

## Research Article

# Pareto Assumption for Constrained PSO-Based Locomotive Resistance Minimization

Mine Sertsöz  and Mehmet Fidan 

*Eskişehir Technical University, Voc. School of Transportation 26140, Eskişehir, Turkey*

Correspondence should be addressed to Mine Sertsöz; [msertsoz@eskisehir.edu.tr](mailto:msertsoz@eskisehir.edu.tr)

Received 14 April 2020; Revised 11 June 2020; Accepted 13 June 2020; Published 16 July 2020

Academic Editor: Nunzio Salerno

Copyright © 2020 Mine Sertsöz and Mehmet Fidan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The mechanical resistance of a locomotive is crucial for power consumption. It is desirable to maintain this resistance at a minimum value for energy efficiency under optimal operation conditions. The optimal conditions can be found by particle swarm optimization with constraints. The particle swarm optimization method is a highly preferred type of heuristic algorithm because of its advantages, such as fewer parameters, faster speed, and a simpler flow diagram. However, fast convergence can be misleading in finding the optimum solution in some cases. Pareto analysis is used in this proposed study to prevent missing the target. When the literature is searched, it is seen that there are various studies using this method. However, in all of these studies, the results of the particle swarm method have been interpreted as whether or not they complied with Pareto's 80/20 rule. The validity of the Pareto analysis is taken as an assumption, and with the help of this assumption, the coefficients of a locomotive's mathematical equation were changed, and finally the results were found by applying the particle herd optimization method. Finally, a novel hybrid method has been created by including the Pareto optimality condition to particle swarm optimization. The results are compared with this innovative hybrid method of Pareto and particle swarm and the results found using only the particle swarm method.

## 1. Introduction

Optimization is used in many fields of life. Firstly, the mathematical model of the system is introduced, then this model is solved with different optimization methods. Optimization methods are classified as classical methods and heuristic optimization methods. Classical methods depend on the analytical and derivation of the problem. There are two types of classical methods. They are Gradient and nongradient based methods. When the problem size increases, classical methods may be insufficient. A study [1] proves that classical methods are not sufficient in finding the optimum working points of the locomotive.

Heuristic optimization methods use reflecting of nature. There are many algorithms that have been produced that reflect the behavior of living beings in nature. These include the Ant System [2], the Max-Min Ant System [3], Particle Swarm Optimization [4], Artificial Bee Colony [5], the Fruit Fly Optimization Algorithm [6], Cuckoo optimization based

on Levy Flight [7], the Krill Herd Optimization Algorithm [8], Bakeri Foraging Behavior [9], the Bat Algorithm [10], the Firefly Algorithm [11], the Lion Algorithm [12], the Gray Wolf Algorithm [13], the Dolphin Algorithm [14], the Bush Colony Algorithm [15], the Artificial Algae Algorithm [16], the Virus Colony Search Algorithm [17], the Shark Olfaction Optimization Algorithm [18], and the Social Spider Algorithm [19]. Among all these algorithms, particle swarm optimization is the most cited intuitive intelligence algorithm with the cited number being 5721 [20].

Particle Swarm Optimization (PSO) is a heuristic optimization method based on social information sharing, which, to summarize briefly, was developed jointly by a social psychologist and an electrical engineer [4]. This method is an algorithm designed by considering the behaviors that fish and birds exhibit in order to survive as a swarm. The algorithm of the PSO method is based on the communication between individuals in the swarm. The search is performed by the number of generations as in the

genetic algorithm. In fact, individuals are called particles, and the community of these particles is called the swarm.

The aim is that each particle tries to approach the particle with the best position in the swarm, based on the experience of its previous position. This approach speed is random, and the assumption is that the next step is better than the previous one. This approach continues until it reaches the target (best position). Although it is similar to the genetic algorithm (GA) method, it is easier to use and respond better in some studies than GA.

The original PSO algorithm was designed for unconstrained optimization problems. However, real-life problems generally need constraints on the input elements of the problem. According to this need, some constrained PSO approximations were applied in the literature. One of these studies is a dynamic particle swarm optimizer with escaping prey (DPSOEP), which was designed for constrained non-convex and piecewise problems [21]. In another recent research, partially and fully constrained improved particle swarm optimization methods (PCIPSO and FCIPSO) were proposed for planning the optimal usage of a water reservoir [22]. A moth swarm algorithm with arithmetic crossover modification (MSA-AC) was also developed for solving constrained optimization problems and tested on most common benchmark functions [23]. In a different study, the inequalities for the determination of constraints were explained as a fuzzy set, and new constraint definitions for this fuzzy set were improved, which can be interpreted by the constrained PSO [24]. Constrained PSO was also used for optimal trajectory planning of a space robot in a recent research [25]. For hyperspectral images, the vertices of shapes, which are represented by highly mixed pixels, were found by Linear Mixture Model Constrained PSO (LMMC-PSO) in another recent research [26]. In the most recent research for constrained PSO, a novel PSO method called CMPSOWV was developed by extracting velocity vector [27]. As seen in the literature, many optimization problems were handled by constrained PSO approximation. In this study, locomotive resistance is also handled as a constrained optimization problem.

The Pareto analysis was found by the Italian economist Vilfredo Frederico Damaso Pareto (1848–1923); it is a bar diagram used to distinguish the important causes of a problem from the less important causes. This analysis started with Pareto observing that 80% of the land is owned by 20% of the population in Italy. This distribution has been tried in many different areas reaching the same findings. This statistical rate (80/20), which can also be applied to engineering, is called the “Pareto Law” especially in economic fields.

Hwang and Masud established the necessity of decision-making and classified the existing decision-making approaches to optimization into three categories: a priori, a posteriori, and progressive articulation of preferences [28]. A priori articulation of preferences involves the aggregation of objective functions using weights and, thereafter, determines the optimal solution. A posteriori articulation of preferences, in which decision-making is applied after an efficient set of solutions is obtained, is the selection of the best solution among the obtained solutions. A progressive

articulation of preferences, which involves the process of decision-making along with the search process, is a guide for the search effort toward the regions of the Pareto front, where the probability of getting the compromise solution is the highest.

Normalization is performed using different related targets within a single purpose function. Another way is to find the Pareto optimal front that summarizes all Pareto optimal solutions. A Pareto optimal solution, by definition, is the best solution that can be achieved for the one target, while keeping at least one other target without a disadvantage [29]. In fact, normalization gives the same importance to all variables, whereas Pareto gives different levels of importance to all variables. Studies using Swarm and Pareto in the literature generally include a posteriori articulation as in the study of Ulungu and Teghem [29] and the study of Di Barba et al. [30]. Although there are many studies in the literature including both Pareto and Swarm, specific studies can be listed as follows.

In one of the studies, where Swarm and Pareto were used together, the PSO algorithm was applied to multiobjective optimization problems with two and more objectives [31]. The algorithm revealed an excellent performance concerning the number of solutions of the forward problem, completed with a very reliable representation of the Pareto optimal front. In another study, the Pareto dominant was adapted to evaluate particles found throughout the search process in the swarm; with this approach, the nondominant characters are stored in an external replenishment that constantly renews itself [32]. This paper [33] presents an approach in which Pareto dominance is incorporated into particle swarm optimization (PSO) in order to allow this heuristic to handle problems with several objective functions. In another study, a simple evolution scheme for multiobjective optimization problems was introduced, called the Pareto Archived Evolution Strategy (PAES). It was argued that PAES may represent the simplest possible nontrivial algorithm capable of generating diverse solutions in the Pareto optimal set [34]. The objective of another paper was to introduce a novel Pareto-frontier Differential Evolution (PDE) algorithm to solve multiobjective functions. The solutions provided by the proposed algorithm for two standard test problems outperform the Strength Pareto Evolutionary Algorithm, one of the state-of-the-art evolutionary algorithms for solving multiobjective functions [35]. In another research, particles of PSO were classified as nondominated, dominated, and Pareto-front particles for finding the optimal placement and sizing configuration of distributed generation in the radial distribution system [36]. A similar Pareto-front PSO approximation was applied for discrete time-cost trade-off problem [37]. A Pareto-glowworm swarm optimization method was developed for a quality of service problem in a wireless sensor network [38]. A Pareto improved artificial fish swarm algorithm (IAFSA) was improved for a disassembly line balancing problem [39]. In another research, the Pareto analysis was used to select the best solver in the particle swarm [40]. A virtual Pareto front (vPF) structure was included to improve the multiobjective PSO method in a recent research [41]. A novel Pareto-active learning method

was included to surrogate assisted-PSO for solving multi-objective optimization problems in one of the most recent researches [42].

