

Received August 12, 2021, accepted August 27, 2021, date of publication August 31, 2021, date of current version September 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3109286

Reducing CO₂ Emissions of an Airport Baggage Handling Transport System Using a Particle Swarm Optimization Algorithm

GABRIEL LODEWIJKS¹, YULIAN CAO¹, (Member, IEEE), NING ZHAO², AND HAN ZHANG³

¹School of Aviation, University of New South Wales, Sydney, NSW 2052, Australia

²School of Mechanical Engineering, University of Science and Technology Beijing, Beijing 100083, China

³Institute of Nuclear and New Energy Technology, Tsinghua University, Beijing 100084, China


Corresponding author: Yulian Cao (yulian.cao@unsw.edu.au)

ABSTRACT Optimizing the design of an airport baggage handling transport system (BHTS) with respect to the minimization of the total costs and energy consumption is essential to reduce costs and Carbon dioxide (chemical formula CO₂) emissions in airport operations. This paper introduces a mathematical model that comprehensively considers relevant costs regarding the operation of belt conveyors in a BHTS. Specifically, the **Capital Expenditure (CapEx)** and **Operational Expenditure (OpEx)** are considered in the airport BHTS cost function. Furthermore, to include the impact of CO₂ emissions, the offsetting costs of CO₂ emissions are included in the airport BHTS cost function. This function forms the basis of an objective function that can be used to optimize the airport BHTS's design by metaheuristic algorithms. Three state-of-the-art particle swarm optimization (PSO) algorithms are utilized to solve the airport BHTS optimization problem. The results of experiments show that the three PSO variants can solve the optimization problem effectively and efficiently. The self-regulation PSO algorithm performed the best in terms of CPU time and has been used for the case studies. Extensive tests of the impact of key parameters, e.g., capacity and system length, on the optimized solutions have been conducted. Experiments show that a system with several belt conveyors of shorter lengths performs better than a system with one long conveyor. In reality however, more parameters play a role like the varying baggage throughput per hour and therefore the BHTS problem needs to be optimized case-by-case. Optimizing an airport BHTS design leads to a significant reduction in CO₂ emission and thus costs.

INDEX TERMS Baggage handling transport system, airport operations, belt conveyors, CO₂ emission, particle swarm optimization.

I. INTRODUCTION

The baggage handling transport system (BHTS) at airports is crucial as it majorly impacts the passenger's perception of the quality of their journey. This quality is determined by the minimum allowable time between baggage check-in and the departure of the flight as well as the time it takes to transport passengers' baggage upon arrival. Both these times are affected by the reliability and the capacity of the BHTS. A combination of fully automated, mechanized systems and human operators controls baggage handling at major airports. Most automated and mechanized systems utilize a combination of continuous and discontinuous conveyor systems.

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung .

The main difference between these two systems is the momentarily transport capacity. Assuming that both systems are up and running, a continuous conveyor system always provides enough capacity, whereas the capacity of a discontinuous system depends on the availability of equipment. An example of equipment used in a continuous conveyor system is a belt conveyor that is a mechanical conveyor using a continuous belt supported by pulleys and, in most cases, idler rolls. Belt conveyors are driven by electrical motors that, in many cases, are speed controlled.

The energy required to power a belt conveyor is determined by the friction in the conveyor. This friction mainly depends on the so-called indentation rolling resistance that depends on the structural composition of the belt, the weight of the belt, and its load [1]. The energy consumption of a BHTS

forms a significant percentage of the total energy consumption of an airport, with the conveying equipment accounting for 55% to 70% of that percentage [2]. The carbon emission is directly proportional to the energy consumption of a conveying system. Therefore, the conveying system of a BHTS emits a significant portion of the carbon emitted by an airport. In literature, energy efficiency and reducing carbon emissions in aviation is a theme that is growing [3]–[6]. To reduce carbon emissions in the aviation industry, the International Civil Aviation Organization initiated the Carbon Offsetting and Reduction Scheme for International Aviation (CORSIA) in 2016, which has taken effect in January 2021 [7]. To comply with the CORSIA, it is vital to minimize the energy consumption of belt conveyors in BHTS since belt conveyors form the backbone of most BHTS at major airports.

Earlier research showed that reducing the delay time t_d of a belt conveyor has a significant impact on the energy consumption of a BHTS [8]. The delay time is the time a belt conveyor runs empty or idle after the last piece of baggage left the conveyor. It was shown that the maximum energy savings depend on the capacity of the conveyor and the conveyors' length [8]. In general, it can be said that possible energy savings for shorter conveyors are larger than that for longer conveyors. Therefore, selecting the optimal length of a belt conveyor allows for lower energy consumption of the conveying system of the BHTS. In [8], a study was carried out identifying the impact of the capacity, length, and delay time of a belt conveyor on its energy consumption. This paper will further explore methods to minimize energy consumption, thus Carbon dioxide (chemical formula CO₂) emissions, by optimizing the airport BHTS from the perspective of the total costs.

It should be realized that, by reducing the length of a belt conveyor in the conveying system of a BHTS, the number of belt conveyors increases since the conveying system still has to span a certain distance. An increase in the number of conveyors will lead to an increase in the Capital Expenditure (CapEx) of the system, even though an overall reduction of the energy consumption of the system will lead to a reduction of the Operational Expenditure (OpEx). The total cost of running a conveyor system over a certain period is made up by the CapEx and the OpEx as a function of time. Therefore, instead of optimizing to a minimization of the energy consumption of the conveyor system, in this paper, an optimization to a minimization of the total cost is studied. Furthermore, to include the impact of CO₂ emissions, the offsetting costs of CO₂ emissions are included in the airport BHTS cost function.

It is worthwhile to review the existing studies on airport baggage handling systems to better build our airport BHTS model and select suitable solving methodologies. [9] considers the planning and scheduling of inbound baggage that is picked up by passengers at the baggage reclaim hall. As the inbound baggage handling problem turns out to be NP-hard, they propose a hybrid heuristic approach combining a greedy randomized adaptive search procedure with a guided fast

local search and path relinking. [10] considers the handling of baggage from passengers changing aircraft at an airport. The transfer baggage problem is to assign the bags from each arriving aircraft to an infeed area, from where a network of conveyor belts will bring them to the corresponding outbound flight. Its main objective is to minimize the number of missed bags. Firstly, a static mixed integer programming model is presented for the transfer baggage problem. Furthermore, to handle the uncertainty related to aircraft arrival time, transportation time from aircraft to baggage handling facility, and capacity use in the baggage handling system, a stochastic model is developed. [11] reviews different baggage models and their usage in the BHTS line shared by multiple airlines. The suggested algorithm reduces the imbalances for the airlines sharing the BHTS collection conveyor, while maintaining overall BHTS performance at an acceptable level. [12] shows how a state-of-the-art model-based design framework has been successfully used for model-based design of supervisory controllers for an actual industrial baggage handling system, and for a real-time emulation model of an actual international airport. The high-level modeling elements of the applied model-based design framework allow the modeler to concentrate on implementing the BHTS design requirements. [13] does a part of large airport's BHTS modeling and simulation which deals with more than 20,000 bags per day using a Colored Petri Net. [14] presents a detailed discrete event model of inbound baggage handling at a large regional Italian airport. [15] details the investigation into the design and control of merging bottlenecks of conveyor based BHTS, encompassing the merging control algorithm and the impact of the merge's physical layout. In [16] an alternative transport and scheduling method, as well as the application of a prototype of a partly automated baggage loading and unloading vehicle have been investigated using simulation. [17] presents a microscopic simulation model for a BHTS that fully integrates all baggage-related subsystems. These include passenger arrival to check-in queues, baggage check-in, security screening, sorting, transport to the aircraft and loading. To sum up, a literature review shows that there are studies into BHTS in general but not with respect to an optimization of the design of a belt conveyor system accounting for its environmental impact and total cost.

