# Implementing Dynamic Air Transport Slot Trading Through Secure Auction Mechanism

Emre Koyuncu*, Baris Baspinar*, Guney Guner*, N. Kemal Ure*,
Massimiliano Zanin‡, Vaishali Mirchandani†, Emilio Álvarez Pereira†, Cengiz Paşaoğlu§ and Gokhan Inalhan*
*Istanbul Technical University, Istanbul, Turkey
‡Innaxis Foundation & Research Institute, Madrid, Spain
†Team & Cloud, Madrid, Spain
§General Directorate of State Airports Authority of Turkey (DHMI), Ankara, Turkey

***Abstract*—Future business cases based on air transportation market will need applying secure information sharing and calculation that allows untrusted parties to perform computation over a data set. In this work, we have structured a secure market mechanism that is specific to conceptual airport slot trading based on secure multi-party computation. Considering the needs of a secure information sharing and slot market, we have developed a web-based portal enabling participant to see the open auctions and put their bids. To demonstrate the feasibility of such mechanism, we have utilized historical data to create a realistic market structure and interests. In the simulations, primary and secondary markets including tactical slot trade have been simulated through the Secure Multiparty Computation (SMC). Moreover, we have also performed the cost-benefit analysis through the computational effort assessing of such SMC-based auction mechanism for the realistic operational environment, provided results of these analyze, and given a detailed discussion.**

## I. Introduction

Information sharing in air transportation systems is becoming a delicate issue, and there is a global interest in US and Europe to transform the current information handling into a highly efficient and secure system through SESAR in Europe, NextGen in the USA. The common idea behind of these programs is: efficiency can be improved only by ensuring a continuous flow of information between stakeholders involved in the operation. While some examples involving tactical information sharing such as flight intent exchange or price negotiations for slot exchange by airlines; some needs for strategical improvements, e.g. analysis of past incidents, thus of historical operational data.

To achieve this continuing flow of information involves two important aspects. First, most data in air transportation systems are considered confidential and sensitive and; hence, private - both for their commercial value and for the political or social consequences. Second, at the same time, data should be stored and processed in a safe and efficient way, which usually implies the use of a cloud-based infrastructure. This usually generates security and trust problems, as the exact location of data in the cloud is generally not known. SESAR's System-Wide Information Management (SWIM) [16], only partially tackle these problems where SWIM allows users only to access those sets of data included in their authorization class. As a result, the idea behind this paradigm is trust between the users and system managers.

Secure Multi-party Computation (SMC) is a set of techniques and algorithms that allow two or more untrusted parties to perform some kind of computation over a data set. The basic principle behind this is that the input information is divided into a number of shares, which are transmitted to different computation server. While each share is not enough to recover the initial information, protocols can be designed to perform operations on them; at the end, the result is collectively calculated by the computation servers, while no one of them has enough information on its own to recover any input. This allows once the computation is over, to recover the output of that computation, without any additional knowledge on the information provided by the parties. In other words, instead of providing any party with the full dataset (and thus creating a security issue to be managed) or denying the access to it (effectively blocking any possibility of using the data), the data owners could allow third parties to run computations on encrypted information, without real access to the full dataset. Secure computation has hitherto been used to solve several real-world problems, from secure sealed-bid auction [10], elections with an electronic voting scheme [21], benchmarking [6], up to defense applications in military operations [17].

Several kinds of research on slot allocation has been conducted focusing on marked based methods. A sealed-bid combinatorial action procedure is developed in [18]. Two different market mechanisms are defined: primary market and secondary market. Since primary market allocation does not make provision for slot demand dependence, the secondary market allows the airlines to exchange the slots taken from the primary market. In [8], three different market-based methods are suggested that are congestion pricing, auctions and secondary trading. However, real flight data and airport capacities are not utilized in the design processes of mechanisms in [8] and [18].

Another methodology, which is seen in [4] and [7], focuses on optimization based or algorithmic approaches. In [7], a slot allocation model is generated by considering the structure of the air traffic network as a mathematical programming problem, so interdependence of the slots at different airports is also considered. By using this methodology, allocation with

grandfather rights are compared with free allocation, and it is suggested that it is possible to remove grandfather rights without significantly penalizing airlines. Before [7], similar methodology is used in [4] where an integer programming model for large-scale instances of the air traffic flow management problem is presented. Further, a network-based air traffic model is developed in [11]. The values of landing slots at airports are estimated using this model. Furthermore, mathematical programmings for recovery problem is also introduced. An algorithm integrated with constraint programming optimization paradigm is presented in [2] to reschedule the flight plan using delays and swaps. In [1], several different meta-heuristic such as hill-climbing, simulated annealing, and genetic algorithm are used to solve the recovery problem.

