Multi Objective Ant Colony Optimisation to obtain efficient metro speed profiles

Pablo Martínez Fernández[a] , Juan B. Font Torres[a], Ignacio Villalba Sanchís[a], Ricardo Insa Franco[a]

Abstract

Obtaining efficient speed profiles for metro trains is a multi-objective optimisation problem where energy consumption and travel time must be balanced. Automatic Train Operation (ATO) systems may handle a great number of possible speed profiles; hence optimisation algorithms are required find efficient ones in a timely manner. This paper aims to assess the performance of a particular meta-heuristic optimisation algorithm, a variation of the traditional Ant Colony (ACO) modified to deal with multi-objective problems with continuous variables: MOACOr. This algorithm is used to obtain efficient speed profiles in up to 32 interstation sections in the metro network of Valencia (Spain), and the convergence and diversity of these solution sets is evaluated through metrics such as Inverse Generational Distance (GD) and Normalised Hypervolume (NH). The results are then compared to those obtained with a conventional genetic algorithm (NSGA-II), including a statistical analysis to identify significant differences. It has been found that MOACOr shows a better performance than NSGA-II in terms of convergence, regularity and diversity of the solution. These results indicate that MOACOr is a good alternative to the widely used genetic algorithm and could be a better tool for rail operation managers trying to improve energy efficiency.

Keywords: Energy efficiency, Optimization, Genetic Algorithm, Ant Colony, metro trains

[a] Department of Transport Engineering and Infrastructure. Universitat Politècnica de València (UPV), Spain.
Corresponding Author: Pablo Martínez Fernández. Universitat Politècnica de València (UPV), Camino de Vera s/n, 46022, Valencia, Spain. Phone: +34 96 387 7007 Ext. 73767. Email: pabmarfe@cam.upv.es

1. Introduction

The growing threat of climatic change and scarcity of resources imposes the need for increasing efficiency at all levels of our society. This includes the transport sector, whose impact on CO_2 emissions is as remarkable as its importance for global economy. Railways are, comparatively, one of the most efficient ways of moving passengers and goods [1] and thus have become an investment priority for many public administrations across the world. However, there is still ample room for reducing the energy consumption of railways, thus contributing to a more sustainable transport.

Many studies have focused on this topic over the past years, addressing energy efficiency on railways through different approaches: from track geometry [2], to better aerodynamics [3] or lighter materials [4]. However, the most common approach to energy efficiency on railways is eco-driving i.e., the study and application of speed profiles that reduce energy consumption. According to Douglas et al. [5], applying eco-driving may achieve a reduction of energy consumption between 15 and 35% with low implementation costs.

Focusing on metropolitan railways (metros and trams), eco-driving has been successfully applied in many networks which are operated automatically with an ATO (Automatic Train Operation) system. For instance, according to Domínguez et al. [6], an energy reduction of 20% was achieved on Metro Madrid (Spain) after applying eco-driving techniques. Similarly, Brenna et al. [7] achieved an energy reduction of 33% on the metro of Milan (Italy) through an analogous approach. These results point out the effectivity of eco-driving on urban railways.

Obtaining efficient speed profiles for metro trains is a multi-objective optimisation problem because there are, at least, two conflicting variables to consider: energy consumption and travel time. Therefore, there is not a single solution to the problem, but a (theoretically infinite) set of optimal solutions, called Pareto Solutions or non-dominated solutions (or, alternatively, the Pareto Front). Each solution is defined by the ATO commands set by the system for each interstation run (e.g., braking rate, holding speed, etc.) and yields a different travel time and energy consumption.

Conventional ATO systems had bandwidth limitations that only allow a few different speed profiles to be pre-programmed for each interstation run. Therefore, it is feasible to calculate and analyse all of them to identify the Pareto solutions [6, 8]. In this case, there is no need for applying an optimisation algorithm, as a systematic calculation of the whole solution space is entirely possible. However, the development of more advanced CBTC (Communication Based Train Control) allows ATO systems to handle many values (theoretically infinite, although this is of course limited by the accuracy of the system to measure and set magnitudes such as train speed or acceleration) for each ATO command when departing from each station. This means many more potential speed profiles to be considered, thus expanding dramatically the solution space and making a direct calculation of the Pareto solutions unfeasible [9]. Even more so if that calculation is repeated for each interstation section covering the whole network, and if other factors are considered (e.g., variations in train load, on-board energy storage or interactions between trains).

The most common way of addressing this problem [10] is using a meta-heuristic algorithms that allows searching the solution space and finding non-dominated solutions in a timely manner. These algorithms have been extensively used in fields such as unmanned vessels [11], fault diagnosis [12] or supply chains [13] and have also been widely applied to obtain efficient train profiles and to optimise train schedules [14–17]. There are many different meta-heuristic algorithms, but in this context the preferred choice by far are Genetic Algorithms (GA), and particularly the NSGA-II (Non-dominated Sorted Genetic Algorithm) variant [10]. GA have been extensively used to identify efficient speed profiles in railways [9, 15, 17–21], to the point that the NSGA-II variant has become a benchmark to assess the effectivity of any other optimisation algorithm. The reason is that NSGA-II yields good results with relatively low computing time, and was one of the first truly effective meta-heuristics applied on railways-related optimisation problems [18].

Other, more recent algorithms have been compared to GA. For instance, Lu et al. [22] found that Ant Colony Optimisation (ACO) converges faster than GA in a single-objective problem. On the other hand, Zhao et al. [23] did not find clear differences between the performance of ACO and GA. Focusing on multi-objective optimisation, Domínguez et al. [9] compared the performance of NSGA-II and MOPSO (Multi-Objective Particle Swarm Optimisation) and found the latter to converge faster and yield a more diverse set of solutions. Conversely, according to He and Xiong [21], NSGA-II offers best results in less time than SPEA-II (Strength Pareto Evolutionary Algorithm).

However, despite these examples, relatively few studies compare the performance of different meta-heuristic algorithms in identifying efficient metro speed profiles, and there are many still unexplored options. NSGA-II remains the preferred choice in this field of research [10], ignoring the development of other algorithms in other fields that may offer certain advantages to cope with multi-objective optimisation.

In this context, this paper aims to assess the performance of a relatively new variant of the well-known ACO algorithm, modified to address multi-objective problems with continuous variables. This variant, called MOACOr (Multi-Objective Ant Colony Optimisation for continuous domains), was first introduced by Garcia-Najera and Bullinaria [24] and has been already tested in other fields [13, 25], but has not yet been used to address railways eco-driving. By testing MOACOr, we aim to expand the range of tools available for railways researchers in a field that it is still somewhat limited to the use of NSGA-II and other GA.

The reason for choosing MOACOr over other, more advanced or complex algorithms is that ACO algorithms are one of the most well-known and tested swarm intelligence algorithms, and they have been scarcely used to identify efficient speed profiles.

