to be updated to a faster speed profile to arrive on time at the next station.

In the previous section, it was described that the Pareto surface of possible speed profiles is calculated by DNSGA-III-F every cycle of t_c seconds. Despite that, the set of driving commands to be applied by the driver are not changed unless it is necessary to perform a different speed profile, i.e. unless a significant delay arises (over a threshold value). Thus, the driving commands are not continuously changed.

When the train delay exceeds the threshold, flat-out driving is applied to check when it is possible to reduce the delay. When the delay starts to decrease, the system selects the new driving commands to be applied from the last updated Pareto front. The speed profile selected is the one that makes up the delay in arrival balancing the energy consumption and the risk of future delays. Therefore, an online decision maker is needed to select the most appropriated driving commands. When a new speed profile is needed, the algorithm to update the driver's commands calculates the objective running time. Thereafter, it selects from the Pareto front the set of solutions whose running time is equal or lower to the objective running time.

The running time associated to each solution is a fuzzy number as was explained before. A necessity of punctuality level N_p is configured to compare the fuzzy running time with the objective running time. Therefore, the set of solutions that fulfils the objective running time are those that meet the following comparison:

$$RT_{\overline{\alpha p}}(A) \leq T_{Objective} \quad (5.21)$$

where $RT_{\overline{\alpha p}}(A)$ is the upper limit of the α -cut of running time of a solution A from the Pareto front corresponding to the necessity of punctuality ($\alpha p = 1 - N_p$). $T_{Objective}$ is the objective running time.

Finally, the selected solution is the one with the minimum value for the following fitness function:

$$fitness = w_{ec} \cdot EC(A) + w_{rd} \cdot RD(A) \quad (5.22)$$

where $EC(A)$ and $RD(A)$ are the core values of the fuzzy energy consumption and risk of delay in arrival of a solution A . The constants w_{ec} and w_{rd} are the weighting factors of energy consumption and risk of delay in arrival.

Parameters w_{ec} and w_{rd} are configured by the operator based on their preferences and the state of the traffic complying $w_{ec} + w_{rd} = 1$.

5.6. CASE STUDY

DNSGA-III-F algorithm has been tested in a case study using real data from a Spanish high-speed line. As in Chapter 4, the stretch analysed runs from Calatayud to Zaragoza and its length is 85.4 km. The high-speed train is a Talgo-Bombardier class 102. This train has two motors of 8 MW and 200 kN of maximum traction effort. The train is 200 m long and its empty weight is 324 t. The operational restrictions v_{min} and x_{min} for the driving commands are set to 150 km/h and 50 km respectively to avoid low speed phases in the middle of the journey.

The tests that have been carried out to check the algorithm performance are based on two threads of simulation. The main thread simulates the whole journey of the train matching the simulation clock with the real-time clock. The secondary thread carries out the dynamic optimisation algorithm that makes many fast simulations to evaluate possible solutions. Both threads were executed using an Intel Core i7-2600 CPU@3.4 GHz and 16GB of RAM.

The DNSGA-III-F tuned parameters used in the case study are presented in Table 5-1. The number of sections in C_m has been set to 4. Therefore, the algorithm has to optimise 8 variables that are contained in the command matrix of the possible solutions. On the other hand, the configuration parameters for the fuzzy numbers are presented in Table 5-2.

Parent population size (n_{pop})	Offspring population size (n_{off})	Number of crossover	Number of mutations	Number of sections in C_m (n_s)	Diversity mechanism parameter (ζ)	Calculation time (t_c)
100	50	20	30	4	15 %	60 s

Table 5-1. Tuned parameters of DNSGA-III-F algorithm

Support of fuzzy holding speed (Δv)	Support of fuzzy anticipation/delay time (Δt)	Necessity for dominance comparison (N_d)	Necessity of punctuality (N_p)
3 km/h	2 s	0.6	0.6

Table 5-2. Configuration parameters of fuzzy numbers

5.6.1. DNSGA-III-F CONVERGENCE ANALYSIS

To analyse the performance of the algorithm presented in this paper, the train is simulated to face a disturbance that leads to accumulate a delay. This disturbance is an unexpected temporary speed limitation of 90 km/h located between 229 km and 230 km. If the driver does not change the nominal speed profile calculated at the beginning of route, the train would arrive at the next station 1 minute and 20 seconds late. Figure 5-11 represents the nominal speed profile affected by the unexpected speed limitation.

DNSGA-III-F has been applied to solve this optimisation problem and the Random Mechanism and Mutation Mechanism are compared in order to select the best configuration of the algorithm.

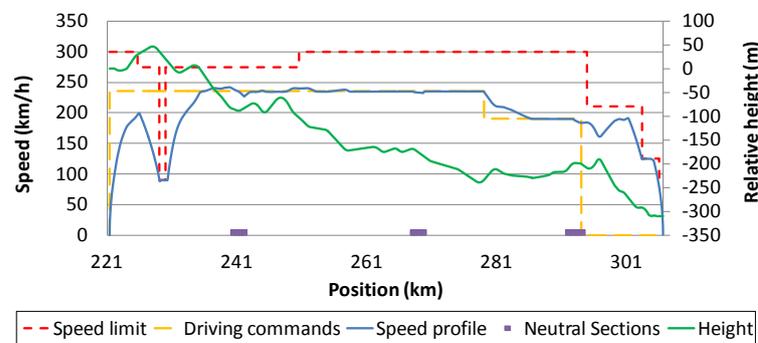


Figure 5-11. Nominal speed profile observing the temporary speed limitation

Initially, the nominal speed profile is calculated before departure (first execution of the algorithm), the train starts the journey following these initially optimised nominal commands, and the Pareto front is updated along the trip. Then, the train faces an unexpected speed limitation and when the delay is detected, a new solution is selected and the new commands are displayed to the driver when the train has left the temporary speed restriction. At this moment, 30 iterations are allowed for the execution of the algorithm with the objective of studying in detail its convergence

Hypervolume metric (Jiménez et al., 2013) is used to compare the quality of the Pareto fronts obtained. The Hypervolume measures the piece of the search space that is not dominated by any solution of the Pareto front. It is a quality metric that assesses not only the proximity of the result to the real Pareto front, but also the spread of the solutions obtained. The Hypervolume is calculated in this paper using the algorithm for calculating a Lebesgue measure proposed in (Fleischer, 2003).

Due to the random search nature of evolutionary algorithms, different solutions can be obtained for similar situations. To obtain a robust analysis, the Hypervolume results provided are the mean value of 20 tests in the same journey.

The performance of DNSGA-III-F using Random and Mutation mechanisms is analysed and compared with the application of a static version of DNSGA-III-F where the population is generated randomly at the beginning of each execution.

Figure 5-12 shows the evolution of the mean value of the Hypervolume metric through the iterations of DNSGA-III-F algorithm. Furthermore, the performance of the algorithm using $t_c = 60$ s is compared with $t_c = 100$ s. The algorithm can perform 5 iterations using a calculation time of 60 s whereas it can perform 12 iterations using a calculation time of 120 s. As shown in Figure 5-12, at iteration 5 the DNSGA-III-F algorithm, using both initialisation mechanisms, has practically converged for $t_c = 60$ s. For $t_c = 100$ s, it can be seen that DNSGA-III-F algorithm has converged at iteration 12. However, the results obtained for both versions of DNSGA-III-F at iteration 5 using 60 s of calculation time are better than the results obtained for both versions of DNSGA-III-F at iteration 12 using 100 s of calculation time. It can be observed in the figure that the *HV* of the initial population is better for $t_c = 60$ s because the result of the previous execution was obtained using more recent initial conditions. For that reason, it achieves better results in less number of iterations. Moreover, the use of a lower value of t_c presents the advantage of refreshing the Pareto front more frequently.

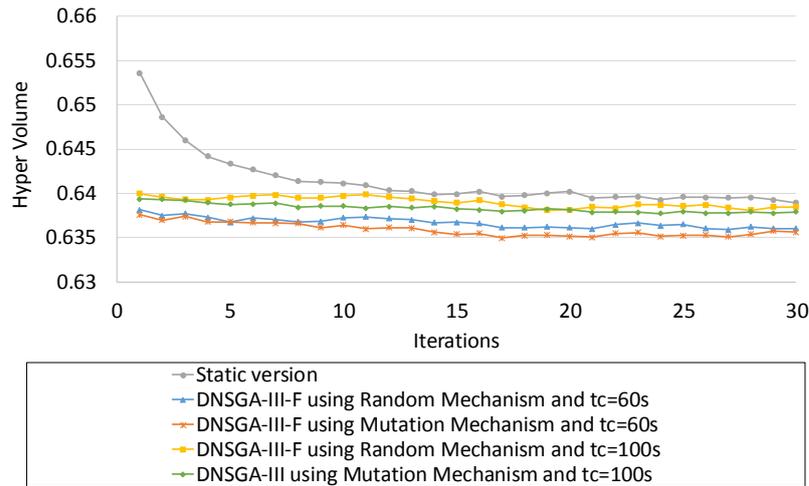


Figure 5-12. Mean value of Hypervolume during the evolution of the DNSGA-III-F

On the other hand, the static version is far from converging at that iteration 5 and it is close to stabilise at iteration 12. The static version of the algorithm starts from a randomly generated population as usual in genetic algorithms and its results are independent of the calculation time given that it does not depend on previous populations. Therefore, it starts from worse initial populations than the dynamic version and takes much longer to converge to a good solution. The use of the population of previous execution allows the algorithm to start from a population closer to the real optima accelerating the optimisation process. This way, the dynamic algorithm only takes a few iterations to update the previous Pareto front to the new train conditions during the travel. Thus, the comparison of the static version with the dynamic version demonstrates that, in 5 iterations, the dynamic algorithm converged for $t_c = 60$ s.

The comparison of the two diversity mechanisms in DNSGA-III-F shows that the results of Mutation Mechanism outperformed results of Random Mechanism using both calculation time values. Thus, it is preferable to introduce diversity in the first population of each execution by means of mutated individuals rather than random individuals. During the train travel, changes in Pareto front are smooth and, therefore, the Pareto front from an execution will be similar to the Pareto front of the following executions. As a consequence, mutated individuals from the previous execution will be closer to real optima than random individuals.

The computation cost of the optimisation depends mainly on the number of solution evaluations required. Each solution evaluation costs 0.15 s on average because of the detailed simulation. The computational cost of the evaluation module at each iteration is 7.5 s on average. This value can be obtained multiplying the evaluation cost of one solution by 50 members of the offspring population. Therefore, the computational cost of the other modules of the algorithm can be neglected compared with the time required by the evaluation module.

5.6.2. DELAY RESPONSE ANALYSIS FACING ONE TEMPORARY SPEED LIMITATION

This section compares the performance of the train facing a delay using DNSGA-III-F algorithm or applying the “immediate” delay recover strategy. The delayed situation described in Figure 5-11 is used to perform this analysis.

The “immediate” delay recover strategy is the typical behaviour of drivers when the train is delayed (Sicre et al., 2014). It consists in driving the train as fast as possible when the delay is detected until the train is circulating on time. At this moment, the driver coasts to link with the nominal driving commands. This delay recovery strategy is not energy efficient because the driver does not make an optimal use of the time margins.

In the application of the DNSGA-III-F algorithm, Mutation Mechanism is used because, as explained before, it is the configuration that provides better results.

The mean value of energy consumption and delay at the end of the journey for 20 tests are presented in Table 5-3. Regenerative energy is taken into account to obtain the net energy consumption, and it is calculated at pantograph and estimated at substations considering the electrical network loses.

		Pantograph net energy consumption (MWh)	Substation net energy consumption (MWh)	Delay at the arrival (s)
Immediate delay recover		1.043	1.170	-38.0
Mean value	$w_{ec} = 1 \ w_{rd} = 0$	0.975	1.089	-14.4
	$w_{ec} = 0.7 \ w_{rd} = 0.3$	0.986	1.038	-14.6
	$w_{ec} = 0.5 \ w_{rd} = 0.5$	1.030	1.152	-21.75
Mean value of the variations compared with the immediate delay recover	$w_{ec} = 1 \ w_{rd} = 0$	6.48%	6.92%	
	$w_{ec} = 0.7 \ w_{rd} = 0.3$	5.40%	5.65%	
	$w_{ec} = 0.5 \ w_{rd} = 0.5$	1.23%	1.47%	

Table 5-3. Mean value of energy savings and delays obtained by DNSGA-III-F for the complete journey

The results demonstrate that the application of DNSGA-III-F algorithm to make up a delay provides energy savings compared with the typical “immediate” delay recover strategy (from 6.9 % to 1.5 %). These energy savings are higher if the weighting factor of the energy consumption in the algorithm to update driver’s commands is high. Table 5-3 shows that the energy savings are reduced when the risk of delay in arrival is penalised more. Besides, it can be observed that the time that the train arrives in advance is also higher as higher is the weighting factor of the risk of delay in arrival. That means that the train is capable of overcoming higher delays during the journey.

The results of a single test are used hereafter to show the benefits of taking into account the risk of delay in arrival when selecting a speed profile. Figure 5-13 shows the Pareto surface used by the Algorithm to update driver’s commands.

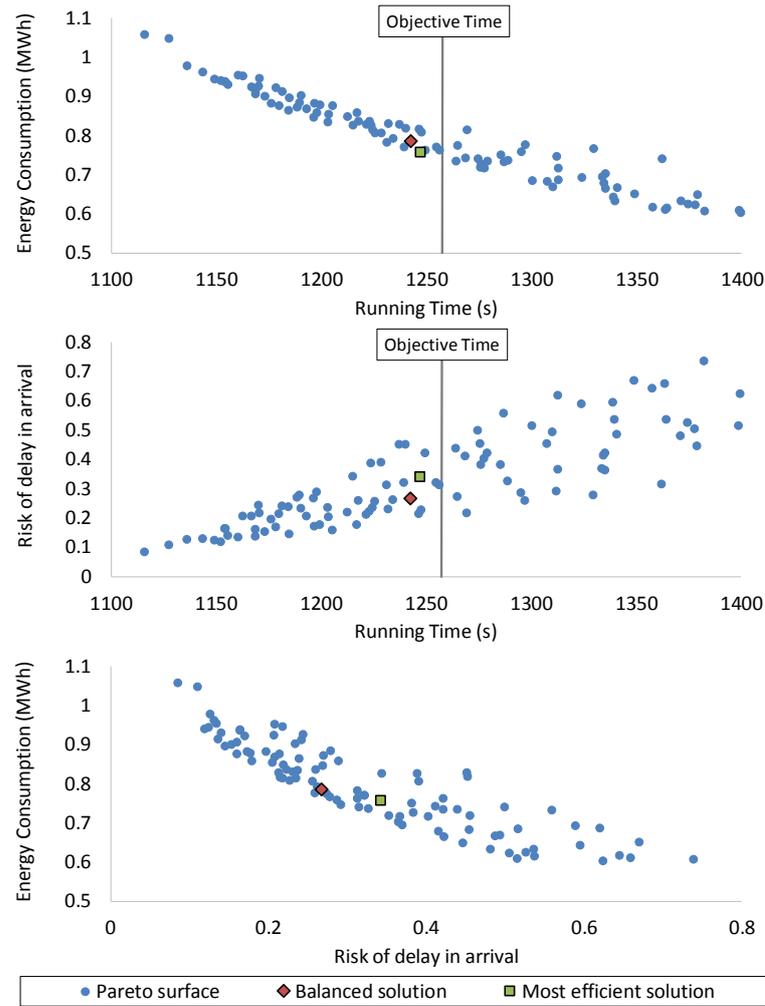


Figure 5-13. Pareto front used to obtain the speed profile for recovering accumulated delay

The green square represents the solution with the lowest energy consumption associated (most efficient solution) that fulfils the fuzzy punctuality requirement (imposed necessity level $N_p = 0.6$). In other words, it is the solution that the system would select if $w_{ec} = 1$ and $w_{rd} = 0$. The red diamond represents the solution that the system would select if $w_{ec} = 0.7$ and $w_{rd} = 0.3$ (balanced solution). The energy consumed by these solutions from the update of the commands up to the destination differs just 3%.

However, the solution obtained with $w_{ec} = 0.7$ presents an RD of 0.27 while solution obtained with $w_{ec} = 1$ presents an RD of 0.34.

Figure 5-14 shows the impact of these solutions on the risk of delay variable. It represents the speed profile and the evolution of the time margin for 3 different solutions: the “immediate” delay recover strategy, the most efficient solution (obtained with $w_{ec} = 1$) and the balanced solution (obtained with $w_{ec} = 0.7$). As can be seen, the “immediate” delay recover strategy runs as close as possible to the speed limits until the delay is recovered and thus, its energy consumption is higher. However, in the balanced solution and in the efficient solution, the speed of the train is usually far from the speed limits in order to save energy. The delay is also recovered in these cases

because the train is driven faster than the nominal speed profile presented in Figure 5-11.

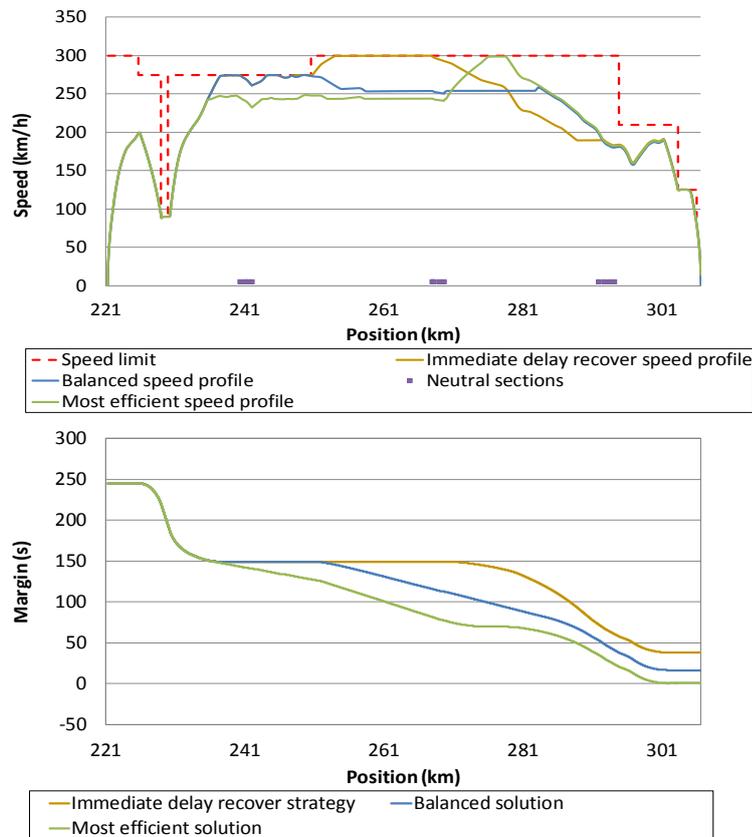


Figure 5-14. Speed profiles and delay evolution of “immediate” delay recover strategy and DNSGA-III-F

At the beginning of the journey the 3 solutions start with 4 minutes of time margin. However, the unexpected speed restriction drastically reduces the margin to 2 minutes and 30 seconds. From this point, the different solutions applied to make up the delay present different time margin evolution. The most robust solution is the “immediate” delay recover strategy because it maintains the margin of 2 minutes and 30 seconds up to position 270 km. Furthermore, it arrives 38 seconds in advance. However, this solution has the highest energy consumption, as shown in Table 5-3.

Solutions obtained by DNSGA-III-F are more energy efficient than the previous one. The most efficient solution is continuously reducing the time margin while the balanced solution keeps the time margin (2 minutes and 30 seconds) constant up to position 250 km. After that position, the balanced solution always maintains 30 seconds over the margin of the most efficient solution up to 280 km. That means that the balanced solution could recover delays 30 seconds higher than the most efficient solution. At the end of the journey the balanced solution arrives 16 seconds in advance while the most efficient solution practically arrives at the commercial arrival time.

It will depend on the preference of the railway operator to give more importance to energy consumption or to ensure the punctuality, and thus, the tuning of the weighting factors should reflect this preferred balance.