Swarm intelligence algorithms are useful in solving optimization problems and have been used in many engineering applications. [18] introduces the origin and background of particle swarm optimization (PSO) and carries out the theoretical analysis of PSO. Then, it analyzes its present situation of research and application in algorithm structure, parameter selection, topology structure, discrete PSO algorithm and parallel PSO algorithm, multi-objective optimization PSO and its engineering applications. [19] presents an evolutionary optimization approach named grey wolf optimization (GWO), which is based on the behavior of grey wolves, for the optimal operating strategy of economic load dispatch. [20] proposes a GWO combined with PSO to tackle the disadvantages of the existing GWO, such as slow

convergence speed and low precision. In [21], a water cycle algorithm (WCA) is inspired from nature and based on the observation of water cycle process and how rivers and streams flow to the sea in the real world. In [22], the multi-objective WCA is presented for solving constrained multi-objective problems. Whale Optimization Algorithm [23] mimics the social behavior of humpback whales. Among the family of swarm intelligence algorithms, PSO is a powerful one and shows its superiority in solving many problems. PSO has experienced a multitude of enhancements since it was proposed. Researchers further proposed some improved PSO versions aiming to different improvement demands. PSO has been widely applied to many engineering problems due to its strong global optimization ability and its easy implementation. Thus, we will utilize PSO to solve the airport BHTS design problem, which will shed a light to study similar problems in the future.

This paper is structured as follows. In Section 2, the characteristic times used in the baggage handling conveyor cost model is defined. Section 3 introduces the cost model used for the optimization process. This cost model is extended with the cost for offsetting the carbon emissions in Section 4. Section 5 defines the optimization objective function, whereas PSO algorithms are introduced in Section 6. Section 7 illustrates the results of the application of PSO in the airport BHTS optimization problem. In section 8, the impact of reducing CO₂ emissions by the optimized solution is discussed. Conclusions are drawn in Section 9.

II. CHARACTERISTIC TIMES

In order to optimize costs, a reference case will be defined and the variation in the total cost with variations in the number of conveyors, conveyor speed and delay time will be studied. Let's assume that the total baggage handling conveyor system in the reference case consists of one belt conveyor with a length L_{ref} . Please note that the system's capacity C is independent of the length and the number of belt conveyors in the system. Let N_f be the number of belt conveyors in the system. If all N_f conveyors have the same length L_f , then the length of each conveyor is:

$$L_f = \frac{L_{ref}}{N_f}, \quad \forall N_f \geq 1, N_f \in I \quad (1)$$

Fig. 1 shows a real belt conveyor system in an airport BHTS utilizing multiple short conveyors.

Fig. 2 shows a schematic diagram of a BHTS with one belt conveyor with length L_{ref} and with N_f ($N_f > 1$) belt conveyors with lengths L_f . Note that the total span of the system remains the same.

Assume that the belt speed, which is assumed to be the same for each conveyor in the system, is v . The time it takes a piece of baggage to travel over the conveyor with length L_f is t_o , which is called the operation time:

$$t_o = \frac{L_f}{v} \quad (2)$$



FIGURE 1. Belt conveyor system in an airport baggage handling transport system with multiple short conveyors.

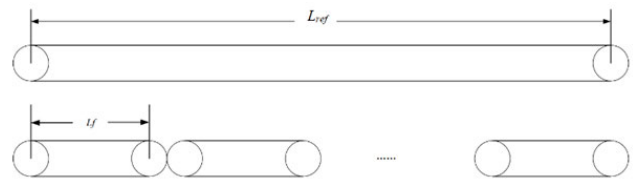


FIGURE 2. Schematic diagram of baggage handling conveyors with one conveyor and N_f conveyors.

Each unique piece of baggage in a BHTS may have a unique destination, also called a lateral. To allow the system to handle each piece of baggage individually, they need to be separated by a minimum gap with a length L_g . The time it takes a piece of baggage to travel over this gap is called the gap time t_g :

$$t_g = \frac{L_g}{v} \quad (3)$$

The time it takes a piece of baggage to travel the conveyor plus the gap is called the processing time t_p :

$$t_p = t_o + t_g \quad (4)$$

Assume that a belt conveyor switches on t_b seconds before a piece of baggage arrives at the conveyor. Time t_b is also called the before time. Furthermore, it is assumed that the belt conveyor switches off t_d seconds after a piece of baggage leaves the conveyor. Time t_d is called the delay time. The total minimum running time t_r of a belt conveyor then is:

$$t_r = t_p + t_b + t_d \quad (5)$$

III. COST MODEL

The cost of a belt conveyor with length L_f (denoted by K_f) consists of CapEx (symbolized by K_{Cf}) and OpEx (symbolized by K_{Of}). It is assumed that, after the economic life T_d of the conveying system, its economic value is zero. The CapEx, therefore, includes the depreciation costs of the conveyor system per year.

During the design of a belt conveyor, a DIN standard is used [24]. In this standard, a distinction is made between the head and tail of the conveyor on the one hand, and the section between the head and the tail on the other. The first section includes the head and tail pulleys, the drive and/or brake,

and the take-up of the system. The second section includes the idlers. Both sections are mounted on a frame as shown in Figure 1. Like the terminology used in the DIN standard, a distinction is made between the CapEx of the head and tail section (symbolized by $K_{Cf,N}$) and the CapEx of the section in between (symbolized by $K_{Cf,H}$):

$$K_{Cf} = \frac{1}{T_d}(K_{Cf,H} + K_{Cf,N}) \quad (6)$$

If it is assumed that an airport standardizes on one type of belting, then $K_{Cf,H}$ is proportional to the length of conveyor:

$$K_{Cf,H} = \alpha_1 L_f \quad (7)$$

where α_1 is the cost for a section between head and tail per meter. As stated earlier, the energy required to power a belt conveyor is determined by the friction in the conveyor. According to [8], the required drive power P_D of a belt conveyor is:

$$P_D = k L_f v \quad (8)$$

where k is the factor that gives the motional resistance per meter belt conveyor on the motor side. Using the data provided in [8], the equation for the regression of factor k versus L_f is:

$$k = \alpha_2 L_f^{\alpha_3} \quad (9)$$

Combining the equations (8) and (9) yields the drive power requirement as a function of the conveyor length and speed:

$$P_D = \alpha_2 L_f^{1+\alpha_3} v \quad (10)$$

Assuming that an airport standardizes the pulleys and frames, the CapEx of the head and tail section depends only on the amount of drive power required, and therefore on the motor to be installed. The CapEx of the head and tail section $K_{Cf,N}$ then can be defined as:

$$K_{Cf,N} = \alpha_4 + \alpha_5 + \alpha_6 P_D + \alpha_7 P_D^2 \quad (11)$$

where α_4 is the cost of the frame, and α_5 to α_7 determine the cost of the drive system. A combination of the equations (6) to (11) results in the total CapEx for a belt conveyor per year:

$$\begin{aligned} K_{Cf} &= \frac{1}{T_d} \left(\alpha_1 L_f + \alpha_4 + \alpha_5 + \alpha_6 \left(\alpha_2 L_f^{1+\alpha_3} v \right) + \alpha_7 \left(\alpha_2 L_f^{1+\alpha_3} v \right)^2 \right) \end{aligned} \quad (12)$$

The CapEx of the reference system with one long belt conveyor with length L_{ref} is K_{Cref} that can be calculated using equation (12) by replacing L_f with L_{ref} . Then, the additional CapEx for a system with N_f smaller conveyors is:

$$\Delta K_C = N_f K_{Cf} - K_{Cref} \quad (13)$$

The OpEx of a conveyor system depends primarily on the energy consumption of the system and its maintenance.

A belt conveyor can be deactivated when there temporarily is no supply of baggage. This time is called the waiting time.