There are also other ACO variants for multi-objective problems, such as iMOACOr for problems with four or more objectives [26], or EMOACO with elitist mechanisms to improve solution diversity [27], but they only offer slight improvements over MOACOr and are devised to cope with certain aspects (such as many optimisation objectives) that are not applicable to this study.

To evaluate the performance of MOACOr, we applied it to calculate efficient speed profiles in the metro network of Valencia (Spain), operated by *Ferrocarrils de la Generalitat Valenciana* (FGV). The results have been assessed using certain metrics to measure the convergence and diversity of the solutions provided by the algorithm. A conventional NSGA-II has been also used for comparison purposes owing to its widespread use in the field as explained above.

The paper is divided as follows: First, a brief description of the Valencia metro network and the data used for this study is given. Then, the driving simulator used as a tool for modelling speed profiles, based on previous published works, is described. Afterwards, the main characteristics of both algorithms (NSGA-II and MOACOr), as well as the metrics used for their evaluation, are explained. Then the process of obtaining efficient speed profiles for up to 32 interstation sections is described, and the results obtained are thoroughly analysed and discussed. Finally, the main conclusions regarding the performance of MOACOr compared to NSGA-II are presented.

2. Materials and methods

2.1. Description of the Valencia metro network

The metro network of Valencia (Spain), operated by FGV, comprises 9 metro and tram lines with a total length of 156 km. 64 million passengers use the network annually, and the total energy consumption is about 78 GWh/year [28]. Trains are powered by a 1500 V DC system (750 V for trams).

Metro lines (1, 2, 3, 5, 7 and 9) are partially operated with an ATO system. The rolling stock used is the metro series 4300 (Vossloh), with a maximum speed of 80 km/h and 1480 kW of power. There are two different train configurations: one with four carriages (588 passengers) and one with five (750 passengers). Tram lines (4, 6 and 8) are omitted from this study due to their different characteristics.

Our analysis focuses on the underground parts of lines 1, 2, 3 and 5, where the ATO controls the trains. As shown in Figure 1, lines 1 and 2 share the same route between 'Empalme' and 'València Sud', as do lines 3 and 5 between 'Aeroport' and 'Alameda'. Only line 5 is completely underground; the rest have parts on the surface where trains are manually driven. Omitted from Figure 1 are the tram lines (4, 6 and 8) as well as metro lines 7 and 9. The latter share most of their layout with the other metro lines except from two short track stretches from which there was no data available.

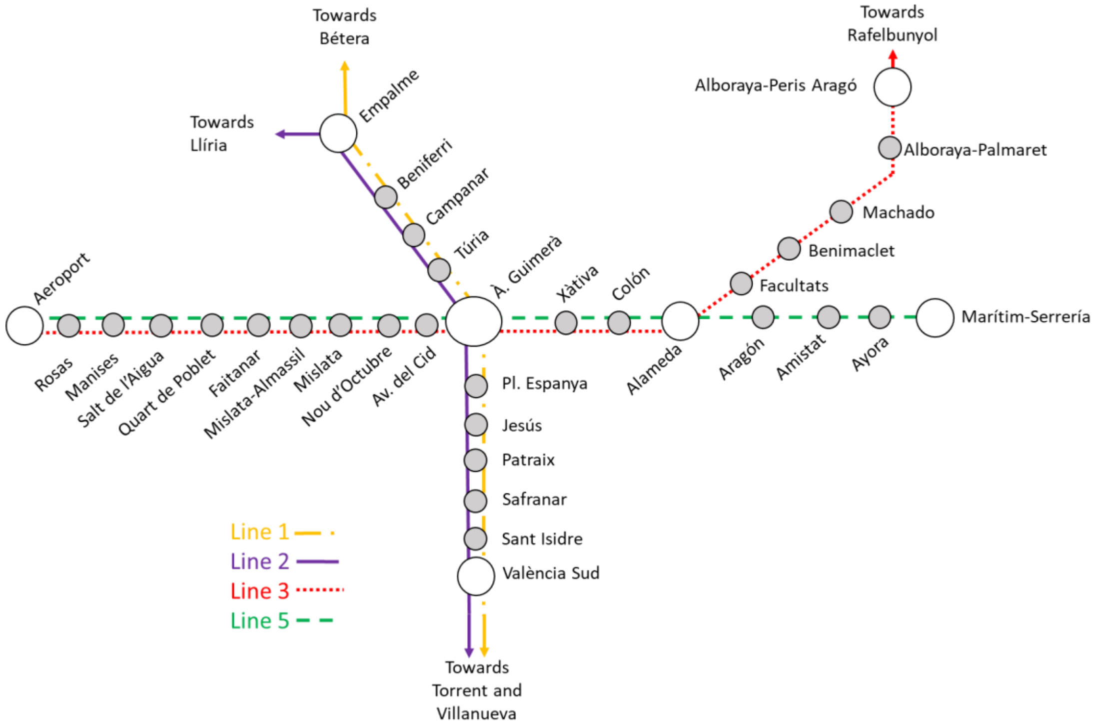

FIGURE 1: Valencia Metro network (only routes operated with ATO).

Real energy consumption, travel time and speed profile data were measured along these lines to calibrate and validate the simulation model as well as to define the optimisation problem that constitutes the framework to compare MOACOr and NSGA-II. More details regarding data gathering and processing can be found in Martínez Fernández et al. [29].

2.2. *Problem description*

The optimisation problem defined to compare MOACOr and NSGA-II follows the same model already used by other authors [9, 14]. It aims at obtaining efficient speed profiles for automatically operated metro trains, balancing two opposing objectives: travel time and energy consumption.

Travel time depends directly on the train speed profile and has a lower limit (defined by the fastest profile a train may apply along a given route) and usually an upper one as well (defined by the operator based on what amount of delay is acceptable). Energy consumption, on the other hand, depends on several factors such as train characteristics (e.g., engine configuration and efficiency, train load), track layout and condition, the presence of regenerative braking, etc. It has also at least an upper limit, that of the fastest profile at a given route.

As we are considering a metro line operated automatically, each solution is defined by the values of the ATO commands (see section 2.3), which in turn define the speed profile that the train will follow from one station to the next. Each profile yields a value of travel time and energy consumption.

There are certain constrains that affect the speed profile and thus both optimisation objectives. The ones considered in this research are:

- Speed limits.
- Limited braking force (comfort requirement).
- Speed differences of at least 20 km/h between coasting cycles (comfort).
- Jerk limited to 0.5 $m/s^2/s$ (comfort requirement).

Variations of train load, energy storage and interactions with other trains have not been included.