In this work, we have structured a conceptual secure airport slot trading mechanism utilizing Secure Multi-party Computation (SMC). Such mechanism enables to stakeholders secure their sensitive commercial information, e.g. how much they are willing to pay. SMC based trading system does their calculations in several steps to protect essential part of the data coming from the participants. Considering the needs of a secure trading system, we have also developed a web-based slot trading portal enabling participants to see the open auctions and put their bids. In order to conceptually demonstrate the feasibility of such trade system, we have structured a hypothetic market and market interest models through the historical data analysis. We have also performed analyses by the means of computational effort assessing of SMC-based slot trading for the realistic operational environment. The results of these analyses are provided and discussed in details.

The paper is organized as follows: first, a brief explanation for Secure Multi-Party Computation is given in Section II. Then, slot market structure is introduced in Section III. Section IV presents data provisions, market interest models and example simulations for different market scenarios. Section V provides computational effort analysis for the SMC-based slot trade mechanism. Finally, Section VI presents conclusions and remarks.

## II. Brief Secure Multi-Party Computation

In the last decade, the increasingly use of cooperative computation, as well as the new ways of decentralized and distributed computing, *i.e.* peer to peer networks and cloud computing, has fostered the need of such technology, in order to solve problems in which many parties need to provide inputs for a computation, however, no mutual trust can be ensured. Some examples include secure decentralized elections [19], [9], secure auctions [3], secure benchmarking or retrieval of private information, *e.g.* biomedical records [14].

The evolution of cryptographic needs, from simple data security to identity verification, reached its last step in recent years, as some applications required combining data security with the possibility of executing calculations upon them. One example of such problem is the so-called *Yao's Millionaires' problem* [22]. Suppose two millionaires, Alice and Bob, who are interested in knowing which one of them is richer without revealing their actual wealth. More generally, this is equivalent to a problem of evaluating the inequality $a \geq b$ for two numbers $a$ and $b$, without revealing their actual values. To better introduce the reader to this research field, we propose a very simple SMC algorithm as an initial example, depicted in Fig. 1. Consider three people, *e.g.* Alice, Bob and Charlie, each holding a secret number (say $x_a$, $x_b$ and $x_c$). Due to confidentiality reasons, they cannot share these numbers with the other parties; nevertheless, they need to calculate their sum, that is $x = x_a + x_b + x_c$. The solution to this problem is the following. Firstly, Alice chooses a random number $r$ and privately sends $r + x_a$ to Bob. Afterwards, Bob adds his secret number and privately sends $r + x_a + x_b$ to Charlie. Finally, Charlie does the same with his personal number and sends $r + x_a + x_b + x_c$ back to Alice. At the end of this process, Alice can recall the random number $r$, subtract it from the received value $r + x_a + x_b + x_c$, and announce the result. Notice how none of them learns the input of the other parties: for instance, Alice's random number $r$ prevents Bob from knowing her private number.

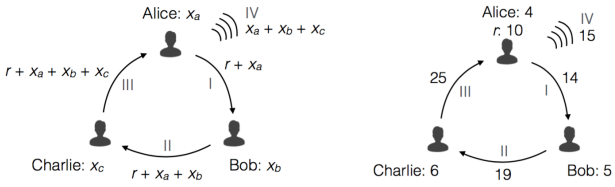


Fig. 1: Example of the secure calculation of a sum.

Different approaches, or *primitives*, have been used to implement SMC protocols for different applications. Independently on the problem to be solved, *e.g.* ranking, auction or set intersection problems, the protocol has to be constructed by means of a combination of these primitives, being therefore the building blocks of any SMC solution. The four combinations that have by and large been used in real-world applications are *Secret Sharing* [20], [5], *Oblivious Transfers* [19], *Garbled Circuits* and *Homomorphic Encryption* [13].

As its name suggests, Secret Sharing is a set of techniques aimed at distributing a secret, *i.e.* private information that should be concealed, among a group of participants, each one of them receiving just one piece of the secret. The secret can then be reconstructed only when a sufficient number of participants work collaboratively, as individual shares are of no use on their own.

The following section gives the slot market structures envisioned for the simulations.

## III. Slot Market Structure

Considering the slot market structure, it is supposed that all the participants in a slot trade auction are considered semi-honest parties (honest but curious) such that the participants surely follow the required protocol, and send well-formed messages, thus not affecting the outcome of the computation.

However, they might try to understand the market and learn the private information by examining all the data they can get.

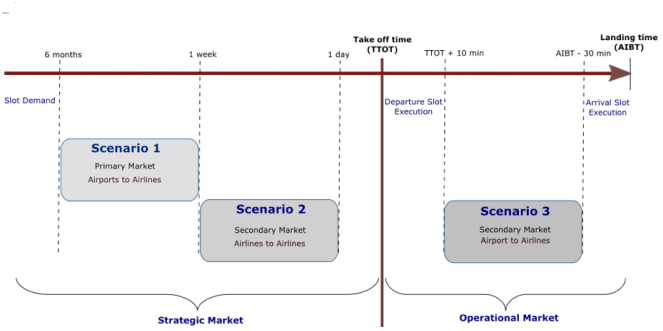


Fig. 2: Market structure through primary, secondary, strategic and tactical(operational) slot trading

In our demonstration, it is envisioned that the market structure involves different scenarios with different time-scales (i.e. in strategic and operational) and different type of markets (i.e. primary and secondary markets). Slots (i.e. airport slots), as a commodity, may be sold by airports, or by an airline owning them (due to the historical rights or previous trades). Moreover, slots may be sold for a whole time-window (e.g. season, months or days), corresponding to long-term and pre-planned auctions (strategic transactions), or just for one single operation in the case of dynamical landing queues (operational transactions). To provide a potential market prototype for the market structure, the following scenarios have been taken into account:

- Airlines try to buy slots or slot packages from the primary market, i.e. from airports.
- Airlines try to buy slots or slot packages from secondary market, i.e. from other airlines/
- An airline tries to buy a specialized priority approach from an airport.