The time between arrival of pieces of baggage on the conveyor is the interarrival time (symbolized by t_i). The waiting time for an individual belt conveyor, i.e., t_{wf} , can be defined by equation (14):

$$t_{wf} = \begin{cases} t_i - t_{rf}, & \forall t_i \geq t_{rf} \\ 0, & \forall t_i < t_{rf} \end{cases} \quad (14)$$

where t_{rf} is running time of one belt conveyor. In [8], it was found that the interarrival time of a piece of baggage can be modeled as a negative exponential Poisson distribution. Thus, the total waiting time of a belt conveyor per hour is as follows [8]:

$$C t_{wf} = 3600 e^{-\frac{C t_{rf}}{3600}} \quad (15)$$

where C is the capacity of the system. Normally, the required capacity of a conveyor is defined per hour and the airport is open X hours a day. This means that a day can be divided into X blocks of one hour. Therefore, the total waiting time per day for a belt conveyor is:

$$T_{wf} = \sum_{i=1}^X 3600 e^{-\frac{C_i t_{rf}}{3600}} \quad (16)$$

where C_i represents the capacity of the BHTS in the i -th hour block. Stopping the belt conveyor when it is not needed creates waiting time, therefore, energy is saved. If the required drive power for an empty conveyor is P_D , then the possible total OpEx savings per belt conveyor per day coming from waiting time is:

$$K_{Of,E} = \alpha_8 P_D T_{wf} \quad (17)$$

where α_8 is the cost of one kilowatt-hour energy (kWh). Therefore, the total OpEx savings for the whole baggage handling conveyor system is:

$$K_{O,E} = \sum_{i=1}^{N_f} K_{Ofi,E} \quad (18)$$

With an increase in belt conveyors in a BHTS and an increase in the number of starts and stops, the maintenance costs rise. It is assumed that the maintenance costs per year are a percentage of the CapEx (i.e., K_{Cf}) of one belt conveyor:

$$K_{Of,M} = \alpha_9 K_{Cf} \quad (19)$$

where α_9 is a percentage. For the whole BHTS, the maintenance costs become:

$$K_{O,M} = \sum_{i=1}^{N_f} K_{Ofi,M} \quad (20)$$

IV. CO₂ EMISSIONS

When changing the length (and thus number) of belt conveyors in the airport BHTS, generally, the capital costs go up, while the operational costs go down due to an increase in waiting time. Hence, this not only results in a reduction of energy costs but also greenhouse gasses (e.g., CO₂) emissions. To study the impact of different BHTS design on the reduction of greenhouse gasses emissions, it can be quantitatively represented by the carbon offsetting costs.

The total drive energy E_D , required to drive a belt conveyor is delivered by an energy source (denoted by E_s). This energy source is electrical energy for an electrical motor. The efficiency of the delivery of the energy source for the total drive energy is the engine efficiency η_E . The relationship between the drive energy and the source energy is as follows:

$$E_D = F_D L_f = \eta_E E_s \quad (21)$$

where F_D is the total drive force. Therefore,

$$E_s = \frac{F_D L_f}{\eta_E} \quad (22)$$

Please note that a typical engine efficiency η_E is 95%. The power generated by the energy source (symbolized by P_s) is the amount of energy used as a function of time.

$$P_s = \frac{E_s}{\Delta t} = \frac{F_D L}{\eta_E \Delta t} = \frac{F_D v}{\eta_E} \quad (23)$$

The total amount of source power saved during the waiting time is $P_s T_{wf}$. The mass of the emission output of greenhouse gasses is linearly related to the amount of energy consumption E_s . The relation between the mass of emission output and the energy consumption is given by the specific emission factor (abbreviated as *s.e.f.*):

$$\begin{aligned} m_{substance} &= s.e.f. \cdot substance P_s T_{wf} \\ &= s.e.f. \cdot substance ((F_D v) / \eta_E) T_{wf} \end{aligned} \quad (24)$$

where:

$m_{substance}$	emitted mass of a substance [g]
$s.e.f. \cdot substance$	specific emissions factor of a certain substance [g/kWh]

For example, the emission output of carbon dioxide (CO₂) calculated with equation (24) is:

$$m_{CO_2} = s.e.f. \cdot CO_2 ((F_D v) / \eta_E) T_{wf} \quad (25)$$

The cost of offsetting this emission K_{CO_f} for a specific belt conveyor is:

$$K_{CO_f} = \alpha_{10} m_{CO_2} \quad (26)$$

where α_{10} is the cost of the emission of CO₂ in \$ per ton.

V. OBJECTIVE FUNCTION

The objective of the airport BHTS optimization problem is to maximize the difference in capital, operational, maintenance, and carbon emission costs between the reference case, where one belt conveyor with length L_{ref} is used, and the case where N_f belt conveyors are used with length L_f . These differences are as follows:

$$\Delta K_C = N_f K_{Cf} - K_{Cref} \quad (27a)$$

$$\Delta K_{O,M} = N_f K_{Of,M} - K_{Oref,M} \quad (27b)$$

$$\Delta K_{O,E} = N_f K_{Of,E} - K_{Oref,E} \quad (27c)$$

$$\Delta K_{CO} = N_f K_{CO_f} - K_{COref} \quad (27d)$$

Equation (27a) defines the difference in CapEx, equation (27b) the difference in OpEx in terms of maintenance costs, equation (27c) the difference in OpEx in terms of energy costs, and equation (27d) the difference in carbon emission costs. With the components of equation (27), the objective function Z can be defined as:

$$Z = (\Delta K_{O,E} + \Delta K_{CO}) - (\Delta K_C + \Delta K_{O,M}) \quad (28)$$

Remember that the energy and emission costs decrease with an increase in the number of conveyors, whereas the CapEx and maintenance costs increase. Therefore, Z should be maximized. To better utilize a heuristic algorithm to solve the airport BHTS optimization problem, this problem will be transferred from a maximization problem to be a minimization problem. Thus, the objective is to minimize $G = -Z$. The decision variables are N_f , v and t_d . The total objective function can be written as:

$$\begin{aligned} G &= (1 + \alpha_9) N_f \frac{1}{T_d} \left(\alpha_1 \frac{L_{ref}}{N_f} + \alpha_4 + \alpha_5 + \alpha_6 \left(\alpha_2 \left(\frac{L_{ref}}{N_f} \right)^{1+\alpha_3} v \right) \right. \\ &\quad \left. + \alpha_7 \left(\alpha_2 \left(\frac{L_{ref}}{N_f} \right)^{1+\alpha_3} v \right)^2 \right) \\ &\quad - \frac{(1 + \alpha_9)}{T_d} \left(\alpha_1 L_{ref} + \alpha_4 + \alpha_5 + \alpha_6 \left(\alpha_2 L_{ref}^{1+\alpha_3} v \right) \right. \\ &\quad \left. + \alpha_7 \left(\alpha_2 L_{ref}^{1+\alpha_3} v \right)^2 \right) - \left(N_f * \alpha_8 \alpha_2 \left(\frac{L_{ref}}{N_f} \right)^{1+\alpha_3} v \right. \\ &\quad * 365 * \sum_{i=1}^X e^{-\frac{c \left(\frac{L_{ref}}{N_f v} + \frac{l_g}{v} + t_b + t_d \right)}{3600}} - \alpha_8 \alpha_2 L_{ref}^{1+\alpha_3} v * 365 \\ &\quad * \sum_{i=1}^X e^{-\frac{c \left(\frac{L_{ref}}{v} + \frac{l_g}{v} + t_b + t_d \right)}{3600}} \left. \right) - \left(N_f * s.e.f. \cdot CO_2 \right. \\ &\quad * \alpha_2 \left(\frac{L_{ref}}{N_f} \right)^{1+\alpha_3} v * 365 * \sum_{i=1}^X e^{-\frac{c \left(\frac{L_{ref}}{N_f v} + \frac{l_g}{v} + t_b + t_d \right)}{3600}} / \eta_E \\ &\quad \left. - s.e.f. \cdot CO_2 * \alpha_2 L_{ref}^{1+\alpha_3} v * 365 * \sum_{i=1}^X e^{-\frac{c \left(\frac{L_{ref}}{v} + \frac{l_g}{v} + t_b + t_d \right)}{3600}} / \eta_E \right) \\ &\quad * \alpha_{10} * 10^{-6} \end{aligned} \quad (29)$$

VI. PARTICLE SWARM OPTIMIZATION

A. STANDARD PSO ALGORITHM

Particle Swarm Optimization (PSO) is a popular heuristic algorithm and is widely used to solve engineering problems due to its good performance, easy implementation, and adaptability to different problems. Thus, in this study, PSO is selected to solve our airport BHTS optimization problem.

Originally proposed by James Kennedy and Russell Eberhart, PSO was inspired by the foraging behavior of bird flocks [25]. A swarm is stochastically initialized in the search space. Each particle is a candidate solution of the problem, which is represented by the velocity v and the location x .