2.3. Driving simulator

To carry out an optimisation scheme and find out Pareto speed profiles, a simulation tool is usually required. The optimisation algorithms search across the solution space, composed of every feasible set of ATO commands, and use such tool to calculate the actual travel time and energy consumption of any potential solution [6, 9, 10].

Moreover, as the purpose of this study is not to carry out an optimisation study but to compare between different meta-heuristic algorithms, the simulator is also needed to obtain the real Pareto front (that is, the subset of solutions that are non-dominated) for each interstation stretch through systematic simulation. In this way, the real Pareto front may be used to assess the performance of each optimisation algorithm (in fact, as explained in section 2.5, certain performance metrics require the real Pareto front to be known beforehand).

There are different ways of modelling train dynamics, from mathematical formulations based on the Pontryagin Principle [30–32] to stochastic models [33] or fuzzy logic [15, 34]. However, the most common approach [10] is to use a time-step simulator based on simple motion equations, which yields accurate results with reasonable computing effort [35, 36].

This is the approach chosen for this study, which is based on the time-step simulator presented by Domínguez et al. [35] but also incorporates a neural network to obtain the energy consumption [29].

The combined simulator thus developed has been calibrated and validated using real data measured in the metro of Valencia [29], and yields an average error of 2.9% for travel time and 3.6% for energy consumption. This degree of accuracy is in line with the results obtained by other authors with similar time-step simulators [2, 35, 37, 38]. The simulator was implemented in MATLAB R2018a (The Mathworks, Inc.).

Only two types of speed profiles were considered: holding profiles (i.e., the train accelerates to a pre-set speed, holds that speed for as long as possible and then brakes to stop at the next station) and coasting profiles (i.e., the train coasts and accelerates between two pre-set speed limits). This is because these two types are the most energy efficient [32, 39] to the point that certain ATO systems are designed based exclusively on holding and coasting [8]. This choice determines the simulator input, which consists of four ATO commands: braking ratio (b), holding speed (v_h), coasting speed (v_c) and recovery speed (v_r). As the two types of speed profiles are mutually exclusive (i.e. a train will either hold or coast between two consecutive stations), the ATO system never sets values for the four commands at once. For holding profiles, positive values for braking ratio and holding speed are set (with $v_c = v_r = 0$). For coasting, positive values for braking ratio, coasting and recovery speeds are set (with $v_h = 0$ and $v_c > v_r$).

2.4. Optimisation algorithms

As explained before, the main objective of this study is to assess the performance of a variant of the ACO algorithm called MOACOr (Multi-Objective Ant Colony Optimisation for continuous domains). Socha and Dorigo [40] proposed a variant of the traditional ACO called ACO-R, or Ant Colony Optimisation for Continuous Domains. This variant modifies the basic mechanism of ACO to search through continuous variables by means of a probability density function. Conversely, Garcia-Najera and Bullinaria [24] proposed adding an archive to store the best, non-dominated solutions after each iteration in order to cope with multi-objective problems. These two developments are the source of the MOACOr algorithm used in this study. Although simpler versions of ACO have been used for railway optimisation problems with a single objective [22], or for multi-objective problems with discrete variables [41], the MOACOr version has not been yet tested to obtain energy efficient speed profiles in metro networks.

The ACOr as defined by Socha and Dorigo [40] uses an archive to store the best solutions after each iteration. Each stored solution (s_l) has a weight (w_l) defined as:

$$w_l = \frac{1}{qk\sqrt{2\pi}} e^{\frac{(l-1)^2}{2q^2k^2}} \tag{1}$$

Where k is the number of ants, subindex l denotes belonging to the l-th solution (s_l) and q is a parameter that controls the selection from the archive (the higher the value of q, the higher the probability of choosing the best solutions already in the archive). On the other hand, as the algorithm deals with continuous variables, for each decision variable (i.e., the four ATO commands) a Gaussian kernel is defined, consisting of the weighted sum of several one-dimensional Gaussian density functions (as many as the size of the archive). During each iteration, each ant chooses and samples one of these distributions, and this choice is ruled by a probability p_l determined by the weights of the solutions stored in the archive:

$$p_l = \frac{w_l}{\sum_{r=1}^{k} w_r} \tag{2}$$

Where l denotes once again belonging to the l-th solution and w_l/w_r are weights as defined in equation (1). Please note that weights have different subindexes to denote different numerations (where subindex r is tied to the summation in the denominator of equation (2)). Each Gaussian distribution is defined by its mean and standard deviation, formulated as:

$$\mu^i = s_l^i \tag{3}$$

$$\sigma^i = \xi \sum_{e=1}^{k} \frac{\left|s_e^i - s_l^i\right|}{k-1} \tag{4}$$

Where index i denotes the i-th decision variable (in our case, each of the four ATO commands), index l indicates the l-th solution and s_e/s_l are the mean values as defined by equation (3). Additionally, ξ is the pheromone parameter, which regulates the convergence of the algorithm.

This ACOr structure is modified to cope with two objectives by applying a non-dominance criterion to choose the solutions that will be stored in the archive. A Crowding Distance criterion has also been applied to trim the archive when there is an excess of solutions, saving the ones that offer more diversity.

The Genetic Algorithm used as a benchmark to assess the performance of the MOACOr is the conventional Non-dominated Sorted Genetic Algorithm (NSGA-II); a widespread algorithm extensively used in railways optimisation problems, as explained in section 1. This algorithm includes crossover and mutation mechanisms to generate new solutions during each iteration, uses non-domination criteria to choose the best ones and incorporates a Crowding Distance mechanism to improve diversity [9]. As the NSGA-II algorithm is a well-known and widely used method, a more detailed description will be omitted.

Both algorithms were implemented in MATLAB R2018a (The Mathworks, Inc.) based on open-source versions available in www.yarpiz.com, with only minor modifications to adapt them to this context.

Table 1 shows the main parameters of both MOACOr and NSGA-II used for this study, calibrated through a sensitivity analysis.

Table 1 Main parameters for both algorithms, range and chosen value after sensitivity analysis

NSGA-II		
Variable	*Range*	*Value*
Population size	10-60	30
Number of iterations	10-60	25

Crossover percentage	0.20-0.60	0.50
Mutation percentage	0.15-0.50	0.15
Mutation rate	Fixed	0.02
MOACOr		
Variable	*Range*	*Value*
Population size	20-50	30
Number of iterations	20-40	25
Archive size	Fixed	30
Weight parameter (q)	0.05-0.80	0.80
Pheromone parameter (ξ)	0.50-1.00	1

For the sensitivity analysis, a single, representative interstation section was chosen: Aragon-Amistat, in line 5 (see Figure 1). This is because there are not speed limits or other factors in this section that may restrict some of the potential speed profiles, and it is a completely straight and mostly flat track stretch. Additionally, the obtained Pareto front convers the full range of both optimisation variables (i.e., time and energy) and includes both types of speed profiles (i.e., holding and coasting).

