Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 6328016, 14 pages
https://doi.org/10.1155/2020/6328016

Research Article

Optimizing Airport Land Side Operations: Check-In, Passengers' Migration, and Security Control Processes

Ludovica Adacher[1] **and Marta Flamini**[2]

[1] *Roma Tre University, Via della Vasca Navale 79, Rome 00146, Italy*
[2] *International Telematic University Uninettuno, C.so Vittorio Emanuele II 39, Rome 00186, Italy*

Correspondence should be addressed to Ludovica Adacher; adacher@dia.uniroma3.it

Received 6 August 2019; Revised 17 January 2020; Accepted 10 February 2020; Published 19 March 2020

Academic Editor: Roberta Di Pace

This paper deals with the optimization of the Check-in, passenger migration, and Security Control processes in an airport land side terminal. Given the layout of the terminal, the passengers' flow, and the scheduled flights in a given time interval, the number and the position of Check-in counters and Security Control gates to be opened are output. The objective function is the minimization of the costs to activate the Check-in counters and the Security Control gates plus the costs that measure the passengers' discomfort. The stochastic passengers' behaviour and their preferences are simulated by a discrete event model, while the managing costs and the passengers' discomfort are optimized by the Surrogate Method. Capodichino Airport, located in Naples (IT), has been considered for the test phase. Results show the effectiveness and efficiency of the solutions of the Surrogate Method compared with the performances of other algorithms.

1. Introduction

In this paper a decision support system for different processes carried out in the land side area of an airport is proposed. The focus is on the connected processes of Check-in management, passengers' migration to the Security Control barrier, and Security Control operations. These processes usually involve several types of resources and services, affecting the costs and the performances of the airport management and of airline companies, together with the passengers' satisfaction. The paper addresses the problem of minimizing the Check-in and Security Control costs and the passengers' discomfort in terms of waiting times in line.

The increasing growth in the demand of air transportation services highlights the limited capacity of the involved systems and the necessity of developing decision support systems able to manage such growth, optimizing the performances, and limiting the effective costs. The problems arising in the airport environment involve both of the air side area and the land side area and affect each other. Moreover, most of the issues arising in this environment are complex, since they are marked out by a large variety of resources and services; they imply complex interactions among the involved processes and are affected by the stochastic nature of the behaviour of the actors operating and using the airport.

Several models have been developed over the last decade to support management decision making in optimizing the use of the resources and performing optimal services. There are different models and tools that can be classified on the basis of the levels of details [1]: microscopic models (high level of details), mesoscopic models (medium level), and macroscopic models (low level). In practice, macroscopic models help to make a very preliminary analysis of the solution effects; however they often lack realism and are not able to reproduce the complexity, variability, and stochastic nature of the airport. Microscopic models take into consideration several real details and can represent specific processes involving only some parts or aspects of the airport. One of the main shortcomings is that they neglect the interaction among different processes. Mesoscopic models have a suitable level of realism, since they are able to model interaction among processes integrated with the passengers' behaviour. They also allow inferring an accurate measure of the performance of subsets of connected services from the point of view of different actors, such as passengers, companies, and airport management, becoming a good support for airport planning.

In this paper authors combine simulation and optimization techniques to solve the problem of optimizing the Check-in, passenger migration, and Security Control processes in an airport land side terminal. They provide the optimization of a multicriteria cost function that models the costs of opening Check-in counters and Security Control barriers and discomfort of the passengers waiting in queue. Afterwards, the interaction between simulation and optimization phases provides the objective function value and its minimization. For this kind of problems, traditional optimization methods based on derivatives cannot be applied. Most known approaches are based on some form of random search or ordinal optimization approach. In addition, also because the simulation is computationally expensive, the extensive exploration of the entire solution domain would imply unacceptable calculation time. To avoid these problems, the Surrogate Method is introduced. This Method combines the advantages of stochastic approximation type of algorithm with the ability of obtaining sensitivity estimates. The gradient information necessary to drive the stochastic approximation part of the Surrogate Method is simplified considering the estimation of the objective function for a selection set (it will be described in Section 4).

This paper presents a mesoscopic model composed by an optimization module and a simulation module. The former minimizes the direct costs for the airport management and airline companies and the indirect costs modelling passengers' dissatisfaction. The latter cyclically interacts with the optimization module to compute the objective function value that is affected by the stochastic behaviour of the passengers. The simulator takes into consideration passengers with different needs, preferences, and behaviour.

More specifically, as shown in Figure 1, the input of the problem is the layout of the airport, the passengers' flow, and the scheduled flights during a given time interval. The main output is the number and the position of Check-in counters (for each airline) and Security Control gates to be opened. The objective is the minimization of the total costs: the cost for activating the Check-in counters plus the cost measuring the passengers' discomfort. Other output information is concerned with the average waiting times and the number of passengers in line at the Check-in counters and Security Control gates. The simulation module consists of a discrete event simulator: given the layout of the airport, the passengers' flow, the number and the position of Check-in counters (for each airline), and the active Security Control gates, it simulates the passengers' behaviour and computes the objective function value. The optimization module, according to the value of the objective function provided by the simulation module, finds the best solution (number of Check-in counters and Security Control gates that must be opened). The system has been tested on the case study of

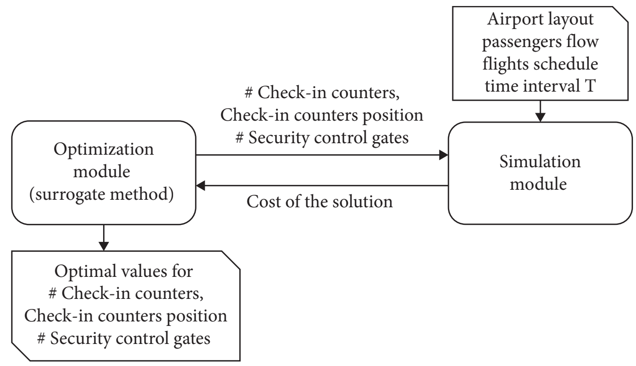

Figure 1: General schema of integration of optimization and simulation modules.

Naples Capodichino Airport (Italy) and the Surrogate Method has been compared to different versions of OptQuest, used in standard simulators such as Arena [2], ProModel [3], Simul8 [4], and Simio [5]. OptQuest has been developed by Glover et al. at the University of Colorado [6, 7]. The first version of OptQuest was customized to optimize discrete event simulation systems modelled with Micro Saint 2.0. OptQuest main optimization engine is based on the scatter search methodology with the tabu search strategy.

The main contribution of this paper can be summarized as follows:

(i) It provides an optimization/simulation decision support system

(ii) It considers the integration of Check-in, passengers' migration, and Security Control processes

(iii) It considers several types of passengers and services to make the model as much realistic as possible

(iv) It adapts the Surrogate Method to the addressed problem

(v) It compares the Surrogate Method performances to those of the algorithms used by standard simulators

The paper is organized as follows: Section 2 reports on the related literature review. Section 3 presents the problem formulation and an analysis of the objective function. In Section 4, the Surrogate Method and its adaptation to the problem are described. In Section 5, problem parameters, assumptions, and instance classes, processed by the simulation module, are specified. The results on the comparison between the Surrogate Method and OptQuest algorithm are reported in Section 6. Section 7 is dedicated to the conclusions.

2. Literature Review

In this section the authors provide a review of the recent literature focused on three principal aspects of the presented study, namely, the integration of the optimization and simulation techniques, the costs' minimization objective, and the solution methods. These three features are not singularly addressed by the researchers. Multiobjective approaches are very suitable for solving this kind of problems, but, given their complexity of the problem, it is difficult to formulate pure mathematical models to evaluate their performances. Simulation has been more widely used as the tool to reproduce the complexity, variability, and stochastic nature of different passengers. When the system operates in a stochastic environment and no closed form expression for the objective function is available, the problem is further complicated by the need to estimate the objective function values. Many approaches in the literature propose the combination of optimization and simulation to solve such issues. In particular on the basis of different level of details of the scenarios, the optimization is solved by linear programming or metaheuristic approaches: the former can address detailed cases while the latter is able to produce fast solutions when applied to limited models. The evaluation of different objective functions is computed by queue theory or simulation tools that are able to represent interaction among variables and stochastic aspects, such as the passengers' behaviour or their preferences.