5.6.3. DELAY RESPONSE ANALYSIS FACING TWO TEMPORARY SPEED LIMITATION

This section analyses in more detail the effect of taking into account the risk of delay when choosing a solution from the Pareto front. A case study of a train facing two delays during the journey is used to illustrate how energy optimal solutions can lead to delays if the risk is not taken into account. When the train departs, it faces immediately an unexpected temporary speed limitation of 70 km/h between 221 km and 223 km positions. This first speed limitation produces 1 minute of delay if the nominal speed profile is not changed. After that, the train faces a new unexpected delay caused by a temporary speed limitation of 90 km/h between 273 km and 274 km positions. This new restriction increases 2 minutes the delay of the train at the arrival if no actions are taken. In Figure 5-15 it is represented the nominal speed profile in the journey described.

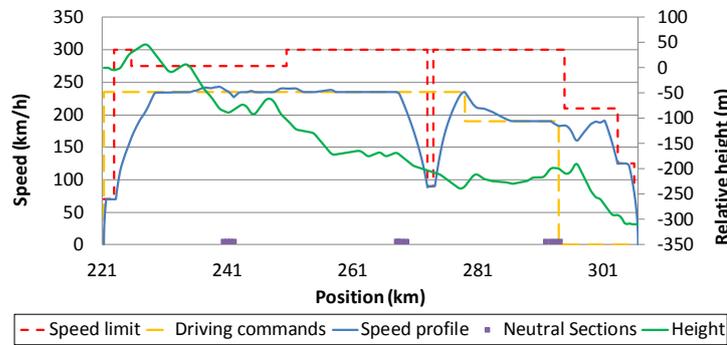


Figure 5-15. Nominal speed profile observing two temporary speed limitations

The proposed DNSGA-III-F algorithm using the Mutation Mechanism is applied during this journey to change the speed profile of the train when the delays are detected.

When the train departs its speed is limited by 70 km/h and this situation causes an increase of the delay. A new speed profile is selected from the Pareto surface obtained by the algorithm to reduce the delay. This surface is presented in Figure 5-16. In this figure, the green square represents the most efficient solution that can be selected to recover the delay, i.e. the solution obtained with $w_{ec} = 1$ and $w_{rd} = 0$. The red diamond represents a balanced solution that takes into account not only energy consumption but also risk of delay. The balanced solution is obtained using $w_{ec} = 0.7$ and $w_{rd} = 0.3$.

From this point, two scenarios are studied. In the first scenario, the train changes its nominal speed profile and applies the most efficient solution after the first speed limitation. In the second scenario, the train changes its nominal speed profile and applies the balanced solution after the first speed limitation. Figure 5-17 and Table 5-4 show the results obtained at this point where the second speed limitation is not expected.

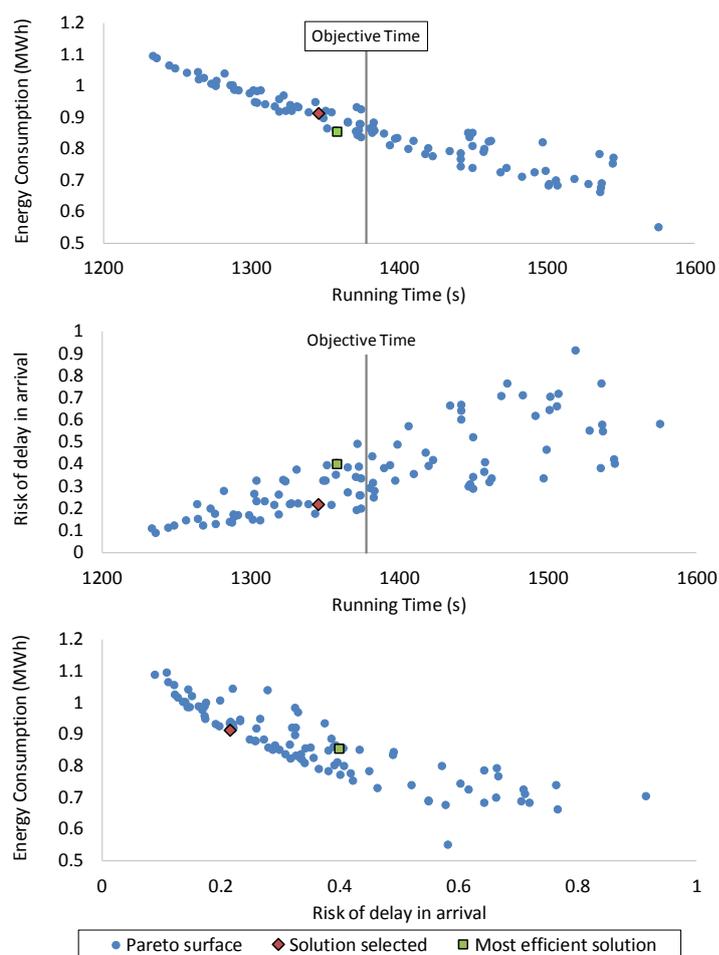


Figure 5-16. Pareto front used to obtain the speed profile for recovering the first accumulated delay

As can be seen, both solutions would lead to recover the delay at the next station if the second delay does not exist. The most efficient solution would result in less energy consumption than the balanced solution. However, it can be seen looking at the Pareto surface depicted in Figure 5-16 that the balanced solution has associated a less value of *RD*. The *RD* value of the balanced solution is 0.216 whereas the *RD* value of the most efficient solution is 0.399. This will be an important factor when facing the second delay of the journey.

	Pantograph net energy consumption (MWh)	Substation net energy consumption (MWh)	Expected Delay at the arrival (s)
Most efficient speed profile	0.956	1.041	-23
Balanced speed profile	1.001	1.100	-35

Table 5-4. Results of energy consumption and delays obtained by DNSGA-III-F for the complete journey of a train observing only a first speed limitation of 70 km/h

After the application of one the new speed profile, the train continues its journey until it faces a new speed limitation of 90 km/h in 273 km position. Due to this limitation, the delay increases again and a different speed profile is needed to arrive on time. In the case of the application of the most efficient solution, the train has not available any solution that fulfils the required arrival time. This is because it consumed an important

amount of its margin during the journey to save energy. Therefore, after the second delay, the only possible solution is to apply flat-out driving and the train will arrive late.

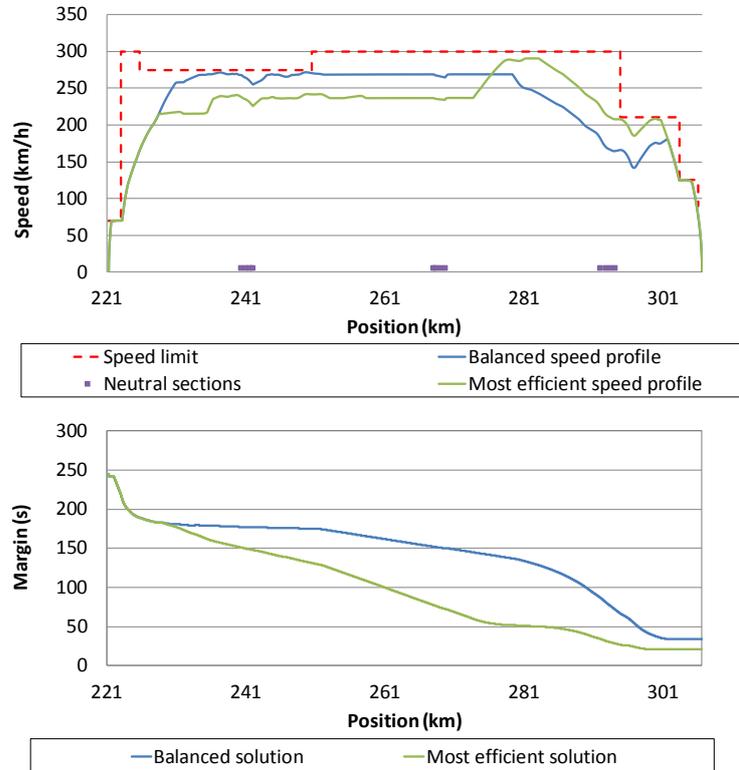


Figure 5-17. Speed profiles and margin evolution of DNSGA-III-F solutions when the train faces only a first speed limitation of 70 km/h

On the other hand, in the case of the application of the balanced solution, the train has available several solutions to recover the delay. Therefore, the algorithm selects one of those. In Figure 5-18 it is represented the resulting speed profiles at the end of the journey for the two cases analysed.

As can be seen, the application of the balanced solution makes the train run faster than the most efficient solution before the second limitation. That maintains the train margin above 150 s before it faces the second speed limitation. This way, after surpassing the 90 km/h limitation, the train has enough margin to arrive on time at the next station. Moreover, it can be observed that the margin allows coasting before the final braking from 286 km.

On the other hand, the most efficient solution runs slower to save energy but it is consuming its margin constantly. When the train faces the second limitation, the margin is totally consumed and, as a consequence, there is no solution that makes up the delay at the arrival. For this reason, the flat-out driving is applied from this point to minimise the final delay.

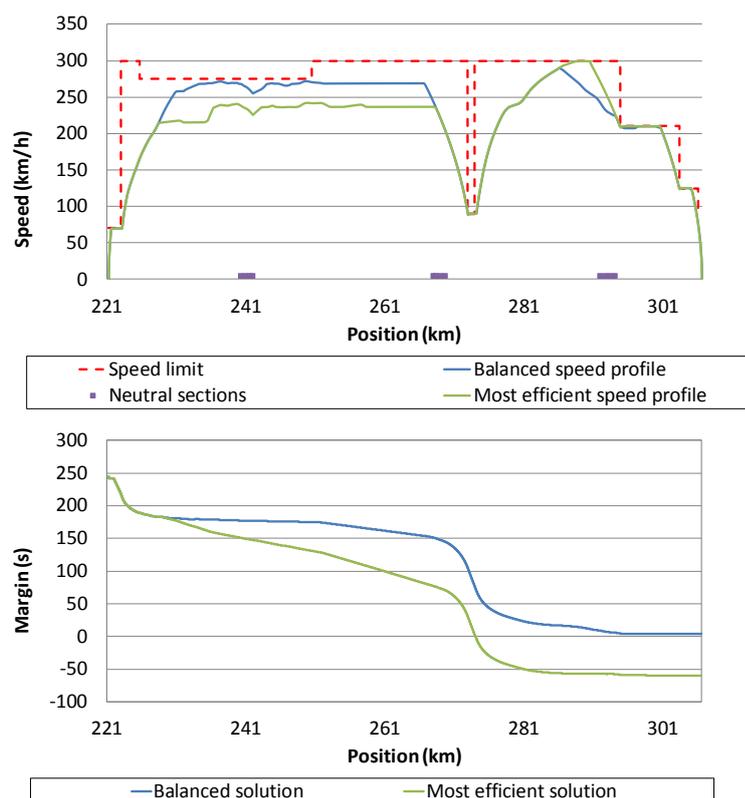


Figure 5-18. Speed profiles applied to recover the two delays faced by the train at the end of the journey

Table 5-5 shows the results obtained at the end of the journey in terms of energy consumption and delay at the arrival produced by the two cases studied. The results indicate that the most efficient solution produces less energy consumption. However, that solution arrives at the next station practically 1 minute late. On the other hand, the balanced solution that takes into account the risk of delay is able to arrive on time with 6 seconds in advance.

	Pantograph net energy consumption (MWh)	Substation net energy consumption (MWh)	Delay at the arrival (s)
Most efficient speed profile	1.097	1.213	58
Balanced speed profile	1.147	1.276	-6

Table 5-5. Results of energy consumption and delays obtained for the complete journey of a train observing two speed limitations

Figure 5-19 depicts the Pareto surface obtained by the DNSGA-III-F algorithm after the second speed limitation in the case of application of the balanced solution. The red diamond represents the solution selected by the algorithm to recover the second delay. As can be seen, the difference in risk of delay value for solutions with similar running time is not very significant because of the short distance to arrival and because the margin is small at that position. Therefore, the weight of the risk of delay in the selection of the speed profile is limited and a solution with lower energy consumption that fulfils the objective time is chosen in that case.

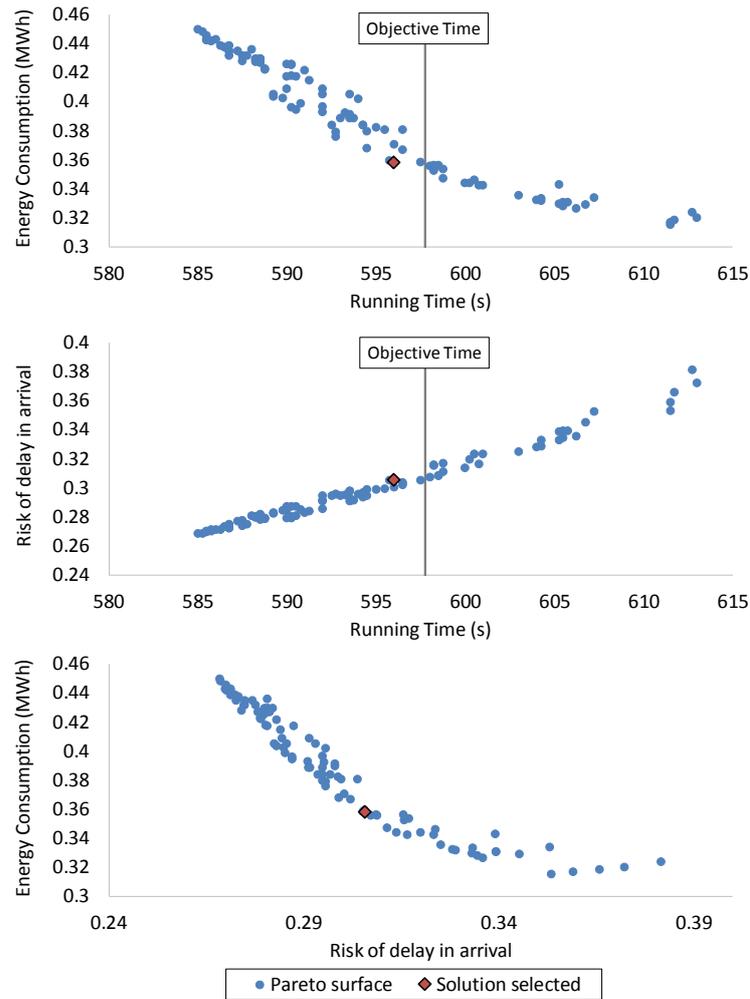


Figure 5-19. Pareto front used to obtain the speed profile for recovering accumulated delay at the second speed limitation of 90 km/h

5.7. CONCLUSIONS AND CONTRIBUTIONS

A regulation algorithm for high-speed trains has been proposed to obtain efficient speed profiles during the train travel. The algorithm calculates the Pareto front of the possible speed profiles before the departure of the train. After that, during the journey, this Pareto front is updated periodically. Using this method, a set of energy efficient speed profiles is always available to be executed when it is necessary to change the nominal driving commands due to an accumulated delay.

The proposed algorithm optimises not only the energy consumption and running time (typical in the literature), but also a new objective named as risk of delay in arrival. This objective is firstly introduced in this thesis and measures the robustness of a speed profile to arrive on time at the next station. The risk of delay in arrival is calculated based on the evolution of the time margin during the train travel. Running time, energy consumption and risk to delay in arrival are conflicting objectives and the result of the optimisation is a three-dimensional Pareto front.

A new optimisation algorithm has been proposed, DNSGA-III-F, that is a 3 objective dynamic algorithm with fuzzy parameters. This dynamic algorithm is able to calculate the Pareto front of optimal solutions and also to track its changes during the train travel. It is executed periodically to update the Pareto front and it uses solutions from previous executions to accelerate the optimisation process.

Furthermore, the uncertainty associated to the manual driving is modelled by means of fuzzy numbers and hybridised with the algorithm.

To ensure the real applicability of the algorithm solutions, a detailed simulator is used by the algorithm to evaluate the performance of the possible speed profiles.

The energy benefits of the proposed DNSGA-III-F algorithm have been analysed applying it to obtain a new set of driving commands when an unexpected delay affects the train. These results have been compared with the typical behaviour of drivers: the “immediate” delay recover strategy. Using DNSGA-III-F important energy savings can be obtained (6.9%). However, the energy savings will depend on the preference of the operator and the importance given to risk of delay in arrival.

The method proposed is flexible and the operator can reflect the preferred balance when the algorithm selects a solution from Pareto front. Thus, the operator could give priority to solutions that have more capacity to recover delays or to solutions that are more energy efficient. In other words, the railway operator could reduce the energy consumption at the minimum level without compromising the quality of the service.

The main contributions of this chapter are:

- The definition of risk of delay in arrival.
- The introduction of risk of delay in arrival as a third objective in the eco-driving problem.
- A dynamic three-objective algorithm with fuzzy parameters (DNSGA-III-F) to solve the online eco-driving of high-speed trains.
- The fuzzy delay response mechanism based on a continuously updated Pareto front.
- The analysis of the different energy savings that can be obtained giving different degrees of importance to the risk of delay.

ECO-DRIVING IN ATO OVER ERTMS

6.1. INTRODUCTION

There are many years of experience in the development and application of ATO systems in urban railways. During these years, ATO has been expanding in many mass transit systems because of the benefits that it provides. The use of ATO provides more regular and predictable travel times compared with the manual operation. This leads to an increase of the transport capacity in the system as well as an increase of the punctuality results. Moreover, the energy consumption is reduced using ATO systems because eco-driving can be executed more easily (Emery, 2017). The passenger comfort is also improved using automatic operation given that the speed profiles performed are smoother and calculated off-line to fulfil this type of constraint.

Benefits provided by ATO are also common objectives in mainline operation. However, the application of ATO is more complex in a mainline system. There are several differences between mainline and mass transit system that affect the implementation of automatic operation. Urban railways are usually operated by a single administration while multiple operators could participate in a long distance train travel. Besides, the number of different types of trains are very limited compared with mainline operation where the types of trains, composition and characteristics are multiple. Another important issue is that, typically, there is a single vendor that provides the signalling system and the automatic control in an urban line. Contrary, mainlines are

characterised by having multiple companies providing signalling equipment in different track sections.

These differences are some of the reasons why the ATO implementation in mainline systems is still under development. In view of these difficulties, interoperability arises as the keystone in the development of ATO for long distance travels.

In terms of interoperability there is much work done thanks to ERTMS (European Railway Traffic Management System). ERTMS was born as part of a project undertaken by the European Commission to develop the specification of a standardised signalling system (Council Directive, 1996). The aim of this system is to ensure interoperability of European trains and to improve safety, capacity and economic effectiveness. The ERTMS is composed by two main subsystems: ETCS (European Train Control System) and GSM-R (Global System for Mobile Communications - Railways). ETCS is in charge of the control and safety of the traffic while GSM-R is in charge of communications. ERTMS is a successful system, which is implemented in many European railway lines. Furthermore, it is continuing its expansion reaching countries from other continents.

Thus, linking ATO to ERTMS is a good opportunity to solve the interoperability problems for ATO systems. With this aim, a TEN-T project (TEN-T - ATO project, 2016) has been developed to include ATO in the ERTMS specification and the group of signalling companies, UNISIG, is working on it. The new ATO over ERTM standard will specify the requirements that this system must comply to drive the train automatically and to be interoperable.

It is expected that ATO over ERTMS real applications will be in service in the next years. Several projects are currently under development to implement ATO over ERTMS in real lines as the Mexico-Toluca project developed by CAF Signalling (Villalba, 2016) and the Thameslink project in London developed by Siemens (Burton, 2009). The commissioning of these projects is expected from 2019.

The requirements of the ATO over ERTMS establish new functions that are usually assigned to human drivers. Among them, it can be found speed control, accurate stopping and door opening and closing. Different equipment is needed to perform these tasks. Thus, it is necessary to differentiate between the trackside equipment and the on-board equipment. The trackside equipment is in charge of supplying the information of the track profile in the route, the operational restrictions and the timetable assigned to the train. On the other hand, the on-board equipment is in charge of collecting all this information, generating the speed profile to fulfil the timetable, driving the train and informing about the train status.

As can be seen, the on-board equipment will need algorithms to calculate the train speed profile. Furthermore, as energy efficiency is one of the main goals of ATO over ERTMS, these algorithms have to be designed using eco-driving principles.

The way in which the speed profile must be calculated will depend on the timetable and how it is defined. The time information sent to the on-board equipment could be the nominal timetable, or a timetable generated by complex algorithms for traffic

regulation purposes. Timetable information is provided by means of timing points. These points are defined as a set of the following information:

- Type of point: Departure, passing or arrival.
- Position of the point.
- Time assigned.