In this track section, each algorithm was tested several times to obtain sets of optimal speed profiles, varying its parameters within the ranges shown in Table 1. A systematic analysis was carried out, where each parameter was modified along its range while keeping the rest fixed. In order to account for interactions between parameters, this process was repeated several times modifying the values of the fixed parameters before each simulation, hence yielding several combinations of values across all ranges. Moreover, for each combination of parameters, the algorithm was run six times and the average results for each performance metric were calculated. The combination that better minimised these metrics was then chosen. Ranges for each parameter (and the decision of fixing some of them beforehand, such as mutation rate in NSGA-II and archive size in MOACOr) were defined based on previous research [23, 42, 43].

2.5. Performance metrics

To compare the performance of both meta-heuristic optimisation algorithms, two metrics were used. The objective of these metrics is to carry out a quantitative evaluation of three aspects of the solution front provided by each algorithm: convergence (i.e., proximity to the real Pareto front), diversity (i.e., how spread across the solution space is the calculated front) and regularity (i.e., how evenly distributed is the calculated front). These three aspects are essential for any set of solutions provided by an optimisation algorithm, as the ATO system may thus choose among speed profiles that are optimum (as they are part of the Pareto front) and may cover a wider range of possibilities, from faster profiles to more conservative ones depending on traffic needs.

These metrics require the real Pareto front to be known beforehand, hence the need for a systematic simulation of speed profiles using the driving simulator. This simulation consisted, for each interstation stretch, of up to 5832 calculations of the speed profile (666 corresponding to holding profiles and 5166 to coasting profiles). This was made by sampling the full range of each ATO command in equal increments. Evidently, this is only a fraction of the total number of potential solutions (which is virtually infinite), and thus the Pareto front obtained in each track section is but an approximation to the real one. However, as the purpose of this paper is to carry out a comparison between both algorithms in a realistic setting, and not obtaining optimal profiles, this deviation was deemed acceptable as it provided the required front to obtain the performance metrics with a reasonable calculation time.

The metrics are:

Inverse Generational Distance (IGD)

The IGD is an improvement over the more traditional Generational Distance (GD) metric. GD measures the proximity of the solution provided by the algorithm to the real Pareto front by means of calculating an average distance. It is thus an evaluation of convergence. However, GD is not Pareto-compliant and is sensitive to the size of the solution set provided by the algorithm. To address this, IGD inverts the terms of GD so that it calculates the proximity of the real Pareto front to the solution set provided by the algorithm. IGD is calculated as follows [44]:

$$IGD = \sqrt{\frac{\sum_{j=1}^{NP} d_j^2}{NP}} \tag{5}$$

Where d_j is the Euclidean distance between the j-th element of the real Pareto front and the nearest point belonging to the solution set obtained by the algorithm, and NP is the number of elements in the real Pareto front. The lower the value of IGD, the closer the solution is to the real Pareto front.

Normalised Hypervolume (NH)

The Hypervolume, which was first proposed by Thiele and Zitzler [45], is a Pareto-compliant metric that measures the portion of the objective space weakly dominated by the set of solutions given by the algorithm. Calculating this metric is a rather complex process, particularly when the number of optimisation objectives increases [46]. In this case, we have used an open source algorithm programmed in Python [47], based on the algorithm proposed by Fonseca et al. [48]. Moreover, the metric given by each algorithm is normalised using the hypervolume calculated for the approximated Pareto front. As it is expressed as a fraction of the Pareto front hypervolume, the closer to 1, the better.

3. Results and discussion

3.1. Pareto fronts

The first step to compare both algorithms is to obtain the real Pareto front for each of the 32 interstation stretches analysed (which, considering both directions of circulation, yields 64 cases). This systematic simulation would not be necessary in a real optimisation problem, as the main reason of using meta-heuristic algorithms is precisely to avoid such a time-consuming task. However, as explained before, it is necessary in this case as the evaluation metrics are otherwise impossible to calculate. Therefore, a systematic simulation of speed profiles was carried out in each interstation stretch, using up to 5832 combinations of ATO commands (666 holding profiles and 5166 coasting profiles). This set of combinations is based on the values shown in Table 2.

Table 2 Maximum, minimum and step values for ATO commands used for systematic simulation

	Braking rate, b (m/s^2)	Holding speed, v_h (km/h)	Coasting speed, v_c (km/h)	Recovery speed, v_r (km/h)
Maximum value	0.80	80	80	60
Minimum value	0.55	25	40	20
Step	0.05	0.5	1	1

It is evident that a more thorough simulation could be done, as there are theoretically an infinite number of possible combinations of ATO commands within the limits in Table 2. However, those 5832 combinations where enough to identify a good approximation to the real Pareto front with a feasible computing time: 100 minutes on average for each interstation stretch (while both optimisation algorithms only require a few minutes at worst). Figure 2 shows an example of this simulation, namely the results obtained between 'Rosas' and 'Manises' stations (Lines 3 and 5), with the Pareto front fully identified. The figure shows the 5832 calculated speed profiles, divided between holding and coasting profiles, and marks those that form the approximated Pareto front considering a non-dominance criterion.