To sum up, three scenarios are considered that they are using similar secure multi-party computation. Each scenario needs secure auction as the sensitive information sharing is essential. The Figure 2 demonstrates the timeline for the whole market.

Trade process, which requires secure information sharing and secure computation, begins when at least one auction is present in the market. An airport (i.e. scenarios 1 and 3) or an airline (i.e. scenario 2) willing to sell empty slots sends a request to the referee. The referee determines the start and end times of the auction, evaluates the applications of the sellers and creates an auction in the market. When an auction starts, participants send their bids by using the "Slot Trading Portal".

Three different outcomes may occur at the end of an auction. In the first outcome, no buyer fulfills the minimum asked price, as set by the seller, hence, there is no winner, and the relevant information is sent to all participants. If there is enough time left by the end of the auction, the seller may decide to put the slot in the market again and may also regulate its minimum selling price. Then a new auction begins, as the slot is available in the market again. In the second outcome, there are at least two potential winners with the same highest price exceeding the minimum selling price of the seller. In this case, the referee informs the participants and accepts new offers in the second round only from the matching bidders of the previous round. In the third outcome, there is only one winner that bids higher both than the other bidders and than the minimum selling price. Then the slot schedule of the airport for a given time window is updated with the new outcomes, and all stakeholders are conveniently informed.

Following section explains data set, market interest models and example simulation runs for each scenario.

## IV. Data Provision and Simulations for Slot Trading

The general architecture for the simulations for slot trading is composed of the Market Generation and Market Interest Generation. Figure 3 depicts this process cycle for the slot trade simulations.

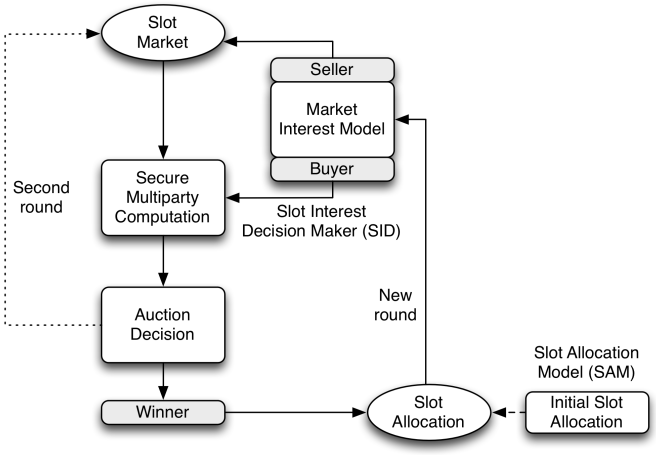


Fig. 3: General simulation process for slot trading based on secure multiparty computation (SMC)

The first market action in the timeline, Scenario 1, begins as the airport opens its slots to trade. The interest on the open slots in the primary market is based on the flight heritage and future vision of the airlines. To generate realistic simulation results, the Slot Demand Allocation Model (SAM) algorithm, which utilizes the historical flight data (i.e. ALLFT+) and airport capacity reports of EUROCONTROL, has been developed.

Let $\mu_a$ be an arrival service capacity, and $\mu_d$ be a departure service capacity of an airport per hour. Let $\lambda_a$ and $\lambda_d$ denotes the arrival demand and departure demand respectively. It is obvious that, whenever demand exceeds the service rate, such that $\lambda_a > \mu_a$ and $\lambda_d > \mu_d$, it generates a delay in the arrival and departure queues respectively.

At this point, a formal definition for the airport slot can be given as a right granted by an airport allowing the slot holder to schedule a landing or departure during a specific time period [12]. Throughout of this paper, we have accepted 15 minutes time windows as a single slot time for both departure and arrival in our calculations and simulations. Hence, the arrival and departure service rates per slot (i.e. per 15 minutes) for an airport will be $\sigma_a = \mu_a/4$ and $\sigma_d = \mu_d/4$ respectively, while demands will be $\lambda_a/4$ and $\lambda_d/4$. It is obvious that an airport can sell a departure slot $\sigma_d$ times, and an arrival slot $\sigma_a$ times, such that one airline might be interested in having it more than one times only for its different flights. The minimum price for a single slot mainly depends on the airport and slot time such that the price for the peak hours or midnight might be higher than the others. In that simulation model, non-flat rate implementation for the pricing depends on peak hours has been chosen.