A particle flies to a better position as it evolves. Particles have memory and share their information with each other, which imitates social behavior. In the original PSO, velocities and locations of particles are iteratively updated as follows:

$$v_{id}^{t+1} = v_{id}^t + c_1 r_1 (pbest_{id}^t - x_{id}^t) + c_2 r_2 (gbest_d^t - x_{id}^t) \quad (30)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (31)$$

where t denotes the current number of iterations. i represents the particle index, and d represents the dimension index. r_1 and r_2 are random numbers uniformly distributed in the range of $[0,1]$. **pbest** denotes the best previous position of a particle. **gbest** denotes the best position of the swarm so far. c_1 is a learning factor that refers to the impact of a particle's history information, and c_2 is a learning factor reflecting the social impact from the swarm.

In this paper, the original PSO is called the standard PSO. Three parts, i.e., personal current velocity, pbest, and gbest affect the velocity update. The standard PSO suffers from premature convergence and is easy to trap into a local optimum caused by the loss of population diversity. To tackle the premature convergence issue, a PSO with inertia weight (ω PSO) was proposed to balance the global search ability and local search ability of PSO. Its velocity is updated using Eq. (32) [26].

$$v_{id}^{t+1} = \omega_i v_{id}^t + c_1 r_1 (pbest_{id}^t - x_{id}^t) + c_2 r_2 (gbest_d^t - x_{id}^t) \quad (32)$$

$$\omega_i = \omega_{ini} - (\omega_{ini} - \omega_{end}) \cdot t / M_{Iter} \quad (33)$$

The inertia weight (denoted by ω_i) linearly decreases over time using Eq. (33), which is good for balancing the exploration and exploitation ability. ω_{ini} is the initial inertia weight, and ω_{end} is the end inertia weight. M_{Iter} represents the total number of iterations. A large value of ω_i at the early stage provides a strong global search ability by exploring a wider space. As ω PSO evolves to the end, ω_i tends to be a small value and the algorithm focuses on local exploitation around *gbest* and *pbest*.

B. COMPREHENSIVE LEARNING PSO

Comprehensive Learning PSO (CLPSO) algorithm [27] was proposed to deal with the premature convergence problem, which is a very successful improvement. Its velocity updates by Eq. (34) where $fi(d)$ represents the index of the particle that the d -th dimension of the i -th particle select to learn. Everyone chooses a **pbest** from the whole swarm, and different dimensions of a particle can select different learning objects. This learning strategy effectively maintains the population diversity [27], [28] so that the global search ability of CLPSO is greatly improved compared to the standard PSO.

$$v_{id}^{t+1} = \omega_i v_{id}^t + c_1 r_i^d (pbest_{fi(d)d}^t - x_{id}^t) \quad (34)$$

C. UNIFIED PSO

The Unified Particle Swarm Optimization (UPSO) scheme was proposed to combine the exploration and exploitation

properties of both the local and global PSO versions [29]. In the global version of PSO, the neighborhood of each particle is the whole swarm. In the local version, the neighborhoods are smaller, and they usually consist of several particles.

Let G_i^{t+1} and L_i^{t+1} denote the velocity update of the i -th particle x_i for the global and local PSO version, respectively.

$$G_i^{t+1} = \Phi [v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (gbest^t - x_i^t)] \quad (35)$$

$$L_i^{t+1} = \Phi [v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (nbest_i^t - x_i^t)] \quad (36)$$

where $i = 1, \dots, N$, Φ is the constriction factor. For the t -th generation, $pbest_i^t$ is the best history position of particle i , $gbest^t$ is the best position of the whole swarm so far (global version), and $nbest_i^t$ is the best particle in the neighborhood of x_i (local version). The search directions defined by Eqs. (35) and (36) are aggregated in a single equation, resulting in the main UPSO scheme in Eqs. (37) and (38).

$$U_i^{t+1} = (1 - u) L_i^{t+1} + u G_i^{t+1}, \quad u \in [0,1] \quad (37)$$

$$x_i^{t+1} = x_i^t + U_i^{t+1} \quad (38)$$

where u is named as unification factor to balance the influence of the global and local search directions.

D. SELF-REGULATION PSO

A self-regulated theme provides better exploration and exploitation, which motivates the development of self-regulating and self-perception strategies in the standard PSO algorithm. It is called Self-Regulation PSO (SRPSO) [30].

In SRPSO, self-regulating means that the best particle will be given a higher acceleration of its exploration process through an increased inertia weight. It performs the search without any information from other particles and will return to the normal search strategy once it is not the global best position.

Self-perception is another important strategy in SRPSO. The best particle uses his direction as the best direction and is not influenced by its experience or other experiences. The particles who are not the best use their perception based on the global best position to find the appropriate direction in the current iteration. SRPSO updates velocity via Eq. (39).

$$v_i^{t+1} = \omega_i v_i^t + c_1 r_1 p_{id}^{se} (pbest_i^t - x_i^t) + c_2 r_2 p_{id}^{so} (gbest^t - x_i^t) \quad (39)$$

where ω_i is the inertia weight of i -th particle. p_{id}^{se} is the perception for the *self-cognition* and p_{id}^{so} is the perception for the *social-cognition*. p_{id}^{se} and p_{id}^{so} are defined as follows:

$$p_{id}^{se} = \begin{cases} 0, & \text{for the best particle} \\ 1, & \text{otherwise} \end{cases} \quad (40)$$

and

$$p_{id}^{so} = \begin{cases} 0, & \text{for the best particle} \\ \gamma, & \text{otherwise} \end{cases} \quad (41)$$

where γ is binary (i.e., 0 or 1) depending on the threshold value for defining the confidence.

PSO is a popular heuristic algorithm in solving engineering problems and further modifications have been proposed to improve the performance of PSO. Among the improved PSO variants, SRPSO, UPSO and CLPSO are powerful variants that have demonstrated good performance in many benchmark functions and applications. Thus, the three mentioned PSO algorithms are selected to solve our airport BHTS model. Since each algorithm may exhibit different performance, it is worthwhile to leverage several algorithms to check their capabilities in solving the airport BHTS model.

VII. APPLICATION OF PSO IN BHTS OPTIMIZATION

A. PARAMETER SETTINGS

The parameters and their values shown in Table 1 are the inputs for the airport BHTS optimization problem. These values are based on the practical experience of the authors and data from component suppliers.

TABLE 1. Key parameter settings of the airport BHTS model.

Name	Value	Name	Value
α_1	1000	α_8	0.16
α_2	1.2373	α_9	0.1
α_3	-0.241	α_{10}	48
α_4	1500	T_d	5
α_5	136.78	$s.e.f \cdot CO_2$	540
α_6	-5.49	η_E	0.95
α_7	13.27		

CLPSO, UPSO and SRPSO are three state-of-the-art PSO variants [27]–[30]. To learn their performance on the airport BHTS optimization problem, all three PSO algorithms are tested to obtain comparison results. Table 2 presents the main parameter settings for the numerical experiments, including the number of test times, population size, and the maximum number of iterations. Considering the random factors in heuristic algorithms, all tests run 30 independent trials to obtain their statistical results.

B. SYSTEM CAPACITY AND L_{ref}

It is assumed that the required capacity of the airport BHTS varies over the day but is a constant per hour [8]. Table 3 shows the capacity C_i of a total of X=15 hours. The first row represents the start to the end time of the indicated hour. The second row shows the number of bags processed in that hour.

TABLE 2. Experiment settings.

Name	Symbol	Value
Number of test times	$Test_Times$	30
Population size	Pop_Size	20
Maximum number of iterations	Max_Gen	70

TABLE 3. Capacity at each hour.

Time Slot	5am-6am	6am-7am	7am-8am	8am-9am	9am-10am
C	115	70	10	30	30
Time Slot	10am-11am	11am-12pm	12pm-1pm	1pm-2pm	2pm-3pm
C	10	40	90	30	75
Time Slot	3pm-4pm	4pm-5pm	5pm-6pm	6pm-7pm	7pm-8pm
C	105	75	45	40	10

The impact of the variation of capacity is illustrated by comparing the results of the optimization with those of a system with a constant capacity of 110 bags per hour. This comparison is shown in Table 4 for different lengths of the system. In this comparison, the length L_{ref} starts from 15 meters and ends at 75 meters with an increment of 15 meters.