The on-board ATO equipment generates the speed profile that the train must perform to comply with the timing points. By this way, the interoperability of train is ensured. Each train is responsible of generating its own speed profile following its own rules or driving commands to meet the target times.

Compared with the studies presented in previous chapters of this thesis, the algorithms needed by the on-board ATO system must be capable of generating speed profiles that not only meet a target running time minimising energy consumption as usual, but also meet intermediate timing points. This introduces new constraints to the speed profile optimisation problem that can be defined in the form of target time windows.

Some works in the literature studied similar problems where the train control is restricted by intermediate time targets. In (Pudney et al., 2011) a method is proposed to modify optimal speed profiles in order to meet intermediate timing points. This method starts with an optimal speed profile obtained in the basis of Pontryagin's Maximum Principle that meets the arrival time to the next station. Then, a constructive algorithm modifies the initial speed profile adding new holding speed phases to meet the intermediate timing points. Although the method finds feasible speed profiles, the energy efficiency is not guaranteed.

In (T. Albrecht et al., 2013) the problem of speed profile optimisation subject to intermediate time windows or target points is discussed. Time restrictions are usually located in junctions, passing stations or positions where a faster train overtakes a slower one. Thus, intermediate target times are needed to meet the timetable. The train path envelope is proposed in this paper to define the train driving. Train path envelope defines a set of bounds to the feasible train driving that can be formed by intermediate target times and/or target speed ranges. These bounds can be obtained as a result of the planned timetable or provided by a traffic control and conflict resolution system as proposed in ON-TIME project (Quaglietta et al., 2016) where train path envelope is applied. The application of Pontryagin's Maximum Principle and dynamic programming are proposed in that paper to solve the eco-driving problem subject to a train path envelope.

The problem of obtaining the optimal speed profile of two trains travel in the same direction in a flat track was studied in (Albrecht et al., 2015). The safe separation of the two trains is ensured not allowing them to occupy the same track section at the same time. With this objective, the track is divided into sections defined by signal positions and section clearance times are specified. The clearance time of a section can be based on a timetable and defined the latest allowed exit time for the leading train and the earliest possible entry time for the following train. As a result of this definition, clearance times establish target times in the middle of the journey for both trains. The

problem is solved applying Pontryagin's Maximum Principle to derive the optimal train control regimes and obtaining the switching points between them. This work was extended to non-level tracks in (Albrecht et al., 2018).

Other authors have also studied the optimal train control given a train path envelope for the case of single and multiple trains (Wang and Goverde, 2017, 2016a, 2016b). The characteristics of the signalling system are taken into account and the so-called green wave policy is applied to avoid interferences between trains. In these cases, the pseudospectral method is used to transform the problem into a mixed-integer linear programming problem and to solve it using commercial solvers.

As can be seen, all these works make use of analytical procedures to obtain optimal speed profiles that meet intermediate time constraints. Previous chapters of this thesis have mentioned the limitations imposed to the train dynamic model by analytical methods. Therefore, in this chapter of the thesis a Nature Inspired Technique combined with simulation is proposed to solve the eco-driving problem subject to final and intermediate timing points.

This chapter presents the ATO over ERTMS eco-driving optimisation problem as well as the Nature Inspired algorithm proposed to solve it. In Section 6.2, the eco-driving problem is described. Then, Section 6.3 presents the proposed algorithm to solve the optimisation problem and the constraint handling technique. Section 6.4 presents the case study and analyses the performance of the algorithm and the effect of the targets imposed by the timing points. Finally, Section 6.5 presents the main conclusions obtained in this chapter.

6.2. PROBLEM DESCRIPTION

ATO over ERTMS can be applied either urban railways or mainline systems. However, its main purpose is the application in the complex operation of mainline system. ERTMS is the mandatory signalling system for European high-speed trains although it has been applied in a variety of systems. For this reason, this study will continue the trend followed in Chapters 4 and 5 of this thesis of focusing on high-speed railways.

ATO over ERTMS specification establishes that the on-board equipment must calculate the speed profile to be followed in order to meet the timing points. Thus, the eco-driving defined is a mono-objective optimisation problem with the aim of minimising the energy consumption. The optimisation problem is subject to time constraints. Time constraints are derived from the timing points and impose several time objectives in the middle and the end of the journey. According to the ATO over ERTMS specification, the objective times imposed by the timing points must be met with a certain time window. Therefore, there is some flexibility in the timetable meeting.

Thus, the eco-driving problem defined above can be formulated as:

Find X which minimises

$$EC(X) \tag{6.1}$$

subject to

$$|OT_l - t_l(X)| < \varepsilon_l, \quad l = 1, \dots, L \tag{6.2}$$

where $EC(X)$ is the energy consumption of solution X , $t_l(X)$ is the resulting passing time of solution X in the timing point l located at position s_l , OT_l is the objective time for the point l , ε_l is the time window allowed to the passing point l and L is the number of timing points including the arrival point at the station.

In this piece of research, each possible solution X is the set of holding speed without braking commands and a final coasting command collected in the command matrix C_m presented in Chapter 4 (Sicre et al., 2012).

A C_m is composed by n_s sections where the ATO applies certain driving command. The first $n_s - 1$ sections are holding speed without braking commands. Although holding speed without driving command was proposed to be applied manually, it also can be applied by an ATO. It is a more energy efficient way to negotiate steep downhill compared with the common holding speed commands. Furthermore, it ensures the comfortability of the speed profile limiting the sudden changes in train acceleration. The use of several holding speed commands is needed to meet the intermediate timing points because it is difficult to meet several intermediate time goals using only one holding speed command. The final driving command defines a coasting period before the final braking.

The commands contained in C_m are subject to a minimal speed restriction for the holding speed without braking commands (vc_{min}) and a minimum position from which the train can start the final coasting command (sc_{min}). Thus, it is ensured that the train does not run at low speed that can be perceived as unpleasant by passengers.

The problem defined can be classified as a Constrained Numerical Optimisation Problem (CNOP). CNOPs are optimisation problems in the presence of constraints (Mezura-Montes and Coello Coello, 2011). As was pointed out in the introduction of this chapter, this piece of research studies the application of Nature Inspired Techniques to the ATO over ERTMS eco-driving problem.

Several algorithms have been applied in the literature to solve CNOPs such as Genetic Algorithm (GA), Particle Swarm Optimisation (PSO), Evolutionary Strategies (ES) or Differential Evolution (DE). It can be found in the study of the literature that there is a preference for the application of Differential Evolution algorithm to solve CNOPs compared with other algorithms (Mezura-Montes and Coello Coello, 2011). For this reason, DE has been selected to solve the ATO over ERTMS eco-driving problem. The next chapter explains in detail the characteristics of the DE algorithm proposed.

6.3. DIFFERENTIAL EVOLUTION ALGORITHM FOR THE ATO OVER ERTMS ECO-DRIVING PROBLEM

Storn and Price proposed the DE algorithm for solving non-linear, non-convex, multi-modal and non-differentiable functions defined in the continuous space (Storn and Price, 1997). DE and its variants are considered one of the most competitive and versatile technique among evolutionary algorithms. Moreover, it has been applied to a variety of engineering and real-life problems (Das and Suganthan, 2011).

Among the advantages of DE compared with other methods it can be highlighted its simplicity and its straightforward implementation. Furthermore, the parameters to be tuned are very few. However, DE has provided excellent results solving not only non-constrained optimisation problems (Das et al., 2009) but also CNOPs (Li et al., 2010).

The procedure of DE algorithm is similar to other evolutionary algorithms. It starts from a population of solutions that are randomly created. This population evolves through iterations creating new solutions from previous individuals and selecting the best ones. The main difference of DE is that the new solutions are generated using a mutation operator based on scaled differences of distinct members of the population. Thus, the search moves of the algorithm tend to be self-adapted to the difference between surviving solutions.

6.3.1. DIFFERENTIAL EVOLUTION FLOWCHART

The DE procedure starts with the initialisation process. The initial population is created producing random individuals that respects the minimum speed and minimum position of the final coast command. This population is evaluated using the train simulator to obtain the energy consumption, passing times at the timing points and the fitness of each individual. Then, the population is sorted in rising fitness value.

After that, the algorithm creates for each individual a donor/mutant vector. The donor vector is created perturbing the best solution of the population with the difference between two randomly selected individuals (Das et al., 2016). It can be obtained using the following expression:

$$DV_j = X_{best} + F(X_{R_1} - X_{R_2}) \quad (6.3)$$

where DV_j is the donor vector of the individual j , X_{best} is the best individual of the population, F is a parameter called scaling factor, R_1 and R_2 are two different randomly generated integer numbers, X_{R_1} is the individual in the position R_1 in the population and X_{R_2} is the individual in the position R_2 in the population.

Crossover operator is applied after obtaining the donor vectors for all the individuals in the population. Crossover consists in mixing the components of the donor vector with the components of the individual associated, also called target vector. Thus, the trial or offspring vector is generated. The crossover mechanism is performed on each of the components of the solutions. At each component, a randomly generated number between 0 and 1 is compared with a parameter C_r , called crossover rate. If the random

number is less or equal to the crossover rate, the trial vector will inherit the corresponding component of the donor vector. Otherwise, it inherits the component of the trial vector. The scheme can be formulated as follows:

$$TV_j^h = \begin{cases} DV_j^h & \text{if } h = K \text{ or } rand[0,1] \leq C_r \\ X_j^h & \text{otherwise} \end{cases} \quad (6.4)$$

where TV_j^h is the component h of the trial vector associated with the individual j , DV_j^h is the component h of the donor vector associated with the individual j , X_j^h is the component h of the individual j and K is a randomly generated number between 1 and the number of components. The equality $h = K$ ensures that, at least, one component of the donor vector will be present in the trial vector.

Once all the trial vectors are generated, they are simulated to obtain the results on energy consumption, the passing times at the timing points and the fitness of each of them. The simulation is carried out using the simulation model detailed in Chapter 4. After that, the selection process is performed. Selection consists in comparing the fitness of each individual with the fitness of its associated trial vector. If the fitness of the trial vector is less or equal to the fitness of the individual, the trial vector replaces the individual in the population. Otherwise, the individual is retained and the trial vector is erased.

The process described before is repeated until a maximum number of iterations is reached. In Figure 6-1 the flowchart of the algorithm is presented.

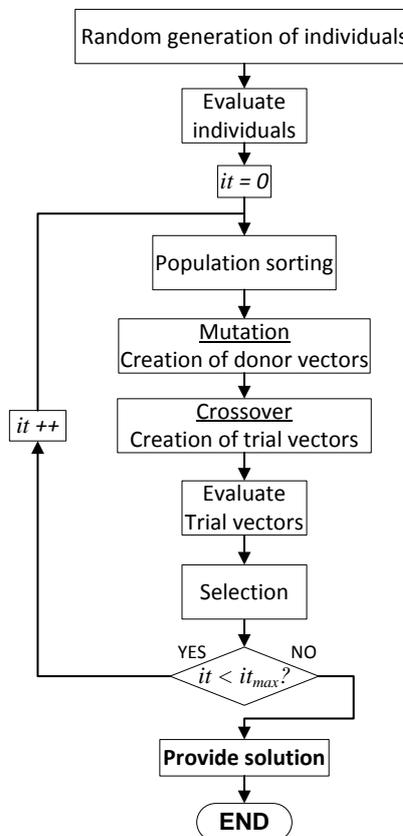


Figure 6-1. DE flowchart.

6.3.2. CONSTRAINT-HANDLING PROCESS

Nature Inspired Techniques are unconstrained optimisation procedures. For this reason, there is a research field that investigates ways to incorporate the constraint-handling into these algorithms (Coello Coello, 2002). Several constraint-handling techniques have been proposed in the literature to be incorporated in Nature Inspired algorithms.

One of the most common techniques are the penalty functions. Penalty functions consist in including in the fitness function the non-compliance with the constraints multiplied by a penalty factor (Baeck et al., 1997). By this way, solutions with better fitness are solutions closer to be feasible. This is a straightforward way to incorporate constraints. However, penalty factors require a careful tuning that increases the number of control parameters of the algorithms.

Other authors have proposed the separation of objective function and constraints. Thus, if the solution is feasible, its fitness value is equal to the objective function value. Otherwise, its fitness function is calculated based on the constrain violation (Powell and Skolnick, 1993). The idea is that feasible solutions always present better fitness values and move the population to the feasible space.

Another popular approach are the feasibility rules. In (Deb, 2000) the application of the binary tournament selection combined with three feasibility criteria in genetic algorithms was proposed. These feasibility rules are:

- The individual with the best value in the objective function is selected when two feasible solutions are compared.
- The feasible individual is selected when a feasible solution is compared with an infeasible solution.
- The solution with the lowest constraint violation is selected when two infeasible solutions are compared.

Other authors used and adapted the feasibility rules for their application to other algorithms (Brest et al., 2006; Mezura-Montes et al., 2005; Zielinski and Laur, 2006).

The use of decoders to solve CNOPs was proposed in (Kim and Husbands, 1998a, 1998b). Decoders map the feasible region of the search space in an easier-to-sample space where the algorithm can perform better. Decoders are one of the most competitive techniques to handle constraints. However, their implementation is complex and requires a high computational cost to transform the search space.

The approach presented in this thesis is a combination of feasibility rules with the separation of objectives and constraints. The fitness function is calculated differently for feasible and infeasible solutions. Thus, a feasible solution presents a fitness equal to its energy consumption. On the other hand, the fitness of infeasible solutions is calculated as sum of constraint violation plus the energy consumption of the flat-out driving (EC_{flat}), i.e. the most energy consuming speed profile. By this way, an infeasible

solution will always present a higher fitness value than a feasible individual. The fitness calculation is expressed using equation (6.5).

$$fitness(X) = \begin{cases} EC(X) & \text{if feasible} \\ EC_{flat} + \sum_{l=1}^L \max(|OT_l - t_l(X)| - \varepsilon_l, 0) & \text{otherwise} \end{cases} \quad (6.5)$$

The use of this fitness expression in the selection procedure will always present a similar result than feasibility rules. This way, a feasible solution will be selected over an infeasible one when the fitness of trial and target vectors are compared. Furthermore, the most energy efficient vector will be chosen between two feasible solutions while the closer to feasibility will be chosen between two infeasible solutions.

When ordering the population to find the best solution, the fitness value will split the population into two groups. The feasible solutions will appear in the first group and the infeasible ones will appear thereafter in the second group. Inside these groups the feasible solutions will be ordered in rising energy consumption order while the infeasible ones will be ordered in rising distance to feasibility. Thus, the best solution will appear always in the first place.

6.4. CASE STUDY

This section analyses the performance of the DE algorithm solving the problem proposed. As in Chapters 4 and 5, the case study uses real data from the Spanish high-speed line that runs from Calatayud to Zaragoza. The length of the journey studied is 85.4 km and the train is a Talgo-Bombardier train class 102. It is 200 m long and its empty weight is 324 t. The operational restrictions vc_{min} and sc_{min} for the driving commands are set to 150 km/h and 50 km respectively to avoid low speed phases in the middle of the journey.

The proposed method is used to obtain the energy optimal speed profile that fulfils the arrival hour as well as the intermediate target points. Two cases are analysed: a journey with one intermediate timing point and a journey with two intermediate timing points. For the shake of simplicity, all the time windows ε_l have equal value.

The performance of the DE algorithm proposed is compared with the performance of GA using the same constraint handling process. GA algorithm has been selected for this comparison because is one of the most used Nature Inspired Techniques solving the classical mono-objective eco-driving problem (Bocharnikov et al., 2010, 2007; Cucala et al., 2012b; Lu et al., 2013; Sicre et al., 2012; Wong and Ho, 2004b, 2003; Yang et al., 2012). As DE algorithm, GA evolves a population of solutions, called individuals, through iterations. At the end of each iteration only the best individuals of the population survive while the rest of the population is eliminated. These surviving individuals are called the elite group. The population eliminated is replaced in the next iteration by the offspring individuals generated by means of mutation and crossover of the elite individuals.

Table 6-1 and Table 6-2 show the tuned parameters used in the DE algorithm and the GA applied to the case study.

Population size	Scaling factor (F)	Crossover rate (C_r)	Number of sections in C_m (n_s)	Number of iterations
80	0.5	0.9	4	24

Table 6-1. Tuned parameters of DE algorithm

Population size	Elite group size	Number of crossover	Number of mutations	Number of sections in C_m (n_s)	Number of iterations
120	40	26	52	4	24

Table 6-2. Tuned parameters of GA

As in previous chapters of this thesis, different solutions can be obtained in different executions of the same algorithm because of the random search nature of evolutionary algorithms. For this reason, the performance of the algorithms has been evaluated executing 20 tests for the same journey.

6.4.1. ONE INTERMEDIATE TIMING POINT

In this case a journey with one intermediate timing point beside the arrival timing point is studied. The intermediate timing point is located between the departure point (221.3 km) and the arrival point (306.7 km) at position 264 km. The target times chosen for the timing points are based on the commercial timetable and are listed in Table 6-3 given that the departure time is at 0:00:00.

Location of timing point	Target time
264 km	00:12:03
306.7 km	00:26:00

Table 6-3. Location and target of timing points in the case of one intermediate timing point

Taking into account these restrictions, the algorithms are executed to obtain the speed profile of minimum consumption that fulfils the timing points. Three values for the time window have been used to study the influence of the constraint relaxation. These values are 1 second, 5 seconds and 10 seconds.

Figure 6-2 depicts the evolution of the average value of the fitness of the best solution found by the GA throughout the iterations. In this figure, the 3 cases of time window value are represented. Furthermore, the area of this figure is divided into two areas by a horizontal line with value located in the position Fitness = 1.52. As was explained in Section 6.3.2 infeasible solutions have associated a fitness value equal to the energy consumption of the flat-out driving plus the constrain violation. In this case, the energy consumption of the flat-out driving is equal to 1.52 MWh. Therefore, if the fitness is greater than 1.52, i.e. it is located in the red area, the best solution found by the algorithm at that iteration is infeasible. Contrary, if the fitness is lower than 1.52, i.e. it

is located in the green area, the best solution found by the algorithm at that iteration fulfils the timing points restrictions.

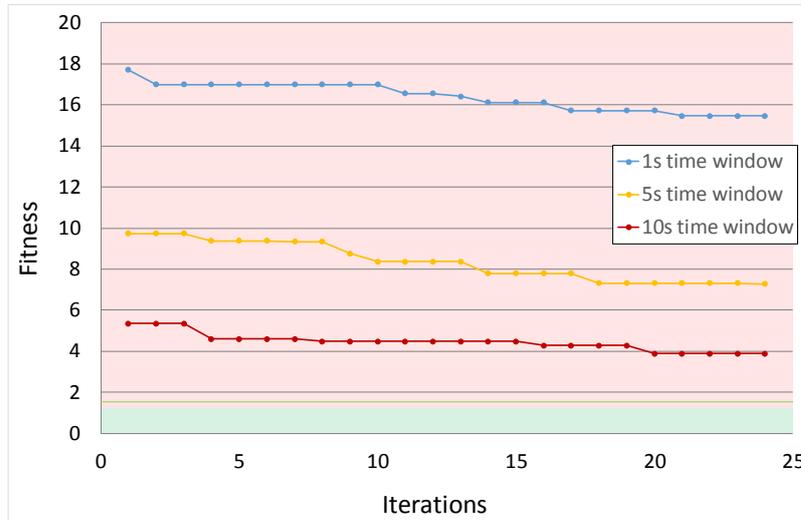


Figure 6-2. Evolution of the mean value of the fitness of the best solution found by the GA for the case of one intermediate timing point

As can be observed, the GA is not capable of finding a feasible solution in 24 iterations. The evolution of the algorithm improves the fitness value of the best solution. In other words, the solutions found are closer to the feasible region. However, the fitness improvement between two iterations is too small to find a feasible solution in the iterations given. It can be seen, comparing the results for the different values of the time window, that solutions found are closer to the feasible region as the time window is increased. Increasing the time window gives more flexibility to fulfil the timing points.

On the other hand, the results provided by the DE algorithm present a better performance. In Figure 6-3 the evolution of the mean value of fitness of the best solution found by the DE at each iteration is shown. The figure is divided into the infeasible red area and the feasible green area as in the previous case.