The SAM algorithm initiates the Scenario 1 with an initial demand of the airlines, and Scenario 2 and Scenario 3 follows the timeline. Slot Interest Decision Model (SID) is an algorithm that generates a particular demand in the market for each scenario, including randomness, to initiate the simulation. Such demand is independent of the problem, and the demand profile can arbitrarily be placed without considering bottom layers of the simulation. Following subsections give descriptions for the different market scenarios.

### A. *Simulation Process for Scenario 1*

Scenario 1 focuses on the initial slot trade such that the airports sell their slots to the airlines. In that case, an airline might want to begin operating a new route, and to assure that it gets two slots, in the departure and the arrival airport respectively.

The price for a single slot mainly depends on the airport and the slot time. For instance, the price for the peak hours or midnight might be higher than the others. In the simulation model, a non-flat rate implementation for the pricing that depends on peak hours has been chosen. Figure 4 demonstrates the entire simulation model of the Scenario 1 where red frames indicate the seller (i.e. airports) and the other frames are buyers in the scenario.

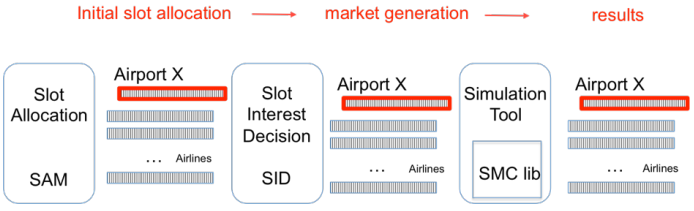


Fig. 4: Simulation process for Scenario 1

Scenario 1 begins with the Slot Demand Allocation Algorithm (SAM). SAM is a data-driven algorithm that utilizes historical flight data and airport capacity declaration data (from DDR2 data repository of EUROCONTROL). Specifically, the algorithm collects the flight history for scheduled flights for each airline and normalize their departure and arrival times to find their slot usage trend. These trends are evaluated from the real data that provides the scheduled arrival and departure time for each scheduled (periodically flying) flight. The algorithm also processes the capacity declarations to assess a service rate (capacity) of an airport for the each slot. It is seen that the capacity declarations are not always complete or reflects actual service capacities for each airport. Therefore, the algorithm also performs statistical parameter extraction to find the actual service rates from the real flight data. Once the service rates per slot (for both arrival and departure) are computed, the slot allocation is executed using first-in-first-out (FIFO) basis. The slot is assigned to actual of block time (AOBT) and actual in-block time (AIBT) of each flight for departure and arrival respectively.

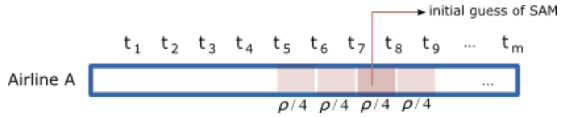


Fig. 5: Random slot demand selection of SID in Scenario 1 based on initial slot allocation

SAM Algorithm ensures that a slot is assigned for each flight of the airline depending on the airport capacity. Market competition begins when at least two airlines wish to get the same slot exceeding its capacity. The airlines typically compete at the peak hours mostly preferred by the passengers. As the SAM algorithm hypothetically allocates the best slots for each flight, it also eliminates the competition factor. To bring competition, which is natural in trade, a random factor to slot demand is added through Slot Interest Decision Maker (SID). Figure 5 demonstrates a hypothetic interest of an airline willing to bid. Specifically, the airline hypothetically may be interested in to compete for the previous two slots (early actions) or the next slot (late actions) of its initial guess. This approach is an arbitrary that one can come up with completely different policy. In this scenario, the SID algorithm chooses an interest profile in a probabilistic manner, which their likelihoods for four slots is equal. This pure randomness, which moves demand randomly to the nearby slots, readily generates the need for auctions.

Once the slot trade begins, the SMC Engine collects all the offers through the Slot Trade Portal (seen in Figure 6 and 7). Only a single auction is performed and the time window is closed. The secure computation starts after the auction is closed. Finally, the result (including the winning price as well as the winner) is disclosed to all participants. If the result is a tie between two or more participants, the referee provides a notification to the participants and creates a new secure auction between them. If the minimum price that the seller asks for has not been reached, participants have informed accordingly.

At each iteration of the slot trading process, both the seller and the buyer should put their price values in common, respectively the minimum price the seller would accept and the maximum price the buyer is willing to pay. According

to the main SMC objective, at the end of the computation, such information will not be available to any party, as each participant will only know his/her own input and the final result (win or not).