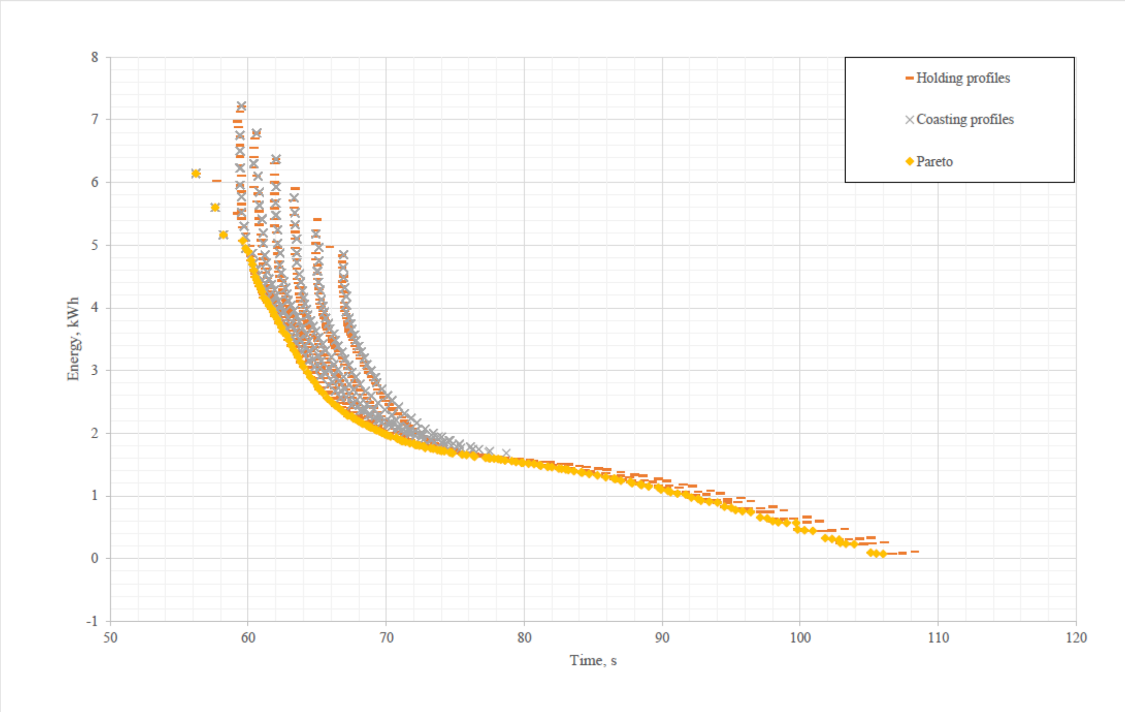

FIGURE 2: Example of extensive simulation to obtain the real Pareto front. Rosas-Manises (Lines 3-5).

3.2. Metric results

Once all Pareto fronts were calculated, both algorithms were used to obtain a set of optimised speed profile in each of the 64 cases, and the corresponding metrics defined in section 2.5 were obtained. Figure 3 shows a few examples of the results provided by each algorithm, compared with the real Pareto front. Overall, both algorithms yield a good approximation to the Pareto front, particularly in terms of convergence, although MOACOr seems to cover better the full extent of the real Pareto front and hence offers a more diverse solution.

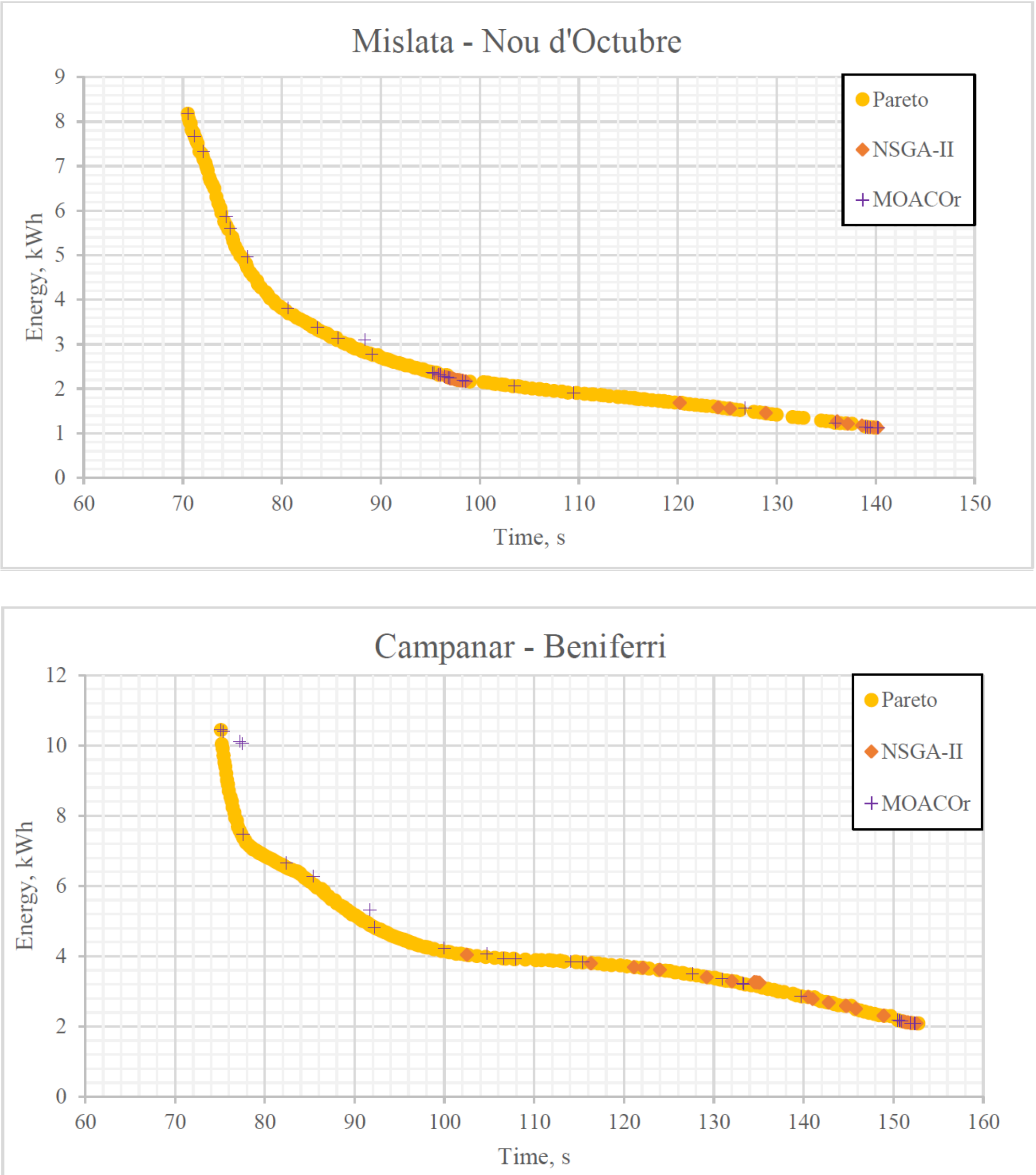

FIGURE 3: Examples of algorithms' results vs Pareto front.

Table 3 shows the average results for each metric, considering all 64 cases. It seems that MOACOr performs better than NSGA-II in both metrics, as it yields an average smaller IGD and a NH closer to that of the Pareto front. Therefore, MOACOr may perform better in terms of convergence, diversity and regularity. Moreover, MOACOr outperforms NSGA-II in 76% of the cases with respect to IGD, and in 87% of the cases with respect to NH.

Table 3 Average results for each metric and algorithm

		Metric	
		IGD	NH
NSGA-II	Mean	3.417	0.881
	Median	2.338	0.896
	Std. dev.	2.991	0.070
	Asymmetry	6.428	-5.324
	Kurtosis	6.623	6.744
MOACOr	Mean	1.910	0.937
	Median	1.599	0.949
	Std. dev.	1.427	0.056
	Asymmetry	8.217	-12.625
	Kurtosis	13.935	36.805

Figure 4 shows the box and whiskers plot for each metric and algorithm, considering all 64 cases. MOACOr yields a smaller value of IGD on average and shows much lower dispersion than NSGA-II. Additionally, MOACOr offers a value of NH closer to one on average, and again with less dispersion than NSGA-II. Therefore, there is a trend for an overall better performance of MOACOr regarding both diversity and convergence.