The settings of the range of the variable v is [0.5, 1] and the range of t_d is [5], [40] for all tests in this subsection. Through using the three PSO algorithms, the optimized solution of v is its minimum value 0.5 m/s, and the optimized solution of t_d is its minimum value 5 second for all independent runs of each case. However, the optimized solution of N_f is different for different cases. Thus, we focus on the discussion of the optimized solution of N_f .

For the results of the analyses in Table 4, the SRPSO algorithm is used. The columns in Table 4 labelled ‘Objective value’ represents the value of the object function.

As can be seen from the results of the constant capacity $C = 110$ case (right column), the optimized solution of N_f shows a constant ratio, between the length L_{ref} and the optimized solution N_f , i.e., 1.36. Hence, for a BHTS with the parameters given in Table 1, the optimum conveyor length is 1.36 m when C is a constant capacity of 110 bags per hour. However, this is not true for the case with transport capacity varying over the day, as shown in Table 3. For a system with a length of 15 m, the optimum conveyor length is 1.25 m. If the system length increases to 75 m, the optimum conveyor length increases to 1.29 m. Therefore, it demonstrates that in reality, where system capacity does change over the day, an optimization exercise is required to find the optimum conveyor length rather than using a standard length like the 1.36 m found in this study.

It is necessary to check the performance of different PSO algorithms for the airport BHTS optimization problem. All the five inputs of L_{ref} have been tested but, since their results turn out to be the same, only the results of a 15 m system length are discussed in detail.

TABLE 4. Comparison of solutions between cases with C_i and constant C per hour.

L_{ref}	C_i		$C=110$	
	N_f	Objective value	N_f	Objective value
15	12	-5003.9	11	-5353.1
30	23	-14246.2	22	-15362.0
45	35	-25098.0	33	-26656.4
60	47	-36842.5	44	-38319.0
75	58	-49137.7	55	-50016.8

Statistical results of the value of the objective function, including the minimum value, maximum value, median value, mean value, and standard deviation value obtained by CLPSO, SRPSO and UPSO algorithms are shown in Table 5. This table demonstrates that all three PSO variants can find the best solution by looking into their minimum values. Furthermore, the results show that SRPSO and UPSO achieve the global optimum solution every trial since all 5 metrics are the same and the standard deviation value is very small. The maximum value of CLPSO is larger than that of the other two PSO algorithms, which means that some trials tested by CLPSO do not achieve the global optimum solution.

TABLE 5. Statistical results of objective value for $L_{ref} = 15$ with $Max_Gen = 70$.

Algorithm	Min	Max	Median	Mean	Std.
CLPSO	-5003.948	-4999.804	-5003.948	-5003.672	1.05E+00
SRPSO	-5003.948	-5003.948	-5003.948	-5003.948	9.25E-13
UPSO	-5003.948	-5003.948	-5003.948	-5003.948	9.25E-13

To further investigate the efficiency of different PSO algorithms, their CPU time is recorded, and the statistical results are shown in Table 6. This table shows that SRPSO is the fastest one from all 5 metrics.

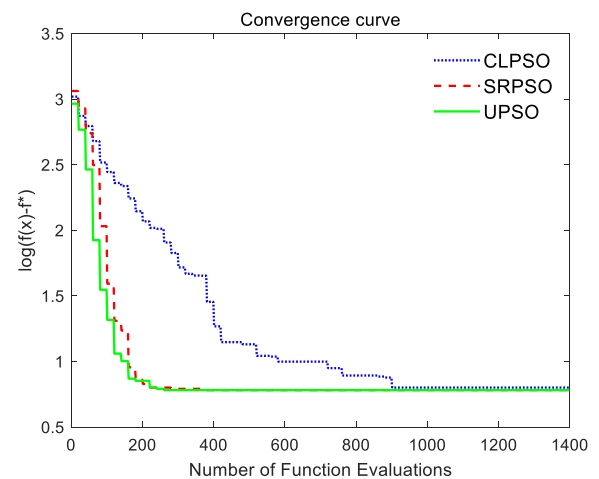
TABLE 6. Statistical results of CPU time for $L_{ref} = 15$ with $Max_Gen = 70$.

Algorithm	Min	Max	Median	Mean	Std.
CLPSO	0.0131	0.0269	0.0149	0.0160	0.0029
SRPSO	0.0054	0.0101	0.0072	0.0073	0.0013
UPSO	0.0313	0.0368	0.0332	0.0334	0.0016

With $Max_Gen = 70$, the convergence curves of CLPSO, SRPSO and UPSO are presented in Fig. 3. The meaning of y-axis is the log operation result of the difference between $f(x)$ and f^* . Conducting the log operation is to easily see the obvious converge differences among various algorithms. If the difference is too small, it is not easy to figure out when

the log value is not used. Using $(f(x) - f^*)$ as the input of log other than using the objective function $f(x)$ is to check the gap between the current objective value and the optimal one. Thus, the evolve process is directly presented. To do this, we have solved the problem first and obtained a best solution as f^* . Then the output of each iteration is subtracted with f^* . The x-axis is the number of function evaluations. For each generation, its number of function evaluations is Pop_Size . Thus, after t -th generation, the total number of function evaluations consumed so far equals $t * Pop_Size$.

From Fig. 3, it is evident that SRPSO and UPSO perform similarly in terms of convergence rate. CLPSO converges slower at the early stage, which is because of its strong global search ability and its main mechanism of maintaining diversity at its early stage. When the number of function evaluations is around 900, all three algorithms find the best solution for this airport BHTS optimization case.

**FIGURE 3.** Convergence curves for $L_{ref} = 15$ with $Max_Gen = 70$.

C. PERFORMANCE OF PSO WITH LIMITED Max_Gen

When the maximum number of iterations is limited, for example, it is reduced to 10, some algorithms cannot obtain the best solution. So, it is worthwhile to analyse the performance of different PSO algorithms in the airport BHTS optimization problem with limited Max_Gen . Table 7 shows the statistical results of the value of the objective function obtained by the three PSO variants when $L_{ref} = 15$ with $Max_Gen = 10$. UPSO performs better than CLPSO and SRPSO from the perspective of the mean value and standard deviation value.

TABLE 7. Statistical results of objective value for $L_{ref} = 15$ with $Max_Gen = 10$.

Algorithm	Min	Max	Median	Mean	Std.
CLPSO	-5003.948	-4082.921	-4967.460	-4844.823	2.52E+02
SRPSO	-5003.948	-4793.930	-4999.804	-4972.302	5.58E+01
UPSO	-5003.948	-4967.460	-5003.948	-5002.456	6.69E+00

By looking into the CPU time, as shown in Table 8, SRPSO is the fastest one, which is consistent with that of the situation when *Max_Gen* is 70.

TABLE 8. Statistical results of CPU time for *L_{ref}* = 15 with *Max_Gen* = 10.

Algorithm	Min	Max	Median	Mean	Std.
CLPSO	0.0017	0.0042	0.0019	0.0022	0.0007
SRPSO	0.0007	0.0024	0.0009	0.0010	0.0003
UPSO	0.0046	0.0080	0.0057	0.0059	0.0011

With *Max_Gen* = 10, the convergence curves of CLPSO, SRPSO and UPSO are presented in Fig. 4. As can be seen in this figure, UPSO obtains the best solution among the three PSO variants. CLPSO evolves with the slowest speed, which is due to its updating strategy emphasizing on maintaining swarm diversity and focusing on the global search at the early stage. So, when the given number of iterations is limited, CLPSO is still conducting a global search of its early stage and has not start the later converge stage. Thus, CLPSO does not converge to the best solution with limited iterations.