The main scope of research in this field is to minimize the total costs. The considered costs can be both operational and social, when passengers' preferences or discomfort is considered and mapped into appropriate cost functions.

The cost minimization for what concerns Check-in operations has been addressed by several authors.

In [8] a counter assignment problem for an airport is considered with the possibility of application in both planning and operations mode. At the beginning, they develop an integer programming optimization model to solve the counter assignment problem to optimality. Then they propose a decomposition algorithm which solves the problem in reasonable computer time. The efficiency of both model and algorithm is also discussed. In [9], a binary linear programming formulation is developed and solved by CPLEX, for the workforce planning at the Check-in counters for real-world demand scenarios. In [10], a combinatorial optimization model to balance the operative costs of the service and the passenger waiting time at the terminal is proposed. In [11], the author proposes a linear programming model to minimize the total number of counters in each Check-in area, since in real-life counters for one flight should be adjacent and the remaining number of counters in each area should be fixed during Check-in operations. Stochastic aspects are then modelled by simulation, and the effects of various parameters, such as number of passengers on a flight and Check-in counters opening and closing time, have been studied. In [12], authors minimize the operational costs, by proposing a methodology which combines optimization based on integer linear programming and simulation. The aim is to determine the optimal number, location, and schedule of Check-in counters to be opened for departing flights, such that a given service level is ensured. In [13] the authors present a hybrid methodology to simulate the Check-in problems using an evolutionary algorithm that considers several constraints combined with a discrete event simulator that models the passengers' behaviour. In [14], the authors propose a static policy for the optimal allocation of a fixed number of dedicated Check-in counters serving a single flight. The objective is the minimization of the total cost of waiting counter operations and passenger delay. An improvement has been proposed in [15] where the authors consider the queuing optimization while developing a stochastic dynamic programming model able to determine the optimal numbers of counters to open in a given time window. In [16], the authors consider a more complex and complete problem suggesting a novel methodology which combines an evolutionary algorithm and simulation. The algorithm solves the allocation problems considering the minimum and maximum number of Check-in counters per flight, load balance in the Check-in areas, opening times of Check-in counters, and other restrictions imposed by the level of service agreement. Once the solutions are obtained through the algorithm, a second evaluation is performed by using a simulation model of the terminal that takes into account the stochastic aspects of the problem, with the objective of determining the most efficient allocation that maintains the quality indicators at the desired level.

As for the integration of service performance and passengers' expectation and satisfaction, many papers study the passengers' profiles and the perceived service quality.

In [17] the authors provide information on the low cost airlines passengers' behaviours, as well as managerial and research implications for effective passenger relationship management at two major British airports. In [18] authors provide passengers' assessments by considering some not observable features that depend on the management and the airport characteristics. They reach the conclusion that the quality assessment improves when price competition and private attendance affect the management. In [19] the authors compare different multicriteria evaluation methods to analyze passengers' preferences and satisfaction with respect to the performed passenger service. In [20] the authors combine optimization and simulation to model the airport passenger flow and to assign the resources to tasks in order to improve the level of services.

The specific problem of optimizing the Check-in process and customer satisfaction has been examined by several authors.

The paper [10] deals with the optimization of the Check-in desks schedule by determining the optimal number of the Check-in gates to balance the operative costs of the service and the passenger waiting time. In [21] the authors propose a mathematical model for the optimal management of the Check-in process, in which the objective function represents a measure of personnel costs associated with the delivery of the service. Available shift systems are taken into account, as a constraint for the model. The paper [22] presents a linear programming model to define an operators schedule that also meets the passenger arrival which varies during the day. In [14] the objective is to minimize the (expected) total cost of waiting, counter operation, and passenger delay costs which they show to be convex in the number of counters allocated. Only a few authors analyze the problem of minimizing costs related to the Security Control barriers. In [23] a model is presented for the security gates of a typical American airport. The model includes parameters such as the flow profile of the passengers and the counters activated in every procedure. In [24] the minimum walking distances of passengers, the minimum idle time variance of each gate, the minimum number of flights at parking apron, and the most reasonable utilization of large gates are selected as the optimization objectives. Probably the reason of the little interest in optimizing of the Security Control process lays in the fact that the objective function is convex that makes the problem easy to solve (as shown in Subsection 3.2).

3. Problem Description

In this section the addressed problem is described. It consists in optimizing the management of Check-in counters, passengers' migration, and Security Control gates that characterize the airport land side area departure operations. Departing passengers go through the Check-in controls and then migrate towards the Security Control gates. The objective of airline companies is to minimize the costs of activating the Check-in counters; the aim of the airport management is to minimize the cost of activating the Security Control gates. On the other hand, passengers prefer a limited queue both at the Check-in counters and at the Security Control gates and a fast path between the two areas. Hence the average length of the path between the Check-in area and the Security Control gates should be minimized by assigning the Check-in counters closer to the Security Control area to the companies with the highest number of passengers.

Figure 2 reports on a simple schema of the passenger flow in the land side area of the terminal. Departing passengers can either go to the Check-in area or use the Check-in kiosks or directly go to the Security Control gates, if they have passed the Check-in online. The first group of passengers can be *tourist* or *business* passengers and access either the common Check-in area or the dedicated one. As the name suggests, a *common* Check-in counter processes passengers going to several destinations of a given airline company, while a *dedicated* Check-in counter processes only the passengers of a specific flight. Business and tourist passengers can travel in a group and/or can carry a certain number of suitcases. Service time by Check-in operators depends on the number of people in the group and on the number of luggage. *Smart* passengers use the online Check-in procedures. The three main flows of passengers can either directly go to the Security Control gates or pass through the shopping area before reaching the Security Control barriers. Such passenger behaviour determines the arrival profile at the Security Control gates and affects the main travel time of the passengers from the Check-in area to the Security Control area. Service time at the Security Control gates increases if the passenger has a hand luggage.

Given the terminal layout, a stochastic passenger arrival and type distribution, and a set of departing flights in a time slot, the decision support system determines the number of Check-in counters to be activated for each company, their type (common or dedicated), their position, and the number of the Security Control gates which minimize the activation costs and, at the same time, maximize the passengers' satisfaction, allowing the passenger flows to be as much fluid as possible.

In this paper, Naples Capodichino International airport, located in Naples, Italy (NAP in the international IATA code), has been chosen as case study. The land side of NAP is composed by Terminal 1 (T1) and Terminal 2 (T2) (see Figure 3). The passengers' flow on T1 is modelled (since T2 is used only in spring and summer season and for charter flights). Passengers entering T1 access the Check-in operations on the ground floor where Check-in counters and kiosks are located. Afterwards, passengers migrate to the Security Control gates located on the first floor that also presents some shops and food services that passengers can visit before accessing the Security Control gates.

The problem has been modelled and solved for both a single airline company and several airline companies. When the authors examine just one airline company, Alitalia is considered, since 61% of the total number of the flights per day in NAP belongs to it.

3.1. Problem Formulation. This subsection defines parameters and variables used to formulate the complete problem. As highlighted above, each passenger can belong to one of the following five different categories of the set C: $C = \{d, c, dB, cB\}$, where *sm* passengers can directly access the Security Control gates, d passengers are served by dedicated Check-in counters, c are served by common Check-in counters, dB passengers have a business class ticket and are served by a dedicated Check-in counter, and cB passengers have a business class ticket and are served by a common Check-in counter.

The parameters are as follows.

(i) N is the number of airline companies

(ii) F is passenger distribution, which involves arrival times and categories

(iii) Q_{cat} is the tolerable length of the queue for a passenger belonging to category cat $\in C$

Decision variables related to the i – th airline company are as follows.

(i) c_i: number of common Check-in counters activated

(ii) d_i: number of dedicated Check-in counters activated

(iii) cB_i: number of common Check-in counters activated for business passengers

(iv) dB_i: number of dedicated Check-in counters activated for business passengers

In addition, scc identifies the number of Security Control gates activated and Q_{sc} represents the tolerable length of the queue for a passenger at the security gates. The variables are all integer and nonnegative.