Start a New Auction

**List of Available Slots**

| ID | Slot | Seller | Apply | Result |
|---|---|---|---|---|
| 17 | LTBA 20:00-21:00 | peer 01 | End Auction | Computation Not started |
| 18 | LTBA 20:00-21:00 | peer 01 | End Auction | Computation Not started |
| 19 | LTAC 17:00-17:30 | peer 03 | End Auction | Computation Not started |
| 20 | LTAF 12:00-13:00 | peer 01 | End Auction | Computation Not started |
| 21 | LTBA 08:00-08:45 | peer 01 | End Auction | Computation Not started |
| 22 | LTAC 20:00-21:00 | peer 01 | End Auction | Computing |
| 23 | LTBA 23:00-23:30 | peer 01 | End Auction | Computation Not started |
| 38 | LTAI 09:00 - 10:00 | peer 01 | End Auction | [peer03, 320] |
| 39 | LTAC 05:20 - 06:30 | peer 01 | End Auction | Computation Not started |
| 40 | LTAC 14:00 - 15:00 | peer 02 | End Auction | [0] |

Fig. 6: Slot trade web portal prepared for the simulations – referee's auction list page

**Selected Slot**

| ID | Slot | Seller | Bid Price | Execute |
|---|---|---|---|---|
| 47 | LTAC 14:15 - 14:30 | peer 02 | 1500 | Send Price |

Fig. 7: Slot trade web portal prepared for the simulations – participant's bidding page

*Example auction:*
An example secure auction is given through the log file of the SMC Engine. The auction (for slot 43) is initiated by the seller airport LTBA. The minimum price set by LTBA is 32806. Throughout this paper, we have removed the units from the slot prices. Airlines DLH, THY and BAW joins the auction and offers randomly generated prices. As seen in the logs, BAW offers the highest price for the first location, which also exceeds the minimum slot price of the airport; hence, the location is allocated to BAW. This process is repeated for each location in the slot, thus making the number of auctions held for each slot equal to its number of locations. THY and DLH continue the auction for the second location and THY wins the auction. For the third location, THY continues to make offers. Based on the degree of interest in the slot, an airline might want to allocate more than one location for that particular slot. In this situation, the airline continues to make offers after acquiring a location in the slot. The process is repeated until all locations in the slot are allocated.

```
Referee created auction for slot-43 flight-0
Seller LTBA price: 32806. Seller sent price to slot-43 flight-0
DLH price: 31524. Participant sent price for DLH1299 to slot-43 flight-0
THY price: 32008. Participant sent price for THYE9K to slot-43 flight-0
BAW price: 36648. Participant sent price for BAW679 to slot-43 flight-0
Referee closed auction for slot-43 flight-0
Waiting for result...
Winner BAW-BAW679 for slot-43 flight-0
...
Referee created auction for slot-43 flight-1
Seller LTBA price: 32806. Seller sent price to slot-43 flight-1
DLH price: 35908. Participant sent price for DLH1299 to slot-43 flight-1
THY price: 36977. Participant sent price for THYE9K to slot-43 flight-1
Referee closed auction for slot-43 flight-1
Waiting for result...
Winner THY-THYE9K for slot-43 flight-1
...
Referee created auction for slot-43 flight-2
Seller LTBA price: 32806. Seller sent price to slot-43 flight-2
DLH price: 35020. Participant sent price for DLH1299 to slot-43 flight-2
THY price: 34830. Participant sent price for THYB3M to slot-43 flight-2
Referee closed auction for slot-43 flight-2
Waiting for result...
Winner DLH-DLH1299 for slot-43 flight-2
...
Referee created auction for slot-43 flight-3
Seller LTBA price: 32806. Seller sent price to slot-43 flight-3
THY price: 35827. Participant sent price for THYB3M to slot-43 flight-3
Referee closed auction for slot-43 flight-3
Waiting for result...
Winner THY-THYB3M for slot-43 flight-3
...
```

### *B. Simulation Process for Scenario 2*

In scenario 2, we have focused on the secondary and strategic market in which airlines trade their slot between them. This scenario allows airlines to benefit from their unused slots and creates an alternative business for them. For simulation purposes, this scenario immediately follows and uses the results of Scenario 1. Figure 8 depicts the simulation model of the Scenario 2 where red frames indicate the seller (i.e. airlines) and the other frames are buyers in the scenario.

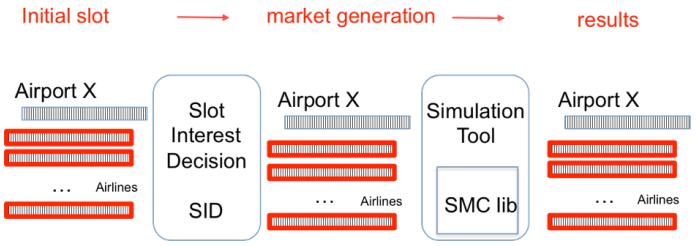


Fig. 8: Simulation process for Scenario 2

When the Scenario 1 ends, each airline will have hypothetical slots that they can use for their scheduled flights. In that case, the Slot Interest Decision Maker (SID), which builds a

hypothetic market and the traders with their interests, utilizes flight frequencies. The idea behind the conceptual model is that the airlines are interested in selling less frequently used slots of their scheduled flights, and buying the slots that match with their hypothetical interests. Specifically, similar to Scenario 1, an airline is assumed to be interested in two preceding and one succeeding available slots from the slots of their most frequent flights, as depicted in Figure 13, which indicates their hypothetical interests. Flight frequency rates are calculated from the historical flight data (i.e. ALLFT+ from the EUROCONTROL's DDR2 repository).