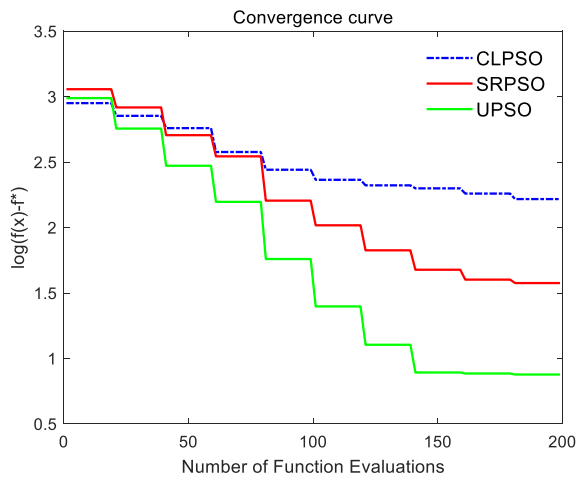


FIGURE 4. Convergence curve for *L_{ref}* = 15 with *Max_Gen* = 10.

D. OPTIMIZED SOLUTIONS WITH DIFFERENT *t_d* AND *C*

The delay time *t_d* affects the running time *t_r*, as shown in Eq. (5). Thus, the waiting time is also affected. Therefore, different values of *t_d* will have a different output in terms of the optimized solution of *N_f* and the parameter values. As illustrated in Section VII.B, the capacity *C* also affects the results of the optimization. It is important to understand how *t_d* and *C* together affect the optimized results to better understand the optimization of the airport BHTS. To analyze this, the following values of *C* are used [90, 100, 110, 120, 130, 140, 150, 200, 250, 300]. The optimized solution of the delay time is its minimum value of the range. Thus, different inputs of the lower bound are set to learn the impact of delay

time on the airport BHTS model. The lower bound of delay time *t_d* is set to be 5, 10 and 15 seconds, respectively. All upper bound of delay time is 40. In this subsection, SRPSO is used because it is the fastest one among the three PSO algorithms tested for this application.

In this subsection, the focus is on determination the changes to the value of the objective function by changing the lower bound thus the optimized solution of *t_d* and *C*. To intuitively learn the changes, the values of the objective function are normalized by dividing these values with the maximum value among all objective values of a set. Please note that a set includes all results from an optimization using all values of *C* at a specific lower bound value of *t_d*.

As seen in Fig. 5, the normalized values of the objective function decrease with an increase in the capacity. Furthermore, as *t_d* increases, the slope gets greater, which means that a large value of *t_d* has more impact on the value of the objective function. This is caused by the fact that, with a larger *t_d*, the difference between the performance of the reference system and the optimized system is smaller due to a larger *t_d* resulting in less waiting time decreasing the value of the objective function.

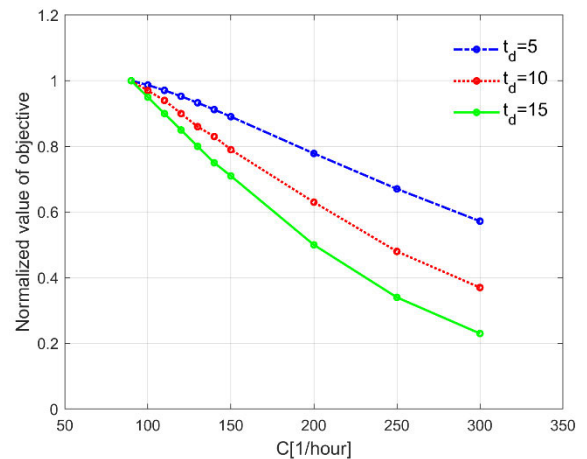


FIGURE 5. Normalized objective values when *t_d* and *C* change.

It is interesting to determine the impact on the value of the objective function when capacity and delay time change with a constant *N_f* with an input of *t_d*. To check different scenarios of delay time, three situations of *t_d*, i.e., *t_d* = 5, 10 and 15, are tested. For each scenario, *N_f* is set to be the same value for different capacity *C*. The optimized solution from the minimum input of *C* (see Table 3) is picked for each *t_d*. Thus, when *C* is 90, *N_f* is 56, 49 and 42 for corresponding scenarios when the optimized solution of *t_d* is 5, 10 and 15, respectively. Since the optimized solution of *t_d* is its minimum value within its defined range, the scenario of a 5 second solution is with a setting of the range of [5, 40], the 10 second scenario is from the range of [10, 40] and the 15 case is from the range of [15, 40]. For other capacity values, *N_f* is kept the same instead of using its optimized solution for that case to create Figure 6.

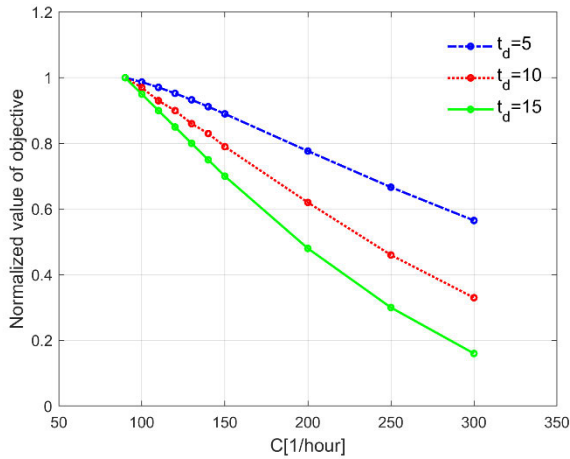


FIGURE 6. Normalized objective value with a constant $N_f t$ when t_d and C change.

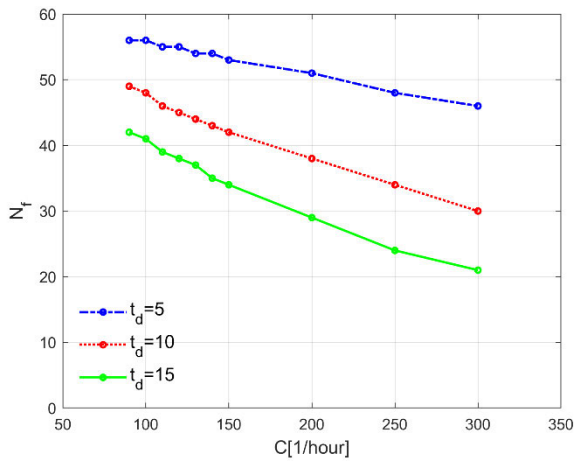


FIGURE 7. Optimized solutions of N_f when t_d and C change.

Figure 6 illustrates that the trend is similar to the results shown in Fig. 5. However, the slope of the lines is steeper than those in Fig. 5. This is caused by the fact that at different C values, the optimized solutions in terms of the number of conveyors, will be lower than the used number, e.g., 42 for a $t_d = 15$ case. A lower value of N_f means a longer length of the conveyor, which will further decrease the value of the objective function.

The relationships of the optimized solution in terms of the number of conveyors N_f with varying t_d and C are shown in Fig. 7. The number of conveyors in the optimized solution decreases when C increases. This is caused by the fact that a higher capacity leads to less opportunity for waiting time and thus energy savings. Therefore, the impact of the increase in CapEx with an increase in the number of conveyors plays a dominant role. As t_d increases, the slope gets greater for the same reason as explained earlier, but not as significant as Figs. 5 and 6.

VIII. CO₂ OFFSET COST REDUCTION

In the introduction section, it is stated that reducing the delay time t_d of a belt conveyor has a significant impact

TABLE 9. Carbon emission reduction analysis.

L_{ref}	Reference System		Optimized System		Difference of reduced carbon emission cost
	N_f	Reduced Carbon emission cost	Optimized N_f	Reduced Carbon emission cost	
15	1	\$ 1,081	12	\$ 2,357	\$ 1,276
30	1	\$ 1,527	23	\$ 4,662	\$ 3,135
45	1	\$ 1,761	35	\$ 7,019	\$ 5,258
60	1	\$ 1,884	47	\$ 9,376	\$ 7,492
75	1	\$ 1,943	58	\$ 11,681	\$ 9,738

on the energy consumption of an airport BHTS. In practice, belt conveyors in BHTS at some major airports run continuously since the systems lack the sensors to monitor baggage flow. Thus, introducing a delay time will automatically impact energy consumption and, consequently, the emission of greenhouse gasses like CO₂. Assume an airport is required to financially offset the carbon dioxide emission of its operation. Hence, a reduction in the emission of CO₂ immediately leads to a cost reduction.