Due to the complexity of the problem and the randomness of the passengers' behaviour, the objective function value and the arrangement of the passengers in the queues are computed by a discrete event system simulator. In particular, the value indicating the arrangement of the

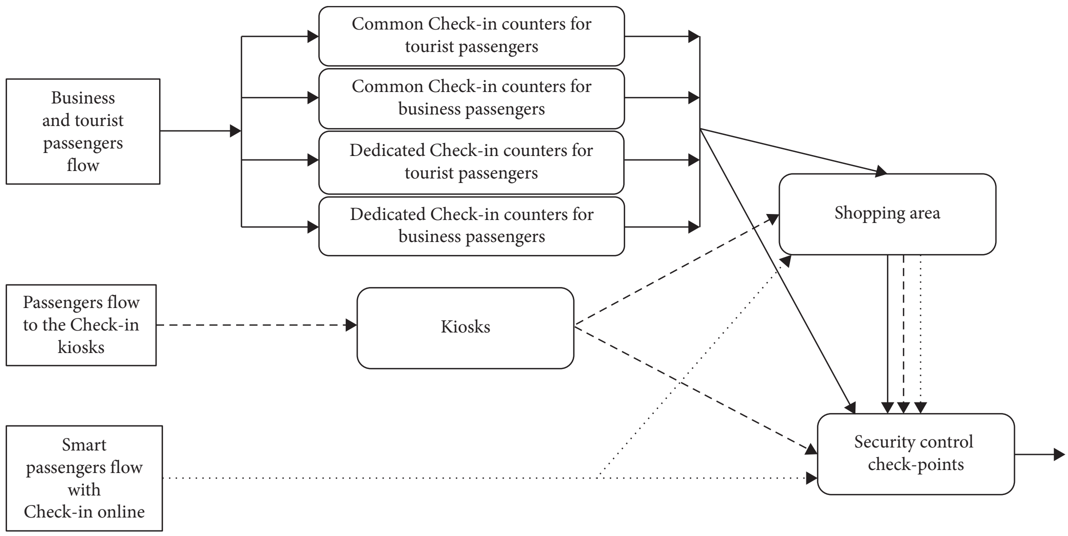

Figure 2: Schema of passengers' flow in the land side area.

passengers in the Check-in area is denoted by PinQ_{cat} and identifies the average number of passengers belonging to the category cat $\in C$ in a queue. Analogously PinQ_{sc} identifies the average number of passengers in line at the Security Control gates.

3.2. The Objective Function. The objective function involves the following cost items: (C_0) is the cost for activating a Check-in counter; (C_{sc}) is the cost for activating a Security Control gate; (CP_{sc}) is the discomfort cost of the passenger at the Security Control queue; (CP_{cB}) is the discomfort cost of a business passenger at a common Check-in counter queue; (CP_{dB}) is the discomfort cost of a business passenger at a dedicated Check-in counter queue; (CP_c) is the discomfort cost of a tourist passenger at a common Check-in counter queue; and (CP_d) is the discomfort cost of a tourist passenger at a dedicated Check-in counter queue [10, 20]. Hence, the objective function OF, to be minimized, is

$$\begin{aligned} \text{OF} = {} & \sum_{i=0}^{N} C_0\left(c_i + d_i + cB_i + dB_i\right) + C_{\text{sc}}\text{scc} + CP_{\text{sc}}\max\left(\text{PinQ}_{\text{sc}} - \text{Q}_{\text{sc}}, 0\right) + CP_{cB}\max\left(\text{PinQ}_{cB} - \text{Q}_{cB}, 0\right) \\ & + CP_{dB}\max\left(\text{PinQ}_{dB} - \text{Q}_{dB}, 0\right) + CP_{c}\max\left(\text{PinQ}_{c} - \text{Q}_{c}, 0\right) + CP_{d}\max\left(\text{PinQ}_{d} - \text{Q}_{d}, 0\right). \end{aligned} \tag{1}$$

Constraints are

$$\sum_{i=1}^{N}\left(c_i + cB_i + d_i + dB_i\right) \leq \text{Max}_{ck-in}, \tag{2}$$

$$\text{scc} \leq \text{Max}_{\text{sc}}, \tag{3}$$

$$c_i, d_i, cB_i, dB_i, \text{scc} \in \mathbb{Z}^{+}. \tag{4}$$

Parameters Max_{ck-in} and Max_{sc} identify the maximum number of Check-in counters and Security control gates, respectively. Hence, the set of constraints [25] bounds the number of Check-in counters due to the physical capacity of the airport. Analogously [26] bounds the number of Security control gates. These sets of constraints are referred to as capacity constraints.

X and Y are the sets of integer decision variables: $X = \{c_i, d_i, cB_i, dB_i\}$ is the set of variables related to the Check-in counters and $Y = \{\text{scc}\}$ is the set of variables related to the Security Control gates.

The analysis of the objective function can be done by considering its trend when one of the decision variables varies and the others are constant. Figure 4 reports on the trend of the objective function for only one airline company. The trend of the objective function with respect to the Security Control Check gates (Y) is convex, while its trend, as a function of the Check-in counters variables (X), presents many valleys and then many local minima.

This characteristic requests an optimization approach that is able to get away from local minima.

4. Optimization Module: The Surrogate Method

The Surrogate Method was proposed by Gokbayrak and Cassandras to solve stochastic discrete optimization problems with not negative integer decision variables [27]. The Surrogate Method provides good results in various application areas finding good or suboptimal solutions for the original discrete problem and assuring very fast convergence [25–28]. In a previous work [29], the authors firstly test the

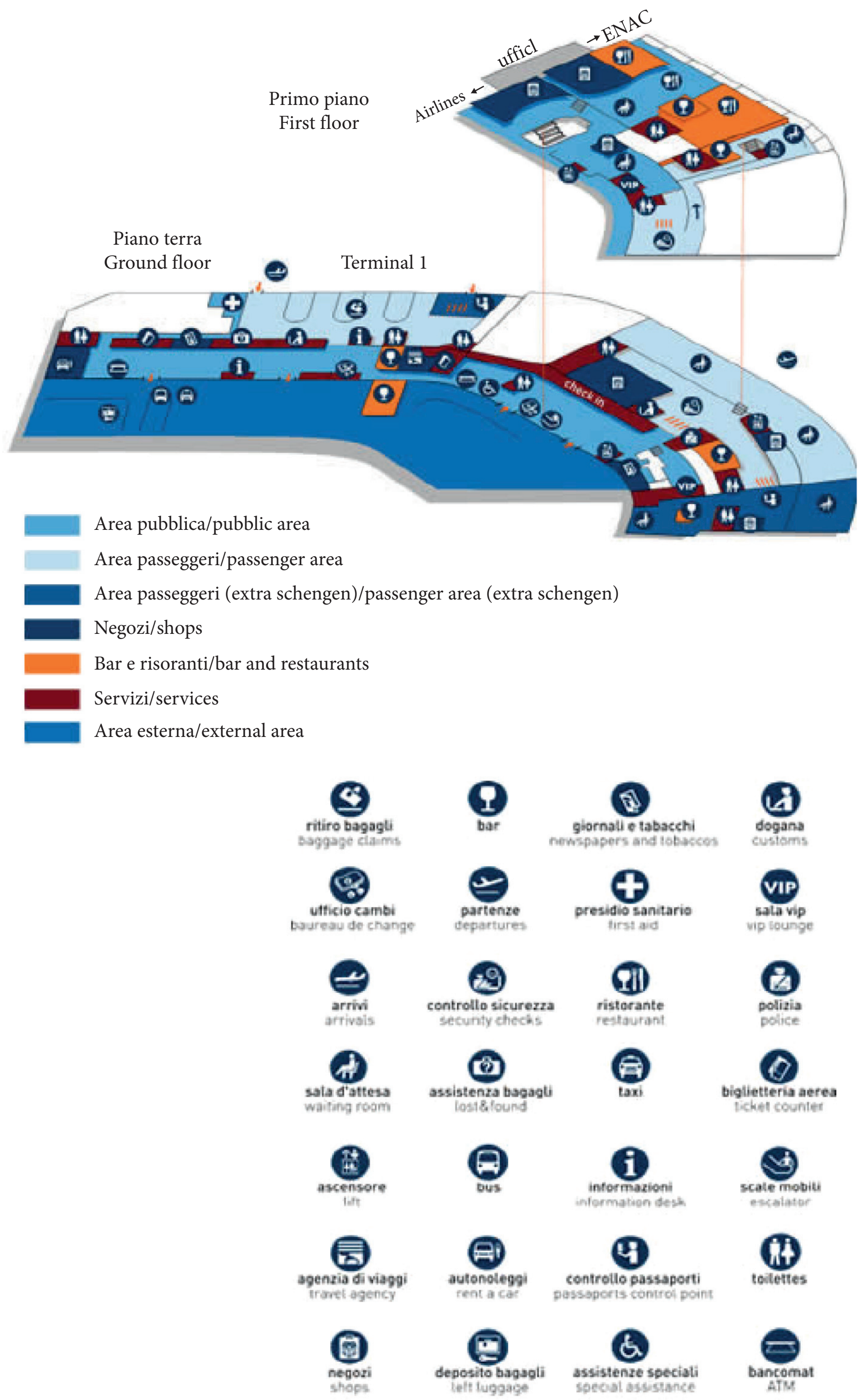

Figure 3: Terminal 1 of Naples Capodichino Airport.