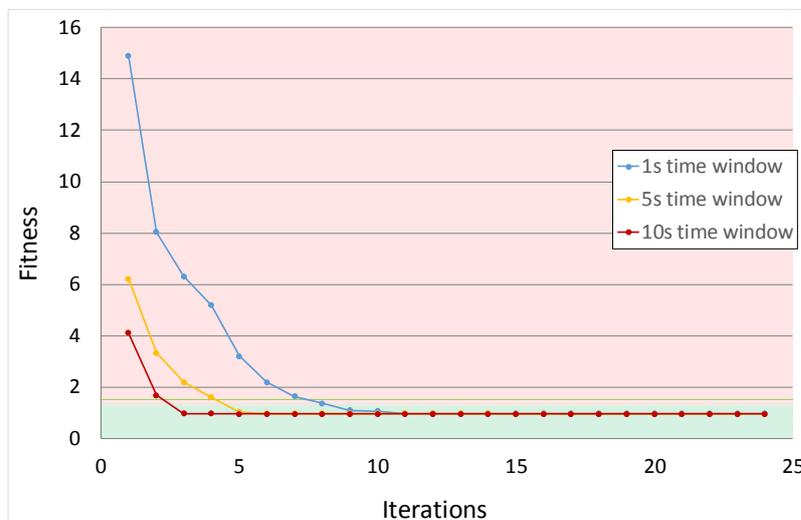


Figure 6-3. Evolution of the mean value of the fitness of the best solution found by the DE algorithm for the case of one intermediate timing point

The results provided by the algorithm has shown its capability to find feasible solutions. In all the 20 tests the algorithm was capable to find a speed profile that meets the target times. As in the case of the GA, the greater the time window the easier for the algorithm to find a feasible solution. As can be seen in Figure 6-3, with a time window of 1s the algorithm found a feasible solution at iteration 8, with a time window of 5s the feasible solution is found in iteration 5 and with a time window of 10s the feasible solution is found at iteration 3.

The DE algorithm is capable not only of finding a feasible solution but also of finding an energy efficient solution. Figure 6-4 shows the same results than in Figure 6-3 but zooming in the feasible region. This figure shows how the energy consumption of the best solution found by the algorithm is improved during the iterations. Furthermore, it can be observed how these improvements are stabilised at around 15 iterations. Therefore, it can be concluded that in 24 iterations the algorithm has converged to the optimal solution.

If the three different time windows are compared, it can be seen that the energy consumption of the speed profile is better as higher is the value of the time window. This is mainly because the train can perform higher running time for the whole journey if there is more flexibility. A second order effect is that the more flexibility gives more freedom to perform different speed profiles that can be more efficient.

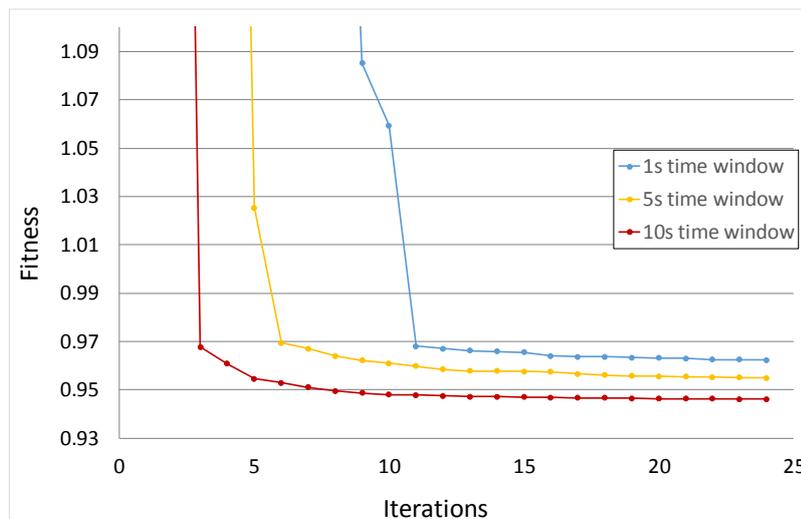


Figure 6-4. Evolution of the mean value of the fitness of the best solution found by the DE algorithm in the feasible region for the case of one intermediate timing point

The energy improvements found by the algorithm are non-trivial. It could be thought that once a speed profile that meets the timing points is found, the rest of feasible solutions will be similar. However, it is possible to obtain a variety of energy consumption results among the feasible speed profiles. For instance, in the case of 1 second of time window (which is the most restrictive case), the most energy consuming solution found by the algorithm in all the executions and in all the iterations presents an energy result of 1.04 MWh while the most energy efficient solution presents a result of 0.96 MWh. This is a difference of 7.7%, which is a relevant figure in eco-driving studies.

Figure 6-5 shows the speed profile of the most efficient and most energy consuming solutions found by the algorithm as well as the timing points. Both solutions meet the target times but in different ways. The most efficient solution presents a smoother speed profile that coasts at 284 km position. On the other hand, the most energy consuming solution reduces its speed once it has passed the intermediate timing point. This causes that the train must coast later, at 292 km position, to meet the arrival target time. The difference of energy consumption is caused mainly because of the difference in the length of the coasting period.

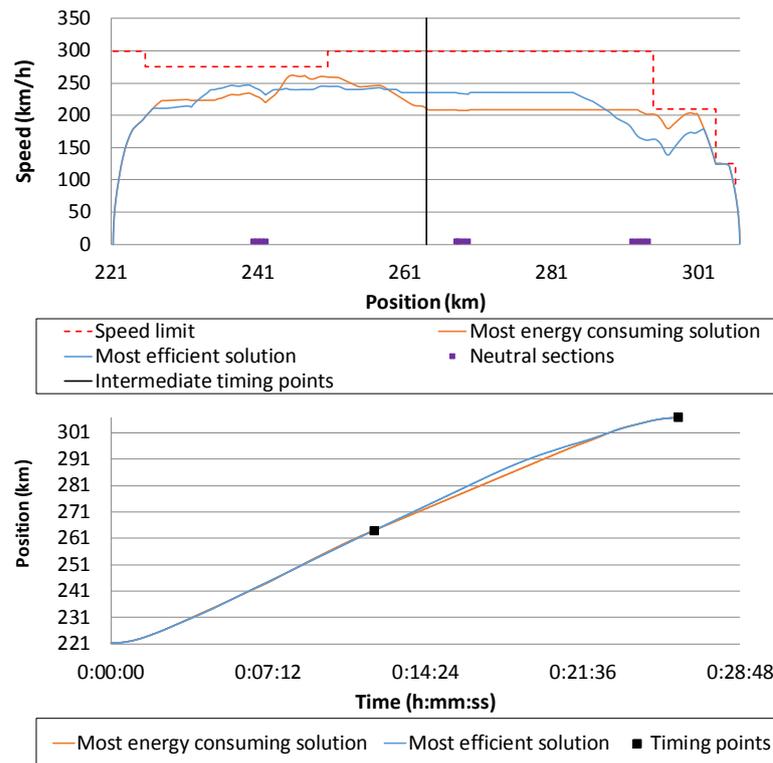


Figure 6-5. Speed profile and position evolution with time of the most efficient and most energy consuming solutions found by the algorithm for the case of one intermediate timing point

Concerning the computational time, the DE algorithm needs nearly 6 minutes to complete the 24 iterations demanded. This time is excessive for an online application where the train would need to react to changes in timing points during the journey. However, it is adequate for an offline application where the target times are considered constant.

6.4.2. TWO INTERMEDIATE TIMING POINTS

In this case a journey with two intermediate timing points beside the arrival timing point is studied. The intermediate timing points are located at positions 250 km and 278 km. The target times chosen for the timing points are based on the commercial timetable and are listed in Table 6-4 given that the departure time is at 0:00:00.

Location of timing point	Target time
250 km	00:09:10
278 km	00:16:20
306.7 km	00:26:00

Table 6-4. Location and target of timing points in the case of 2 intermediate timing points

Taking into account these objectives, the DE algorithm and the GA are executed to obtain the speed profile of minimum consumption that fulfils the timing points. As in the previous case, three values for the time window have been used to study the influence of the constraint relaxation. These values are the same than in the case of one intermediate timing point (1 second, 5 seconds and 10 seconds).

Figure 6-6 shows the evolution of the average value of the fitness of the best solution found by the GA throughout the iterations for the 3 time window values. The figure is divided into the red infeasible area and the green feasible area. The results obtained are similar to those obtained in the previous case. The GA is not capable of finding a feasible solution in 24 iterations. Besides, solutions found are closer to the feasible region as the time window is increased.

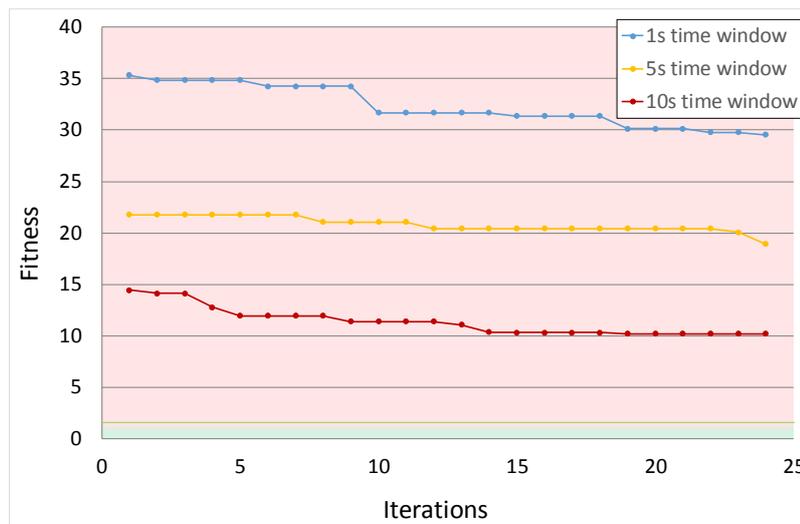


Figure 6-6. Evolution of the mean value of the fitness of the best solution found by the GA for the case of two intermediate timing points

The DE is capable of finding feasible solutions although the complexity of the problem has increased with one more restriction. In Figure 6-7 the evolution of the mean fitness value of the best solution found by the DE at each iteration is shown. The figure is divided into the infeasible red area and the feasible green area as in the previous examples.

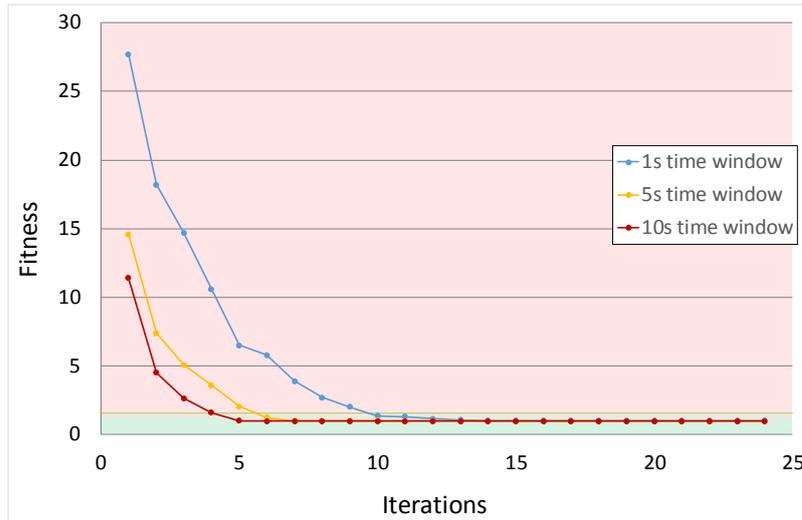


Figure 6-7. Evolution of the mean value of the fitness of the best solution found by the DE algorithm for the case of two intermediate timing points

In all the 20 executions of the algorithm, it was capable of obtaining a feasible solution. The relaxation of constraints facilitates the algorithm task. Figure 6-7 shows that with a time window of 1s the algorithm found feasible solutions at iteration 10, with a time window of 5s the feasible solutions are found at iteration 6 and with time window of 10s the feasible solutions are found at iteration 5. Comparing these results with those obtained with one intermediate timing point, it can be seen that with the new timing point the algorithm needs more iterations to find a feasible solution. This is because the new restriction reduces the feasible region and it is harder to reach.

Figure 6-8 shows the same results as Figure 6-7 but zooming in the feasible region. Once the algorithm finds a feasible solution it is improved in the remaining iterations reducing the energy consumption. Furthermore, the fitness improvement is stabilised before reaching 24 iterations in all the cases. That means that the number of iterations given to the algorithm is enough to converge to the optimal solution. Figure 6-8 presents also the effect of increasing the time window in the energy the solution obtained energy consumption.

As in the previous case, it can be observed a variety of energy consumption results among the feasible speed profiles. For instance, in the case of 1 second of time window (which is the most restrictive case), the most energy consuming solution found by the algorithm in all the executions and in all the iterations presents an energy result of 0.99 MWh while the most energy efficient solution presents a result of 0.96 MWh. This is a difference of 3.1%, which cannot be neglected. This difference is lower compared to the one obtained in the one intermediate timing point case. This is because the higher the number of restrictions, the lesser flexibility to drive the train and the more similar are the feasible speed profiles.

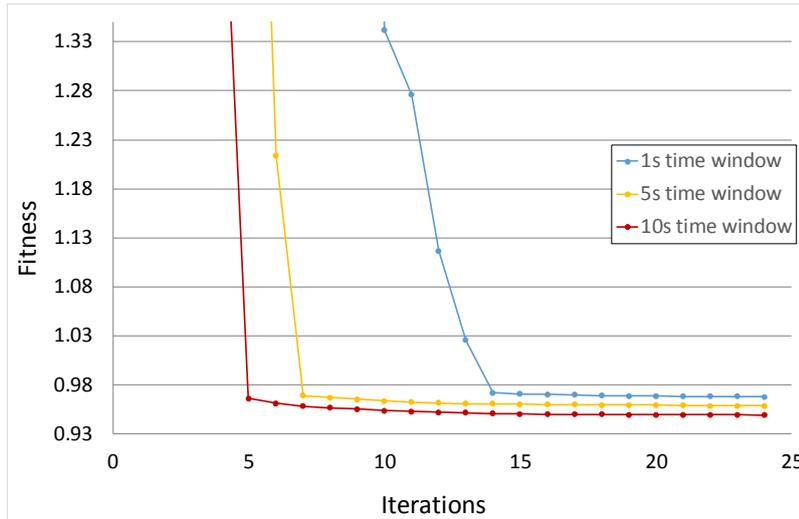


Figure 6-8. Evolution of the mean value of the fitness of the best solution found by the DE algorithm in the feasible region for the case of two intermediate timing point

Figure 6-9 shows the speed profile of the most efficient and most energy consuming solutions found by the algorithm as well as the timing points. As can be observed both solutions meet the target times. These two speed profiles are more similar than those shown in the previous section in Figure 6-5. The main difference between these two solutions is that the efficient speed profile coasts at 285 km position while the most energy consuming solution does not apply coast and regulates speed. The difference of energy consumption is caused mainly by the coasting period.

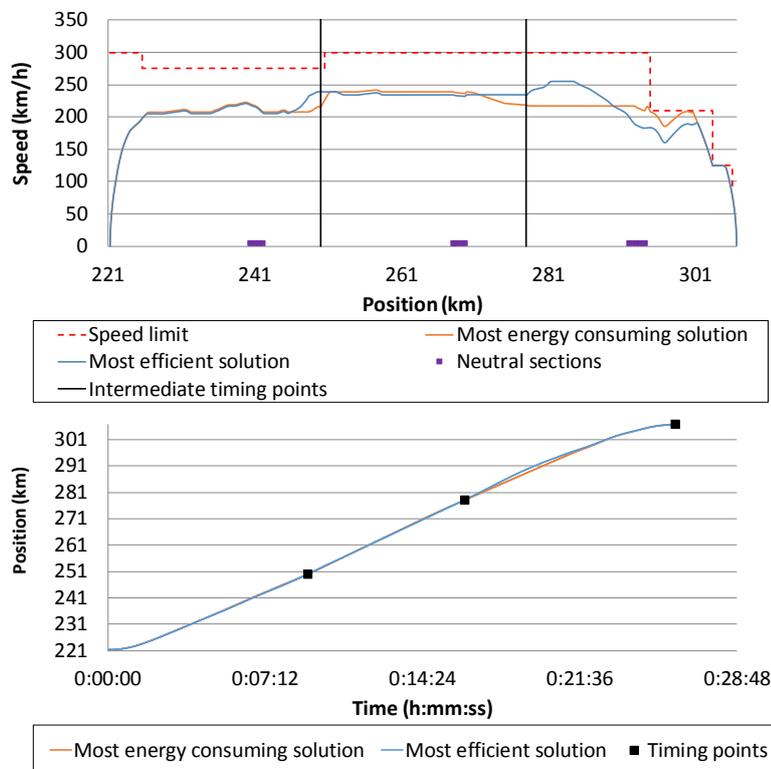


Figure 6-9. Speed profile and position evolution with time of the most efficient and most energy consuming solutions found by the algorithm for the case of two intermediate timing points

The same result than in the previous section has been obtained concerning the computational time. DE algorithm needs nearly 6 minutes to complete the 24 iterations demanded.

6.4.3. EFFECT OF THE TARGET TIMES ON THE ENERGY CONSUMPTION

In the two previous subsections, the energy variation that can be observed in different feasible solutions has been analysed. In this section, it is going to be studied the effect of the target selected for the timing points in the energy result.

It seems clear that the arrival timing point has a decisive role in the energy consumed by a train because it determines the running time of the journey. The inverse relationship between the running time and the energy consumption has been widely studied in the literature and in this thesis. Usually, this time is imposed by the timetable so there is no much flexibility to modify it.

On the other hand, intermediate target times can be selected with more flexibility. The intermediate timing points are planned to ensure the order of crossing trains and to ensure the safe separation between trains that run on the same track. Therefore, any intermediate target time can be selected provided that the previous criteria are met and the train is capable of fulfilling it.

The intermediate timing points determine the average speed in the different sections of the track. They have an impact on the way the train can be driven and affect directly to the final energy consumption. For this reason, it is necessary to select these target times bearing in mind the effect on the consumption apart from the operational requirements.

It can be taken the case of one intermediate timing point to illustrate the effect of the intermediate target time value on the energy consumption. If the intermediate target time is delayed, from the nominal 0:12:03 to 0:14:40, the train will have to perform a different speed profile to meet the new objectives although the arrival target time is maintained. Figure 6-10 shows the solution obtained by the algorithm with the delayed timing point compared with the solution obtained using the nominal timetable in Section 6.4.1.

It can be seen how the new target time forces a slower driving in the first section of the track before the intermediate timing point compared with the nominal timetable solution. In the second section, the new timing point forces the application of flat-out driving. For this reason, the energy consumption obtained is greater (1.14 MWh) than the energy consumption obtained by the nominal timetable speed profile (0.96 MWh). This difference is 18.7% of energy consumption increase of the delayed timing point solution with respect to the nominal conditions.