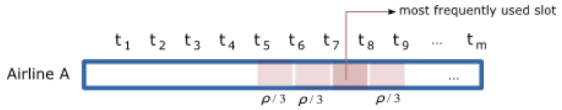


Fig. 9: Random slot demand selection of SID in Scenario 2 based on most frequently used slots

In this simulation, airlines consider to sell the slots that are used by flights once a week or less; on the other hand, airlines are interested in buying slots that are neighboring of their nearly everyday flights. This is the chosen behavior of the Slot Interest Decision Maker (SID) algorithm, which generates a potential market and buyers.

For this scenario, in the strategic secondary market, purchases and sales are planned for one week to one day in advance, and the decision to buy a slot is guided by mid-term market considerations. Therefore, one SMC Engine is sufficient, as there are no critical restrictions in the computation time. Similar to Scenario 1, daily departure slot schedules of the airlines have been chosen to perform the simulations. In this case, slot pricing for airline slot swapping is very much airline dependent. Each airline can indeed offer price over a pricing range as long as it is consistent with the mean base slot value. Each airline can do pricing scaling up and down on the mean base slot value. For the sake of simplicity, in the simulations for scenario 2, flat rate (i.e. mean pricing) is used, which means that airline put randomly generated values around a mean price, as it does not affect the calculations.

*Example auction:*
An example auction in Scenario 2 is given through the logs of SMC Engine. In this example, BAW tries to sell the location in slot 57 and THY and PGT compete to take this location. As the offered price exceeds the price set by BAW, THY takes this location.

```
Referee created auction for slot-57 flight-0
Seller BAW price: 63233. Seller sent price to slot-57 flight-0
THY price: 58442. Participant sent price for THYW4E to slot-57 flight-0
PGT price: 59970. Participant sent price for PGT2815 to slot-57 flight-0
Referee closed auction for slot-57 flight-0
Waiting for result...
There is no winner in auction, auction will be opened again...
...
Referee created auction for slot-57 flight-0
Seller BAW price: 63233. Seller sent price to slot-57 flight-0
THY price: 63590. Participant sent price for THYW4E to slot-57 flight-0
PGT price: 61260. Participant sent price for PGT2815 to slot-57 flight-0
Referee closed auction for slot-57 flight-0
Waiting for result...
Winner THY-THYW4E
...
```

### C. Simulation Process for Scenario 3

The third simulation scenario focuses on buying a priority landing slot from an airport during the approach. We suppose that the airport has an additional runway, which is open to trade. This particular runway can be utilized in normal operations; however, bidder customers should have priority utilizing this capacity as needed. Hence, the airport can economically benefit from that runway while airlines might have the opportunity to compensate for the losses associated with delayed flights. For this scenario, Slot Interest Decision Maker (SID), which utilizes ALLFT+ data, assesses the arrival delays for each landing aircraft. If the delay is higher than 30 minutes, the airline interests in buying a priority landing.

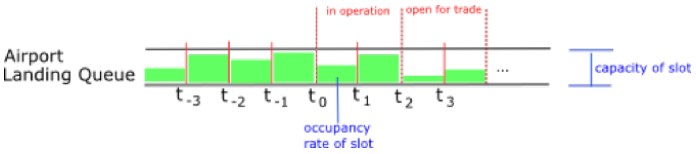


Fig. 10: Airport priority landing queue with sliding time window for Scenario 3

For this scenario, in the operational (tactical) secondary market, purchases are made 30 minutes before landing, and the decision to buy a priority landing is led by a real-time judgment of the Airline Operation Centers (AOC). The priority-landing queue is an infinite queue as its trade window shifts as time increases, in the other words, operates in an online manner. This decision is mainly governed by the induced delay in the current flight. Therefore, the decision must be assessed in a very short time, such that several SMC Engines might be running at the same time. For this scenario, SMC is used for the trade of priority landing to schedule the landing queue subject to trade. For simplification, flat-rate in pricing policy has been chosen, since it is independent of the calculations.

*Example auction:*
An example auction in scenario 3 is given through the logs of SMC Engine. In this example, an auction for slot 59 and slot 60 in LOWW is opened by the airport. In the simulation of

Scenario 3, the 10% of the capacity of each runway is defined as the dynamical landing queue service capacity, and the minimum price of this slot is set as three times the minimum price of the related slot. The capacity of LOWW is 11; hence, it has one priority landing service capacity in each slot if it is available. The airport starts the auction for slot 59 and 60. The referee sees that the slot 59 is full, and slot 60 has an empty location. For slot 60, AUA and NLY offer prices and NLY wins the auction.