Surrogate Method to a similar problem with a single airline company and some restrictions on passenger categories.

With respect to the cited paper in this work the authors better fit the Surrogate Method parameters to the addressed problem. To this aim, a detailed study of the gradient step size is here reported. Moreover, the authors now provide a comparison with different version of commercial tools that validate the effectiveness of the Surrogate Method. Finally, the solution approach is generalized up to four airline companies (instead of only one).

In general, the Surrogate Method transforms the discrete problem into a surrogate continuous problem. The discrete problem is also denoted as the original one that is the problem that has to be solved, while the surrogate

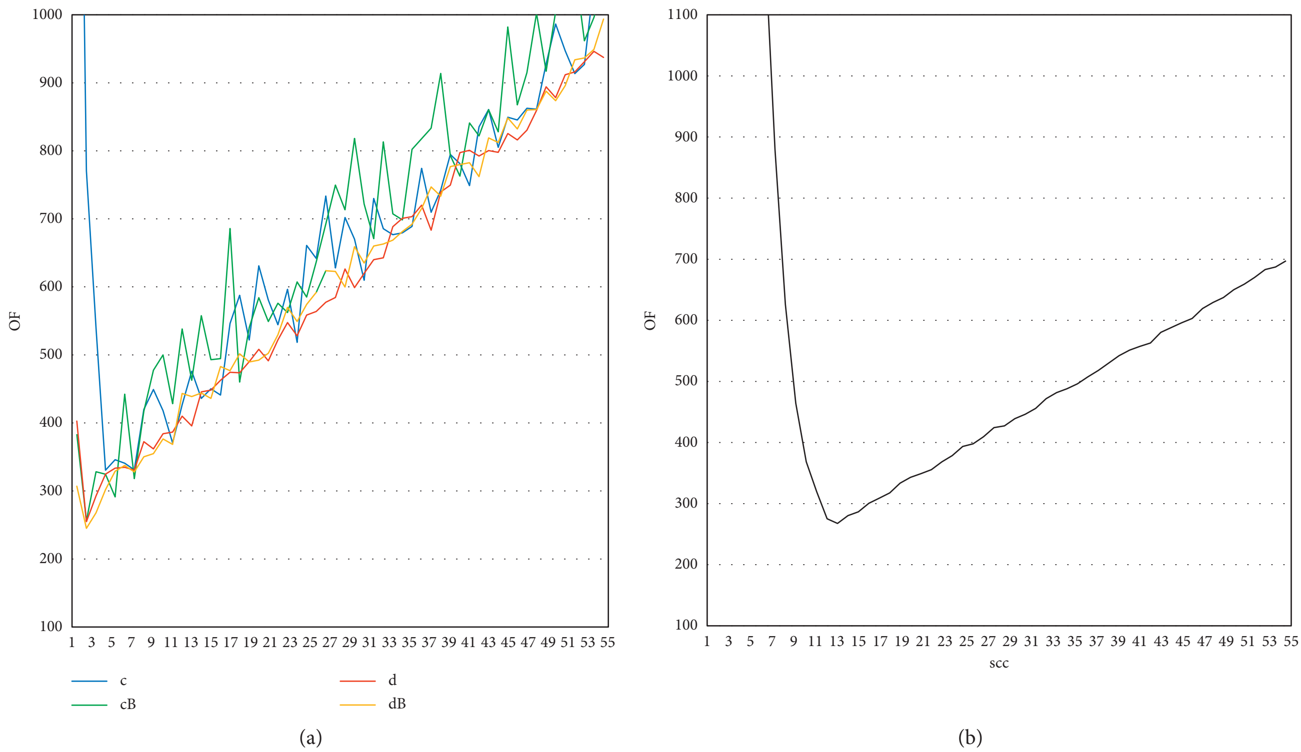

Figure 4: Trend of the objective function with respect to the Check-in counters variables and Security Control gates for only one company.

continuous problem is obtained by the relaxation of the integer constraints of the original problem. The continuous problem is solved by applying the standard gradient-based procedure. This procedure presents two distinctive features that make the Surrogate Method more efficient and effective as follows.

(i) The cost of the discrete system is cyclically adjusted; i.e., the updating of the continuous and discrete states is computed at each iteration of the algorithm, rather than at the end when the best solution in the continuous field has been checked (step 14 of Algorithm 1). Since the best solution in the continuous field does not necessarily correspond to the best solution in the discrete field, this feature allows evaluating the effectiveness of the solution in the discrete field and increasing the probability of finding a good solution for the original problem.

(ii) In the classical gradient procedure, when the continuous solution has been updated, the gradient is computed in the continuous field, while in the Surrogate Method the gradient is evaluated in the discrete field (step 16 of Algorithm 1). Hence, the gradient direction is influenced by the trend of the solution in the discrete field.

The specific application of the Surrogate Method to the problem addressed in this paper has been sketched in Algorithm 1, where the reader can find the steps of the algorithm on the left side and the comments on the right side. The comments in bold represent the main phases of the algorithm.

Parameters H and K represent the maximum number of consecutive iterations where the solution does not improve, and the maximum number of iterations of the algorithm, respectively.

Vector $Z = (X, Y)$ is an integer M-dimensional decision vector where each component denotes the number of resources that have to be activated, subject to the capacity constraints [25, 26, 28]. $\mathrm{OF}(Z)$ is the cost of the solution when the state is Z. The integer constraint is relaxed and a resulting surrogate problem is obtained. Steps from 4 to 13 form the selection set $S(\rho_k)$ that is the set of the discrete vectors (not necessarily feasible for the original problem) used to compute the gradient (see [27] where the authors prove the effectiveness of this procedure).

When and if a solution of the surrogate problem ρ_k is obtained, it is possible to map it through the transformation function f into a discrete point $z_k = f(\rho_k)$, which is the solution of the discrete problem. Function f selects the integer d that minimizes the difference $|d - \rho_k|$ as reported in step 14.

Note however that the sequence $\{\rho_k\}$, $k = 1, 2, \ldots$, generated by an iterative scheme to solve the relaxed problem, consists of real-valued solutions which are unfeasible for the discrete problem.

Here, the key feature of the Surrogate Method that at each iteration k of the scheme updates the discrete state z_k through the function f: $z_k = f(\rho_k)$ as ρ_k is highlighted. Then the state is updated. Notice that the choice of the step size (η_k, in step 16 of Algorithm 4) determines the performance of the convergence rate, as usual in the classical gradient procedure.

Steps from 17 to 23 update the optimal solution in the discrete field (step 24).

In Figure 5, the integration of the simulation module into the optimization module is represented. Given the capacity constraints of Check-in and Security control areas and the number of companies as input, the Surrogate Method starts from a random feasible solution and determines the selection set $S(\rho_k)$ described in Steps 5–13 of Algorithm 4. The input to the simulation module consists of information related to the airport layout, the flights' schedule, and the passenger flow, since it is supposed to simulate the stochastic behaviour of the passengers.

Note that, for each updating step of the Surrogate states (step 16 of Algorithm 1) the objective function value is computed $M + 1$ times by the simulation module. Hence, it becomes crucial to develop an efficient simulation module which performs an acceptable computation time. A discrete event simulator is implemented with Java.