Table 9 shows the possible CO₂ offset cost reduction per year for an airport BHTS with different lengths after introducing a delay time of 5 seconds instead of letting the conveyors run continuously. Remember that the delay time only affects the time a belt conveyor runs empty. The parameters that play a role are the system length and the number of conveyors. The first column of Table 9 shows the reference length of the conveyor system. In the reference case only one conveyor is used, hence, the N_f of 1 in the second column. Column 3 shows the cost reduction per year when a delay time of 5 seconds is used. As can be seen, the cost reduction varies from \$1,081 for a system with a length of 15 m to \$1,943 for a system with a length of 75 m. Fig. 8 shows that the CO₂ offset cost reduction decreases from \$72 per meter to \$26 per meter of a conveyor. This is caused by the fact that, with an increase in conveyor length and a given interarrival time of pieces of baggage, the conveyor's running time increases and the chances that the interarrival time exceeds the running time decrease.

The last 4th and 5th columns in Table 9 show the possible CO₂ offset cost reduction per year for an optimized BHTS. The optimum number of conveyors, determined using the SRPSO algorithm, is given in column 4. Column 5 shows the cost reduction per year when a delay time of 5 seconds is used. As shown in the reference case, this cost reduction is relative to a system that is up and running all the time. The cost reduction ranges from \$2,357 for a system with a length of 15 m to \$11,681 for a system with a length of 75 m. Column 6 finally shows the difference in cost reduction between the reference case and the optimized case. Fig. 8 shows that the CO₂ offset cost reduction in the optimum case is almost a constant at \$156 per meter. This was to be expected as SRPSO optimizes the object function to maximizing the impact of the delay time, and irrespective of the system's length.

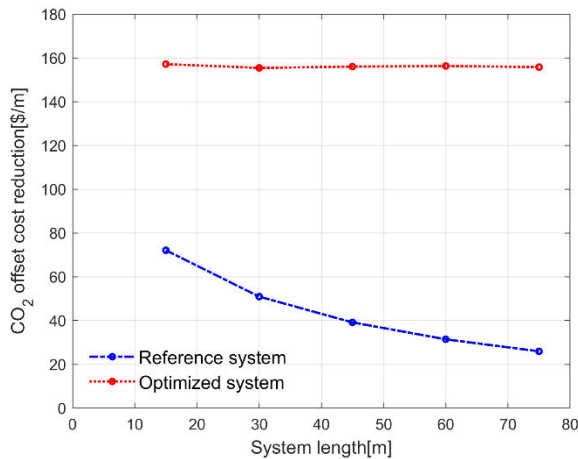


FIGURE 8. CO₂ offset cost reduction per meter for the reference system and optimized system.

A possible cost reduction of \$11,681 per year for a conveyor system with 75 m long may seem trivial. However, consider that large airports have BHTSs with lengths covering kilometres instead of meters. In that case, the yearly reduction of CO₂ offset costs becomes \$156,000 per kilometre of system. At \$48 per tonne, the reduction of CO₂ emission is 3,250 tonne per year per kilometre of system, which is significant.

IX. CONCLUSION

This paper investigates optimizing the design of the airport baggage handling transport system by minimizing the total costs, including CapEx, OpEx and the costs for offsetting CO₂ emissions. Based on this study, the following conclusions can be drawn:

1) In the international aviation industry, a minimization or elimination of CO₂ emissions is crucial, as highlighted by the CORSIA scheme. Unfortunately, there are hardly any scientific papers that can assist in the environmentally friendly design of airport systems, i.e., baggage handling transport systems.

2) It is possible to develop a detailed total cost model based on the characteristics and specifics of belt conveyors used in an airport BHTS. This cost model needs to include the CapEx and OpEx of the transport system accounting for the costs for offsetting CO₂ emissions.

3) Converting the cost model into an objective function aids the optimization process. The primary parameters in the optimization process are the total length of the belt conveyor system, interarrival time of baggage, capacity of BHTS, drive power of a belt conveyor, etc. The decision variables are the number of conveyors, the delay time, and the belt conveyor speed.

4) The particle swarm optimization algorithm can be used for the airport BHTS optimization problem. All three PSO variants tested in this study can determine the optimum design. From a computational efficiency point of view, SRPSO is the best one and therefore utilized.

5) For different lengths of the system, where system capacity does change over the day, an optimization exercise is required to find the optimum conveyor length rather than using a standard length when the capacity is a constant.

6) Optimization of the design of an airport baggage handling transport system, especially in terms of the number of conveyors, leads to a significant CO₂ emission reduction and consequently an offset cost reduction.

APPENDIX A

LIST OF ABBREVIATIONS

BHTS	Baggage Handling Transport System
CapEx	Capital Expenditure
OpEx	Operational Expenditure
PSO	Particle Swarm Optimization
UPSO	Unified PSO
CLPSO	Comprehensive Learning PSO
SRPSO	Self-Regulation PSO

APPENDIX B

LIST OF SYMBOLS

C	capacity of the BHTS	[bax/hour]
C_i	capacity of the BHTS in the i -th hour	[bax]
E_D	drive energy	[J]
E_s	source energy	[J]
I	integer	[-]
G	total objective function	
ΔK_C	capital cost difference per year between both systems	[\$/year]
$\Delta K_{O,M}$	maintenance cost difference per year between both systems	[\$/year]
$\Delta K_{O,E}$	operational cost difference per year between both empty systems	[\$/year]
ΔK_{CO}	emission offset cost difference per year between both systems	[\$/year]
K_{Cf}	capital cost per year of one belt conveyor in the BHTS	[\$/year]
K_{Cref}	capital cost per year of the reference system	[\$/year]
$K_{Cf,H}$	capital cost of the main section of one belt conveyor	[\$]
$K_{Cf,N}$	capital cost of the head and tail section of one belt conveyor	[\$]
$K_{CO,f}$	cost of offsetting CO ₂ emissions for one belt conveyor	[\$/year]
K_{COref}	cost of offsetting CO ₂ emissions for the reference conveyor	[\$/year]
K_f	total cost of one belt conveyor in the BHTS	[\$]
K_{of}	operational cost per year of one belt conveyor in the BHTS	[\$/year]
$K_{of,E}$	operational cost per year of an empty belt conveyor in the BHTS	[\$/year]
$K_{O,E}$	operational cost per year of an empty conveyor system in the BHTS	[\$/year]
$K_{Oref,E}$	operational cost per year of an empty reference conveyor in the BHTS	[\$/year]
$K_{O,M}$	maintenance cost per year of the conveyor system in the BHTS	[\$/year]
$K_{Oref,M}$	maintenance cost of the reference system	[\$/year]
$K_{of,M}$	maintenance cost per year of a belt conveyor in the BHTS	[\$/year]
L_f	length of each belt conveyor in the BHTS	[m]
L_g	gap length	[m]
L_{ref}	length of reference belt conveyor	[m]
N_f	number of belt conveyors in the BHTS	[-]
P_d	drive power of one belt conveyor in the BHTS	[kW]
P_s	source power	[kW]
T_d	economic life of the transport system in the BHTS	[years]
T_{wf}	total waiting time of one conveyor in the BHTS	[s]
X	openings time of the airport	[hours]
Z	objective function	[-]
bax	number of pieces of baggage	[-]
k	motional resistance per meter of belt conveyor	[N/m]
$m_{substance}$	emitted mass of a substance	[g]
$s.e.f_{substance}$	specific emission factor of a substance	[g/kWh]
t_b	before time	[s]
t_d	delay time	[s]
t_g	gap time	[s]
t_i	interarrival time	[s]
t_o	operation time	[s]
t_p	processing time	[s]
t_r	running time of the reference system	[s]
t_{rf}	running time of one belt conveyor	[s]
t_w	waiting time of the reference system	[s]
t_{wf}	waiting time of one belt conveyor	[s]
v	belt speed	[m/s]