In this study, the Pareto analysis was used for the determination of the places where the probability of the solution is maximum, that is, in which *progressive articulation of preferences* is made, and then the results were found by introducing additional limits to the particle swarm optimization of a locomotive's mathematical equation. The advantages of this innovative method are as follows:

- (1) The improvement of the algorithm is simple to implement by including a Pareto Optimality Check process to the PSO structure
- (2) The improvement reduces the necessary number of iterations to reach the optimal solution
- (3) Although the iteration number is small, the hybrid system gives better results

In *Methods* section, firstly, reaching minimal locomotive motion resistance in the past is given in brief, then the mathematical expression of the original PSO and a general explanation of the Pareto Assumption are explained. The optimization problem for the locomotive resistance is explained in the *YHT 65000 High-Speed Train* section. Embedding the Pareto assumption in a constrained PSO algorithm is introduced in the *Application of Pareto Assumption on Constrained PSO* section. The optimization performances of pure constrained PSO and constrained PSO with the Pareto assumption are benchmarked in the *Comparative Results* section. Finally, the results are interpreted and the suggestions are presented in the *Conclusions* section.

## 2. Methods

Some studies generally depend on Davis' polynomial to find minimal locomotive motion resistance in the literature [43, 44]. Another study uses the PSO to avoid highest locomotive motion resistance. However, not be obtained correctly with the PSO method, the method needs hybridizing with a different method [45]. In a study, the flow chart is presenting the logical steps that were followed in order to obtain optimal speed profiles regarding minimum energy consumption. The program takes about 10 seconds to run [46]. This study considers the theoretical analysis, modeling technique, and experimental verification of the effects of the gradient, running, and curve resistance on train performance [47].

In our study, a new hybrid method was created by combining the advantages of two different methods: one is a metaheuristic algorithm and the other is a statistical rule. Studies related to this have been performed before, and the main studies are given hereafter. However, in previous studies, different from this study, the results of constrained PSO have been evaluated as to whether or not constrained PSO is a good method for the study by looking at whether it complies with the 80/20 rule.

**2.1. Mathematical Expression of PSO.** In a PSO, a sequence of particles with random positions and velocities is initiated at

size  $D$ . Dimension  $D$  is also equal to the unknown number in the conformity function. The goal here is to find the best value by updating its generations. At each iteration, each particle is updated according to the two "best" values. In fact, there are two aspects to the best value. The first is the best suitability value a particle has ever found. This value is stored in memory for use later and is referred to as "pbest," the best value of the particle. The second best value is the best fitness value that any particle in the swarm has ever achieved. This value is the global best value for the swarm and is called "gbest." The speeds and positions are changed according to these new assigned values. The swarm particle matrix is  $n \times D$  in size, where  $n$  is the number of particles:

$$X = \begin{bmatrix} X_{11} & \cdots & X_{1D} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nD} \end{bmatrix}. \quad (1)$$

Particle updates velocity (amount of change of position in each size) and position according to (2) and (3):

$$V_i^{k+1} = V_i^k + c1 * \text{rand}_1^k (pbest_i^k - x_i^k) + c2 * \text{rand}_2^k (gbest^k - x_i^k), \quad (2)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1}. \quad (3)$$

In order to better understand the situations described in Figures 1 and 2, they are shown as working principle and flow chart, respectively.

**2.2. Pareto Assumption.** The Pareto analysis can be generalized by three different approximations in the literature. These approximations belong to Burell [48], Egghe [49], and Chen et al. [50]. In this study, Egghe's equation was chosen, which is explained by (4) and (5):

$$X = X(n)^2, \quad (4)$$

$$Q = Q(n), \quad (5)$$

where  $n$  is the number of examples,  $X$ : 20% rate,  $Q$ : 80% rate.

The Pareto analysis is explained in terms of the 80/20 rule, which was also mentioned in Introduction. The application of the 80/20 rule is demonstrated in Figure 3. In Figure 3, the distribution of the input parameters is given by a decreasing Probability Mass Function. The cumulative rate determines which parameters are critical and which parameters can be neglected.

From the information presented in Figure 3, it is seen that problems 1 and 2 intersect with the cumulative distribution curve. This curve explains that 80% of the problems will be solved by finding solutions to problems 1 and 2.

To obtain the results presented in Figure 3, the following preparations should be applied:

- (1) Define and list the parameters of the optimization problems
- (2) Sort the parameters in order of priority
- (3) Score the sorted parameters in descending order

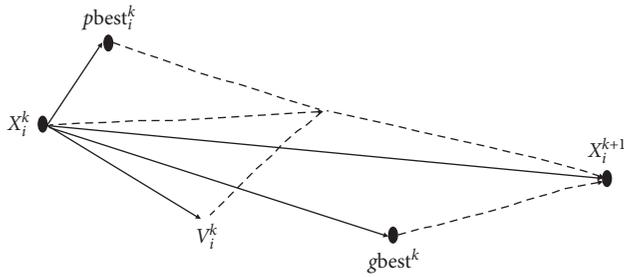


FIGURE 1: Working principle of particle swarm optimization [1].

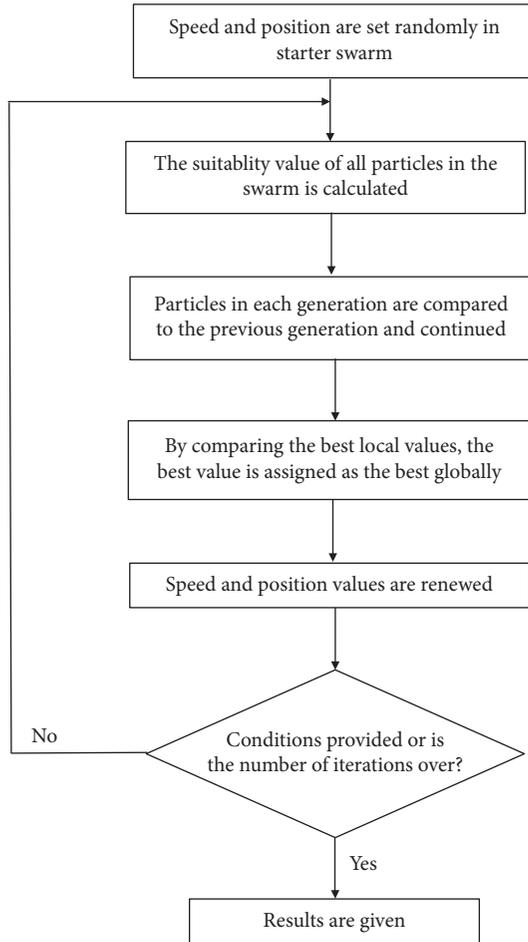


FIGURE 2: PSO flow chart.

- (4) Separate the most critical parameters from the others
- (5) Construct the diagram

The result may not always be 80/20. This ratio is sometimes 70/30, 90/10, etc. It does not seem possible to interpret the ratio without creating a diagram in terms of the solution.