5. Simulation Module: The Parameters

The section examines the problem parameters and assumptions for the instances processed by the simulation module.

The model is a discrete event simulation model: the state of the system changes when asynchronous events occur. Each event occurs at a particular time; thus the simulation can directly jump in time from one event to the next. The computation of the objective function value is possible thanks to the simulation of the passenger flow, represented by the formation of the queues at the Check-in counters or at the Security Control gates (see [30]). The simulator provides also the waiting time of the passengers at lines and the routing to reach Security Control gates. In order to simulate the passenger flow it is necessary to introduce some simplifying assumptions as follows.

(i) Discretization of the problem: the time horizon T is divided into intervals with constant duration t. The problem becomes a discrete problem, and all the parameters and variables are referred to each interval t. The simulation time is divided into 32 time slots, each of 15 minutes.

(ii) Arrival distribution: Check-in service demand can be expressed in terms of passengers, represented by stochastic variable. The passengers' arrival distribution is shown in Figure 6. In each time slot the interarrival time is uniformly distributed. Three types of passengers are here considered: *business*, travelling alone and with one baggage only; *tourist* travelling alone or in groups up to 4 people carrying from 0 to 4 luggage; and *smart* who skips the Check-in operations and goes directly to the Security Control gates. 10% of the passengers are business passengers while 90% are tourist passengers. It is assumed that 75% of passengers use the airport counters, 20% use the Check-in online (smart passengers), and 5% use the kiosk. 60% of passengers proceed directly from facility to facility and the remaining 40% spend time in bar or shops (a uniform distribution from 5 to 25 minutes has been considered). Moreover 60% of passengers know the airport or/and have no orientation or physical impediments and take the shortest path to reach their destination. 25% spend 10% more and the remaining 15% spend 20% more to reach their destination with respect to the shortest path travel time (this might be due to some moving speed limitations by old people or families with babies and/or to the preference of visiting shops before reaching the Security Control barriers).

(iii) Service time: service time represents the time needed to process the passenger. As already mentioned, this study takes into consideration two types of Check-in counters: *common* to the flights of the same airline company or *dedicated* to a specific flight of an airline company. 60% of the Check-in counters are common, whereas 40% are dedicated. At the Check-in counter, the processing time is based on the number of bags. Each bag needs 0.5 mins to be processed. 60% of business passengers have no luggage and 40% have only one suitcase. Let G be the number of people of the group each passenger belongs to. N_{lug} is the number of luggage for each passenger. Four categories of tourist passengers, each identified in a column of Table 1, are considered.

(iv) At the Security Control gate, it is assumed that the processing distribution is k-Erlang distribution with $k = 0$ and mean $= 0.5$.

(v) Unit costs to evaluate the objective function are provided in Table 2 [31].

The maximum number of Check-in counters that can be opened is 56, while the maximum number of Security Control gates is 25. The tolerance threshold for the length of the line is set to 10 for business passengers and 15 for the others.

The results reported in the following are the mean values calculated on 20 different runs with the same probability distribution of the simulation parameters.

6. Results Analysis

This section reports on the test results highlighting the efficacy and effectiveness of the solutions provided by the Surrogate Method. The rate of convergence of the Surrogate Method depends not only on the characteristics of the objective function but also on the choice of the step size (η_k in Algorithm 4, step 16). For this reason, different step sizes and two different procedures are considered: one static (just one value for η_k) and one dynamic (η_k changes considering the gradient's value). In the static procedure, η_k varied from 0.01 to 0.9 with step 0.01, whereas the choice of the dynamic procedure is reported in Table 3, where five different options are examined.

(1) Initialize $\rho_0 = z_0$ satisfying constraints [25, 26] (ρ_0 is a continuous vector, z_0 is a discrete vector, both of dimension $M + 1$)
(2) Initialize $\rho^* = \rho_0$, $z^* = z_0$ (ρ^* is the optimal solution of the continuous problem)
(3) Initialize $h = 0$
(4) **while** $((k \leq K) \vee (h \leq H))$ **do** (K and H integer parameters, **Form the selection set** $S(\rho_k)$ (steps 5–13): $S(\rho_k)$ is a set of discrete vectors)
(5) Initialize $I = \{1, \ldots, M\}$ and $v = \rho - [\rho]$ (I is the set of M integers, v is a continuous vector the component $v[i]$ is the decimal part of the $\rho[i]$ component)
(6) **while** $I \neq \theta$ **do**
(7) $i = \arg\min_{j \in I}(v[j])$
(8) $y[i] = v[i]$
(9) $W_i = \sum_{j \in I} e_j$ (W_i integer vector, e_j the versor with j-th component equal to 1)
(10) $v = v - y[i]W_i$
(11) $I = I \backslash \{i\}$
(12) **end while**
(13) $S(\rho_k) = \{W_i - \lfloor \rho \rfloor, i = 0, \ldots, M\}$ (**Transform the continuous problem to the discrete problem**, D is the set of the discrete vectors that satisfy constraints [25, 26, 28])
(14) $z_k = f(\rho_k) = \arg\min_{d \in D} \|d - \rho_k\|$ (Transformation function f, **Gradient estimate)**
(15) $\nabla \mathrm{OF}(\rho_k) = [\nabla_1 \mathrm{OF}, \ldots, \nabla_M \mathrm{OF}]^T$ (OF Objective function declared in [29], where $\nabla_j \mathrm{OF}(\rho_k) = \mathrm{OF}(p) - \mathrm{OF}(q)$, where k satisfies $p - q = e_j$ and $p, q \in S(\rho_k)$, **Update state**)
(16) $\rho_{k+1} = f[\rho_k - \eta_k \nabla \mathrm{OF}(\rho_k)]$ (η_k is the step size of the gradient method, **Optimal solution update**)
(17) **if** $\mathrm{OF}(\rho_k) \leq \mathrm{OF}(\rho^*)$ **then**
(18) $\rho^* = \rho_k$
(19) $h = 0$
(20) **else**
(21) $h = h + 1$
(22) **end if**
(23) **end while** (**Return the optimal solution** z^*)
(24) **Return** $z^* = \arg\min_{z_k} \mathrm{OF}(z_k)$

Algorithm 1: The Surrogate Method.

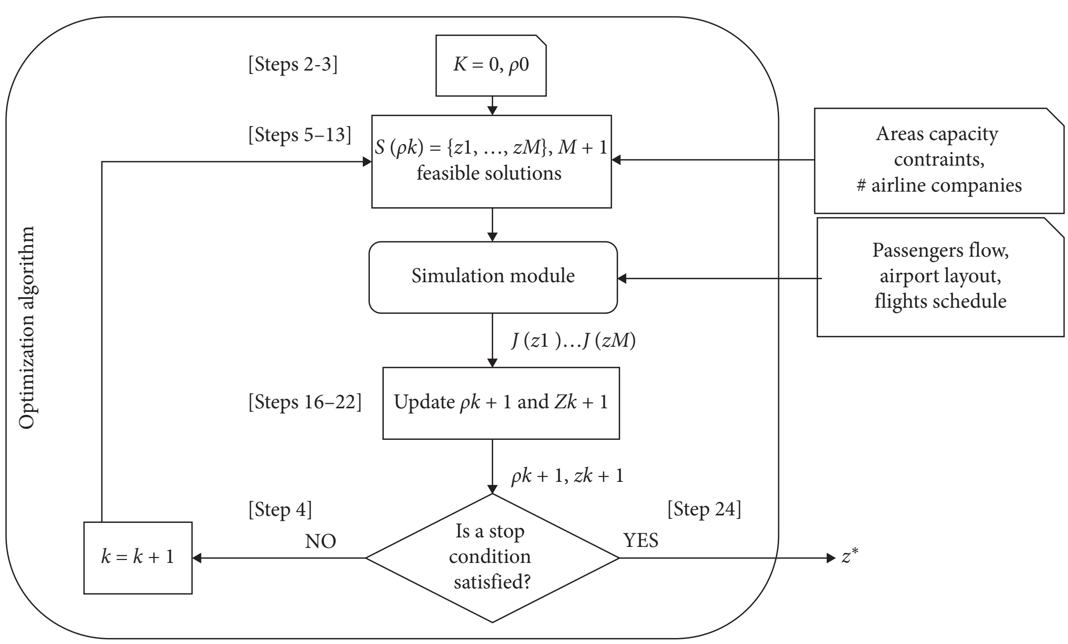

Figure 5: General scheme of the proposed approach.