REFERENCES

- [1] G. Lodewijks, "The rolling resistance of conveyor belts," *Bulk Solids Handling*, vol. 15, no. 1, pp. 15–22, 1995.
- [2] E. van Enter, "Energy consumption of baggage handling systems," M.S. thesis, Dept. Maritime Transp. Technol., TU Delft, Delft, The Netherlands, 2018.
- [3] X. Olsthoorn, "Carbon dioxide emissions from international aviation: 1950–2050," *J. Air Transp. Manage.*, vol. 7, no. 2, pp. 87–93, Mar. 2001.
- [4] I. J. Smith and C. J. Rodger, "Carbon emission offsets for aviation-generated emissions due to international travel to and from New Zealand," *Energy Policy*, vol. 37, no. 9, pp. 3438–3447, Sep. 2009.
- [5] A. T. H. Chin and P. Zhang, "Carbon emission allocation methods for the aviation sector," *J. Air Transp. Manage.*, vol. 28, pp. 70–76, May 2013.
- [6] M. N. Büyükbay, G. Özdemir, and E. Üstündağ, "Sustainable aviation applications in turkey: Energy efficiency at airport terminals," in *Proc. Sustain. Aviation*, 2016, pp. 53–60.
- [7] H. Chao, D. B. Agusdinata, D. DeLaurentis, and E. B. Stechel, "Carbon offsetting and reduction scheme with sustainable aviation fuel options: Fleet-level carbon emissions impacts for US airlines," *Transp. Res. D, Transp. Environ.*, vol. 75, pp. 42–56, Oct. 2019.
- [8] G. Lodewijks, "Energy efficient use of belt conveyors in baggage handling systems," in *Proc. 9th IEEE Int. Conf. Netw., Sens. Control*, Beijing, China, Apr. 2012, pp. 97–102.
- [9] M. Frey, F. Kiermaier, and R. Kolisch, "Optimizing inbound baggage handling at airports," *Transp. Sci.*, vol. 51, no. 4, pp. 1210–1225, Nov. 2017.
- [10] T. C. Barth, J. T. Holm, J. L. Larsen, and D. Pisinger, "Optimization of transfer baggage handling in a major transit airport," *Social Netw. Oper. Res. Forum*, vol. 2, no. 2, pp. 1–35, Jun. 2021.
- [11] R. J. Ekeocha and S. A. Ushe, "Application of queuing process in the optimization of baggage handling system of murtala muhammed international airport," *Int. J. Sci.*, vol. 7, no. 5, pp. 97–103, 2018.
- [12] L. Swartjes, D. A. van Beek, W. J. Fokkink, and J. A. W. M. van Eekelen, "Model-based design of supervisory controllers for baggage handling systems," *Simul. Model. Pract. Theory*, vol. 78, pp. 28–50, Nov. 2017.
- [13] D. Hafilah, A. Radja, N. Rakoto-Ravalontsalama, and Y. Lafdail, "Modeling and Simulation of baggage handling system in a large airport," in *Proc. 18th Asia Pacific Ind. Eng. Manage. Syst. Conf.*, 2017, pp. 1–7.
- [14] C. Malandri, M. Briccoli, L. Mantecchini, and F. Paganelli, "A discrete event simulation model for inbound baggage handling," *Transport. Res. Procedia*, vol. 35, pp. 295–304, Jan. 2018.
- [15] M. Johnstone, D. Creighton, and S. Nahavandi, "Simulation-based baggage handling system merge analysis," *Simul. Model. Pract. Theory*, vol. 53, pp. 45–59, Apr. 2015.
- [16] J. C. Rijsenbrij and J. A. Ottjes, "New developments in airport baggage handling systems," *Transp. Planning Technol.*, vol. 30, no. 4, pp. 417–430, Aug. 2007.
- [17] J. P. Cavada, C. E. Cortés, and P. A. Rey, "A simulation approach to modelling baggage handling systems at an international airport," *Simul. Model. Pract. Theory*, vol. 75, pp. 146–164, Jun. 2017.
- [18] M. Pradhan, P. K. Roy, and T. Pal, "Grey wolf optimization applied to economic load dispatch problems," *Int. J. Elect. Power Energy Syst.*, vol. 83, pp. 325–334, Dec. 2016.
- [19] Z.-J. Teng, J.-L. Lv, and L.-W. Guo, "An improved hybrid grey wolf optimization algorithm," *Soft Comput.*, vol. 23, no. 15, pp. 6617–6631, 2019.
- [20] H. Eskandar, A. Sadollah, A. Bahreinejad, and M. Hamdi, "Water cycle algorithm—A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Comput. Struct.*, vols. 110–111, pp. 151–166, Nov. 2012.
- [21] A. Sadollah, H. Eskandar, and J. H. Kim, "Water cycle algorithm for solving constrained multi-objective optimization problems," *Appl. Soft Comput.*, vol. 27, pp. 279–298, Feb. 2015.
- [22] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.
- [23] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: An overview," *Soft Comput.*, vol. 22, no. 2, pp. 387–408, 2017.
- [24] *Continuous Conveyors—Belt Conveyors for Loose Bulk Materials—Basis for Calculation and Dimensioning*, Standard DIN 22101, 2011.
- [25] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neur. Net.*, Nov. 1995, pp. 1942–1948.
- [26] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. Int. Conf. Evol. Program.* Berlin, Germany: Springer, 1998, pp. 591–600.
- [27] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.
- [28] Y. Cao, H. Zhang, W. Li, M. Zhou, Y. Zhang, and W. A. Chaovalitwongse, "Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 718–731, Aug. 2019.
- [29] K. E. Parsopoulos and M. N. Vrahatis, "Unified particle swarm optimization in dynamic environments," in *Proc. Workshops Appl. Evol. Comput.* Berlin, Germany: Springer, 2005, pp. 590–599.
- [30] M. R. Tanweer, S. Suresh, and N. Sundararajan, "Self regulating particle swarm optimization algorithm," *Inf. Sci.*, vol. 294, pp. 182–202, Feb. 2015.



GABRIEL LODEWIJKS received the B.S. degree in transport engineering and logistics from the University of Twente, The Netherlands, in 1990, and the M.S. degree in transport engineering and logistics and the Ph.D. degree in dynamics of transportation systems from Delft University of Technology, The Netherlands, in 1992 and 1996, respectively.

He was a Professor of transport engineering and logistics at Delft University of Technology, from 2001 to 2017, and the Chief Technology Officer of Schiphol Airport Group, from 2007 to 2010. He has been a Full Professor and the Head of the School of Aviation, University of New South Wales, Sydney, Australia, since 2017. He has about 300 publications, including two books and more than 140 journal articles. His research interests include optimization of maintenance, repair and overhaul processes, automation of air cargo handling systems, tracking, and tracing of equipment, components, and people at airports and in aviation related companies, optimization of gate processes and baggage handling procedures to reduce the turnaround time of aircraft, maintaining safety and security in airport logistic processes, and the improvement of passenger experience by streamlining airport logistics.



YULIAN CAO (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in logistics engineering from Wuhan University of Technology, Wuhan, China, in 2009, 2012, and 2018, respectively.

From 2012 to 2014, she worked as a Visiting Scholar with the University of Washington (UW), Seattle, WA, USA. She has been a Postdoctoral Fellow with the University of New South Wales, Sydney, Australia, since 2018. Her research interests include evolutionary computing, PSO algorithm, memetic algorithm, aviation operation and optimization, and logistics system optimization.



NING ZHAO received the B.S. and M.S. degrees in mechanical engineering from Shandong University, Jinan, China, in 1999 and 2002, respectively, and the Ph.D. degree in mechanical engineering from Beijing Institute of Technology, Beijing, China.

He is currently a Professor with the University of Science and Technology Beijing, Beijing. His research interests include the simulation modeling and scheduling of logistics systems, factory planning and optimization, digital twin application in manufacturing, and logistics systems.



HAN ZHANG received the Ph.D. degree from Tsinghua University, Beijing, China, in 2015.

From 2013 to 2014, he worked as a Visiting Scholar with the University of Michigan, Ann Arbor, USA. From 2015 to 2020, he worked as a Scientific Fellow at Karlsruhe Institute of Technology, Germany. He has been an Assistant Professor at Tsinghua University, since 2020. His research interests include metaheuristic optimization algorithm, PSO algorithm, nonlinear/linear iteration algorithm, and numerical partial differential equations solution.

• • •