### 3. YHT 65000 High-Speed Train

This train is a high-speed train model that is currently used in Turkey. It is a high-speed train set produced by the

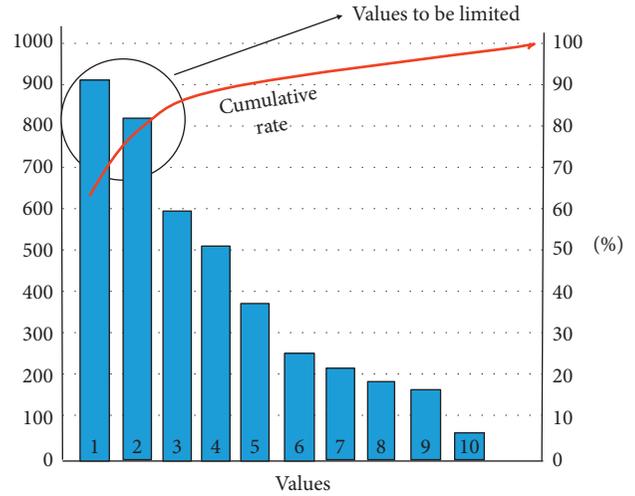


FIGURE 3: Diagram of the Pareto analysis.

Spanish railway manufacturer Construcciones y Auxiliar de Ferrocarriles (CAF).

YHT 65000 trains are based on the trains that Red Nacional De Ferrocarriles Espanoles (RENFE) Class with 120 trains uses in Spain. A set consists of 6 cars as standard. In this study, it was thought that there were 6 cars. However, it has a modular structure, and 2 more cars can be added if desired. In addition, 2 sets can be combined to form a total of 12 cars. Figure 4 shows the YHT 65000 train in Turkey. Before this study, the technical information of the high-speed train used in the study is given in Table 1.

For to find the mathematical model of YHT 65000 High-Speed Train, four resistances, which prevent the movement of high-speed trains, are calculated (aerodynamic resistance is neglected). These are listed as combined resistance of cruise and locomotive, curve resistance, ramp resistance, and acceleration resistance. Since there is also a fifth resistance, which is wind resistance, and it has a probabilistic behavior due to the characteristics of environmental conditions, the effect of this fifth resistance is neglected in the constructed model. It should be noted that these resistors directly affect power consumption. In this case, it is possible to express the total resistance (RT) equation as follows ( $G$  is the total load carried in tons):

#### 3.1. Combined Resistance of Cruise and Locomotive.

$$R_S = (1.3953 - 0.0071 \cdot V + 0.0006 \cdot V^2)G. \quad (6)$$

Since combined resistance of cruise and locomotive exists for a high-speed train set, it consists of a single resistance provided by the manufacturer, which is inseparable as car and locomotive resistance.

#### 3.2. Curve Resistance.

$$R_k = \left( \frac{650}{k - 55} \right) G. \quad (7)$$



FIGURE 4: YHT 65000 high-speed train [51].

TABLE 1: Technical information of high-speed YHT 65000 train\*.

Main characteristics	YHT 65000
Power	38400 kW
Locomotive load	297.25 ton
Axle load	17 ton
Axle type	—
Maximum velocity	275 km/h
Line gap	1435 mm
Catenary type	AC 25 kV, 50 Hz
Traction motor power	AC 4800 kW

\*Obtained from Turkish State Railways (TCDD).

This curve resistance formula is known as the Röcki formula and is used for 1435 mm line length. The  $k$  value in the equation is the curvature radius of the line in  $m$ .

### 3.3. Ramp Resistance.

$$R_r = rG, \quad (8)$$

where  $r$  is the ramp value in %. This value is taken as positive when climbing the ramp and negative when descending.

In order for the train set to move, it must overcome these resistances. A train can only move with “Steady” velocity after overcoming these resistances. At steady velocity, there is no acceleration ( $a$ ) (it is equal to zero) according to Newton’s 1st Law. There is one more resistance that must be overcome when the train wants to change its velocity. This resistance is called acceleration resistance.

### 3.4. Acceleration Resistance.

$$R_a = \left( \frac{4V}{S} \right) G. \quad (9)$$

The acceleration resistance given above is the acceleration resistance of the train set.  $S$  refers to the line.

While there is no acceleration in the first three resistance types, in the fourth one, there is a rise in acceleration resistance when the speed changes. The power consumption is determined by applying the PSO method to the total resistance formula:

$$R = R_s + R_k + R_r + R_a. \quad (10)$$

The equation in full is written as

$$R = \left[ V^2 \left( 0.0006 + \frac{4}{D} \right) - 0.0071V + 1.3953 + \frac{650}{k-55} + r \right] G, \quad (11)$$

$$V = at,$$

$$R = \left[ a^2 t^2 \left( 0.0006 + \frac{4}{D} \right) - 0.0071at + 1.3953 + \frac{650}{k-55} + r \right] G. \quad (12)$$

Constraints are

$$-28 \leq a \leq 28 \text{ (m/s}^2\text{)}, \quad (13)$$

$$1800 \leq t \leq 3600 \text{ (s)}, \quad (14)$$

$$-26 \leq r \leq 26 \text{ (\%_0)}, \quad (15)$$

$$130 \leq k \leq 530 \text{ (m)}, \quad (16)$$

$$0 \leq D \leq 300000 \text{ (m)}, \quad (17)$$

$$287 \leq G \leq 328.425 \text{ (ton)}. \quad (18)$$

## 4. Application of Pareto Assumption on Constrained PSO

The Pareto assumption of speed and load parameters was obtained by using  $E = (1/2)mV^2$ , the most general equation of the amount of energy used. While deriving these assumptions, the speed was progressively decreased from 240 km/h to 10 km/h by 10 km/h each time, and 24 levels of the speed were obtained. The load was also progressively decreased from 328.425 tons to empty weight of cruise, which is 287 tons, by 1.5 tons in each time, and 24 levels of load were obtained. The Pareto analysis of speed and load is demonstrated in Figures 5 and 6 respectively.

In Figure 5, it is seen that the share of the first 9 speed levels, including 160 km/h, in energy consumption is 60%. These 9 levels maximize the energy consumption. This curve explains that 60% of the energy consumption will be solved by finding solutions to problems from 1 to 9. They should be excluded from the search of PSO, which means that the speed must be lower than 160 km/h or 44.44 m/sec.

In Figure 6, it is seen that the share of the first 15 load levels, including 309.294 ton, in energy consumption is 63.73%. These 15 levels maximize the energy consumption. This curve explains that 63.73% of the energy consumption will be solved by finding solutions to problems from 1 to 15. They should be excluded from the search of PSO, which means that the load must be lower than 309.294 ton.

From the inference made according to Figures 5 and 6, it is seen that the correction to be made only for the speed is contribution to the energy consumed, because 15 levels of the load mean big limitation. This limitation is not so likely.

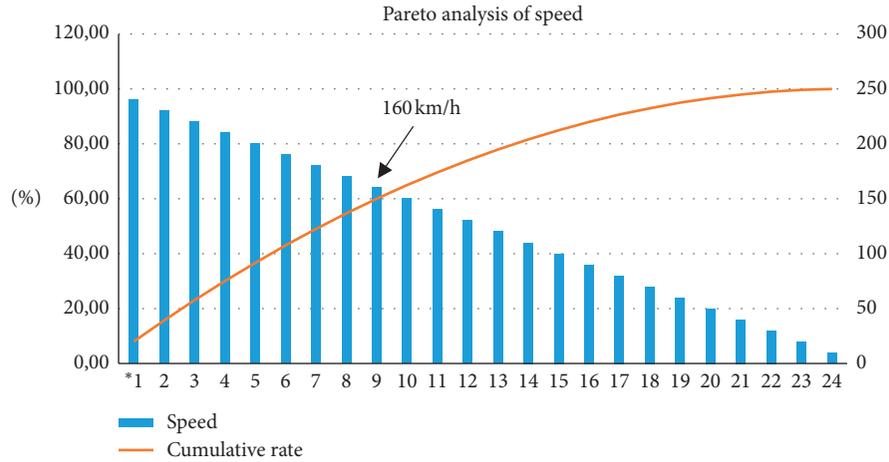


FIGURE 5: Pareto diagram of speed. \*1: 240 km/h, 2: 230 km/h,..., 24:10 km/h.