Current Time: 052200
Auctions are opened for p59 & p60
---
Dynamic landing queue capacity for slot p59 is full
Referee created auction for slot p60 flight-0
Seller LOWW price: 15000. Seller sent price to slot p60 flight-0
AUA price: 10310. Participant sent price for AUA728D to slot p60 flight-0
NLY price: 15293. Participant sent price for NLY131E to slot p60 flight-0
Referee closed auction for slot p60 flight-0
Waiting for result...
Winner NLY-NLY131E
...

## V. Computational Cost

One of the mains challenges limiting the applicability of SMC to real-world problems is the large computation cost required to perform even simple analyses. Comparing two numbers using SMC requires multiple computational steps, from dividing the initial data into shares to manipulating them in separate servers. For instance, the computational cost of a protocol based on secret sharing scheme of $n$ players usually implies the creation of $n^2$ shares, representing a cost of operation of $O(n^2)$. The situation is even more complicated when non-linear operations are included in the mix, like comparisons and multiplications, which greatly increase the computational complexity and the evaluation cost. In order to assess the feasibility of an SMC paradigm, a set of simulations have been run, using the data models. The charts report the results of a set of velocity tests performed on the functional secure servers, as a function of the data input the number of clients (i.e. of participants).

Three distinctive metrics have been defined, as part of the total execution time of each analysis:

- *Computation cost (blue bars)*: Time required to create and manipulate the shares.
- *Communication cost (green bars)*: Time spent by the SMC servers to transmit information among themselves, as required to perform the secure computation.
- *Communication overhead (yellow bars)*: Any other time cost, including the initial setup of the system, authentication of the clients, network discovery, synchronization between servers, etc.

In the following charts, an average of the results for an increasing number of flights per airline (left) and a growing number of participating airlines (right) can be seen for analysis by route.

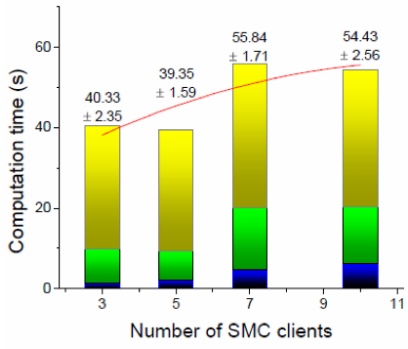


Fig. 11: Computational time as a function of the number of participant

Considering the results of computational time analysis, one can come up with following results:

- Using more computation servers increases the computation time of the SMC Libraries. This rise is not so large as to be a time-related problem, whereas it presents a clear security advantage: in order to decrypt the Secret Sharing Protocol, a harmful party needs to access all servers at one time.
- In a complete secure system it is important to have complete control over the execution times.

Overall, all secure computations can be executed in acceptable times, even when the number of participants increases beyond what initially estimated. Thus, the obtained results confirm the feasibility of SMC solutions in air transport environment as the total execution times are, in average, under the one-minute bar.

## VI. Conclusions and Remarks

In this paper, we have presented a conceptual secure system for slot trading mechanism utilizing Secure Multi-party Computation (SMC) library. Considering the needs of a reporting system, web-based slot trading portal has been developed enabling participants to see the open auctions and put their bids. In the simulation phase, we have utilized historical ALLFT+ data to create a realistic market structure and interests. In the simulations, primary and secondary markets including real-time (priority landing) for slot trade have been simulated through the Secure Multiparty Computation (SMC) library. Considering the needs of such trade, a Web-based Simulation Portal has been developed, enabling referee and participants to manage each type of trade operations, such as logging into the system, putting a slot into a market, bidding for slots, etc. Through the batch simulations, we have demonstrated the feasibility of such business for slot trade. In addition to secure information sharing through the SMC tools, structuring highly dynamic market adds benefits to the airlines, allowing to make their operations more flexible according to the current needs. For example, airlines can easily generate profits from their slots, in those cases where it would be unfeasible for operations. To assess the advantage

of this business model, one can run these simulations through the real commercial interests, which is not open to others. Nevertheless, the simulation results show that slot exchange rate could be relatively high even by governing market through the small random factors. Adding realistic business interests of the airlines would add additional dynamism in such slot trade. In such case, the pricing model, that in our simulations only depends on utilization and occupancy, would be highly complex.

Beyond the specific results here discussed, this contribution aimed at highlighting the necessity and feasibility of applying SMC techniques in AT and ATM. Any practitioner in air transport could easily identify a plethora of different scenarios in which private information cannot be shared, and yet some computation should be collaborative performed on it. One of the mains challenges limiting the applicability of SMC to real-world problems is the large computation cost required to perform even simple calculations. In this work, we have demonstrated that even with high numbers of participants, all computations here described can be carried out in less than one minute, well below the time constraints set by a slot trading problem.