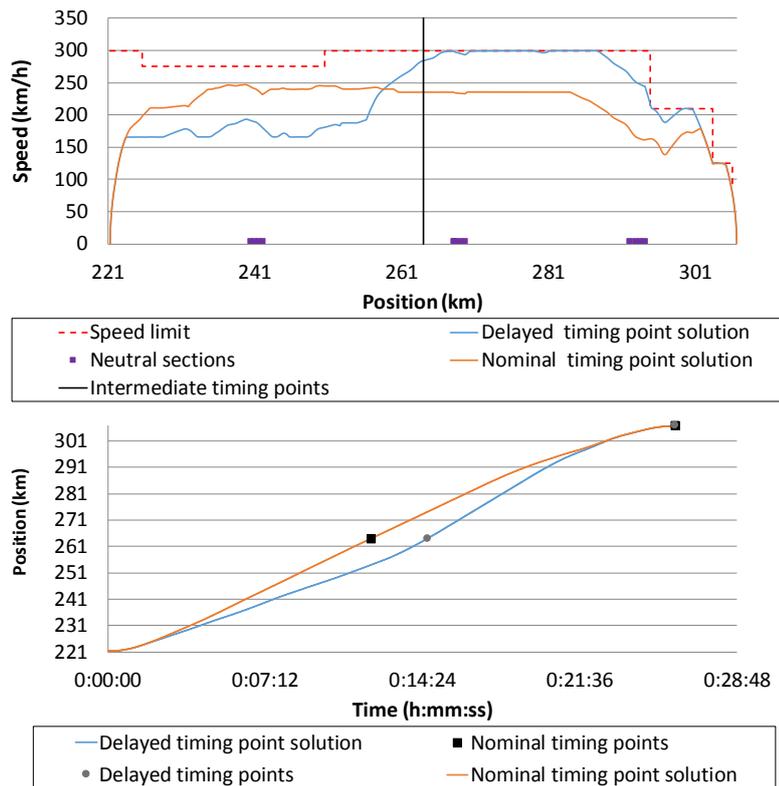


Figure 6-10. Speed profile and position evolution with time of the solution found by the algorithm for the case of one intermediate delayed timing point

Another example can be shown in the case of two intermediate timing points. If the target time of the point located at 278 km is reduced from the nominal timetable 0:16:20 to 0:15:00, the running time available between the two intermediate timing points will be decrease. As the arrival timing point is not modified, the running time between the second intermediate timing point and the arrival will be increased. Figure 6-11 depicts the effect of the modified timing points on the speed profile of the solution obtained by the DE algorithm compared with the nominal solution studied in Section 6.4.2.

The first section of the journey is very similar for both speed profiles because the first intermediate timing point has not changed. However, the modification of the second target time forces the train to run near the speed limits until it starts a long coast at 270 km position. This long coast period allows the train to meet the increased time difference between the second timing point and the arrival target. On the other hand, the nominal solution applies a lower holding speed without braking command in the second section and starts to coast at 285 km position. The results show that the modified timing point solution presents higher energy consumption (0.99 MWh) than the nominal speed profile (0.96 MWh). This is 3.1% energy consumption increase due to the modifications in the timing points. The energy consumption increases due to the period that the modified timing point solution drives close to the maximum speed limits. This period is more energy consuming because of the higher speed, and because the train cannot perform the small coasting sections associated to the holding speed without braking commands. The extra of energy consumed during that period is not compensated by the larger coasting period performed later.

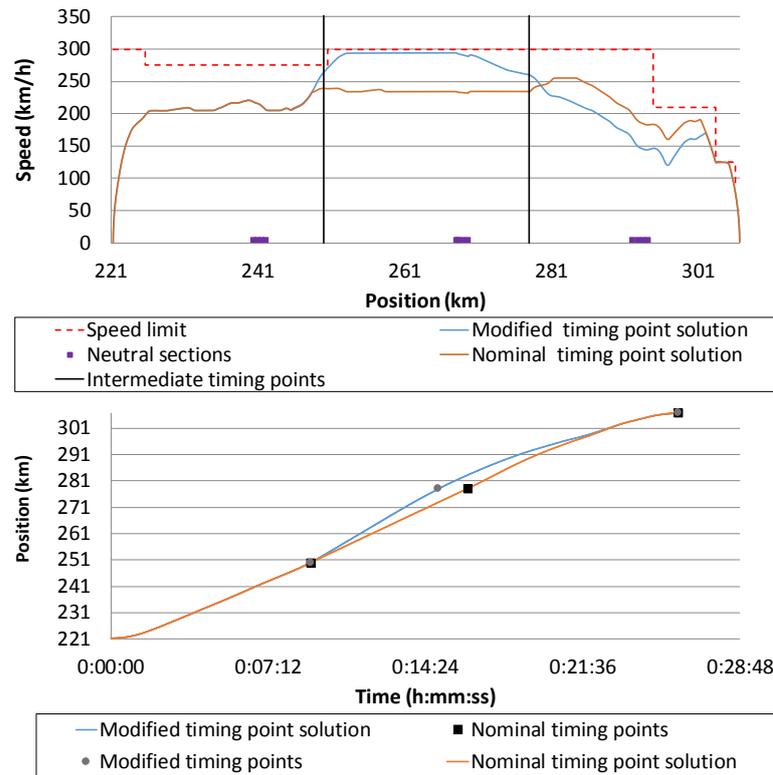


Figure 6-11. Speed profile and position evolution with time of the solution found by the algorithm for the case of 2 intermediate timing points with modifications respect to the timetable

These two examples have illustrated that it is important to find the most efficient speed profile given a set of timing points. However, the optimal train operation can only be achieved choosing wisely the target times of intermediate timing points.

6.5. CONCLUSIONS AND CONTRIBUTIONS

This chapter has presented the eco-driving problem defined by the requirements of the new ATO over ERTMS system. This eco-driving problem aims to minimise the energy consumption fulfilling the commercial arrival time to the next station. Furthermore, it is subject to new constraints compared with the classical eco-driving because of the intermediate target times imposed by the timing points during the journey.

A Nature Inspired algorithm combined with a detailed simulation model have been proposed to solve this problem. The algorithm proposed to solve this eco-driving problem is the DE algorithm, which has been widely applied in the field of constraint optimisation. Furthermore, a fitness function has been defined to handle the new constraints in the problem.

The performance of the DE algorithm has been compared with the well-known GA. These algorithms have been tested in two case studies where one and two intermediate timing points have been taken into account beside the arrival target time. The results have shown that the GA does not find feasible solutions in this difficult-to-solve

problem. Contrary, the DE algorithm proposed has demonstrated its capacity to find speed profiles that meet all the target times.

Apart from finding feasible solutions, the algorithm is capable of finding which one has the lowest energy consumption. This task is not trivial since important energy variations have been obtained among feasible solutions. Thus, it has been observed 7.7% of energy variation the case of one intermediate timing point and 3.1% in the case of two intermediate timing points. The presence of more constraints reduces the feasible region making more similar the speed profiles that meet all the target times. For this reason, the energy differences observed in the case of two intermediate timing points are smaller compared with the case of one intermediate point.

The time window allowed to meet the target times influences the performance of the algorithm and the energy consumption of the optimal solution. The greater the time window the greater the feasible region and the easier to find valid speed profiles. Furthermore, the energy consumption of the solution obtained is lower as greater is the time window. Increasing the time window flexibility to driving solutions and increases slightly the running time.

The effect of the target time selected for the timing points on the energy consumption has also been shown. Important energy consumption variations have been obtained modifying the nominal timing points. For this reason, the minimum energy consumption on the train operation can be achieved not only obtaining the best speed profile for a given timetable but also choosing the best possible target times (thus, designing an optimal timetable).

Finally, the time required by the algorithm to perform the optimisation is around 6 minutes. This time is adequate for an offline application. However, the calculation time is excessive to be applied online at the on-board equipment of the ATO over ERTMS. In the online application, the timing points can change during the journey and quick response times are required in the speed profile calculation. For this reason, future work is needed to improve the calculation speed of the algorithm.

The main contributions of this chapter are:

- The application of the DE algorithm to solve the eco-driving problem subject to time constraints along the journey to fulfil the ATO over ERTMS requirements.
- The fitness function proposed to handle the constraints related to the timing points.
- The comparison of the DE and the GA solving the constrained eco-driving problem.
- The analysis of the influence of the number of timing points in the performance of the optimisation algorithm and in the energy consumption of the optimal solution obtained.
- The demonstration of the important influence of the designed target times associated with the timing point in the energy consumption.

CONCLUSIONS AND CONTRIBUTIONS

7.1. CONCLUSIONS AND CONTRIBUTIONS

This section collects the main conclusions obtained in the development of this thesis. They are detailed following the main sections in this document.

OPTIMISATION ALGORITHMS FOR THE DESIGN OF ENERGY EFFICIENT ATO SPEED PROFILES IN METROPOLITAN LINES

Nowadays, many metropolitan trains are equipped with Automatic Train Operation systems (ATO). These system drives the train automatically according to a speed profile defined a by pre-programmed set of driving commands in the trackside equipment. At each station, a set of pre-programmed ATO speed profiles are available with different running times and energy consumption.

The new signalling technologies such as CBTC permit a better communication capacity and thus, the possible values of the ATO parameters that can be sent to the train (that is, the associated number of different possible speed profiles) is drastically higher. The exhaustive exploration of all the possible solutions is not advisable because of the high computational cost. Moreover, it has been shown that some of these new possible speed profiles are more efficient from the energy point of view (up to 20%) and are better distributed in a wide running time range. Consequently, it is important to apply algorithms that can find effectively the most efficient solutions to exploit the advantages of the new systems.

In the literature, the eco-driving problem is typically stated as a mono-objective optimisation problem where the result is a speed profile that fulfils a target time with the minimum energy consumption. However, other works state the eco-driving problem as a multi-objective optimisation problem. Usually, this problem has two conflicting objectives, which are energy consumption and running time, and the result is the Pareto front of solutions. Solutions in the Pareto front are the non-dominated solutions. Non-dominated solutions are those that cannot be improved in all the objectives at the same time.

The multi-objective optimisation model has the advantage that the decision maker can select the most appropriate speed profile taking into account the trade-off between energy consumption and running time in view of Pareto front. For solving this task, Nature Inspired techniques and, in particular, population-based algorithms seem to be especially suited due to their abilities to search simultaneously for multiple Pareto optimal solutions and to perform better global search of the search space.

There are many Nature Inspired algorithms developed to solve multi-objective problems. However, only few of them have been applied to the eco-driving problem. In this thesis, MOPSO and NSGA-II algorithms for the optimal design of ATO speed profiles have been applied and compared based on the accurate simulation of the ATO and train motion.

The assessment of the results obtained with both algorithms has been carried out using several metrics that compare the results of the algorithms with the points of the real Pareto front obtained using exhaustive simulation of the search space. These metrics measure the quality of the Pareto front obtained by the algorithms in terms of number of solutions obtained, diversity of solutions and distance to the real optimum.

The results of the application of these algorithms to a case study show that MOPSO algorithm obtains more solutions than NSGA-II. Furthermore, solutions obtained by MOPSO are closer to the real Pareto front than solutions obtained by NSGA-II. Diversity measures indicate that the MOPSO solutions are better distributed in the objective space than the ones obtained by NSGA-II.

From the point of view of the Railway Operator, the difference between the results of the algorithms can be translated in terms of energy consumption. For some running time values, the solutions obtained by MOPSO consume 10% less energy than those obtained by NSGA-II.

The main contributions of chapter 2 are:

- The application of the NSGA-II algorithm to the eco-driving problem defined by real ATO speed commands.
- The use of several metrics to assess the performance of optimisation algorithms when solving the eco-driving problem.
- The comparison between MOPSO algorithm and NSGA-II solving the eco-driving problem.

ROBUST SPEED PROFILES FOR THE AUTOMATIC TRAFFIC REGULATION SYSTEM IN METROPOLITAN LINES

The study developed in chapter 2 does not take into account the uncertainty of the solutions obtained. The main sources of uncertainty of the train operation are the load and the train delays.

The high precision of the ATO equipment in the execution of the pre-programmed driving parameters is practically just affected by the uncertainty in the mass of the train associated with the passengers load. The passenger load varies depending not only on the departure station but also on the moment of the observation. These variations produce different results on energy consumption. Besides, it causes variations in the running time produced by a set of pre-programmed driving parameters because the train resistance is dependant of the train mass. For the same reason, the load variations could cause variations in the shape of the speed profile making an expected comfortable solution to become non-comfortable.

The other main source of uncertainty associated with the traffic operation is the occurrence of delays that must be corrected by the traffic regulator. The statistical distribution of delays determines the frequency with which the controller demands each pre-programmed ATO speed profile at each station. Typically, there are four speed profiles pre-programmed. The set of speed profiles is usually chosen equidistant in time. This approach is not energy optimal because the frequency of use of each speed profile is not taken into account.

A new procedure to design energy efficient speed profiles to be programmed in the signalling equipment of a metropolitan system has been proposed in chapter 3 of the thesis.

The proposed model is based on the calculation of the Pareto curve of the possible speed profiles that are robust against passenger load variations. Then, the set of speed profiles to be programmed in the signalling system is taken from the robust Pareto front by means of a PSO optimisation algorithm, considering energy efficiency and delay distribution in the line.

Two algorithms for obtaining the robust Pareto front have been applied and compared using a case study. The first model is a robust multi-objective optimisation algorithm that makes use of a robust definition as a restriction. The second one is an alternative method based on the robustness of the solution to changes in its driving pattern. It has been shown that pattern-robustness requirement is more restrictive than definition of robustness type-II. Moreover, the pattern-robustness requirement is more useful because it guarantees the comfort of the speed profile. Besides that, the alternative procedure has found more solutions than the standard robust optimisation algorithm. For this reason, the pattern recognition gives to the designer more possible solutions to choose and, some of them, have lower energy consumption for a given running time.

The proposed selection model including train delays information has been compared with the traditional selection method that distributes speed profiles uniformly in time.

Three different cases of delay distributions are considered: decreasing, increasing and uniform distributions. The first one represents stations where delays are frequent, the second distribution represents stations where trains are regulated frequently to reduce their time interval with the following train and the third distribution corresponds to an intermediate situation where running times are demanded with the same probability.

The results show important energy savings, around 3 - 14%. This model has also been used to study the energy benefits obtained from increasing the number of speed profiles in the pre-programmed set. The typical size of this set is 4 and the energy consumption can be reduced 3.5% by the inclusion of two extra speed profiles.

The main contributions of chapter 3 are:

- The application of the robust optimisation method proposed in (Deb and Gupta, 2005) combined with MOPSO algorithm to be applied in the ATO eco-driving problem.
- A process to design robust and efficient ATO speed profiles based on the proposed pattern of the driving model.
- The comparison between the method proposed in (Deb and Gupta, 2005) and the pattern-based method proposed in this thesis.
- A model to select the optimal speed profiles from a Pareto front to be programmed in the ATO equipment based on the delay distribution in the line.
- The application of PSO algorithm to obtain the optimal pre-programmed set of speed profiles of the traffic regulation system.
- The assessment of the energy saving that can be obtained selecting optimally the pre-programmed set, using delay information and the energy savings that can be obtained increasing the number of speed profiles in the pre-programmed set.

REAL TIME ECO-DRIVING OF HIGH-SPEED TRAINS

Chapters 4, 5 and 6 of the thesis studied the case of eco-driving application in long-distance lines, particularly, in high-speed railways. It is important to take this into account because there are important differences between urban and in high-speed railways that affect the eco-driving application.

Metropolitan railways are highly automated systems where the distance between stations is short and the trains are typically driven by automatic train operation (ATO) equipment. The driving strategies applied in ATO equipped trains are basically: speed regulation and coasting-remotoring.

On the other hand, high-speed trains are typically driven manually and the journeys between stations are long-distance travels. The driving strategies applied in HSR are speed regulation and its efficient version, holding speed without braking.

Most of the eco-driving work in the literature is related to the offline planning of railways efficient driving. However, optimal speed profiles can also be obtained in the regulation stage as an on-line calculation. If the train is delayed, the offline eco-driving design is not valid anymore and a new on-line eco-driving calculus is required to recover

the delay in an energy efficient way. The main challenge of the on-line eco-driving application is the low computation time available to carry out the optimisation and the changing situations that may occur along the trip.

Chapter 4 of the thesis has proposed a dynamic multi-objective optimisation model based on accurate simulation of the train motion to handle the real-time regulation of a high-speed train using eco-driving. The model calculates the Pareto front of the possible speed profiles before the departure of the train. After that, during the journey, this Pareto front is updated at time intervals by an optimisation algorithm that adapts it to the new conditions of the train. Using this model, a set of energy efficient speed profiles is always available to be executed when it is necessary to change the nominal driving commands to perform a new speed profile with a different running time associated.

Two algorithms have been tested to solve the dynamic model: DNSGA-II and DMOPSO. These algorithms are the dynamic extension of NSGA-II and MOPSO that have been demonstrated suitable to eco-driving problems. Versions A and B of DMOPSO have been compared with the two versions of DNSGA-II and the static algorithms MOPSO and NSGA-II. The results demonstrate that the dynamic algorithms outperformed their static versions tracking the changes in the Pareto front. Furthermore, the Pareto front provided by DMOPSO-A has an associated better Hypervolume metric and convergence.

To analyse the energy benefits of the dynamic models, MOPSO, DMOPSO and DNSGA-II have been applied to obtain a new speed profile when an unexpected temporary speed limitation affects the train. These algorithms have been compared with the typical behaviour of drivers: the “immediate” delay recover strategy.

The solutions obtained applying a static algorithm such as MOPSO provide important energy savings (5.6% of the whole trip energy) compared with the “immediate” delay recover strategy.

However, these energy savings can be improved significantly using the dynamic version of the algorithms. The best performance is obtained by the DMOPSO, which has associated energy savings of 7.8%.

The main contributions of chapter 4 are:

- A dynamic multi-objective optimisation model of the online eco-driving of high-speed trains.
- The assessment of the performance of two versions of DNSGA-II solving the online eco-driving problem.
- The assessment of the performance of two versions of DMOPSO solving the online eco-driving problem.
- The comparison of the results obtained by DNSGA-II and DMOPSO.
- The delay response mechanism based on a continuously updated Pareto front.

- The analysis of the energy savings that can be obtained by means of the proposed dynamic model compared with the static models and the typical delay response model applied by drivers.

BALANCING ENERGY CONSUMPTION AND RISK OF DELAY IN HIGH-SPEED TRAINS

In chapter 4, a dynamic multi-objective optimisation model was proposed for the real-time regulation of a high-speed train. However, the uncertainty associated to the operation of the solutions obtained was not taken into account in the study. It is important to fulfil the punctuality requirements, apart from using an accurate model. For this reason, in this chapter the model proposed has been extended to incorporate all the main sources of uncertainty that are present in the real operation of a high speed train.

The main uncertainty in high-speed traffic regulation is associated with the contingencies that may occur in the line. Contingencies are usual situations that produce delays and are related to temporary speed limitations and traffic perturbations. The commercial running time between two stations is designed using a time margin in timetables to deal with contingencies. If necessary, this margin is available to make up delays and, if not, it is consumed during the train travel to perform efficient speed profiles. If the margin is quickly consumed during the journey, the train could perform an efficient speed profile but it might not be able to react to unexpected delays. On the other hand, the speed profiles that retain the time margin until the end of the journey are highly energy consuming but more robust to contingencies in the line.

In addition, it is also important to model the uncertainty associated with manual driving, considering that there are always small deviations in the application of driving commands. The uncertainty in these parameters is usually better represented using fuzzy knowledge modelling. The main advantages of using fuzzy modelling are its capacity to work with imprecise or incomplete data and its flexibility and simplicity to be implemented providing fast calculation times. The information of how drivers apply driving commands is usually incomplete or non-existent because each driver has its own driving style.

A regulation algorithm for high-speed trains has been proposed in chapter 5 to obtain efficient speed profiles during the train travel. As in the previous chapter, the problem is stated as a dynamic multi-objective optimisation problem. Thus, the proposed algorithm calculates the Pareto front of the possible speed profiles before the departure of the train and, during the journey, updates it periodically. Therefore, a set of energy efficient speed profiles is always available to be executed when it is necessary.

The proposed algorithm optimises not only the energy consumption and running time (typical in the literature), but also a new objective named as risk of delay in arrival. This objective is firstly introduced in this thesis and measures the robustness of a speed profile to arrive on time at the next station. The risk of delay in arrival is calculated based on the evolution of the time margin during the train travel. Running time, energy

consumption and risk to delay in arrival are conflicting objectives and the result of the optimisation is a three-dimensional Pareto front.

Furthermore, the uncertainty associated to the manual driving is modelled by means of fuzzy numbers and hybridized with the algorithm.

A new optimisation algorithm has been proposed, DNSGA-III-F, that is a 3 objective dynamic algorithm with fuzzy parameters. This dynamic algorithm is able to calculate the Pareto front of optimal solutions and also to track its changes during the train travel. It is executed periodically to update the Pareto front and it uses solutions from previous executions to accelerate the optimisation process.

The energy benefits of the proposed DNSGA-III-F algorithm have been analysed applying it to obtain a new set of driving commands when an unexpected delay affects the train. These results have been compared with the typical behaviour of drivers: the “immediate” delay recover strategy. Using DNSGA-III-F important energy savings can be obtained (6.9%). However, the energy savings will depend on the preference of the operator and the importance given to risk of delay in arrival.