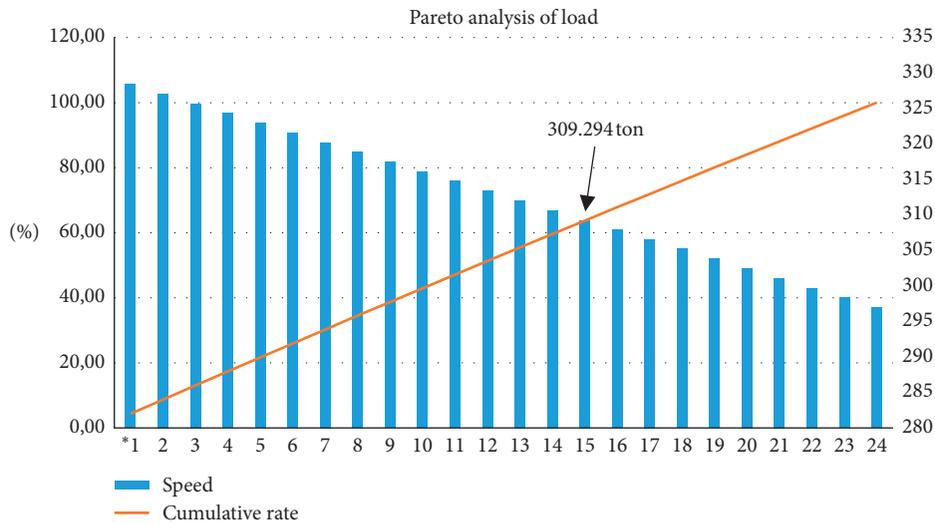


FIGURE 6: Pareto diagram of load. \*1: 328,418 ton, 2: 327.052 ton,..., 24: 297 ton.

Although trains can go at low speed, they cannot go at low load in practice.

For this reason, the algorithm can be made hybrid by only changing the speed in the swarm algorithm used for the optimization of mathematical modeling. Algorithms 1–4 give the details of the application of the proposed model on the locomotive resistance minimization problem. Algorithm 3 gives the details of the improvement on PSO by Pareto analysis. Algorithm 4 shows how the algorithm is restricted by constraints.

The application of pareto optimality conditions to obtain the hybrid Pareto-PSO algorithm can also be expressed by the flow chart shown in Figure 7.

This flow chart is different from PSO flow chart because of the red process. This red process is coming from making constraints according to Pareto analysis.

## 5. Determination of PSO Parameters

In the PSO part of the proposed algorithm, there are two critical parameters that should be chosen, which are the

number of necessary particles ( $N_{\text{particles}}$ ) and the number of necessary iterations ( $N_{\text{iterations}}$ ). The parameter  $c_{\text{training}}$  is the counter parameter, which is used for iteration tracking, which is initialized as 0 at the beginning and incremented after each iteration unless it is reached  $N_{\text{iterations}}$ . The particle update velocity vector  $\vec{V}$  is initialized and updated relatively randomly at each iteration, where the values of its elements are bounded by the difference vectors of optimum solution vector and solution vector of each particle from the previous iteration, as seen in Algorithm 1.  $\vec{V}$  is used to update solution vector  $\vec{X}_{\text{swarm}}(i)$  of each swarm. However, this update step is limited by Algorithm 3 for the Pareto optimality check. Algorithm 3 is only used when Pareto's assumption is handled. In the pure PSO algorithm, this algorithm is not applied. Moreover, for the constraint optimization  $\vec{X}_{\text{swarm}}(i)$  is also limited by Algorithm 4.

The high number of particles up to value will accelerate to reach the optimum point. However, when the number of particles exceeds the upper limit, the complexity of the algorithm increases, and deviations in learning the extremum point occur. Therefore, the particles should be limited by the

```

Parameter Initialization
Repeat,
   $c_{\text{training}} = c_{\text{training}} + 1$ 
   $\vec{R} = f_{\text{Resistance}}(\vec{X}_{\text{swarm}})$ 
   $R_{\text{min}} = \min(\vec{R})$ 
   $X_{\text{swarm,opt}} = \vec{X}_{\text{swarm}}(\arg \min(\vec{R}))$ 
   $\vec{V} = 2 \cdot \text{rand}([0, 1]) \cdot (X_{\text{swarm,opt}} - \vec{X}_{\text{swarm}})$ 
   $\vec{X}_{\text{swarm}} = \vec{X}_{\text{swarm}} + \vec{V}$ 
Pareto Optimality Check
Constraints Check
Until  $c_{\text{training}} == N_{\text{iterations}}$ 

```

ALGORITHM 1: Pareto-PSO algorithm for locomotive resistance minimization.

```

Initialize  $N_{\text{particles}}$ 
Initialize  $N_{\text{Iterations}}$ 
for  $i = 1$  to  $N_{\text{particles}}$ 
   $X_{\text{swarm}}(i).a = \text{rand}([-28, 28])$ 
   $X_{\text{swarm}}(i).t = \text{rand}([1800, 3600])$ 
   $X_{\text{swarm}}(i).D = \text{rand}([0, 300000])$ 
   $X_{\text{swarm}}(i).r = \text{rand}([-26, 26])$ 
   $X_{\text{swarm}}(i).k = \text{rand}([130, 530])$ 

   $X_{\text{swarm}}(i).G = \text{rand}([287, 328.45])$ 
end

 $X_{\text{constraints}} = \begin{bmatrix} -28 & 28 \\ 1800 & 3600 \\ -26 & 26 \\ 130 & 530 \\ 0 & 300000 \\ 287 & 328.425 \end{bmatrix}$ 

 $c_{\text{training}} = 0.$ 

```

ALGORITHM 2: Parameter initialization.

```

for  $i = 1$  to  $N_{\text{particles}}$ 
  if  $(X_{\text{swarm}}(i).a) \times (X_{\text{swarm}}(i).t) > 50$ 
     $(X_{\text{swarm}}(i).a) = 50 / (X_{\text{swarm}}(i).t)$ 
  else if  $(X_{\text{swarm}}(i).a) \cdot (X_{\text{swarm}}(i).t) < -50$ 
     $(X_{\text{swarm}}(i).a) = -50 / (X_{\text{swarm}}(i).t)$ 
  end
end.

```

ALGORITHM 3: Pareto optimality check.

upper value to avoid this computational complexity, redundancy, and distortion in convergence steps.

For the purpose of finding optimum ( $N_{\text{particles}}, N_{\text{iterations}}$ ) couple, the experiment was repeated by changing the number of particles from 1 to 20 and the number of iterations from 1 to 50 for PSO without and with Pareto assumption. Experiments were carried out on the R2020a version of the Matlab program on a computer with 2.8 GHz, Intel Core i7 processor, 16 Gb 1067 Mhz DDR3 RAM, and MacOS High Sierra operating system.

The minimum resistance and elapsed time for the calculation in seconds are obtained as in Figures 8 and 9 for PSO without Pareto assumption.

As can be seen from Figure 8, the best solution for the minimum resistance was obtained with the minimum ( $N_{\text{particles}}, N_{\text{iterations}}$ ) configuration, when the number of particles was chosen as 20 and the number of iterations was chosen as 25 for pure PSO method.