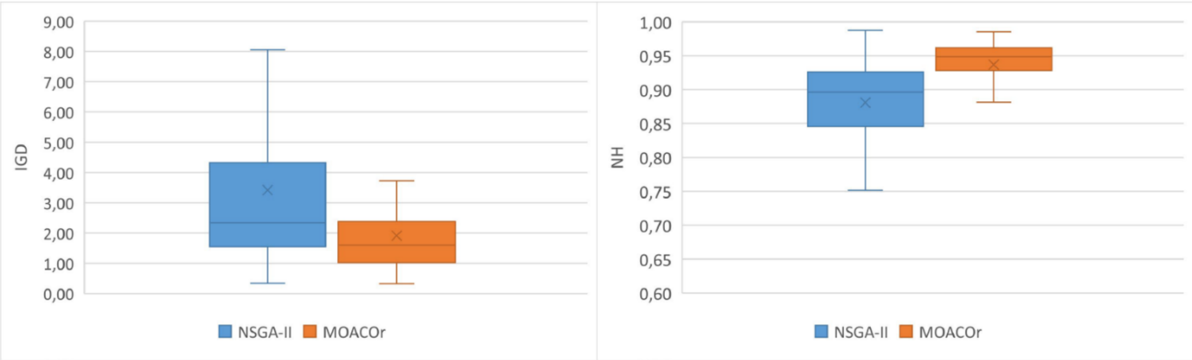

FIGURE 4: Box and whiskers plots for each metric and algorithm.

Note that, apart from convergence and diversity metrics, the algorithms might be compared in terms of running time. However, in this case, the overall running time of each algorithm is greatly affected by the running time of the train simulator, which is accurate and functional [29] but not particularly efficient. As each algorithm runs several iterations, and on each iteration several speed profiles are calculated by calling on the simulator, the total running time may reach several seconds (even up to a few minutes). However, these figures do not represent the actual performance of the algorithms and thus comparing running times could be misleading. A more accurate assessment of running times (clearly separating the time required by each instance of the train simulator from the actual running time of each algorithm) could be carried out in future research steps.

3.3. Statistical Analysis

In order to go further in the comparison between both algorithms, a statistical analysis has been carried out to find whether the observed differences are statistically significant.

Figure 5 shows a flow chart detailing the steps of the analysis, carried out using STATGRAPHICS CENTURION XVII 17.2.04 (Statpoint Technologies, Inc.).

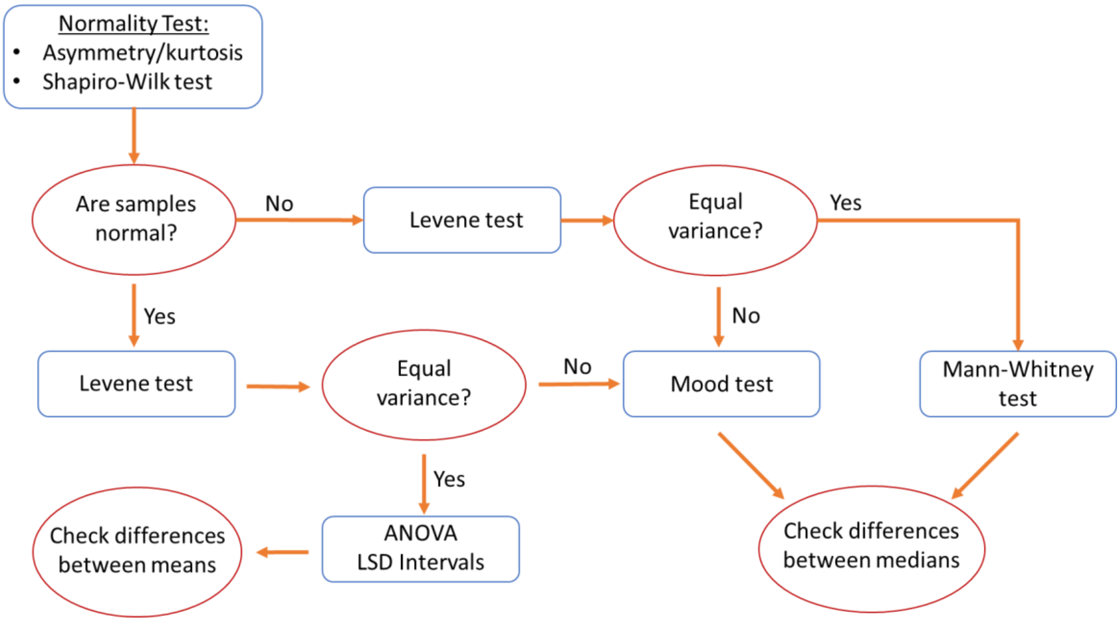

FIGURE 5: Statistical Analysis flow chart

First of all, considering the values of asymmetry and kurtosis shown in Table 3, it is clear that the distribution of IGD and NH values do not follow a Normal distribution as they are clearly outside the [-2, 2] range. A Shapiro-Wilk test confirmed this assessment.

Moreover, a Levene test was applied to test whether each pair of distributions (i.e., IGD and NH) have equal variance. According to the results shown in Table 4, in both cases the hypothesis of equal variance is rejected with a 95% statistical confidence level.

Therefore, to detect potential significant differences between the values of each metric for both algorithms, a non-parametric test was used. Mann-Whitney was discarded as it relies on the assumption of approximately equal variance (which is not acceptable in this case) and thus the Mood Median test was used instead, once again assuming a 95% statistical confidence level, to detect differences in the median values. The results are also shown in Table 4.

Table 4 Levene and Mood Tests results

VARIANCE (Levene)		
Metric	**P-Value**	**Equal variance?**
IGD	0.0001 < 0.05	NO
NH	0.0145 < 0.05	NO
MEDIAN (Mood)		
Metric	**P-Value**	**Equal median?**
IGD	0.0047 < 0.05	NO
NH	1.85E-9 < 0.05	NO

As these results show, there is a statistically significant difference between the median values of IGD and NH obtained by each algorithm. Therefore, in terms of convergence, regularity and diversity, MOACOr outperforms NSGA-II. This further proves that MOACOr offers a small yet significant advantage compared to NSGA-II, as it will offer solutions better spread along the Pareto front as well as closer to it. This allows choosing ATO profiles among a wider range of options, from a faster profile to a slower and less energy-consuming one depending on service needs [9].

As stated in the introduction, MOACOR has never been used before as an optimisation tool for metro speed profiles, and thus it is not possible to strictly compare the results of this paper with other studies. However, other heuristic algorithms based on swarm intelligence (and specifically MOPSO) have already proved to offer better results [9] than the widespread NSGA-II, thus supporting the conclusion that swarm intelligence algorithms may be better suited for optimising speed profiles than genetic ones. This may be due to these algorithms being less affected by dynamic or unstable optimisation problems [49] and being better suited for problems where the objective functions have high conditioning values [50], both circumstances that apply to speed profile optimisation.