While the use of SMC in air transport seems promising, there are still some issues and open problems that have to be tackled in future research activities. Two of them are of special relevance in the context here described. The first one is the kind of attackers the system is able to handle. The work described in this contribution is based on the assumption that parties are honest but curious: they will honestly collaborate in the computation, sending real data, but will try to deduce other parties inputs if the possibility arises. A different scenario may involve the presence of malicious parties, i.e. parties that actively try to break the system by any mean at their disposal. While algorithms and protocols are available to handle such situations, their computational cost is usually extremely high [15], [23]. The second problem is the integration of such computation paradigm into existing air traffic concepts, the most notable being SWIM [16]. Future research work should be devoted to understanding how both concepts could be integrated, thus effectively transforming SWIM into a platform to perform secure computations.

## ACKNOWLEDGMENTS

This work is co-financed by EUROCONTROL acting on behalf of the SESAR Joint Undertaking (SJU) and the EU as part of Work Package E in the SESAR Programme. Opinions expressed in this work reflect the authors' views only and EUROCONTROL and/or the SJU shall not be considered liable for them or for any use that may be made of the information contained herein.

## REFERENCES

[1] B. Aguiar, J. Torres, and A. J. Castro. Operational problems recovery in airlines–a specialized methodologies approach. In *Progress in Artificial Intelligence*, pages 83–97. Springer, 2011.

[2] P. Arias, D. Guimarans, and M. Mújica. A new methodology to solve the stochastic aircraft recovery problem using optimization and simulation. In *International Conference on Interdisciplinary Science for Innovative Air Traffic Management (ISIATM). Toulouse, France*, 2013.

[3] A. Ben-David, N. Nisan, and B. Pinkas. Fairplaymp: a system for secure multi-party computation. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 257–266. ACM, 2008.

[4] D. Bertsimas, G. Lulli, and A. Odoni. An integer optimization approach to large-scale air traffic flow management. *Operations Research*, 59(1):211–227, 2011.

[5] G. R. Blakley et al. Safeguarding cryptographic keys. In *Proceedings of the national computer conference*, volume 48, pages 313–317, 1979.

[6] D. Bogdanov, R. Talviste, and J. Willemson. Deploying secure multi-party computation for financial data analysis. In *Financial Cryptography and Data Security*, pages 57–64. Springer, 2012.

[7] L. Castelli, P. Pellegrini, and R. Pesenti. Airport slot allocation in europe: economic efficiency and fairness. *International journal of revenue management*, 6(1-2):28–44, 2011.

[8] D. Condorelli. Efficient and equitable airport slot allocation. *Rivista di politica economica*, 1:81–104, 2007.

[9] R. Cramer, R. Gennaro, and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. *European transactions on Telecommunications*, 8(5):481–490, 1997.

[10] I. Damgrd, M. Geisler, and M. Krigaard. Efficient and secure comparison for on-line auctions. In J. Pieprzyk, H. Ghodosi, and E. Dawson, editors, *Information Security and Privacy*, volume 4586 of *Lecture Notes in Computer Science*, pages 416–430. Springer Berlin Heidelberg, 2007.

[11] S. Hong and P. T. Harker. Air traffic network equilibrium: toward frequency, price and slot priority analysis. *Transportation Research Part B: Methodological*, 26(4):307–323, 1992.

[12] IATA. *IATA Worldwide Slot Guidelines*. International Air Transport Association, Montreal – Geneva, 19th edition, 2010.

[13] V. Kolesnikov, A.-R. Sadeghi, and T. Schneider. How to combine homomorphic encryption and garbled circuits. *Signal Processing in the Encrypted Domain*, pages 100–121, 2009.

[14] B. A. Malin, K. El Emam, and C. M. O'Keefe. Biomedical data privacy: problems, perspectives, and recent advances. *Journal of the American medical informatics association*, 20(1):2–6, 2013.

[15] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Bacşar, and J.-P. Hubaux. Game theory meets network security and privacy. *ACM Computing Surveys (CSUR)*, 45(3):25, 2013.

[16] J. S. Meserole and J. W. Moore. What is system wide information management (swim)? *Aerospace and Electronic Systems Magazine, IEEE*, 22(5):13–19, 2007.

[17] R. Pathak and S. Joshi. Secure multi-party computation protocol for defense applications in military operations using virtual cryptography. In *Contemporary Computing*, pages 389–399. Springer, 2009.

[18] S. J. Rassenti, V. L. Smith, and R. L. Bulfin. A combinatorial auction mechanism for airport time slot allocation. *The Bell Journal of Economics*, pages 402–417, 1982.

[19] K. Sako and J. Kilian. Secure voting using partially compatible homomorphisms. In *Advances in CryptologyCRYPTO94*, pages 411–424. Springer, 1994.

[20] A. Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

[21] S. Vaya. Realizing secure multiparty computation on incomplete networks. In *Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on*, pages 1–8, July 2010.

[22] A. C. Yao. Protocols for secure computations. In *Foundations of Computer Science, 1982. SFCS '08. 23rd Annual Symposium on*, pages 160–164, Nov 1982.

[23] G. Zhang, Y. Yang, X. Zhang, C. Liu, and J. Chen. Key research issues for privacy protection and preservation in cloud computing. In *Cloud and Green Computing (CGC), 2012 Second International Conference on*, pages 47–54. IEEE, 2012.