As seen in Figure 9, 22.4 milliseconds was spent for the simplest ( $N_{\text{particles}}, N_{\text{iterations}}$ ) configuration of pure PSO

```

for  $i = 1$  to  $N_{\text{particles}}$ 
  if  $(X_{\text{swarm}}(i).a) < X_{\text{constraints}}(1, 1)$ 
     $X_{\text{swarm}}(i).a = X_{\text{constraints}}(1, 1)$ 
  else if  $(X_{\text{swarm}}(i).a) > X_{\text{constraints}}(1, 2)$ 
     $X_{\text{swarm}}(i).a = X_{\text{constraints}}(1, 2)$ 
  end
  if  $(X_{\text{swarm}}(i).t) < X_{\text{constraints}}(2, 1)$ 
     $X_{\text{swarm}}(i).t = X_{\text{constraints}}(2, 1)$ 
  else if  $(X_{\text{swarm}}(i).t) > X_{\text{constraints}}(2, 2)$ 
     $X_{\text{swarm}}(i).t = X_{\text{constraints}}(2, 2)$ 
  end
  if  $(X_{\text{swarm}}(i).D) < X_{\text{constraints}}(3, 1)$ 
     $X_{\text{swarm}}(i).D = X_{\text{constraints}}(3, 1)$ 
  else if  $(X_{\text{swarm}}(i).D) > X_{\text{constraints}}(3, 2)$ 
     $X_{\text{swarm}}(i).D = X_{\text{constraints}}(3, 2)$ 
  end
  if  $(X_{\text{swarm}}(i).r) < X_{\text{constraints}}(4, 1)$ 
     $X_{\text{swarm}}(i).r = X_{\text{constraints}}(4, 1)$ 
  else if  $(X_{\text{swarm}}(i).r) > X_{\text{constraints}}(4, 2)$ 
     $X_{\text{swarm}}(i).r = X_{\text{constraints}}(4, 2)$ 
  end
  if  $(X_{\text{swarm}}(i).k) < X_{\text{constraints}}(5, 1)$ 
     $X_{\text{swarm}}(i).k = X_{\text{constraints}}(5, 1)$ 
  else if  $(X_{\text{swarm}}(i).k) > X_{\text{constraints}}(5, 2)$ 
     $X_{\text{swarm}}(i).k = X_{\text{constraints}}(5, 2)$ 
  end
  if  $(X_{\text{swarm}}(i).G) < X_{\text{constraints}}(6, 1)$ 
     $X_{\text{swarm}}(i).G = X_{\text{constraints}}(6, 1)$ 
  else if  $(X_{\text{swarm}}(i).G) > X_{\text{constraints}}(6, 2)$ 
     $X_{\text{swarm}}(i).G = X_{\text{constraints}}(6, 2)$ 
  end
end.

```

ALGORITHM 4: Constraints check.

solution. The maximum duration was 56.44 and 54.55 milliseconds when the number of particles was selected as 18, the number of iterations was 50 or the number of particles was 20, and the number of iterations was 50. However, when Figure 8 was examined again, it was seen that these maximum configurations did not provide the minimum resistance result. When Figure 9 is examined, it can be said that pure PSO shows a linear computational complexity in terms of both particle number and iteration number.

When the Pareto assumption was applied for the same experiments, the minimum resistances and elapsed computation time were obtained as in Figures 10 and 11.

It can be understood from Figure 10 that the best solution for the minimum resistance was obtained with the minimum  $(N_{\text{particles}}, N_{\text{iterations}})$  configuration, when the number of particles was chosen as 10 and the number of iterations was chosen as 12 for pure PSO method.

As seen in Figure 11, 9.2 milliseconds was spent for the simplest  $(N_{\text{particles}}, N_{\text{iterations}})$  configuration of PSO with Pareto assumption. The maximum duration was 69.9 and 56.02 milliseconds when the number of particles was selected as 14, the number of iterations was 49 or the number of particles was 20, and the number of iterations was 50. When Figure 10 was examined again, the minimum

resistance can also be found when (20, 50) was chosen instead of minimum configuration.

According to Figure 11, it can be said that PSO with Pareto assumption also shows a linear computational complexity in terms of both particle number and iteration number as similar to pure PSO method. The inclusion of Algorithm 3 ensures that the optimum result is achieved with fewer iterations and particles, although there is no significant burden on the computational complexity for the same number of particles and iterations. For optimum particle numbers obtained by experiments  $(N_{\text{particles, PSO}} = 20, N_{\text{particles, Pareto-PSO}} = 10)$ , the comparison of the pure PSO and the Pareto assumption method is presented in detail in the Comparative Results section.

## 6. Comparative Results

In Figure 12, the convergences for the optimal conditions that satisfy the minimum resistance are illustrated for each training iteration of pure constrained PSO with 20 particles, which is the simplest PSO structure that can find minimum resistance as presented in Figure 8. In addition, the values reached for minimum resistance and elapsed time for each iteration are also presented.

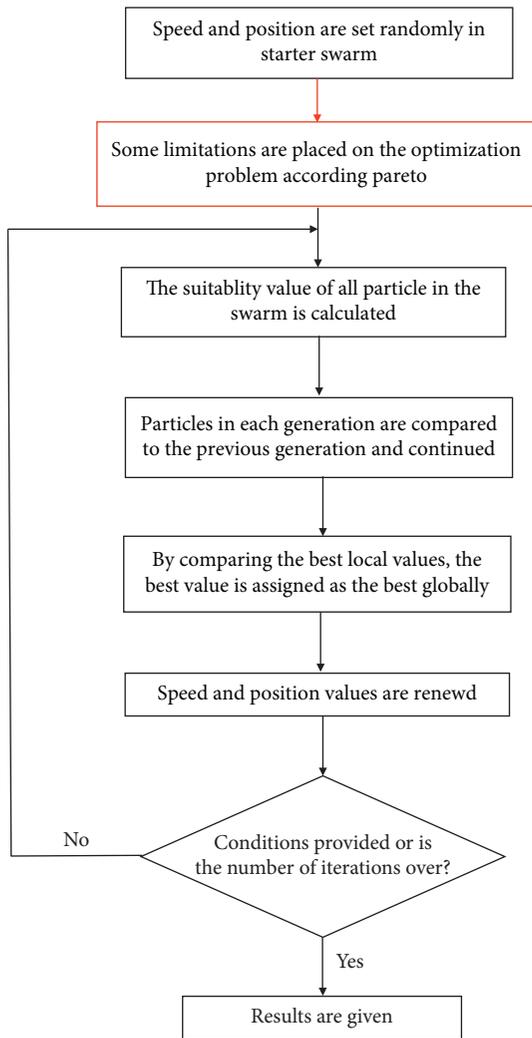


FIGURE 7: Pareto-PSO hybrid system flow chart.

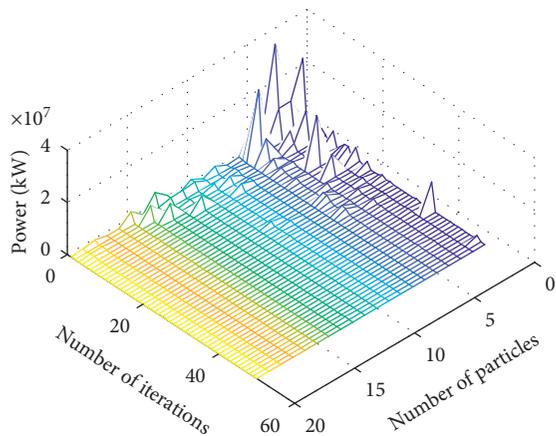


FIGURE 8: Minimum resistances found by different ( $N_{\text{particles}}$ ,  $N_{\text{iterations}}$ ) Configurations of PSO.

In Figure 12, the iterations are limited to 50. The best acceleration, ramp, distance, power, travel time, curve, load, and elapsed time values as found by the PSO

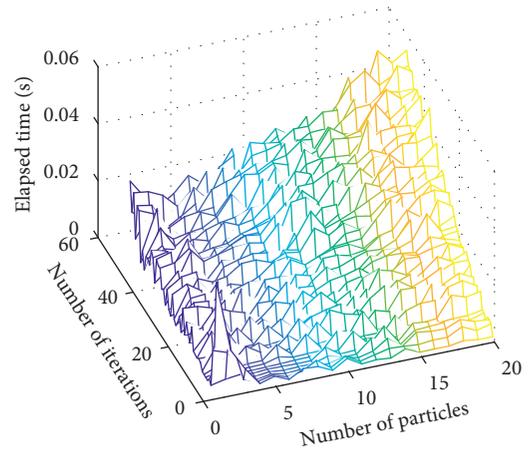


FIGURE 9: Elapsed computation time by different ( $N_{\text{particles}}$ ,  $N_{\text{iterations}}$ ) Configurations of PSO.