To prove the good performance of the Surrogate Method, a comparison with different versions of OptQuest (v1, v2, and v3) is reported. Since the genetic algorithms performances typically depend on the initial population dimension, different numbers of initial individuals have been considered. In v1, the dimension of the initial population is fixed to 100 as proposed by Laguna; in v2, it is increased to 200; and in v3, it is equal to 300. When only one

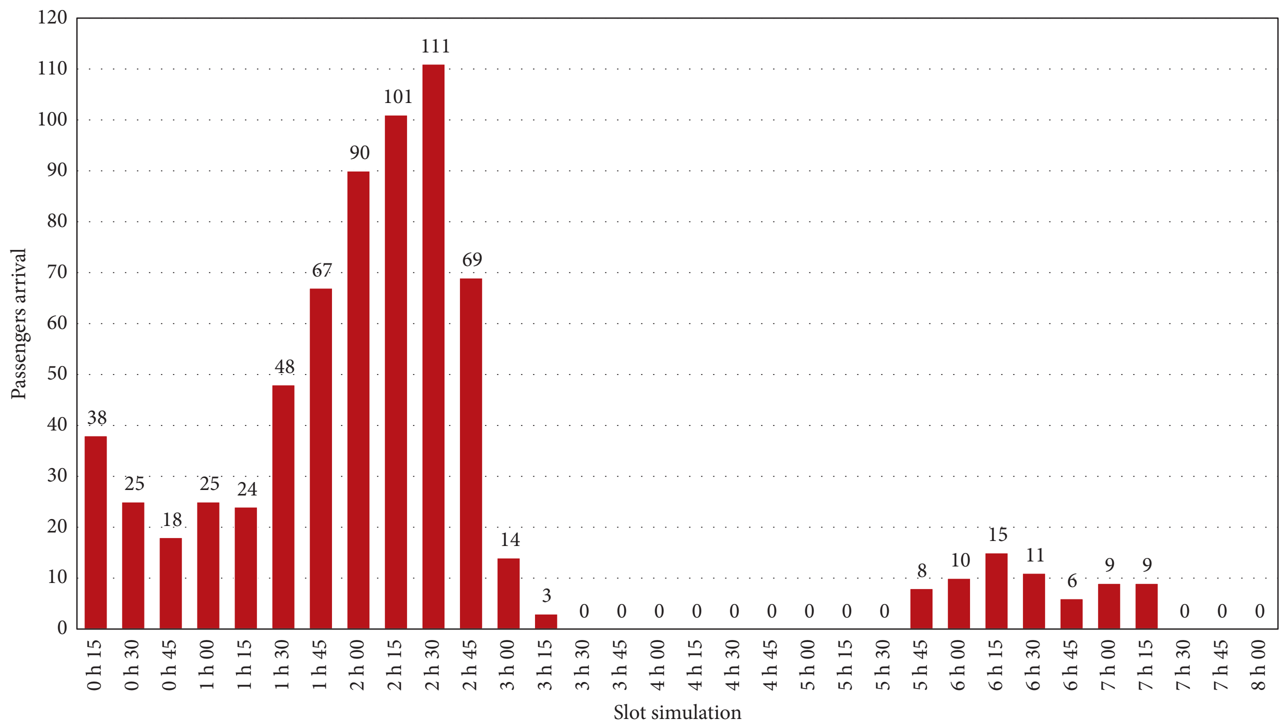

Figure 6: Passengers' arrival distribution.

Table 1: Percentage distribution of variable N_{lug} in the four categories $G1, \ldots, G4$.

N_{lug}	$G1$	$G2$	$G3$	$G4$
0	41	15	10	7
1	43	40	33	14
2	5	40	30	8
3	1	5	27	71

Table 2: Unit costs.

C_0	C_{sc}	CP_{sc}	$CP_{cB} = CP_{dB}$	$CP_c = CP_d$
12.5	10	20	15	10

company is taken into consideration the different optimization algorithms give similar results, but when the problem becomes more complex the dominance of the Surrogate Method is evident, as reported in Table 4. In the column "Surrogate" the values of the objective function and of the processing time of Surrogate Method are reported. It gives the best performances. The percentage results are calculated with respect to this best solution and are given by the difference between the compared and the best solution values, over the value of the best solution, times 100. The costs are measured in thousands of euro; the computation time is measured in seconds.

OptQuest versions give worse performance with respect to the Surrogate Method, from 2.5% to 52.5% for the efficacy and from 33.2% to 287% for the efficiency. When OptQuest is faster with respect to the Surrogate Method, it does not converge to the optimal solution; in fact the cost is 52.5% more expensive with respect to the solution of the Surrogate Method. When four companies are considered the OptQuest approach does not converge at the optimal solution considering the number of iterations limited to 45 (see Figure 7). The number of iterations is limited to 45 since the analysis of the rate of convergence guarantees the convergence of the Surrogate Method for the Check-in problem to 35 iterations (see Figure 8).

When several companies are considered the problem becomes more complex since the dimension of the decision vector is bigger and the objective function presents a higher number of local minima, hard to jump out. As highlighted above, the evaluation of the Security Control solution is not a difficult task (see Figure 4), but it can significantly increase the computation time of the Surrogate Method, since the research space has one more component. In fact, the dimension of the decision variable vector for one company presents $|Z| = 5$ and $|X| = 4$, for two companies $|Z| = 9$ and $|X| = 8$, and for four companies $|Z| = 17$ and $|X| = 16$. For this reason the Surrogate Method needs more iterations to find the optimal solution. It could be a valid solution to apply it to the Check-in problem only and to calculate the best Security Control solution via enumeration.

The problem assuming X as decision variables and only the constraints [25, 26] minimizes the following objective function:

Table 3: Dynamic η values.

η dynamic	$\lvert\nabla OF(\rho_k)\rvert < 25$	$25 < \lvert\nabla OF(\rho_k)\rvert < 50$	$50 < \lvert\nabla OF(\rho_k)\rvert < 100$	$100 < \lvert\nabla OF(\rho_k)\rvert < 200$	$200 < \lvert\nabla OF(\rho_k)\rvert < 100000$
Type 1	$\eta = 0.06$	$\eta = 0.03$	$\eta = 0.02$	$\eta = 0.01$	$\eta = 0.007$
Type 2	$\eta = 0.2$	$\eta = 0.07$	$\eta = 0.03$	$\eta = 0.02$	$\eta = 0.005$
Type 3	$\eta = 0.1$	$\eta = 0.05$	$\eta = 0.025$	$\eta = 0.01$	$\eta = 0.005$
Type 4	$\eta = 0.09$	$\eta = 0.04$	$\eta = 0.02$	$\eta = 0.01$	$\eta = 0.005$
Type 5	$\eta = 0.8$	$\eta = 0.06$	$\eta = 0.03$	$\eta = 0.02$	$\eta = 0.005$

Table 4: Comparison between the Surrogate Method (dynamic η type 1) and OptQuest performances.