The method proposed is flexible and the operator can reflect the preferred balance when the algorithm selects a solution from Pareto front. Thus, the operator could give priority to solutions that have more capacity to recover delays or to solutions that are more energy efficient. In other words, the railway operator could reduce the energy consumption at the minimum level without compromising the quality of the service.

The main contributions of chapter 5 are:

- The definition of risk of delay in arrival.
- The introduction of risk of delay in arrival as a third objective in the eco-driving problem.
- A dynamic three-objective algorithm with fuzzy parameters (DNSGA-III-F) to solve the online eco-driving of high speed trains.
- The fuzzy delay response mechanism based on a continuously updated Pareto front.
- The analysis of the different energy savings that can be obtained giving different degrees of importance to the risk of delay.

ECO-DRIVING IN ATO OVER ERTMS

There are many years of experience in the development and application of ATO systems in urban railways. However, the application of ATO in a mainline system is under development nowadays because its application is more complex. There are several differences between mainlines and mass transit systems that affect the implementation of automatic operation. Urban railways are usually operated by a single administration while multiple operators could participate in a long distance line. The number of different types of trains are very limited in urban lines compared with mainline operation and, typically, there is a single vendor that provides the urban signalling system while mainlines are characterised by having multiple companies providing signalling equipment in different track sections. In view of these difficulties,

interoperability arises as the keystone in the development of ATO for long distance travels.

In terms of interoperability there is much work done thanks to ERTMS. ERTMS was born as part of a project undertaken by the European Commission to develop the specification of a standardized signalling system (Council Directive, 1996). The aim of this system is to ensure interoperability of European trains and to improve safety, capacity and economic effectiveness.

Linking ATO to ERTMS is a good opportunity to solve the interoperability problems for ATO systems. With this aim, a TEN-T project (TEN-T - ATO project, 2016) has been developed to include ATO in the ERTMS specification and the group of signalling companies, UNISIG, is working on it. The new ATO over ERTM standard will specify the requirements that this system must comply to drive the train automatically and to be interoperable.

The requirements of the ATO over ERTMS establish new functions that are usually assigned to human drivers. Among them, it can be found speed control, accurate stopping and door opening and closing. Different equipment is needed to perform these tasks. Thus, it is necessary to differentiate between the trackside equipment and the on-board equipment. The trackside equipment is in charge of supplying the information of the track profile in the route, the operational restrictions and the timetable assigned to the train. On the other hand, the on-board equipment is in charge of collecting all this information, generating the speed profile to fulfil the timetable, driving the train and informing about the train status.

As can be seen, the on-board equipment will need algorithms to calculate the train speed profile. Furthermore, as energy efficiency is one of the main goals of ATO over ERTMS, these algorithms have to be designed using eco-driving principles.

The way in which the speed profile must be calculated will depend on the timetable and how it is defined. Timetable information is provided by means of timing points. Timing points define positions in the track and the target departure/arriving/passing time for these points.

The on-board ATO equipment generates the speed profile that the train must perform to comply with the timing points. By this way, the interoperability of train is ensured. Each train is responsible of generating its own speed profile following its own rules or driving commands to meet the target times.

Compared with the studies presented in previous chapters of this thesis, the algorithms needed by the on-board ATO system must be capable of generating speed profiles that not only meet a target running time minimising energy consumption as usual, but also meet intermediate timing points. This introduces new constraints to the speed profile optimisation problem that can be defined in the form of target time windows.

A Nature Inspired algorithm combined with a detailed simulation model have been proposed to solve this problem. The algorithm proposed to solve this eco-driving problem is the DE algorithm, which has been widely applied in the field of constraint

optimisation. Furthermore, a fitness function has been defined to handle the new constraints in the problem.

The performance of the DE algorithm has been compared with the well-known GA. These algorithms have been tested in two case studies where one and two intermediate timing points have been taken into account beside the arrival target time. The results have shown that the GA does not find feasible solutions in this difficult-to-solve problem. Contrary, the DE algorithm proposed has demonstrated its capacity to find speed profiles that meet all the target times.

Apart from finding feasible solutions, the algorithm is capable of finding which one has the lowest energy consumption. This task is not trivial since important energy variations have been obtained among feasible solutions. Thus, it has been observed 7.7% of energy variation the case of one intermediate timing point and 3.1% in the case of two intermediate timing points. The presence of more constraints reduces the feasible region making more similar the speed profiles that meet all the target times. For this reason, the energy differences observed in the case of two intermediate timing points are smaller compared with the case of one intermediate point.

The time window allowed to meet the target times influences the performance of the algorithm and the energy consumption of the optimal solution. The greater the time window the greater the feasible region and the easier to find valid speed profiles. Furthermore, the energy consumption of the solution obtained is lower as greater is the time window. Increasing the time window flexibility to driving solutions and increases slightly the running time.

The effect of the target time selected for the timing points on the energy consumption has also been shown. Important energy consumption variations have been obtained modifying the nominal timing points. For this reason, the minimum energy consumption on the train operation can be achieved not only obtaining the best speed profile for a given timetable but also choosing the best possible target times (thus, designing an optimal timetable).

Finally, the time required by the algorithm to perform the optimisation is around 6 minutes. This time is adequate for an offline application. However, the calculation time is excessive to be applied online at the on-board equipment of the ATO over ERTMS. In the online application, the timing points can change during the journey and quick response times are required in the speed profile calculation. For this reason, future work is needed to improve the calculation speed of the algorithm.

The main contributions of chapter 6 are:

- The application of the DE algorithm to solve the eco-driving problem subject to time constraints along the journey to fulfil the ATO over ERTMS requirements.
- The fitness function proposed to handle the constraints related to the timing points.
- The comparison of the DE and the GA solving the constrained eco-driving problem.

- The analysis of the influence of the number of timing points in the performance of the optimisation algorithm and in the energy consumption of the optimal solution obtained.
- The demonstration of the important influence of the designed target times associated with the timing point in the energy consumption.

7.2. PUBLICATIONS

The studies carried out in the frame of this thesis have yielded a series of publications in Journals and Conferences. Specifically:

JCR Journals

Fernández-Rodríguez, A., Fernández-Cardador, A., Cucala, A.P., 2018a. Balancing energy consumption and risk of delay in high speed trains: A three-objective real-time eco-driving algorithm with fuzzy parameters. *Transportation Research Part C: Emerging Technologies* 95, 652–678. <https://doi.org/10.1016/j.trc.2018.08.009>

Fernández-Rodríguez, A., Fernández-Cardador, A., Cucala, A.P., 2018b. Real time eco-driving of high speed trains by simulation-based dynamic multi-objective optimization. *Simulation Modelling Practice and Theory* 84, 50–68. <https://doi.org/10.1016/j.simpat.2018.01.006>

Fernández-Rodríguez, A., Fernández-Cardador, A., Cucala, A.P., Domínguez, M., Gonsalves, T., 2015. Design of Robust and Energy-Efficient ATO Speed Profiles of Metropolitan Lines Considering Train Load Variations and Delays. *IEEE Transactions on Intelligent Transportation Systems* 16, 2061–2071. <https://doi.org/10.1109/TITS.2015.2391831>

Domínguez, M., Fernández-Cardador, A., Cucala, A.P., Gonsalves, T., Fernández-Rodríguez, A., 2014. Multi objective particle swarm optimization algorithm for the design of efficient ATO speed profiles in metro lines. *Engineering Applications of Artificial Intelligence* 29, 43–53. <https://doi.org/10.1016/j.engappai.2013.12.015>

Conference presentations

Fernández-Rodríguez, A., Fernández-Cardador, A., Cucala, A.P., 2015. Energy efficiency in high speed railway traffic operation: a real-time ecodriving algorithm, in: 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC). pp. 325–330. <https://doi.org/10.1109/EEEIC.2015.7165181>

7.3. FUTURE WORK

In this section, some research lines are propose to continue the work presented in this thesis.

The first proposal would be the development of algorithms that need less number of configuration parameters. This is a usual concern in the research field of Nature Inspired algorithms. Parameters, such as the size of population or number of mutation/crossovers, are tuned depending on the problem characteristics. This feature demands an adjustment process when there are important changes in the characteristics of the problem to solve. The algorithms with less configuration parameters are more flexible and the tuning process is softer.

Another improvement is the application of parallel computing techniques to accelerate the optimisation process. All the Nature Inspired algorithms applied during the development of this thesis are population based. The most computational expensive process of these algorithms is the population evaluation because it depends on the simulator. The population evaluation process can be parallelised resulting in a drastically reduction of the calculation times.

Other future work can be classified attending to the topics studied in the thesis:

Eco-driving in metropolitan lines

Regarding to the studies on metropolitan railway systems there are some important future lines:

- Optimisation of speed profiles defined by more complex driving parameters. This thesis has studied the typical set of driving parameters: braking deceleration rate, speed holding value, or coasting speed and remotoring speed. However, the improved communication capacity of new signalling systems, such as CBTC, allows not only to transmit driving parameters with more resolution but also to transmit speed profiles defined by more driving parameters. This way, applying the robust optimisation method presented in this thesis, new speed profiles can be found that could improve the energy consumption maintaining the robustness to load variations.
- Improvement of the robust optimisation algorithm proposed: The method proposed demands several executions of several optimisation processes to obtain the Pareto front of each driving pattern identified. The information obtained by the algorithm about the exploration of the search space seeking solutions with one driving pattern could be very useful to speed up the following optimisation processes.
- Development of new algorithms for the traffic regulation. The continuous communication provided by new signalling systems such as CBTC gives more flexibility to control trains movement from the centralised control centre. Thus, it is not necessary to wait until the trains arrive at the stations to change their driving commands if it is necessary. This provides a quicker response to traffic disturbances. With this purpose, new algorithms are needed to modify the driving commands of the train so the speed profile changes adequately to regulate quicker the traffic and in an energy efficient manner.

Eco-driving in high-speed lines

Regarding to the studies on high-speed railway systems, the following future work can be identified:

- The introduction in the simulation model of the uncertainty related to the climate conditions, such as temperature, wind or rain. This could allow to study if the eco-driving speed profiles are valid in all the conditions and obtain an efficient driving under a variety of climate conditions.
- Estimation of the confidence interval of the results provided by simulators. A typical problem of train manufacturers is the confidence of their simulation results especially when making offers of new trains. The uncertainty models studied in this thesis as well as the proposed in the previous bullet point could help to delimit the confidence intervals.

ATO over ERTMS

Some of the future research about ATO over ERTMS has arisen during the analysis of the results obtained in this thesis:

- Improvement of the calculation speed of the algorithm: the time required by the proposed algorithm to perform the optimisation is around 6 minutes. This time is adequate for an offline application. However, the calculation time is excessive to be applied online at the on-board equipment of the ATO over ERTMS. In the online application, the timing points can change during the journey and quick response times are required in the speed profile calculation.
- Optimal intermediate target time selection: The selection of the time restrictions during the journey are not trivial. Important energy consumption variations have been obtained in this thesis modifying the nominal timing points. For this reason, the minimum energy consumption on the train operation can be achieved not only obtaining the best speed profile for a given timetable but also choosing the best possible target times. Thus, a future step would be an optimisation model for the design of timetable that fulfil the ATO over ERTMS requirements. This algorithm would be executed at the traffic control centre which is integrated in the ATO track-side system.

References

- Abril, M., Salido, M.A., Barber, F., 2008. Distributed search in railway scheduling problems. *Engineering Applications of Artificial Intelligence* 21, 744–755. <https://doi.org/10.1016/j.engappai.2008.03.008>
- Acikbas, S., Soylemez, M.T., 2008. Coasting point optimisation for mass rail transit lines using artificial neural networks and genetic algorithms. *IET Electric Power Applications* 2, 172–182. <https://doi.org/10.1049/iet-epa:20070381>
- Albrecht, A., Howlett, P., Pudney, P., Vu, X., 2011. Optimal train control: Analysis of a new local optimization principle, in: *American Control Conference (ACC)*, 2011. Presented at the American Control Conference (ACC), 2011, pp. 1928–1933.
- Albrecht, A., Howlett, P., Pudney, P., Vu, X., Zhou, P., 2018. The two-train separation problem on non-level track—driving strategies that minimize total required tractive energy subject to prescribed section clearance times. *Transportation Research Part B: Methodological* 111, 135–167. <https://doi.org/10.1016/j.trb.2018.03.012>
- Albrecht, A.R., Howlett, P.G., Pudney, P.J., Vu, X., 2013. Energy-efficient train control: From local convexity to global optimization and uniqueness. *Automatica* 49, 3072–3078. <https://doi.org/10.1016/j.automatica.2013.07.008>

- Albrecht, A.R., Howlett, P.G., Pudney, P.J., Vu, X., Zhou, P., 2015. Energy-efficient train control: The two-train separation problem on level track. *Journal of Rail Transport Planning & Management, Optimal (Re)scheduling and Energy Saving* 5, 163–182. <https://doi.org/10.1016/j.jrtpm.2015.10.002>
- Albrecht, T., Binder, A., Gassel, C., 2013. Applications of real-time speed control in rail-bound public transportation systems. *IET Intelligent Transport Systems* 7, 305–314. <https://doi.org/10.1049/iet-its.2011.0187>
- Alvarez-Benitez, J.E., Everson, R.M., Fieldsend, J.E., 2005. A MOPSO Algorithm Based Exclusively on Pareto Dominance Concepts, in: Coello, C.A.C., Aguirre, A.H., Zitzler, E. (Eds.), *Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 459–473.
- Arboleya, P., Coto, M., González-Morán, C., Arregui, R., 2014. On board accumulator model for power flow studies in DC traction networks. *Electric Power Systems Research* 116, 266–275. <https://doi.org/10.1016/j.epsr.2014.06.016>
- Bae, C.H., 2009. A simulation study of installation locations and capacity of regenerative absorption inverters in DC 1500 V electric railways system. *Simulation Modelling Practice and Theory* 17, 829–838. <https://doi.org/10.1016/j.simpat.2009.02.003>
- Baek, T., Fogel, D.B., Michalewicz, Z., 1997. *Handbook of Evolutionary Computation*. CRC Press. <https://doi.org/10.1201/9781420050387>
- Barrero, R., Mierlo, J.V., Tackoen, X., 2008. Energy savings in public transport. *IEEE Vehicular Technology Magazine* 3, 26–36. <https://doi.org/10.1109/MVT.2008.927485>
- Bellman, R.E., Zadeh, L.A., 1970. Decision-Making in a Fuzzy Environment. *Management Science* 17, 141–164. <https://doi.org/10.1287/mnsc.17.4.B141>
- Beusen, B., Degraeuwe, B., Debeuf, P., 2013. Energy savings in light rail through the optimization of heating and ventilation. *Transportation Research Part D: Transport and Environment* 23, 50–54. <https://doi.org/10.1016/j.trd.2013.03.005>
- Binitha, S., Sathya, S.S., 2012. A survey of Bio inspired optimization algorithms. *International Journal of Soft Computing and Engineering* 2, 137–151.
- Bocharnikov, Y.V., Tobias, A.M., Roberts, C., 2010. Reduction of train and net energy consumption using genetic algorithms for Trajectory Optimisation, in: *IET Conference on Railway Traction Systems (RTS 2010)*. Presented at the IET Conference on Railway Traction Systems (RTS 2010), pp. 1–5. <https://doi.org/10.1049/ic.2010.0038>
- Bocharnikov, Y.V., Tobias, A.M., Roberts, C., Hillmansen, S., Goodman, C.J., 2007. Optimal driving strategy for traction energy saving on DC suburban railways. *IET*

Electric Power Applications 1, 675–682. <https://doi.org/10.1049/iet-epa:20070005>

- Bombardier, 2010. Aeroefficient optimised train shaping. Hennigsdorf: Bombardier.
- Brest, J., Zumer, V., Maucec, M.S., 2006. Self-Adaptive Differential Evolution Algorithm in Constrained Real-Parameter Optimization, in: 2006 IEEE International Conference on Evolutionary Computation. Presented at the 2006 IEEE International Conference on Evolutionary Computation, pp. 215–222. <https://doi.org/10.1109/CEC.2006.1688311>
- Burton, A.D., 2009. Automatic train operation (ATO) specification for Thameslink. Electric Railway (London, England : 2001) 54.
- Carvajal-Carreño, W., Cucala, A.P., Fernández-Cardador, A., 2016. Fuzzy train tracking algorithm for the energy efficient operation of CBTC equipped metro lines. Engineering Applications of Artificial Intelligence 53, 19–31. <https://doi.org/10.1016/j.engappai.2016.03.011>
- Carvajal-Carreño, W., Cucala, A.P., Fernández-Cardador, A., 2014. Optimal design of energy-efficient ATO CBTC driving for metro lines based on NSGA-II with fuzzy parameters. Engineering Applications of Artificial Intelligence 36, 164–177. <https://doi.org/10.1016/j.engappai.2014.07.019>
- Cerf, R., 1998. Asymptotic convergence of genetic algorithms. Advances in Applied Probability 30, 521–550. <https://doi.org/10.1239/aap/1035228082>
- Chanas, S., Kołodziejczyk, W., Machaj, A., 1984. A fuzzy approach to the transportation problem. Fuzzy Sets and Systems 13, 211–221. [https://doi.org/10.1016/0165-0114\(84\)90057-5](https://doi.org/10.1016/0165-0114(84)90057-5)
- Chang, C.S., Sim, S.S., 1997. Optimising train movements through coast control using genetic algorithms. Electric Power Applications, IEE Proceedings - 144, 65–73. <https://doi.org/10.1049/ip-epa:19970797>
- Chang, C.S., Xu, D.Y., 2000. Differential evolution based tuning of fuzzy automatic train operation for mass rapid transit system. Electric Power Applications, IEE Proceedings - 147, 206–212. <https://doi.org/10.1049/ip-epa:20000329>
- Chang, C.S., Xu, D.Y., Quek, H.B., 1999. Pareto-optimal set based multiobjective tuning of fuzzy automatic train operation for mass transit system. Electric Power Applications, IEE Proceedings - 146, 577–583. <https://doi.org/10.1049/ip-epa:19990481>
- Chang, C.T., 2010. An Approximation Approach for Representing S-Shaped Membership Functions. IEEE Transactions on Fuzzy Systems 18, 412–424. <https://doi.org/10.1109/TFUZZ.2010.2042961>

- Chevrier, R., Pellegrini, P., Rodriguez, J., 2013. Energy saving in railway timetabling: A bi-objective evolutionary approach for computing alternative running times. *Transportation Research Part C: Emerging Technologies* 37, 20–41. <https://doi.org/10.1016/j.trc.2013.09.007>
- Chiang, T., Chow, Y., 1994. The Asymptotic Behavior of Simulated Annealing Processes with Absorption. *SIAM J. Control Optim.* 32, 1247–1265. <https://doi.org/10.1137/S0363012989166538>
- Chuang, H.-J., Chen, C.-S., Lin, C.-H., Hsieh, C.-H., Ho, C.-Y., 2009. Design of Optimal Coasting Speed for MRT Systems Using ANN Models. *IEEE Transactions on Industry Applications* 45, 2090–2097. <https://doi.org/10.1109/TIA.2009.2031898>
- Chuang, H.-J., Chen, C.-S., Lin, C.-H., Hsieh, C.-H., Ho, C.-Y., 2008. Design of optimal coasting speed for saving social cost in Mass Rapid Transit systems. Presented at the Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, 2008. DRPT 2008, pp. 2833–2839. <https://doi.org/10.1109/DRPT.2008.4523892>
- Ciccarelli, F., Iannuzzi, D., Tricoli, P., 2012. Control of metro-trains equipped with onboard supercapacitors for energy saving and reduction of power peak demand. *Transportation Research Part C: Emerging Technologies* 24, 36–49. <https://doi.org/10.1016/j.trc.2012.02.001>
- Coello, C.A.C., Pulido, G.T., Lechuga, M.S., 2004. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation* 8, 256–279. <https://doi.org/10.1109/TEVC.2004.826067>
- Coello Coello, C.A., 2002. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering* 191, 1245–1287. [https://doi.org/10.1016/S0045-7825\(01\)00323-1](https://doi.org/10.1016/S0045-7825(01)00323-1)
- Coello Coello, C.A., Lechuga, M.S., 2002. MOPSO: a proposal for multiple objective particle swarm optimization, in: *Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02*. Presented at the Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02, pp. 1051–1056. <https://doi.org/10.1109/CEC.2002.1004388>
- Coleman, D., Howlett, P., Pudney, P., Xuan Vu, Yee, R., 2010. Coasting boards vs optimal control. Presented at the IET Conference on Railway Traction Systems (RTS 2010), IET, pp. 1–5. <https://doi.org/10.1049/ic.2010.0020>
- Cornic, D., 2010. Efficient recovery of braking energy through a reversible dc substation. Presented at the Electrical Systems for Aircraft, Railway and Ship Propulsion (ESARS), 2010, pp. 1–9. <https://doi.org/10.1109/ESARS.2010.5665264>