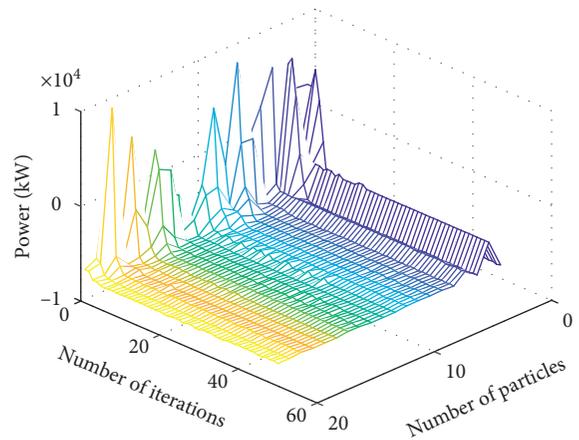


FIGURE 10: Minimum Resistances Found by Different ( $N_{\text{particles}}$ ,  $N_{\text{iterations}}$ ) configuration of Pareto-PSO.

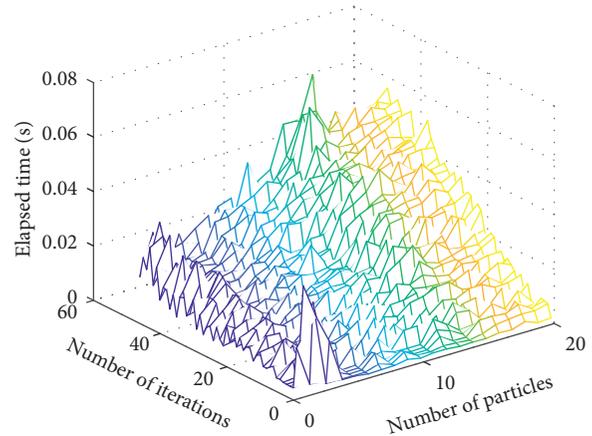


FIGURE 11: Elapsed computation time by different ( $N_{\text{particles}}$ ,  $N_{\text{iterations}}$ ) Configurations of Pareto-PSO.

are detailed. The optimum resistance was reached at the 25th iteration with 22.4-millisecond computation time. When the iterations were continued until the 50th

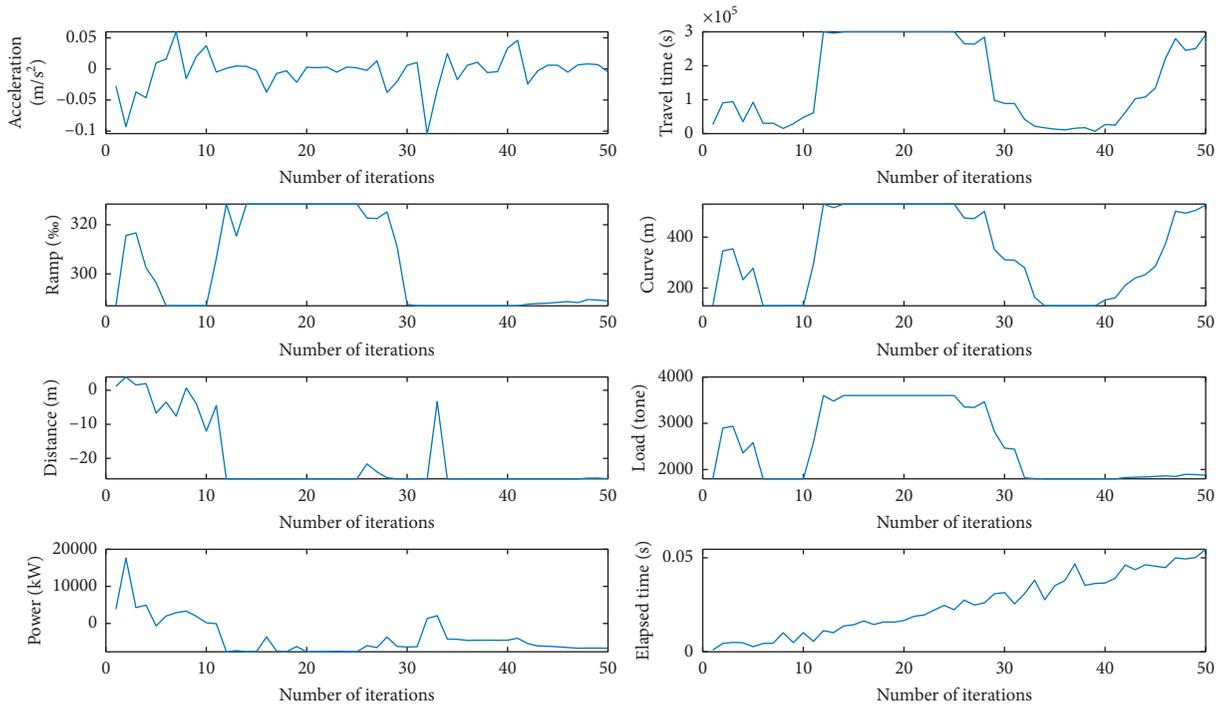


FIGURE 12: Iteration numbers of PSO method for different variables.

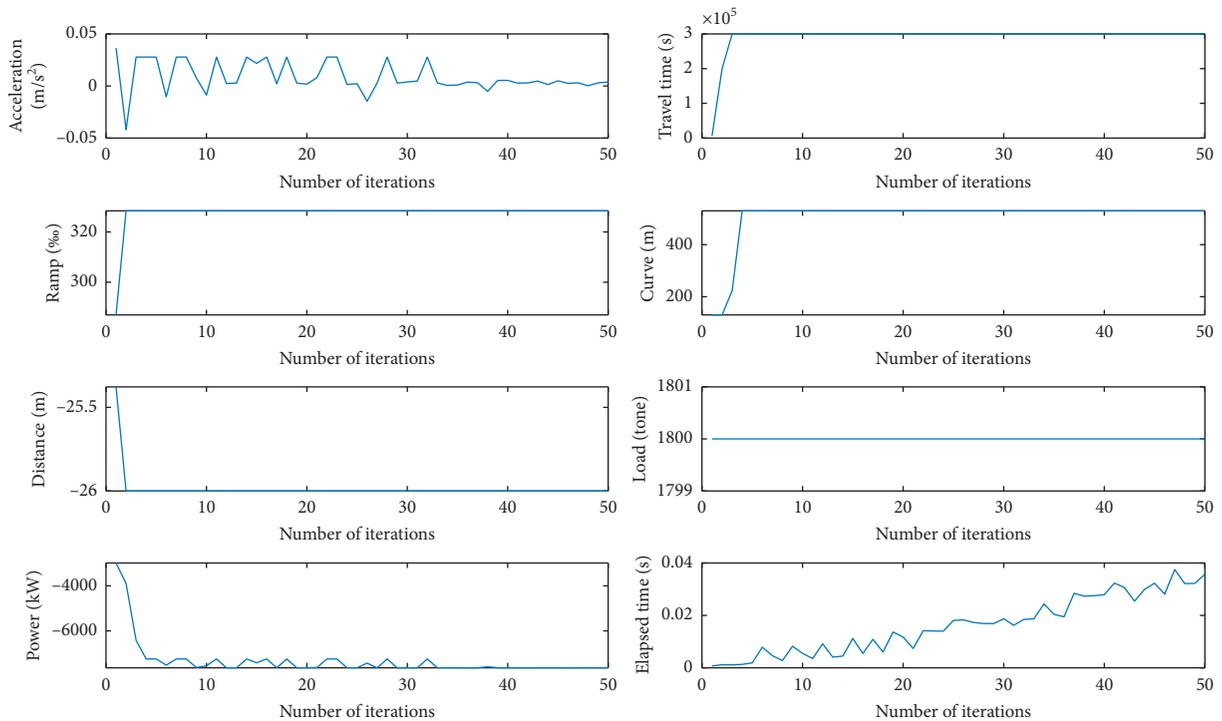


FIGURE 13: Iteration numbers of Pareto-PSO hybrid method for different variables.