Companies	Surrogate Method		OptQuest v1		OptQuest v2		OptQuest v3	
	Cost	Comp. time	Cost	Comp. time	Cost	Comp. time	Cost	Comp. time
1	120	2.92	+11.5%	+139.5%	+5.7%	+217.2%	+2.5%	+287.5%
2	230	14.06	+26.1%	+33.2%	+17.5%	+116.9%	+15.9%	+195.9%
4	484	99.92	+52.5%	-0.94%	+33.3%	+69.6%	+22.3%	+154.8%

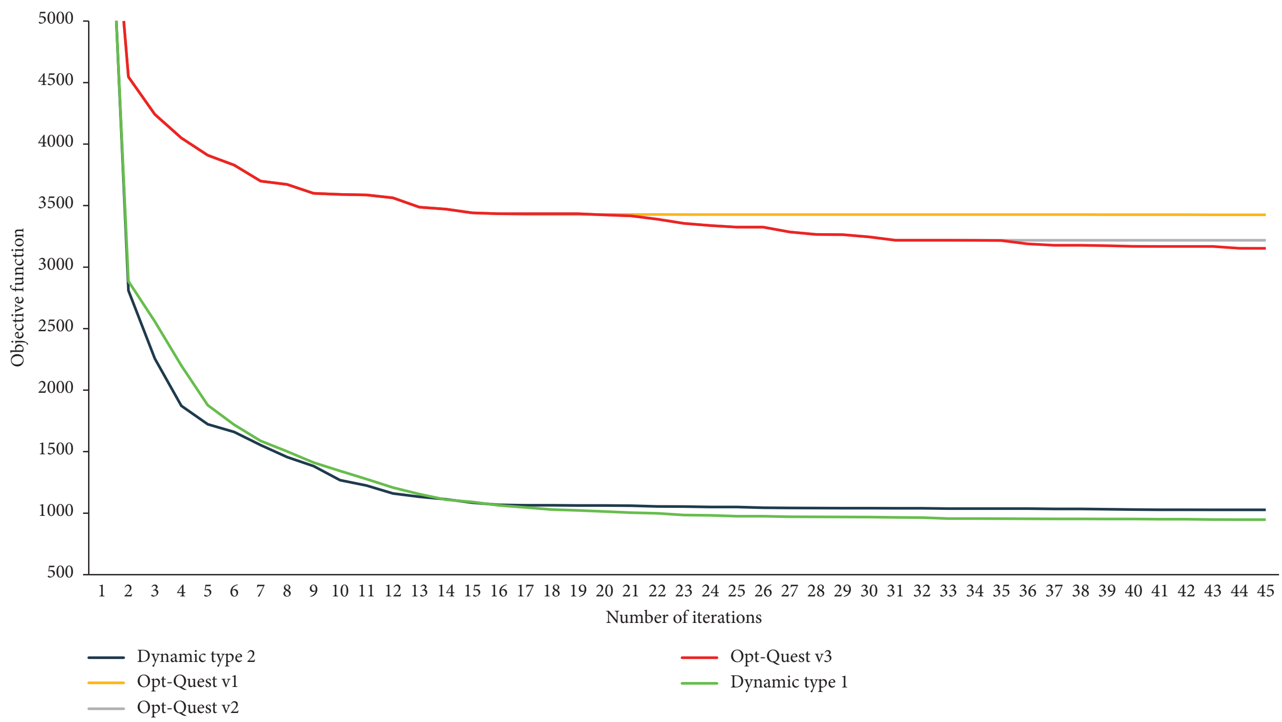

Figure 7: Algorithms' trend considering 4 companies.

$$\mathrm{OF}(X) = \sum_{i=1}^{N} C_0\left(c_i + d_i + cB_i + dB_i\right) + CP_{cB}\max\left(\mathrm{PinQ}_{cB} - Q_{cB}, 0\right) + CP_{dB}\max\left(\mathrm{PinQ}_{dB} - Q_{dB}, 0\right) + CP_{c}\max\left(\mathrm{PinQ}_{c} - Q_{c}, 0\right) + CP_{d}\max\left(\mathrm{PinQ}_{d} - Q_{d}, 0\right). \quad (5)$$

Table 5 reports on the analysis of the convergence time (in seconds) for the dynamic step size type 1. Varying the number of companies on the rows, the second column reports the total convergence time to compute the best solution of the Check-in and Security Control problems, with X and Y decision variables, whereas "Check-in" and "SC (tot. enum)" report on the convergence times to solve two problems of Check-in counters and Security Control gates separately. Notice that the total enumeration is applied after the Surrogate Method has fixed the Check-in solution. The sum of the values in columns three and four is lower than the value in column two, and the percentage of such time saving (with respect to column two) is reported in the fifth column. By observing the values in column three, it can

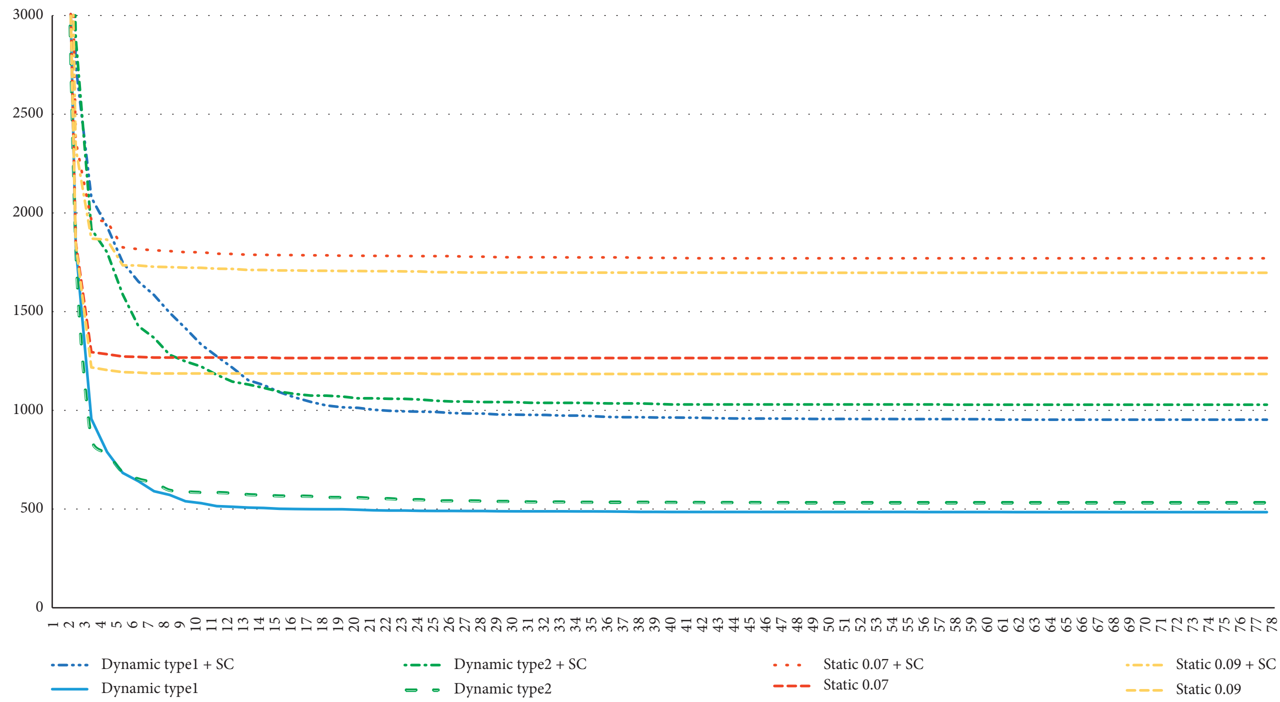

Figure 8: Rate of convergence for different step size with or without Security Control gates.

Table 5: Convergence time analysis for the dynamic type 1.

Companies	Check-in + SC	Check-in	SC (tot. enum)	Time saving%
1	3.91	2.2	0.71	−25.5
2	21.27	10.5	2.42	−39.2
4	140.56	52.7	8.57	−56.4

Table 6: Performance of the Surrogate Method.

Companies	Dynamic type 1		Dynamic type 2		Static 0.07		Static 0.09	
	Cost	Conv. time	Cost	Conv. time	Cost	Conv. time	Cost	Conv. time
Surrogate Method applied to the Check-in problem								
1	120	2.2	+1%	−50%	+1%	−10%	+2%	0%
2	237.4	10.5	+4%	−30%	+32%	−60%	+33%	60%
4	484.3	52.7	+9%	−4%	+160%	−88%	+144%	−86%
Surrogate Method applied to the Check-in and Security Control problems								
1	246.28	3.91	+2%	−50%	+3%	−1%	+4%	−6%
2	487.19	21.27	+4%	−39%	+60%	−14%	+130%	−40%
4	950	140.56	+8%	−4%	+86%	−78%	+85%	−76%

be noticed that the convergence time does not increase with a linear trend with respect to the number of companies (i.e., the convergence time for two companies increases by 443%, rather 100%, with respect to the convergence time of the problem with one company only). The two problems of Check-in and Security Control optimization can be effectively considered separately. In fact when applying the Surrogate Method to the whole problem or the Surrogate Method plus the total enumeration to the separate problems the optimal solutions coincide. Hence it is convenient to solve the two problems' time saving separately until the 56.4% of the convergence time.