3.4. Limitations and future research

The comparison between NSGA-II and MOACOr, albeit based on current trends in the field of railways driving optimisation, could be expanded to other meta-heuristic algorithms. Several algorithms (as well as variants and improvements of existing ones) have been developed and tested in other areas of research over the last years, but only a few have been applied to obtain efficient speed profiles (of which MOPSO is the most noteworthy [9]). Moreover, most of the reviewed papers that compare algorithms in this particular area of research only analyse the performance of one algorithm against the widespread NSGA-II. Therefore, a multiple comparison with many algorithms is the logical next step. Additionally, although metro networks, due to their growing automation, are the most adequate for large-scale optimisation, this study could be expanded to interurban rail lines, both conventional and high-speed ones.

4. Conclusions

The main aim of this study is to assess the performance of a modified version of the ACO algorithm, devised to cope with multi-objective optimisation problems with continuous variables (MOACOr), in obtaining efficient speed profiles for metro lines operated with ATO systems. To do so, the algorithm has been used to obtain a set of efficient solution in terms of energy consumption and travel time in 32 interstation stretches (i.e., 64 cases, considering both directions) within the metro network of Valencia (Spain). The results have been evaluated using two metrics to assess their degree of convergence, diversity and regularity: Inverse Generational Distance (IGD) and Normalised Hypervolume (NH). A genetic algorithm (NGSA-II), which is the algorithm most used to optimise metro speed profiles, has been also used as a reference to assess the performance of MOACOr.

The results show that both algorithms offer, on average, rather good approximations to the real Pareto front (which was calculated through systematic simulation of the ATO commands). However, both in terms of convergence (measured through IGD) and diversity (measured through NH) MOACOr tends to perform better, scoring lower values of IGD in 76% of the cases and lower values of NH in 87% of the cases. Furthermore, a statistical analysis was carried out, using a non-parametric test (Mood Median test) to check whether any difference in the median values of the metrics obtained by each algorithm is statistically significant. The results prove that there is, in fact, a significant difference in the median values of IGD and NH (in both cases with a statistical confidence level of 95%).

Therefore, the MOACOr variant does offer an alternative to the conventional NSGA-II, which has been extensively used in the past years to obtain efficient metro speed profiles in several real-life applications. MOACOr yields solutions with improved degree of convergence, regularity and diversity than those provided by NSGA-II, and thus outperforms it with regard to every criterion considered in the analysis. For a more practical point of view, more diverse solution sets offer the metro manager a wider range of optimum speed profiles to choose when programming the ATO system, from faster (and more energy-consuming) profiles to slower (and more energy-saving) ones, depending on service needs.

Acknowledgements

The authors wish to thank *Ferrocarrils de la Generalitat Valenciana* (FGV) for their kind support during the development of this study, including their assistance during data gathering in their metro network.

Funding statement

This work was supported by the Spanish Ministry of Economy and Competitiveness [grant number TRA2011-26602].

Declaration of conflicting interest

The Authors declare that there is no conflict of interest.

References

[1] García Álvarez A. Consumo de energía y emisiones del tren de alta velocidad en comparación con otros medios de transporte: la extensión de la red AVE mejora la sostenibilidad del transporte. *An mecánica y Electr* 2007; 84: 26–34.

[2] Huang SR, Sung HK, Ma CH. Optimize Energy of Train Simulation with Track Slope Data. *IEEE Conf Intell Transp Syst* 2015; 2015-Octob: 590–597.

[3] Wang DP, He ZK, Li MG, et al. Numerical research on aerodynamic drag reduction optimization of EMU. *J China Railw Soc* 2011; 33: 15–18.

[4] Grasso M, Gallone M, Genovese A, et al. Composite material design for rail vehicle innovative lightweight components. *Lect Notes Eng Comput Sci* 2015; 2218: 731–736.

[5] Douglas H, Roberts C, Hillmansen S, et al. An assessment of available measures to reduce traction energy use in railway networks. *Energy Convers Manag* 2015; 106: 1149–1165.

[6] Domínguez M, Fernández A, Cucala P, et al. Computer-aided design of ATO speed commands according to energy consumption criteria. *WIT Trans Built Environ* 2008; 103: 183–192.

[7] Brenna M, Foiadelli F, Longo M. Application of genetic algorithms for driverless subway train energy optimization. *Int J Veh Technol* 2016; 2016: 1–14.

[8] Domínguez M, Fernández A, Cucala P, et al. Efficient design of Automatic Train

Operation speed profiles with on board energy storage devices. *WIT Trans Built Environ* 2010; 114: 509–520.

[9] Domínguez M, Fernández-Cardador A, Cucala P, et al. Multi objective particle swarm optimization algorithm for the design of efficient ATO speed profiles in metro lines. *Eng Appl Artif Intell* 2014; 29: 43–53.

[10] Martínez Fernández P, Villalba Sanchís I, Yepes V, et al. A review of modelling and optimisation methods applied to railways energy consumption. *J Clean Prod* 2019; 222: 153–162.

[11] Li G, Li Y, Chen H, et al. Fractional-Order Controller for Course-Keeping of Underactuated Surface Vessels Based on Frequency Domain Specification and Improved Particle Swarm Optimization Algorithm. *Appl Sci* 2022; 12: 3139.

[12] Cui H, Guan Y, Chen H. Rolling Element Fault Diagnosis Based on VMD and Sensitivity MCKD. *IEEE Access* 2021; 9: 120297–120308.

[13] Moncayo-Martínez LA, Zhang DZ. Multi-objective ant colony optimisation: A meta-heuristic approach to supply chain design. *Int J Prod Econ* 2011; 131: 407–420.

[14] Fernández A, Fernández-Cardador A, Cucala P, et al. Design of Robust and Energy-Efficient ATO Speed Profiles of Metropolitan Lines Considering Train Load Variations and Delays. *IEEE Trans Intell Transp Syst* 2015; 16: 2061–2071.

[15] Cucala P, Fernández-Cardador A, Sicre C, et al. Fuzzy optimal schedule of high speed train operation to minimize energy consumption with uncertain delays and driver's behavioral response. *Eng Appl Artif Intell* 2012; 25: 1548–1557.

[16] Sicre C, Cucala P, Fernández-Cardador A, et al. Modeling and optimizing energy-efficient manual driving on high-speed lines. *IEEJ Trans Electr Electron Eng* 2012; 7: 633–640.

[17] Wong KK, Ho TK. Dynamic coast control of train movement with genetic algorithm. *Int J Syst Sci* 2004; 35: 835–846.