TABLE 2: Values of  $a$ ,  $D$ ,  $G$ ,  $k$ ,  $r$ , and  $t$  in Different Iteration Numbers for Constrained PSO.

Iteration	$A$	$D$	$G$	$k$	$r$	$t$
1.	-0.0276	26983	287	130	1.1505	1800
2.	-0.0927	90676	315.62	345.97	3.8663	2896
3.	-0.0371	94090	316.69	354.10	1.5462	2937.2
4.	-0.0463	35362	302.45	232.49	1.9330	2357.3
5.	0.0097	92365	296.45	277.32	-6.7304	2581.8
10.	0.0373	47879	287	130	-12.052	1800
20.	0.0027	300000	328.425	530	-26	3600
30.	0.0058	88866	287.56	311.41	-26	2464.5
40.	0.0334	26779	287	152.93	-26	1800
50.	-0.0047	29192	289	525.92	-25.968	1876.7

TABLE 3: Values of  $a$ ,  $D$ ,  $G$ ,  $k$ ,  $r$ , and  $t$  in Different Iteration Numbers for Constrained Pareto PSO.

Iteration	$A$	$D$	$G$	$k$	$r$	$t$
1.	0.0366	6263.7	287	130	-25.378	1800
2.	-0.0419	1993.6	328.425	130	-26	1800
3.	0.0278	300000	328.425	222.45	-26	1800
4.	0.0278	300000	328.425	530	-26	1800
5.	0.0278	300000	328.425	530	-26	1800
10.	-0.0087	300000	328.425	530	-26	1800
20.	0.0018	300000	328.425	530	-26	1800
30.	0.0039	300000	328.425	530	-26	1800
40.	0.0054	300000	328.425	530	-26	1800
50.	0.0038	300000	328.425	530	-26	1800

iteration, the computation time was reached to 54.6 milliseconds.

In Figure 13, the convergences for the optimal conditions that satisfy the minimum resistance are illustrated for each training iteration of the hybrid Pareto-PSO method. The values reached for the minimum resistance and elapsed time are also presented.

In Figure 13, the iterations are also limited to 50. The best acceleration, ramp, distance, power, travel time, curve, load, and elapsed time values as found by the Hybrid system are shown. The minimum resistance was reached at the 12th iteration with 9.2-millisecond computation time. When the iterations were continued until 50th iteration, the computation time was reached to 35.7 milliseconds. According to these results, it can be said that Pareto assumption on constrained PSO method reduced computation time to 41.07% of minimum configuration for the best solution of constrained PSO method. Moreover, the computation time was also reduced to 65.38% by Pareto assumption, when both constrained PSO and constrained Pareto-PSO methods were continued until to the 50th iteration.

Both constrained PSO and constrained Pareto-PSO methods could find the conditions that satisfy the minimum resistance as  $-7638$  kW for their optimum ( $N_{\text{particles}}$ ,  $N_{\text{iterations}}$ ) configurations. This situation can be preserved by the constrained Pareto-PSO method when it is reached until the 50th iteration, but not by the constrained PSO method. At 50th iteration, the minimum resistance found by the constrained PSO increased from  $-7638$  kW to  $-6670$  kW.

Acceleration, distance, load, curve, ramp, and travel time values that can be seen for the increasing number of iterations of the pure constrained PSO are detailed in Table 2.

If the values found by constrained PSO, which are listed in Table 2, are investigated, it can be said that distance, load, and ramp reached relatively stable states between 30th and 40th iterations. Travel time and curve reached a stable state after the 12th iteration until the 25th iteration. After the 25th iteration, computed travel time and curve diverged from their stability point. Acceleration could not reach a stable state until the 50th iteration.

Acceleration, distance, load, curve, ramp, and travel time values can be seen for the increasing number of iterations of constrained Pareto PSO in Table 3.

Distance, load, curve, ramp, and time reached a stable value in iteration 4. On the other hand, acceleration could not reach a stable value until the 50th iteration.

## 7. Conclusions

In this study, a novelty hybrid methodology was developed using the Pareto analyses and the PSO algorithm. Firstly, a mathematical model of a high-speed train in Turkey (YHT 65000) was used as an objective function for optimization. Next, the constrained PSO and this new hybrid system of Pareto and constrained PSO were compared by using the findings of the optimal values of acceleration, ramp, distance, power, travel time, curve, load, and elapsed time for reaching minimal motion resistance.

This new model has some advantages compared with constrained PSO. Firstly, it has a very simple algorithm that only adds one process to the PSO flow chart. Secondly, the number of iterations required is less than PSO. When the experiments were repeated 100 times for both constrained PSO method and proposed hybrid Pareto-PSO method for

their optimum number of particles ( $N_{\text{particles}}$ , PSO = 20,  $N_{\text{particles}}$ , Pareto-PSO = 10), PSO found an average value of  $-6035.7$  kW for minimum resistance, with an average of 30.51 iterations and 29.4-millisecond computation time. On the other hand, the proposed hybrid method found an average value of  $-7483.5$  kW for minimum resistance, with an average of 29.45 iterations and 16.3-millisecond calculation time for finding the optimum  $a$  (acceleration),  $D$  (distance),  $G$  (load),  $k$  (curve),  $r$  (ramp), and  $t$  (time) values. These performance results show that the proposed hybrid method reduces the calculation time of the original PSO to 55.44% on average, obtaining much more reliable values for minimum resistance.

This novel hybrid method can be used for the optimization of other mathematical equation easily. This system not only is simple to implement, but also gives good performance in terms of both result and iteration. The method can also be adapted to conventional locomotives by using their constraints on acceleration, load, duration, travel time, ramp, curve, and travel distance instead of constraints of the chosen high-speed locomotive in this study. This feature of the presented method not only makes the proposed study a special method for high-speed trains, but also gives a general approach applicable to all locomotives for motion resistance optimization.