In Table 6, the first set of results reports on the best solution given by dynamic type 1 that is the values of the objective function (Cost) and convergence time in seconds. In the following columns, values represent the percentage variation with respect to the solution given by dynamic type 1 (hence, if the value is positive, the related solution is worse than dynamic type 1). The table shows the best performances of the Surrogate Method solving the problem with and without the Security Control variable. For a single company, the step size does not affect the convergence time and value of the Surrogate Method. When the number of companies increases the objective function trend presents deeper local minima, hence the static step does not jump out of local minima. Such trend persists with and without the Security Control variable.

In Figure 8, the rate of convergence is reported; 80 iterations are considered. It is evident that the Surrogate Method applied both to the Check-in and to the Security Control problems needs a higher number of iterations to converge to the optimal solution (or get closer to the optimal): for the dynamic step size the whole problem (Check-in and Security Control) requires 65 iterations to converge, whereas for only the Check-in control problem 35 iterations are enough. The trends of the results related to the Surrogate Method with and without the Security Control problem are the same and simply shifted upwards (when considering the Security Control problem the objective function also comprehends the additional costs for activating the Security Control gates; hence, the total costs lines shift upwards due to such additional terms).

With regard to the discomfort costs, the results related to the test instance show that the queues at the Check-in counters, for all the Check-in types, tend to satisfy the tolerance threshold Q_{cat}; on the contrary, there is an additional discomfort cost for the Security Control gates.

7. Conclusions

This paper presents an efficient and effective algorithm to determine the number and the position of critical resources at the airport terminal departure operations. An integrated approach based on discrete event simulation (simulation module) and the Surrogate Method (optimization module) is proposed to minimize a generalized cost function that takes into account not only the managing costs but also the passengers' satisfaction. The Surrogate Method is compared to the well-known OptQuest algorithm, and the results put in evidence the effectiveness of the Surrogate Method, which provides better performances thanks to its ability of jumping out of local minima. The integrated approach might be useful both in optimizing an airport departure area and in designing the size and the layout of the departure area of a new airport.

Data Availability

The data used for the analysis are described in the paper and the data can be available on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] I. E. Manataki and K. G. Zografos, "A generic system dynamics based tool for airport terminal performance analysis," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 4, pp. 428–443, 2009.

[2] https://www.arenasimulation.com/.

[3] https://www.promodel.com/.

[4] https://www.simul8.com/.

[5] https://www.simio.com/index.php.

[6] F. Glover, J. P. Kelly, and M. Laguna, "New advances and applications of combining simulation and optimization," in *Proceedings of the 28th conference on Winter simulation—WSC'96*, J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, Eds., pp. 144–152, Coronado CA, USA, December 1996.

[7] M. Laguna, *Optimization of Complex Systems with Opt-Quest*, A White Paper from OptTek Systems Inc., Boulder, CO, USA, 1997.

[8] M. A. Ornek, C. Ozturk, and I. Sugut, "Model-based heuristic for counter assignment problem with operational constrains: a case study," *Journal of Air Transport Management*, vol. 77, pp. 57–64, 2019.

[9] R. Stolletz, "Operational workforce planning for check-in counters at airports," *Transportation Research Part E: Logistics and Transportation Review*, vol. 46, no. 3, pp. 414–425, 2010.

[10] G. Bruno and A. Genovese, "A mathematical model for the optimization of the airport check-in service problem," *Electronic Notes in Discrete Mathematics*, vol. 36, pp. 703–710, 2010.

[11] A. T. Al-Sultan, "Optimization of airport check-in scheduling at passenger terminal," *International Journal of Applied Business and Economic Research*, vol. 14, no. 5, pp. 3233–3245, 2016.

[12] G. E. Araujo and H. M. Repolho, "Optimizing the airport check-in counter allocation problem," *Journal of Transport Literature*, vol. 9, no. 4, pp. 15–19, 2015.

[13] M. Mujica Mota and C. A. Zuniga, "Allocation of airportcheckn counters using a simulation optimization approach," In Simulation and Optimization: In Logistics, Industrial and Aeronautical Practice M. A. Mujica, I. De La Mota-Flores, and D. A. Guimarans-Serrano, Eds., pp. 203–229, Springer, Berlin, Germany, 2019.

[14] M. Parlar, B. Rodrigues, and M. Sharafali, "On the allocation of exclusive-use counters for airport check-in queues: static vs. dynamic policies," *OPSEARCH*, vol. 50, no. 3, pp. 433–453, 2013.

[15] M. Parlar, B. Rodrigues, and M. Sharafali, "Event-based allocation of airline check-in counters: a simple dynamic optimization method supported by empirical data," *International Transactions in Operational Research*, vol. 25, no. 5, pp. 1553–1582, 2016.

[16] M. Mujica Mota, "Check-in allocation improvements through the use of a simulation-optimization approach," *Transportation Research Part A: Policy and Practice*, vol. 77, pp. 320–335, 2015.

[17] R. K. Akamavi, E. Mohamed, K. Pellmann, and Y. Xu, "Key determinants of passenger loyalty in the low-cost airline business," *Tourism Management*, vol. 46, pp. 528–545, 2015.

[18] A. Suarez-Aleman and J. L. Jimenez, "Quality assessment of airport performance from the passengers' perspective," *Research in Transportation Business and Management*, vol. 20, pp. 13–19, 2016.

[19] W.-H. Tsai, W. Hsu, and W.-C. Chou, "A gap analysis model for improving airport service quality," *Total Quality Management and Business Excellence*, vol. 22, no. 10, pp. 1025–1040, 2011.

[20] Y. Ju, A. Wang, and H. Che, "Simulation and optimization for the airport passenger flow," in *Proceedings of the 2007 International Conference on Wireless Communications,*

Networking and Mobile Computing, IEEE Explore, Shanghai, China, September 2007.

[21] G. Bruno, A. Diglio, A. Genovese, and C. Piccolo, "A decision support system to improve performances of airport check-in services," *Soft Computing*, vol. 23, no. 9, pp. 2877–2886, 2019.

[22] Y. Cao, L. A. Nsakanda, and I. Pressman, "A simulation study of the passenger check-in system at the Ottawa international airport," in *Proceedings of the Summer Computer Simulation Conference*, Montreal, Canada, July 2003.

[23] H. Wang, K. Liu, Z. Qi, and J. Guo, "The establishment and optimization of airport security check process model," in *Proceedings of the 2017 32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 1026–1030, Hefei, China, May 2017.

[24] D. Wu, Z. Huimin, Y. Xinhua, X. Juxia, S. Meng, and L. Bo, "Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment," *Applied Soft Computing*, vol. 59, pp. 288–302, 2017.

[25] L. Adacher, "A global optimization approach to solve the traffic signal synchronization problem," *Procedia-Social and Behavioral Sciences*, vol. 54, no. 4, pp. 1270–1277, 2012.

[26] L. Adacher and C. G. Cassandras, "Lot size optimization in manufacturing systems: the surrogate method," *International Journal of Production Economics*, vol. 155, pp. 418–426, 2014.

[27] K. Gokbayrak and C. G. Cassandras, "A generalized surrogate problem methodology for on-line stochastic discrete optimization," *Journal of Optimization Theory and Applications*, vol. 114, no. 1, pp. 97–132, 2001.

[28] L. Adacher and M. Flamini, "Aircraft ground routing and scheduling optimization," in *Proceedings of the 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, pp. 345–350, Cambridge, UK, March 2014.

[29] L. Adacher, M. Flamini, M. Guaita, and E. Romano, "A model to optimize the airport terminal departure operations," *Transportation Research Procedia*, vol. 27, pp. 53–60, 2017.

[30] S. Janssen, A. Sharpanskykh, and R. Curran, "Agent-based modelling and analysis of security and efficiency in airport terminals," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 142–160, 2019.

[31] E. Romano, T. Murino, and L. C. Santillo, "A simulation decision support model for Check-in counters in an airport," in *International Conference on Harbour, Maritime and Multimodal Logistics Modelling and Simulation*, vol. 1, pp. 16–20, Tenerife, Spain, September 2009.