- Council Directive, 1996. 96/48/EC of 23 July 1996 on the Interoperability of the Trans-European High-Speed Rail System. Official Journal of the European Union L 235, 17/09/1996 6–24.
- Cruz, C., González, J.R., Pelta, D.A., 2011. Optimization in dynamic environments: a survey on problems, methods and measures. *Soft Comput* 15, 1427–1448. <https://doi.org/10.1007/s00500-010-0681-0>
- Cucala, A.P., Fernández, A., Domínguez, M., Ortega, J.M., Ramos, L., Galarraga, A., 2012a. ATO ecodriving design to minimise energy consumption in Metro de Bilbao. Presented at the Thirteenth International Conference on Design and Operation in Railway Engineering (COMPRAIL 2012), New Forest, United Kingdom, pp. 593–601. <https://doi.org/10.2495/CR120501>
- Cucala, A.P., Fernández, A., Sicre, C., Domínguez, M., 2012b. Fuzzy optimal schedule of high speed train operation to minimize energy consumption with uncertain delays and driver's behavioral response. *Engineering Applications of Artificial Intelligence* 25, 1548–1557. <https://doi.org/10.1016/j.engappai.2012.02.006>
- Das, I., Dennis, J., 1998. Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems. *SIAM J. Optim.* 8, 631–657. <https://doi.org/10.1137/S1052623496307510>
- Das, S., Abraham, A., Chakraborty, U.K., Konar, A., 2009. Differential Evolution Using a Neighborhood-Based Mutation Operator. *IEEE Transactions on Evolutionary Computation* 13, 526–553. <https://doi.org/10.1109/TEVC.2008.2009457>
- Das, S., Mullick, S.S., Suganthan, P.N., 2016. Recent advances in differential evolution – An updated survey. *Swarm and Evolutionary Computation* 27, 1–30. <https://doi.org/10.1016/j.swevo.2016.01.004>
- Das, S., Suganthan, P.N., 2011. Differential Evolution: A Survey of the State-of-the-Art. *IEEE Transactions on Evolutionary Computation* 15, 4–31. <https://doi.org/10.1109/TEVC.2010.2059031>
- De Cuadra, F., Fernandez, A., de Juan, J., Herrero, M.A., 1996. Energy-saving automatic optimisation of train speed commands using direct search techniques. Presented at the Computers in Railways V, Wessex Institute of Technology.
- Deb, K., 2010. Recent Developments in Evolutionary Multi-Objective Optimization, in: *Trends in Multiple Criteria Decision Analysis, International Series in Operations Research & Management Science*. Springer, Boston, MA, pp. 339–368. https://doi.org/10.1007/978-1-4419-5904-1_12
- Deb, K., 2000. An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering* 186, 311–338. [https://doi.org/10.1016/S0045-7825\(99\)00389-8](https://doi.org/10.1016/S0045-7825(99)00389-8)

- Deb, K., Bhaskara Rao, N.U., Karthik, S., 2007. Dynamic Multi-objective Optimization and Decision-making Using Modified NSGA-II: A Case Study on Hydro-thermal Power Scheduling, in: Proceedings of the 4th International Conference on Evolutionary Multi-Criterion Optimization, EMO'07. Springer-Verlag, Berlin, Heidelberg, pp. 803–817. https://doi.org/10.1007/978-3-540-70928-2_60
- Deb, K., Gupta, H., 2005. Searching for Robust Pareto-Optimal Solutions in Multi-objective Optimization, in: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (Eds.), Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 150–164.
- Deb, K., Jain, H., 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation* 18, 577–601. <https://doi.org/10.1109/TEVC.2013.2281535>
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 182–197. <https://doi.org/10.1109/4235.996017>
- Dhahbi, S., Abbas-Turki, A., Hayat, S., Moudni, A.E., 2011. Study of the high-speed trains positioning system: European signaling system ERTMS / ETCS, in: 2011 4th International Conference on Logistics. Presented at the 2011 4th International Conference on Logistics, pp. 468–473. <https://doi.org/10.1109/LOGISTIQUA.2011.5939444>
- Domínguez, M., Cucala, A. P., Fernandez, A., Pecharromán, R. R, Blanquer, J., 2011a. Energy efficiency on train control: design of metro ATO driving and impact of energy accumulation devices. Presented at the World Congress on Railway Research.
- Domínguez, M., Fernández, A., Cucala, A.P., Lukaszewicz, P., 2011b. Optimal design of metro automatic train operation speed profiles for reducing energy consumption. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* 225, 463–474. <https://doi.org/10.1177/09544097JRRRT420>
- Domínguez, M., Fernández-Cardador, A., Cucala, A.P., Gonsalves, T., Fernández, A., 2014. Multi objective particle swarm optimization algorithm for the design of efficient ATO speed profiles in metro lines. *Engineering Applications of Artificial Intelligence* 29, 43–53. <https://doi.org/10.1016/j.engappai.2013.12.015>
- Domínguez, M., Fernandez-Cardador, A., Cucala, A.P., Pecharroman, R.R., 2012. Energy Savings in Metropolitan Railway Substations Through Regenerative Energy Recovery and Optimal Design of ATO Speed Profiles. *IEEE Transactions on Automation Science and Engineering* 9, 496–504. <https://doi.org/10.1109/TASE.2012.2201148>

- Dorigo, M., Birattari, M., Stutzle, T., 2006. Ant colony optimization. *IEEE Computational Intelligence Magazine* 1, 28–39. <https://doi.org/10.1109/MCI.2006.329691>
- Douglas, H., Roberts, C., Hillmansen, S., Schmid, F., 2015. An assessment of available measures to reduce traction energy use in railway networks. *Energy Conversion and Management* 106, 1149–1165. <https://doi.org/10.1016/j.enconman.2015.10.053>
- Dubois, D., Prade, H., 1983. Ranking fuzzy numbers in the setting of possibility theory. *Information Sciences* 30, 183–224. [https://doi.org/10.1016/0020-0255\(83\)90025-7](https://doi.org/10.1016/0020-0255(83)90025-7)
- Dullinger, C., Struckl, W., Kozek, M., 2017. Simulation-based multi-objective system optimization of train traction systems. *Simulation Modelling Practice and Theory* 72, 104–117. <https://doi.org/10.1016/j.simpat.2016.12.008>
- Emery, D., 2017. Towards Automatic Train Operation for long distance services: State-of-the art and challenges. 17th Swiss transport research conference.
- Falvo, M.C., Lamedica, R., Bartoni, R., Maranzano, G., 2011. Energy management in metro-transit systems: An innovative proposal toward an integrated and sustainable urban mobility system including plug-in electric vehicles. *Electric Power Systems Research* 81, 2127–2138. <https://doi.org/10.1016/j.epsr.2011.08.004>
- Fay, A., 2000. A fuzzy knowledge-based system for railway traffic control. *Engineering Applications of Artificial Intelligence* 13, 719–729. [https://doi.org/10.1016/S0952-1976\(00\)00027-0](https://doi.org/10.1016/S0952-1976(00)00027-0)
- Feng, X., Feng, J., Wu, K., Liu, H., Sun, Q., 2012. Evaluating target speeds of passenger trains in China for energy saving in the effect of different formation scales and traction capacities. *International Journal of Electrical Power & Energy Systems* 42, 621–626. <https://doi.org/10.1016/j.ijepes.2012.04.055>
- Feng, X., Liu, H., Zhang, H., Wang, B., Sun, Q., 2013a. Rational Formations of a Metro Train Improve Its Efficiencies of Both Traction Energy Utilization and Passenger Transport. *Mathematical Problems in Engineering* 2013, 7 pages. <https://doi.org/10.1155/2013/643274>
- Feng, X., Wang, Q., Liu, Y., Xu, B., Liu, H., Sun, Q., 2014. Ensuring A Reasonable Passenger Capacity Utilization Rate of a Train for Its Sustainably Efficient Transport. *Journal of Applied Research and Technology* 12, 279–288. [https://doi.org/10.1016/S1665-6423\(14\)72344-2](https://doi.org/10.1016/S1665-6423(14)72344-2)
- Feng, X., Zhang, H., Ding, Y., Liu, Z., Peng, H., Xu, B., 2013b. A Review Study on Traction Energy Saving of Rail Transport. *Discrete Dynamics in Nature and Society* 2013, 9 pages. <https://doi.org/10.1155/2013/156548>

- Fernandez, A., Cucala, A.P., Vitoriano, B., Cuadra, F. de, 2006. Predictive Traffic Regulation for Metro Loop Lines Based on Quadratic Programming. Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 220, 79–89. <https://doi.org/10.1243/09544097F00505>
- Fleischer, M., 2003. The Measure of Pareto Optima Applications to Multi-objective Metaheuristics, in: Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science. Presented at the International Conference on Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, pp. 519–533. https://doi.org/10.1007/3-540-36970-8_37
- Fogel, L.J., Owens, A.J., Walsh, M.J., 1966. Artificial Intelligence Through Simulated Evolution. John Wiley & Sons.
- Franke, R., Terwiesch, P., Meyer, M., 2000. An algorithm for the optimal control of the driving of trains, in: Proceedings of the 39th IEEE Conference on Decision and Control, 2000. Presented at the Proceedings of the 39th IEEE Conference on Decision and Control, 2000, pp. 2123–2128 vol.3. <https://doi.org/10.1109/CDC.2000.914108>
- Gao, S., Zhang, Z., Cao, C., 2009. Multiplication Operation on Fuzzy Numbers. Journal of Software 4, 331–338.
- Gill, D.C., Goodman, C.J., 1992. Computer-based optimisation techniques for mass transit railway signalling design. IEE Proceedings B Electric Power Applications 139, 261. <https://doi.org/10.1049/ip-b.1992.0031>
- Givoni, M., 2007. Environmental Benefits from Mode Substitution: Comparison of the Environmental Impact from Aircraft and High-Speed Train Operations. International Journal of Sustainable Transportation 1, 209–230. <https://doi.org/10.1080/15568310601060044>
- Glover, F., Kochenberger, G.A., 2003. Handbook of Metaheuristics. Springer.
- Goldberg, D., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional.
- Goodman, C.J., Siu, L.K., Ho, T.K., 1998. A review of simulation models for railway systems, in: 1998 International Conference on Developments in Mass Transit Systems Conf. Publ. No. 453). Presented at the 1998 International Conference on Developments in Mass Transit Systems Conf. Publ. No. 453), pp. 80–85. <https://doi.org/10.1049/cp:19980101>
- Gu, Q., Lu, X.-Y., Tang, T., 2011. Energy saving for automatic train control in moving block signaling system. Presented at the 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 1305–1310. <https://doi.org/10.1109/ITSC.2011.6082964>

- Gu, Q., Tang, T., Cao, F., Song, Y., 2014. Energy-Efficient Train Operation in Urban Rail Transit Using Real-Time Traffic Information. *IEEE Transactions on Intelligent Transportation Systems* 15, 1216–1233. <https://doi.org/10.1109/TITS.2013.2296655>
- Hasegawa, D., Nicholson, G.L., Roberts, C., Schmid, F., 2016. Standardised approach to energy consumption calculations for high-speed rail. *IET Electrical Systems in Transportation* 179–189. <https://doi.org/10.1049/iet-est.2015.0002>
- Hassannayebi, E., Zegordi, S.H., 2017. Variable and adaptive neighbourhood search algorithms for rail rapid transit timetabling problem. *Computers & Operations Research* 78, 439–453. <https://doi.org/10.1016/j.cor.2015.12.011>
- Helbig, M., Engelbrecht, A.P., 2014. Population-based metaheuristics for continuous boundary-constrained dynamic multi-objective optimisation problems. *Swarm and Evolutionary Computation* 14, 31–47. <https://doi.org/10.1016/j.swevo.2013.08.004>
- Holland, J.H., 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press.
- Howlett, P., 1996. Optimal strategies for the control of a train. *Automatica* 32, 519–532. [https://doi.org/10.1016/0005-1098\(95\)00184-0](https://doi.org/10.1016/0005-1098(95)00184-0)
- Howlett, P., 1990. An optimal strategy for the control of a train. *J. Australian Math. Soc. Ser. B.* 31, 454–471. <https://doi.org/10.1017/S0334270000006780>
- Howlett, P., 1988. Existence of an optimal strategy for the control of a train. *School of Mathematics report 3*.
- Howlett, P.G., Cheng, J., Pudney, P.J., 1995. Optimal Strategies for Energy-Efficient Train Control, in: *Control Problems in Industry*. Birkhäuser Boston, pp. 151–178. https://doi.org/10.1007/978-1-4612-2580-5_7
- Howlett, P.G., Milroy, I.P., Pudney, P.J., 1994. Energy-efficient train control. *Control Engineering Practice* 2, 193–200. [https://doi.org/10.1016/0967-0661\(94\)90198-8](https://doi.org/10.1016/0967-0661(94)90198-8)
- Howlett, P.G., Pudney, P.J., Vu, X., 2009. Local energy minimization in optimal train control. *Automatica* 45, 2692–2698. <https://doi.org/10.1016/j.automatica.2009.07.028>
- Huang, Y., Ma, X., Su, S., Tang, T., 2015. Optimization of Train Operation in Multiple Interstations with Multi-Population Genetic Algorithm. *Energies* 8, 14311–14329. <https://doi.org/10.3390/en81212433>
- Hwang, H.-S., 1998. Control strategy for optimal compromise between trip time and energy consumption in a high-speed railway. *IEEE Transactions on Systems, Man*

- and Cybernetics, Part A: Systems and Humans 28, 791–802. <https://doi.org/10.1109/3468.725350>
- Ibaiondo, H., Romo, A., 2010. Kinetic energy recovery on railway systems with feedback to the grid, in: Proceedings of 14th International Power Electronics and Motion Control Conference EPE-PEMC 2010. Presented at the Proceedings of 14th International Power Electronics and Motion Control Conference EPE-PEMC 2010, pp. T9-94-T9-97. <https://doi.org/10.1109/EPEPEMC.2010.5606545>
- Ichikawa, K., 1968. Application of Optimization Theory for Bounded State Variable Problems to the Operation of Train. Bulletin of JSME 11, 857–865. <https://doi.org/10.1299/jsme1958.11.857>
- IEA, UIC, 2017. Railway Handbook 2017. Energy Consumption and CO2 Emissions.
- IEA, UIC, 2016. Railway Handbook 2016. Energy Consumption and CO2 Emissions.
- IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements, 1999. . IEEE Std 1474.1-1999 i. <https://doi.org/10.1109/IEEESTD.1999.90611>
- Jain, H., Deb, K., 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. IEEE Transactions on Evolutionary Computation 18, 602–622. <https://doi.org/10.1109/TEVC.2013.2281534>
- Ji, H., Hou, Z., Zhang, R., 2016. Adaptive Iterative Learning Control for High-Speed Trains With Unknown Speed Delays and Input Saturations. IEEE Transactions on Automation Science and Engineering 13, 260–273. <https://doi.org/10.1109/TASE.2014.2371816>
- Jia, L.-M., Zhang, X.-D., 1994. Distributed intelligent railway traffic control: A fuzzy-decisionmaking-based approach. Engineering Applications of Artificial Intelligence 7, 311–319. [https://doi.org/10.1016/0952-1976\(94\)90058-2](https://doi.org/10.1016/0952-1976(94)90058-2)
- Jiménez, F., Sánchez, G., Vasant, P., 2013. A multi-objective evolutionary approach for fuzzy optimization in production planning. Journal of Intelligent and Fuzzy Systems 25, 441–455. <https://doi.org/10.3233/IFS-130651>
- Jong, J.-C., Chang, S., 2005. Algorithms for generating train speed profiles. Journal of the Eastern Asia Society for Transportation Studies 6, 356–371. <https://doi.org/10.11175/easts.6.356>
- Karaboga, D., Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Glob Optim 39, 459–471. <https://doi.org/10.1007/s10898-007-9149-x>

- Ke, B.-R., Chen, M.-C., Lin, C.-L., 2009. Block-Layout Design Using MAX-MIN Ant System for Saving Energy on Mass Rapid Transit Systems. *IEEE Transactions on Intelligent Transportation Systems* 10, 226–235. <https://doi.org/10.1109/TITS.2009.2018324>
- Ke, B.-R., Lin, C.-L., Yang, C.-C., 2012. Optimisation of train energy-efficient operation for mass rapid transit systems. *IET Intelligent Transport Systems* 6, 58–66. <https://doi.org/10.1049/iet-its.2010.0144>
- Kennedy, J., Eberhart, R., 1995. Particle swarm optimization, in: , *IEEE International Conference on Neural Networks, 1995. Proceedings. Presented at the , IEEE International Conference on Neural Networks, 1995. Proceedings*, pp. 1942–1948 vol.4. <https://doi.org/10.1109/ICNN.1995.488968>
- Kennedy, J., Eberhart, R.C., 2001. *Swarm intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Kennedy, J., Eberhart, R.C., 1997. A discrete binary version of the particle swarm algorithm. Presented at the , *1997 IEEE International Conference on Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation*, pp. 4104–4108 vol.5. <https://doi.org/10.1109/ICSMC.1997.637339>
- Keskin, K., Karamancioglu, A., 2017. Energy-Efficient Train Operation Using Nature-Inspired Algorithms. *Journal of Advanced Transportation* 2017, 12. <https://doi.org/10.1155/2017/6173795>
- Khmelnitsky, E., 2000. On an optimal control problem of train operation. *IEEE Transactions on Automatic Control* 45, 1257–1266. <https://doi.org/10.1109/9.867018>
- Kim, D.G., Husbands, P., 1998a. Landscape changes and the performance of Mapping Based Constraint handling methods, in: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (Eds.), *Parallel Problem Solving from Nature — PPSN V, Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 221–229.
- Kim, D.G., Husbands, P., 1998b. Mapping Based Constraint Handling for Evolutionary Search; Thurston’s Circle Packing and Grid Generation, in: Parmee, I.C. (Ed.), *Adaptive Computing in Design and Manufacture*. Springer London, pp. 161–173.
- Kim, Y.-G., Jeon, C.-S., Kim, S.-W., Park, T.-W., 2013. Operating speed pattern optimization of railway vehicles with differential evolution algorithm. *Int.J Automot. Technol.* 14, 903–911. <https://doi.org/10.1007/s12239-013-0099-7>
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by Simulated Annealing. *Science* 220, 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Knowles, J., Corne, D., 1999. The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation, in: *Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99. Presented at the*

- Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99, p. 105 Vol. 1. <https://doi.org/10.1109/CEC.1999.781913>
- Knowles, J.D., Corne, D.W., 2000. Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation* 8, 149–172. <https://doi.org/10.1162/106365600568167>
- Ko, H., Koseki, T., Miyatake, M., 2004. Application of dynamic programming to the optimization of the running profile of a train. Presented at the Computers in Railways IX, WIT Press.
- Kondo, M., Miyabe, M., Manabe, S., 2014. Development of a High Efficiency Induction Motor and the Estimation of Energy Conservation Effect. *QR of RTRI* 55, 138–143. <https://doi.org/10.2219/rtriq.55.138>
- Koza, J.R., 1994. Genetic programming as a means for programming computers by natural selection. *Stat Comput* 4, 87–112. <https://doi.org/10.1007/BF00175355>
- Lechelle, S.A., Mouneimne, Z.S., 2010. OptiDrive: a practical approach for the calculation of energy-optimised operating speed profiles. *IET*, pp. 23–23. <https://doi.org/10.1049/ic.2010.0029>
- Lee, H., Jung, S., Cho, Y., Yoon, D., Jang, G., 2013. Peak power reduction and energy efficiency improvement with the superconducting flywheel energy storage in electric railway system. *Physica C: Superconductivity* 494, 246–249. <https://doi.org/10.1016/j.physc.2013.04.033>
- Li, X., 2003. A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization, in: Cantú-Paz, E., Foster, J.A., Deb, K., Davis, L.D., Roy, R., O'Reilly, U.-M., Beyer, H.-G., Standish, R., Kendall, G., Wilson, S., Harman, M., Wegener, J., Dasgupta, D., Potter, M.A., Schultz, A.C., Dowsland, K.A., Jonoska, N., Miller, J. (Eds.), *Genetic and Evolutionary Computation — GECCO 2003, Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 37–48.
- Li, X., Lo, H.K., 2014. An energy-efficient scheduling and speed control approach for metro rail operations. *Transportation Research Part B: Methodological* 64, 73–89. <https://doi.org/10.1016/j.trb.2014.03.006>
- Li, Z., Liang, J.J., He, X., Shang, Z., 2010. Differential evolution with dynamic constraint-handling mechanism, in: *IEEE Congress on Evolutionary Computation*. Presented at the IEEE Congress on Evolutionary Computation, pp. 1–8. <https://doi.org/10.1109/CEC.2010.5586539>
- Lin, W.-S., Sheu, J.-W., 2011a. Optimization of Train Regulation and Energy Usage of Metro Lines Using an Adaptive-Optimal-Control Algorithm. *IEEE Transactions on Automation Science and Engineering* 8, 855–864. <https://doi.org/10.1109/TASE.2011.2160537>

- Lin, W.-S., Sheu, J.-W., 2011b. Metro Traffic Regulation by Adaptive Optimal Control. *IEEE Transactions on Intelligent Transportation Systems* 12, 1064–1073. <https://doi.org/10.1109/TITS.2011.2142306>
- Lin, W.-S., Sheu, J.-W., 2008. Adaptive critic design of automatic train regulation of MRT system. Presented at the IEEE International Conference on Industrial Technology, 2008. ICIT 2008, pp. 1–7. <https://doi.org/10.1109/ICIT.2008.4608405>
- Liu, R., Golovitcher, I.M., 2003. Energy-efficient operation of rail vehicles. *Transportation Research Part A: Policy and Practice* 37, 917–932. <https://doi.org/10.1016/j.tra.2003.07.001>
- López-López, Á.J., Pecharromán, R.R., Fernández-Cardador, A., Cucala, A.P., 2014. Assessment of energy-saving techniques in direct-current-electrified mass transit systems. *Transportation Research Part C: Emerging Technologies* 38, 85–100. <https://doi.org/10.1016/j.trc.2013.10.011>
- Lu, S., Hillmansen, S., Ho, T.K., Roberts, C., 2013. Single-Train Trajectory Optimization. *IEEE Transactions on Intelligent Transportation Systems* 14, 743–750. <https://doi.org/10.1109/TITS.2012.2234118>
- Lu, S., Weston, P., Hillmansen, S., Gooi, H.B., Roberts, C., 2014. Increasing the Regenerative Braking Energy for Railway Vehicles. *IEEE Transactions on Intelligent Transportation Systems* 15, 2506–2515. <https://doi.org/10.1109/TITS.2014.2319233>
- Lv, X., He, B., Li, Y., 2013. Inbound Energy-Saving Slope Design Method of High-Speed Railway. Presented at the Fourth International Conference on Transportation Engineering, Chengdu, China.
- Matsuoka, K., Kondo, M., 2014. Energy Saving Technologies for Railway Traction Motors. *IEEJ Transactions on Electrical and Electronic Engineering* 5, 278–284. <https://doi.org/10.1002/tee.20530>
- Mezura-Montes, E., Coello Coello, C.A., 2011. Constraint-handling in nature-inspired numerical optimization: Past, present and future. *Swarm and Evolutionary Computation* 1, 173–194. <https://doi.org/10.1016/j.swevo.2011.10.001>
- Mezura-Montes, E., Velázquez-Reyes, J., Coello Coello, C.A., 2005. Promising Infeasibility and Multiple Offspring Incorporated to Differential Evolution for Constrained Optimization, in: *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, GECCO '05*. ACM, New York, NY, USA, pp. 225–232. <https://doi.org/10.1145/1068009.1068043>
- Milroy, I.P., 1980. Aspects of automatic train control. PhD. Loughborough University, Loughborough, United Kingdom.

- Mitchell, M., 1996. An introduction to genetic algorithms, Complex adaptive systems. MIT press, Cambridge (Mass.).
- Miyatake, M., Ko, H., 2010. Optimization of Train Speed Profile for Minimum Energy Consumption. *IEEJ Transactions on Electrical and Electronic Engineering* 5, 263–269. <https://doi.org/10.1002/tee.20528>
- Miyatake, M., Matsuda, K., 2009. Energy Saving Speed and Charge/Discharge Control of a Railway Vehicle with On-board Energy Storage by Means of an Optimization Model. *IEEJ Trans Elec Electron Eng* 4, 771–778. <https://doi.org/10.1002/tee.20479>
- Nguyen, T.T., Yang, S., Branke, J., 2012. Evolutionary dynamic optimization: A survey of the state of the art. *Swarm and Evolutionary Computation* 6, 1–24. <https://doi.org/10.1016/j.swevo.2012.05.001>
- Panou, K., Tzieropoulos, P., Emery, D., 2013. Railway driver advice systems: Evaluation of methods, tools and systems. *Journal of Rail Transport Planning & Management* 3, 150–162. <https://doi.org/10.1016/j.jrtpm.2013.10.005>
- Parsopoulos, K.E., Vrahatis, M.N., 2002. Particle swarm optimization method in multiobjective problems, in: *Proceedings of the 2002 ACM Symposium on Applied Computing, SAC '02*. ACM, New York, NY, USA, pp. 603–607. <https://doi.org/10.1145/508791.508907>
- Peña-Alcaraz, M., Fernandez, A., Cucala, A.P., Ramos, A., Pecharromán, R.R., 2011. Optimal underground timetable design based on power flow for maximizing the use of regenerative-braking energy. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* 226, 397–408. <https://doi.org/10.1177/0954409711429411>
- Powell, D., Skolnick, M.M., 1993. Using Genetic Algorithms in Engineering Design Optimization with Non-Linear Constraints, in: *Proceedings of the 5th International Conference on Genetic Algorithms*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 424–431.
- Pudney, P., Howlett, P.G., Albrecht, A., Coleman, D., Vu, X., Koelewijn, J., 2011. Optimal Driving Strategies with Intermediate Timing Points. Presented at the The 10th International Heavy Haul Association Conference, Calgary.
- Quaglietta, E., Pellegrini, P., Goverde, R.M.P., Albrecht, T., Jaekel, B., Marlière, G., Rodriguez, J., Dollevoet, T., Ambrogio, B., Carcasole, D., Giaroli, M., Nicholson, G., 2016. The ON-TIME real-time railway traffic management framework: A proof-of-concept using a scalable standardised data communication architecture. *Transportation Research Part C: Emerging Technologies* 63, 23–50. <https://doi.org/10.1016/j.trc.2015.11.014>
- Raquel, C.R., Naval, Jr., P.C., 2005. An effective use of crowding distance in multiobjective particle swarm optimization, in: *Proceedings of the 2005*

Conference on Genetic and Evolutionary Computation, GECCO '05. ACM, New York, NY, USA, pp. 257–264. <https://doi.org/10.1145/1068009.1068047>

- Roch-Dupré, D., Cucala, A.P., R. Pecharromán, R., López-López, Á.J., Fernández-Cardador, A., 2018. Evaluation of the impact that the traffic model used in railway electrical simulation has on the assessment of the installation of a Reversible Substation. *International Journal of Electrical Power & Energy Systems* 102, 201–210. <https://doi.org/10.1016/j.ijepes.2018.04.030>
- Roch-Dupré, D., López-López, Á.J., Pecharromán, R.R., Cucala, A.P., Fernández-Cardador, A., 2017. Analysis of the demand charge in DC railway systems and reduction of its economic impact with Energy Storage Systems. *International Journal of Electrical Power & Energy Systems* 93, 459–467. <https://doi.org/10.1016/j.ijepes.2017.06.022>
- Rodrigo, E., Tapia, S., Mera, J., Soler, M., 2013. Optimizing Electric Rail Energy Consumption Using the Lagrange Multiplier Technique. *Journal of Transportation Engineering* 139, 321–329. [https://doi.org/10.1061/\(ASCE\)TE.1943-5436.0000483](https://doi.org/10.1061/(ASCE)TE.1943-5436.0000483)
- Salazar Lechuga, M., 2009. Multi-objective optimisation using sharing in swarm optimisation algorithms (d_ph). University of Birmingham.
- Sheu, J.-W., Lin, W.-S., 2012. Energy-Saving Automatic Train Regulation Using Dual Heuristic Programming. *IEEE Transactions on Vehicular Technology* 61, 1503–1514. <https://doi.org/10.1109/TVT.2012.2187225>
- Sheu, J.-W., Lin, W.-S., 2011. Automatic train regulation with energy saving using dual heuristic programming. *IET Electrical Systems in Transportation* 1, 80–89. <https://doi.org/10.1049/iet-est.2010.0074>
- Sicre, C., 2013. Diseño eficiente de servicios ferroviarios y control de la conducción en alta velocidad. Comillas Pontifical University, Madrid.
- Sicre, C., Cucala, A.P., Fernández, A., Lukaszewicz, P., 2012. Modeling and optimizing energy-efficient manual driving on high-speed lines. *IEEJ Transactions on Electrical and Electronic Engineering* 7, 633–640. <https://doi.org/10.1002/tee.21782>
- Sicre, C., Cucala, A.P., Fernández-Cardador, A., 2014. Real time regulation of efficient driving of high speed trains based on a genetic algorithm and a fuzzy model of manual driving. *Engineering Applications of Artificial Intelligence* 29, 79–92. <https://doi.org/10.1016/j.engappai.2013.07.015>
- Srinivas, N., Deb, K., 1994. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation* 2, 221–248. <https://doi.org/10.1162/evco.1994.2.3.221>

- Storn, R., Price, K., 1997. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization* 11, 341–359. <https://doi.org/10.1023/A:1008202821328>
- Strobel, H., Horn, P., 1974. Contribution to optimum computer-aided control of train operation. *2nd Symposium on Traffic Control and Transportation Systems* 377–387.
- Su, S., Li, X., Tang, T., Gao, Z., 2013. A Subway Train Timetable Optimization Approach Based on Energy-Efficient Operation Strategy. *IEEE Transactions on Intelligent Transportation Systems* 14, 883–893. <https://doi.org/10.1109/TITS.2013.2244885>
- TEN-T - ATO project, 2016. ATO over ERTMS Operatinal Requirements (No. v 1.7, 13E137).
- Tian, Z., Weston, P., Zhao, N., Hillmansen, S., Roberts, C., Chen, L., 2017. System energy optimisation strategies for metros with regeneration. *Transportation Research Part C: Emerging Technologies* 75, 120–135. <https://doi.org/10.1016/j.trc.2016.12.004>
- UNIFE/Roland Berger, 2016. World rail market study: Forecast 2016 to 2021.
- United Nations, 2015. Adoption of the Paris Agreement, in: *Framework Convention on Climate Change, 21st Conference of the Parties*. Paris: United Nations.
- Vasant, P., Ganesan, T., Elamvazuthi, I., 2012. Improved tabu search recursive fuzzy method for crude oil industry. *Int. J. Model. Simul. Sci. Comput.* 03, 1150002. <https://doi.org/10.1142/S1793962311500024>
- Villalba, M., 2016. Pioneering ATO over ETCS Level 2. *Railway Gazette International* 172.
- Wang, L., Qin, Y., Xu, J., Jia, L., 2012. A Fuzzy Optimization Model for High-Speed Railway Timetable Rescheduling. *Discrete Dynamics in Nature and Society* 2012, 22 pages. <https://doi.org/10.1155/2012/827073>
- Wang, P., Goverde, R.M.P., 2017. Multi-train trajectory optimization for energy efficiency and delay recovery on single-track railway lines. *Transportation Research Part B: Methodological* 105, 340–361. <https://doi.org/10.1016/j.trb.2017.09.012>
- Wang, P., Goverde, R.M.P., 2016a. Multiple-phase train trajectory optimization with signalling and operational constraints. *Transportation Research Part C: Emerging Technologies* 69, 255–275. <https://doi.org/10.1016/j.trc.2016.06.008>
- Wang, P., Goverde, R.M.P., 2016b. Two-Train Trajectory Optimization with a Green-Wave Policy. *Transportation Research Record: Journal of the Transportation Research Board*.

- Wang, Y., De Schutter, B., van den Boom, T.J.J., Ning, B., 2014. Optimal trajectory planning for trains under fixed and moving signaling systems using mixed integer linear programming. *Control Engineering Practice* 22, 44–56. <https://doi.org/10.1016/j.conengprac.2013.09.011>
- Wang, Y., De Schutter, B., van den Boom, T.J.J., Ning, B., 2013. Optimal trajectory planning for trains – A pseudospectral method and a mixed integer linear programming approach. *Transportation Research Part C: Emerging Technologies* 29, 97–114. <https://doi.org/10.1016/j.trc.2013.01.007>
- Wang, Y., Ning, B., Cao, F., De Schutter, B., van den Boom, T.J.J., 2011. A survey on optimal trajectory planning for train operations. Presented at the 2011 IEEE International Conference on Service Operations, Logistics, and Informatics (SOLI), pp. 589–594. <https://doi.org/10.1109/SOLI.2011.5986629>
- Wei, L., Qunzhan, L., Bing, T., 2009. Energy Saving Train Control for Urban Railway Train with Multi-population Genetic Algorithm, in: *International Forum on Information Technology and Applications, 2009. IFITA '09*. Presented at the International Forum on Information Technology and Applications, 2009. IFITA '09, pp. 58–62. <https://doi.org/10.1109/IFITA.2009.283>
- Wong, K.K., Ho, T.K., 2004a. Coast control for mass rapid transit railways with searching methods. *IEE Proceedings - Electric Power Applications* 151, 365–376. <https://doi.org/10.1049/ip-epa:20040346>
- Wong, K.K., Ho, T.K., 2004b. Dynamic coast control of train movement with genetic algorithm. *Intern. J. Syst. Sci.* 35, 835–846. <https://doi.org/10.1080/00207720412331203633>
- Wong, K.K., Ho, T.K., 2003. Coast control of train movement with genetic algorithm, in: *The 2003 Congress on Evolutionary Computation, 2003. CEC '03*. Presented at the The 2003 Congress on Evolutionary Computation, 2003. CEC '03, pp. 1280–1287 Vol.2. <https://doi.org/10.1109/CEC.2003.1299816>
- Xiao, Z., Xia, S., Gong, K., Li, D., 2012. The trapezoidal fuzzy soft set and its application in MCDM. *Applied Mathematical Modelling* 36, 5844–5855. <https://doi.org/10.1016/j.apm.2012.01.036>
- Xie, T., Wang, S., Zhao, X., Zhang, Q., 2013. Optimization of Train Energy-Efficient Operation Using Simulated Annealing Algorithm, in: Li, K., Li, S., Li, D., Niu, Q. (Eds.), *Intelligent Computing for Sustainable Energy and Environment, Communications in Computer and Information Science*. Springer Berlin Heidelberg, pp. 351–359.
- Xin, T., Roberts, C., He, J., Hillmansen, S., Zhao, N., Chen, L., Tian, Z., Su, S., 2014. Railway vertical alignment optimisation at stations to minimise energy, in: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Presented at the 17th International IEEE Conference on Intelligent

- Transportation Systems (ITSC), pp. 2119–2124.
<https://doi.org/10.1109/ITSC.2014.6958016>
- Yan, X., Cai, B., Ning, B., ShangGuan, W., 2016. Online distributed cooperative model predictive control of energy-saving trajectory planning for multiple high-speed train movements. *Transportation Research Part C: Emerging Technologies* 69, 60–78. <https://doi.org/10.1016/j.trc.2016.05.019>
- Yang, H., Zhang, K.-P., Liu, H.-E., 2015. Online Regulation of High Speed Train Trajectory Control Based on T-S Fuzzy Bilinear Model. *IEEE Transactions on Intelligent Transportation Systems* 17, 1496–1508.
<https://doi.org/10.1109/TITS.2015.2497320>
- Yang, J., Jia, L., Lu, S., Fu, Y., Ge, J., 2016. Energy-Efficient Speed Profile Approximation: An Optimal Switching Region-Based Approach with Adaptive Resolution. *Energies* 9, 762. <https://doi.org/10.3390/en9100762>
- Yang, L., Li, K., Gao, Z., Li, X., 2012. Optimizing trains movement on a railway network. *Omega* 40, 619–633. <https://doi.org/10.1016/j.omega.2011.12.001>
- Yang, L., Lidén, T., Leander, P., 2013. Achieving energy-efficiency and on-time performance with Driver Advisory Systems, in: 2013 IEEE International Conference on Intelligent Rail Transportation (ICIRT). Presented at the 2013 IEEE International Conference on Intelligent Rail Transportation (ICIRT), pp. 13–18.
<https://doi.org/10.1109/ICIRT.2013.6696260>
- Yang, Xin, Chen, A., Ning, B., Tang, T., 2016. A stochastic model for the integrated optimization on metro timetable and speed profile with uncertain train mass. *Transportation Research Part B: Methodological* 91, 424–445.
<https://doi.org/10.1016/j.trb.2016.06.006>
- Yang, X., Li, X., Ning, B., Tang, T., 2016. A Survey on Energy-Efficient Train Operation for Urban Rail Transit. *IEEE Transactions on Intelligent Transportation Systems* 17, 2–13. <https://doi.org/10.1109/TITS.2015.2447507>
- Yang, X.-S., 2014. *Nature-Inspired Optimization Algorithms*. Elsevier.
- Yang, X.-S., 2009. Firefly Algorithms for Multimodal Optimization, in: *Proceedings of the 5th International Conference on Stochastic Algorithms: Foundations and Applications, SAGA'09*. Springer-Verlag, Berlin, Heidelberg, pp. 169–178.
- Yang, X.S., Deb, S., 2009. Cuckoo Search via Levy flights, in: *2009 World Congress on Nature Biologically Inspired Computing (NaBIC)*. Presented at the 2009 World Congress on Nature Biologically Inspired Computing (NaBIC), pp. 210–214.
<https://doi.org/10.1109/NABIC.2009.5393690>
- Yin, J., Chen, D., Yang, L., Tang, T., Ran, B., 2016. Efficient Real-Time Train Operation Algorithms With Uncertain Passenger Demands. *IEEE Transactions on Intelligent Transportation Systems* 17, 1–13. <https://doi.org/10.1109/TITS.2015.2478403>

- Zhao, N., Roberts, C., Hillmansen, S., Tian, Z., Weston, P., Chen, L., 2017. An integrated metro operation optimization to minimize energy consumption. *Transportation Research Part C: Emerging Technologies* 75, 168–182. <https://doi.org/10.1016/j.trc.2016.12.013>
- Zhou, P., Xu, H.Z., Zhang, M.N., 2011. Train Optimal Control Strategy on Continuous Change Gradient Steep Downgrades. *Advanced Materials Research* 267, 211–216. <https://doi.org/10.4028/www.scientific.net/AMR.267.211>
- Zielinski, K., Laur, R., 2006. Constrained Single-Objective Optimization Using Differential Evolution, in: 2006 IEEE International Conference on Evolutionary Computation. Presented at the 2006 IEEE International Conference on Evolutionary Computation, pp. 223–230. <https://doi.org/10.1109/CEC.2006.1688312>
- Zitzler, E., 1999. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD. Swiss Federal Institute of Technology (ETH), Zurich.
- Zitzler, E., Laumanns, M., Thiele, L., 2002. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization, in: *Evolutionary Methods for Design, Optimisation, and Control*. CIMNE, Barcelona, Spain, pp. 95–100.
- Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the Strength Pareto Evolutionary Algorithm.
- Zou, F., Wang, L., Hei, X., Chen, D., Wang, B., 2013. Multi-objective optimization using teaching-learning-based optimization algorithm. *Engineering Applications of Artificial Intelligence* 26, 1291–1300. <https://doi.org/10.1016/j.engappai.2012.11.006>