## Data Availability

No specific data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] M. Sertsoz, *Raylı sistemlerde güneş enerjisi destekli yenilikçi enerji modellenmesi, optimizasyonu ve analizi*, Ph.D. thesis, Bilecik Şeyh Edebali University, Enerji Sistemleri Mühendisliği, Bilecik, Turkey, 2018.
- [2] M. Dorigo, V. Maniezzo, and A. Coloni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 1, pp. 29–41, 1996.
- [3] T. Stützle and H. H. Hoos, "The Max–Min ANT system and local search for combinatorial optimization problems," *Future Generation Computer Systems*, vol. 16, 2000.
- [4] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE, Nagoya, Japan, October 1995.
- [5] D. Karaboga and B. Basturk, "Sayısal fonksiyon optimizasyonu için güçlü ve verimli bir algoritma: yapay arı kolonisi (ABC) algoritması," *Journal Of Global Optimazition*, vol. 39, no. 3, 2007.
- [6] W.-T. Pan, "A new fruit fly optimization algorithm: taking the financial distress model as an example," *Knowledge-Based Systems*, vol. 26, pp. 69–74, 2012.
- [7] X. S. Yang and S. Deb, "Cuckoo search via lévy flights," in *Proceedings of 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, pp. 9–11, IEEE, Coimbatore, India, December 2009.
- [8] A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831–4845, 2012.
- [9] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [10] X. S. Yang, "A new metaheuristic Bat–Inspired algorithm," in *Proceedings of the International Workshop on Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)*, Tenerife Spain, November 2008.
- [11] X.-S. Yang, "Firefly algorithms for multimodal optimization," *Stochastic Algorithms: Foundations and Applications*, vol. 5792, pp. 169–178, 2009.
- [12] B. R. Rajakumar, "The lion's algorithm: a new nature-inspired search algorithm," *Procedia Technology*, vol. 6, pp. 126–135, 2012.
- [13] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [14] A. Kaveh and N. Farhoudi, "A new optimization method: Dolphin echolocation," *Advances in Engineering Software*, vol. 59, pp. 53–70, 2013.
- [15] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.
- [16] S. A. Uymaz, G. Tezel, and E. Yel, "Artificial algae algorithm with multi-light source for numerical optimization and applications," *Biosystems*, vol. 138, pp. 25–38, 2015.
- [17] M. D. Li, H. Zhao, X. W. Weng, and T. Han, "A novel nature-inspired algorithm for optimization: virus colony search," *Advances in Engineering Software*, vol. 92, pp. 65–88, 2016.
- [18] O. Abedinia, N. Amjady, and A. Ghasemi, "A new metaheuristic algorithm based on shark smell optimization," *Complexity*, vol. 21, no. 5, pp. 97–116, 2014.
- [19] J. J. Q. Yu and V. O. K. Li, "A social spider algorithm for global optimization," *Applied Soft Computing*, vol. 30, pp. 614–627, 2015.
- [20] P. Erdoğmuş, "Doğadan esinlenen optimizasyon algoritmaları ve optimizasyon algoritmalarının optimizasyonu," *Düzce Üniversitesi Bilim Ve Teknoloji Dergisi*, vol. 4, pp. 293–304, 2016.
- [21] J. Chen, J. Zheng, P. Wu, L. Zhang, and Q. Wu, "Dynamic particle swarm optimizer with escaping prey for solving constrained non-convex and piecewise optimization problems," *Expert Systems with Applications*, vol. 86, pp. 208–223, 2017.
- [22] R. Moeini and M. Babaei, "Constrained improved particle swarm optimization algorithm for optimal operation of large scale reservoir: proposing three approaches," *Evolving Systems*, vol. 8, no. 4, pp. 287–301, 2017.
- [23] S. Duman, "A modified moth swarm algorithm based on an arithmetic crossover for constrained optimization and optimal power flow problems," *IEEE Access*, vol. 6, pp. 45394–45416, 2018.
- [24] G. E. Tsekouras, J. Tsimikas, C. Kalloniatis, and S. Gritzalis, "Interpretability constraints for fuzzy modeling implemented by constrained particle swarm optimization," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 4, pp. 2348–2361, 2018.
- [25] M. Wang, J. Luo, J. Yuan, and U. Walter, "Coordinated trajectory planning of dual-arm space robot using constrained particle swarm optimization," *Acta Astronautica*, vol. 146, pp. 259–272, 2018.

- [26] M. Xu, B. Du, and Y. Fan, "Endmember extraction from highly mixed data using linear mixture model constrained particle swarm optimization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5502–5511, 2019.
- [27] K. M. Ang, W. H. Lim, N. A. M. Isa, S. S. Tiang, and C. H. Wong, "A constrained multi-swarm particle swarm optimization without velocity for constrained optimization problems," *Expert Systems with Applications*, vol. 140, p. 112882, 2020.
- [28] C. L. Hwang and A. S. M. Masud, "Multiple objective decision making—methods and applications," in *Lecture Notes in Economics and Mathematical Systems*, Springer, Berlin, Germany, 1979.
- [29] E. L. Ulungu and J. Teghem, "Multi-objective combinatorial optimization problems: a survey," *Journal of Multi-Criteria Decision Analysis*, vol. 3, no. 2, pp. 83–104, 1994.
- [30] P. Di Barba, M. Farina, and A. Savini, "An improved technique for enhancing diversity in Pareto evolutionary optimization of electromagnetic devices," *COMPEL-The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 20, no. 2, pp. 482–496, 2001.
- [31] U. Baumgartner, C. Magele, and W. Renhart, "Pareto optimality and particle swarm optimization," *IEEE Transactions on Magnetics*, vol. 40, no. 2, pp. 1172–1175, 2004.
- [32] A. Shubham, Y. Dashora, M. K. Tiwari, and S. Young-Jun, "Interactive particle swarm: a pareto-adaptive metaheuristic to multiobjective optimization," *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, vol. 38, no. 2, pp. 258–277, 2008.
- [33] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.
- [34] J. D. Knowles and D. W. Corne, "Approximating the non-dominated front using the pareto archived evolution Strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149–172, 2000.
- [35] H. A. Abbass, R. Sarker, and C. Newton, "PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 1, pp. 971–978, Seoul, Korea, February 2001.
- [36] K. Mahesh, P. Nallagownden, and I. Elamvazuthi, "Advanced pareto front non-dominated sorting multi-objective particle swarm optimization for optimal placement and sizing of distributed generation," *Energies*, vol. 9, no. 12, p. 982, 2016.
- [37] S. Aminbakhsh and R. Sonmez, "Pareto front particle swarm optimizer for discrete time-cost trade-off problem," *Journal of Computing in Civil Engineering*, vol. 31, no. 1, 2017.
- [38] V. K. Burugari and P. S. Periasamy, "Multi QoS constrained data sharing using hybridized pareto-glowworm swarm optimization," *Cluster Computing*, vol. 22, no. S4, pp. 9727–9735, 2017.
- [39] Z. Zhang, K. Wang, L. Zhu, and Y. Wang, "A Pareto improved artificial fish swarm algorithm for solving a multi-objective fuzzy disassembly line balancing problem," *Expert Systems with Applications*, vol. 86, pp. 165–176, 2017.
- [40] S. G. W. Butcher, J. W. Sheppard, and S. Strasser, "Pareto improving selection of the global best in particle swarm optimization," in *Proceedings of the 2018 IEEE Congress on Evolutionary Computation (Cec)*, pp. 662–669, IEEE, Rio de Janeiro, Brazil, July 2018.
- [41] B. L. Wu, W. Hu, Z. N. He, M. Jiang, and G. G. Yen, "A many-objective particle swarm optimization based on virtual pareto front," *IEEE Congress on Evolutionary Computation*, pp. 78–85, 2018.
- [42] Z. Lv, L. Wang, Z. Han, J. Zhao, and W. Wang, "Surrogate-assisted particle swarm optimization algorithm with Pareto active learning for expensive multi-objective optimization," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 838–849, 2019.
- [43] A. Orellano, "Aerodynamics of high speed trains," *Vehicle Aerodynamics Lecture*, KTH, Stockholm, Sweden, 2010.
- [44] S. Arsene and I. Sebeşan, "Analysis of the resistance to motion in the passenger trains hauled by the locomotive LE 060 EA 5100kW," *INCAS Bulletin*, vol. 6, no. 3, pp. 13–21, 2014.
- [45] M. Sertsöz and M. Fidan, "A critical approach to the particle swarm optimization method for finding maximum points," *Journal of Scientific Perspectives*, vol. 4, no. 2, pp. 111–122, 2020.
- [46] P. Gkortzas, *Study on Optimal Train Movement for Minimum Energy Consumption*, Master thesis, Mälardalen University, Västerås, Sweden, 2013.
- [47] V. N. Rangelov, "Gradient modelling with calibrated train performance models," *Computers in Railways XIII*, vol. 127, pp. 123–134, 2012.
- [48] Q. L. Burrell, "The 80/20 rule: library lore or statistical law?" *Journal of Documentation*, vol. 41, no. 1, pp. 24–39, 1985.
- [49] L. Egghe, "Pratt's measure for some bibliometric distributions and its relation with the 80/20 rule," *Journal of the American Society for Information Science*, vol. 38, no. 4, pp. 288–297, 1987.
- [50] Y.-S. Chen, P. Pete Chong, and Y. Tong, "Theoretical foundation of the 80/20 rule," *Scientometrics*, vol. 28, no. 2, pp. 183–204, 1993.
- [51] [https://tr.wikipedia.org/wiki/TCDD\\_HT65000](https://tr.wikipedia.org/wiki/TCDD_HT65000).