[18] Salim V, Cai X. Scheduling cargo trains using genetic algorithms. In: *Proceedings of 1995 IEEE International Conference on Evolutionary Computation*. Perth, 1995, pp. 224–227.

[19] Kang M-H. GA-based Algorithm for Creating an Energy-Optimum Train Speed Trajectory. *J Int Counc Electr Eng* 2011; 1: 123–128.

[20] Watanabe S, Koseki T. Energy-saving train scheduling diagram for automatically operated electric railway. *J Rail Transp Plan Manag* 2015; 5: 183–193.

[21] He T, Xiong R. Research on Multi-Objective Real-Time Optimization of Automatic Train Operation (ATO) in Urban Rail Transit. *J Shanghai Jiaotong Univ* 2018; 23: 327–335.

[22] Lu S, Hillmansen S, Tin Kin H, et al. Single-Train Trajectory Optimization. *IEEE Trans Intell Transp Syst* 2013; 14: 743–750.

[23] Zhao N, Roberts C, Hillmansen S, et al. A Multiple Train Trajectory Optimization to Minimize Energy Consumption and Delay. *IEEE Trans Intell Transp Syst* 2015; 16: 2363–2372.

[24] Garcia-Najera A, Bullinaria JA. *Extending ACO R to Solve Multi-Objective Problems*. Aberdeen, https://pdfs.semanticscholar.org/d185/8a6f7f73ef5b9b502a741b78bda83c1b6dd3.pdf (2007, accessed 15 May 2019).

[25] Vimal S, Khari M, Crespo RG, et al. Energy enhancement using Multiobjective Ant colony optimization with Double Q learning algorithm for IoT based cognitive radio networks. *Comput Commun* 2020; 154: 481–490.

[26] Falcón Cardona JG, Coello Coello CA. iMOACO R : A New Indicator-Based Multi-Objective Ant Colony Optimization Algorithm for Continuous Search Spaces. In: *Parallel Problem Solving from Nature – PPSN XIV: 14th International Conference*. Edinburgh, pp. 389–398.

[27] Xiangui S, Kong D. A multi-objective ant colony optimization algorithm based on elitist selection strategy. *Metall Min Ind* 2016; 7: 333–338.

[28] FGV. *Informe de gestión, cuentas anuales y memoria de sostenibilidad 2017*. 2017.

[29] Martínez Fernández P, Salvador Zuriaga P, Villalba Sanchís I, et al. Neural networks for modelling the energy consumption of metro trains. *Proc Inst Mech*

Eng Part F J Rail Rapid Transit 2019; 0: 1–12.

[30] Khmelnitsky E. On an optimal control problem of Train Operation. *IEEE Trans Automat Contr* 2000; 45: 1257–1266.

[31] Howlett P, Pudney P, Vu X. Local energy minimization in optimal train control. *Automatica* 2009; 45: 2692–2698.

[32] Albrecht A, Howlett P, Pudney P, et al. The key principles of optimal train control—Part 1: Formulation of the model, strategies of optimal type, evolutionary lines, location of optimal switching points. *Transp Res Part B* 2016; 94: 482–508.

[33] Davydov B, Chebotarev V, Kablukova K. Online Train Traffic Adjustments: Probabilistic Modeling and Estimating. In: Sierpiński G (ed) *Advanced Solutions of Transport Systems for Growing Mobility*. Cham: Springer International Publishing, 2018, pp. 50–60.

[34] Sicre C, Cucala P, Fernández-Cardador A. Real time regulation of efficient driving of high speed trains based on a genetic algorithm and a fuzzy model of manual driving. *Eng Appl Artif Intell* 2014; 29: 79–92.

[35] Domínguez M, Fernández A, Cucala P, et al. Optimal design of metro automatic train operation speed profiles for reducing energy consumption. *Proc Inst Mech Eng Part F J Rail Rapid Transit* 2011; 225: 463–473.

[36] Tian Z, Hillmansen S, Roberts C, et al. Energy evaluation of the power network

of a DC railway system with regenerating trains. *IET Electr Syst Transp* 2015; 6: 1–9.

[37] Jong J-C, Chang S. ALGORITHMS FOR GENERATING TRAIN SPEED PROFILES. *J East Asia Soc Transp Stud* 2005; 6: 356–371.

[38] Salvador Zuriaga P, Martínez Fernández P, Villalba Sanchís I, et al. Modelling energy consumption in diesel multiple units. *Proc Inst Mech Eng Part F J Rail Rapid Transit* 2017; 232: 1539–1548.

[39] Albrecht A, Howlett P, Pudney P, et al. The key principles of optimal train control—Part 2: Existence of an optimal strategy, the local energy minimization principle, uniqueness, computational techniques. *Transp Res Part B* 2016; 94: 509–538.

[40] Socha K, Dorigo M. Ant colony optimization for continuous domains. *Eur J Oper Res* 2008; 185: 1155–1173.

[41] Eaton J, Yang S, Gongora M. Ant Colony Optimization for Simulated Dynamic Multi-Objective Railway Junction Rescheduling. *IEEE Trans Intell Transp Syst* 2017; 18: 2980–2992.

[42] Açikbas S, Soylemez M. Coasting point optimisation for mass rail transit lines using artificial neural networks and genetic algorithms. *IET Electr Power Appl* 2008; 2: 172–182.

[43] Carvajal-Carreño W, Cucala P, Fernández-Cardador A. Optimal design of

energy-efficient ATO CBTC driving for metro lines based on NSGA-II with fuzzy parameters. *Eng Appl Artif Intell* 2014; 36: 164–177.

[44] Schütze O, Esquivel X, Lara A, et al. Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Trans Evol Comput* 2012; 16: 504–522.

[45] Thiele L, Zitzler E. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Trans Evol Comput* 1999; 3: 257–271.

[46] Beume N, Fonseca CM, López-Ibáñez M, et al. On the complexity of computing the hypervolume indicator. *IEEE Trans Evol Comput* 2009; 13: 1075–1082.

[47] Wessing S. Hypervolume algorithm. *TU Dortmund University*, http://ls11-www.cs.tu-dortmund.de/rudolph/hypervolume/start (2010, accessed 10 August 2020).

[48] Fonseca CM, Paquete L, López-Ibáñez M. An improved dimension-sweep algorithm for the hypervolume indicator. In: *2006 IEEE Congress on Evolutionary Computation*. Vancouver: IEEE, 2006, pp. 1157–1163.

[49] Meetei KT. A Survey: Swarm Intelligence vs. Genetic Algorithm. *Int J Sci Res* 2014; 3: 231–235.

[50] El-Abd M. Performance assessment of foraging algorithms vs. evolutionary algorithms. *Inf Sci (Ny)* 2012; 182: 243–263.