Probabilistic Formal Verification of the SATS Concept of Operation

Muhammad Usama Sardar[1], Nida Afaq[1], Khaza Anuarul Hoque[2], Taylor T. Johnson[2], and Osman Hasan[1]

[1]School of Electrical Engineering and Computer Science (SEECS)
National University of Sciences and Technology (NUST)
Islamabad, Pakistan
{usama.sardar,nida.afaq,osman.hasan}@seecs.nust.edu.pk
[2] Department of Computer Science and Engineering (CSE)
University of Texas at Arlington, USA
{khaza.hoque,taylor.johnson}@uta.edu

Abstract. The objective of NASA's Small Aircraft Transportation System (SATS) Concept of Operations (ConOps) is to facilitate High Volume Operation (HVO) of advanced small aircraft operating in non-towered non-radar airports. Given the safety-critical nature of SATS, its analysis accuracy is extremely important. However, the commonly used analysis techniques, like simulation and traditional model checking, do not ascertain a complete verification of SATS due to the wide range of possibilities involved in SATS or the inability to capture the randomized and unpredictable aspects of the SATS ConOps environment in their models. To overcome these limitations, we propose to formulate the SATS ConOps as a fully synchronous and probabilistic model, i.e., SATS-SMA, that supports simultaneously moving aircraft. The distinguishing features of our work include the preservation of safety of aircraft while improving throughput at the airport. Important insights related to take-off and landing operations during the Instrument Meteorological Conditions (IMC) are also presented.

Keywords: Formal Verification, Probabilistic Analysis, Model Checking, SATS, SATS Concept of Operations, Aircraft Safety, Aircraft Separation, Landing and Departure Operations.

1 Introduction

Small Aircraft Transportation System (SATS) [13], developed by NASA, provides access to more communities with less time delays by leveraging upon the recent advances in navigation and communication technologies. When a number of aircraft are in different parts of the airport, aircraft safety has to be ensured through timely separation and sequencing. Traditionally, non-towered non-radar airports rely on procedural separation during Instrument Meteorological Conditions (IMC), i.e., allowing only one aircraft to get access to the airport airspace at a given time, which significantly decreases the potential airport throughput

[23]. The main objective of SATS is to facilitate high volume operations (HVO) of advanced small aircraft at such airports with minimum infrastructure and low cost. Some representative SATS aircraft are Very Light Jet (VLJ) aircraft, an advanced technology Single-Engine (SE), piston-powered aircraft and an advanced technology Multi-Engine (ME), piston-powered aircraft [33].

Conventionally, SATS HVO simulations have been performed using computer programs in which aircraft modules were operated manually by pilots. These simulations develop the human-in-the-loop scenarios to check the effect of SATS procedures in the operational environment, on the pilot's responses in terms of work load and situational awareness [31,12,16,32]. In [12], off-nominal situations were also simulated, in addition to the nominal situations, to check the resulting effect on the pilot's state of mind. Proof-of-concept simulation studies were performed in the Air Traffic Control (ATC) simulation pilot lab at Federal Aviation Administration William J. Hughes Technical Center (FAATC) [30]. These simulations validated that the ATC can accept the SATS procedures, are able to control SATS traffic into and out of the Self Controlled Area (SCA), and support high volume operations. The simulations with pilots were used only for validation purposes and confirmed that SATS procedures are manageable by the airport management module (AMM). AMM's performance during high arrival rates of aircraft into the SCA has also been studied and found to have less delays as compared to one-in-one-out method [27]. Recently, an algorithm has been developed to optimize SATS landing sequence for multiple aircraft in [4], to make it conflict-free and with less delays, using Microsoft VC++ 6.0 simulation environment. However, these piloted simulation methods lack exhaustiveness [14] in terms of coverage of all the possible states as a rigorous piloted simulation of all possible scenarios requires a large number of tests, which in turn demands a significant amount of computational power and time. This leads to another major challenge of simulation-based verification of the SATS Concept of Operations (ConOps), i.e., selection of test vectors. A random selection of test vectors cannot offer a guarantee of correctness of the SATS ConOps since it might miss the meaningful portion of the design space. Moreover, it may not be possible to consider or even foresee all corner cases. Consequently, simulation-based verification of the SATS ConOps is incomplete with respect to error detection, i.e., all errors in a system cannot be guaranteed to be detected, which is a severe limitation considering the safety-critical nature of passenger aircraft.

In order to have a complete analysis, automatic parameterized verification of hybrid automata [20,19] was recently employed to verify properties of the SATS ConOps using model checking principles, while considering position of the aircraft as a continuous variable modeled either as a timer [19] or as a rectangular differential inclusion [20]. While this methodology allows for verification regardless of the number of aircraft, a limitation of this work is that the methodology requires the user to specify inductive invariants sufficient to establish safety. While the process of finding inductive invariants sufficient to establish safety of the SATS ConOps has been successfully automated through an extension of invisible invariants [3], this is an incomplete (heuristic) method that, in general,

may fail to find such inductive invariants [21]. The analysis and formal verification of the timing constraints of SATS was done in [10] using Linear Real-Time Logic (LRTL). The higher-order-logic theorem prover PVS [26] has also been used for the safety verification of the SATS ConOps [13,9,23,29]. In particular, it has been formally verified that SATS rules and procedures can provide minimum required spacing between two and more aircraft. A hybrid modeling technique was also developed in PVS using the PVS tool Besc [25].

In the above-mentioned methods of validation and verification of SATS, only the procedures and transition rules are considered. With these considerations, any model with appropriate conditions can validate that the procedures are enough for the assurance of safe separation between the aircraft. The missed approach transition is dependent on many random factors, for instance, low visibility. In conventional airports, it is mainly caused by the bad weather, increased air-borne traffic density, and ground traffic and its delays [15]. It is also required upon the execution of a rejected landing because of objects, such as men, equipment or animals, on the runway [1]. Due to such uncertainties involved, it is necessary to incorporate the probabilistic considerations of the system into the validation methods and safety verifications of SATS. Hence, we propose to use probabilistic model checking [5,11] for the verification of the SATS ConOps. This paper presents a fully synchronous Discrete-Time Markov Chain (DTMC) model of the SATS ConOps and the verification of the safety properties of SATS, including the landing and take-off procedures, using the probabilistic model checker PRISM [22]. PRISM has been extensively used to formally model and analyze a wide variety of systems, including communication and multimedia protocols, randomised distributed algorithms, security protocols, biological systems and many others, that exhibit random or probabilistic behaviour [2].

The rest of the paper is organized as follows: Section 2 describes the SATS operational concept to facilitate the understanding of the rest of the paper. Section 3 explains the main challenges that we faced in modeling the considered, fully synchronous, system in PRISM and the assumptions used in our DTMC model. In this section, our modeling methodology is also explained through discussion about each module, transition rules and procedures. Section 4 presents the probabilistic verification results of the SATS ConOps and the novel observations made. Finally, Section 5 concludes this paper by drawing conclusions and mentioning some directions of future work.

2 SATS ConOps

The ConOps for SATS is primarily a set of rules and procedures based on an area surrounding the airport, called the SCA, a centralized automated system, called the AMM, data communication between AMM and aircraft and state data broadcast from the aircraft [8,7]. The SCA is typically taken as a region with 12-15 nautical miles radius and 3000 feet above the ground [8,9]. It is arranged in a T structure, consisting of base, intermediate and final zones. It is divided into a number of segments and fixes which are the latitude/longitude points in space.

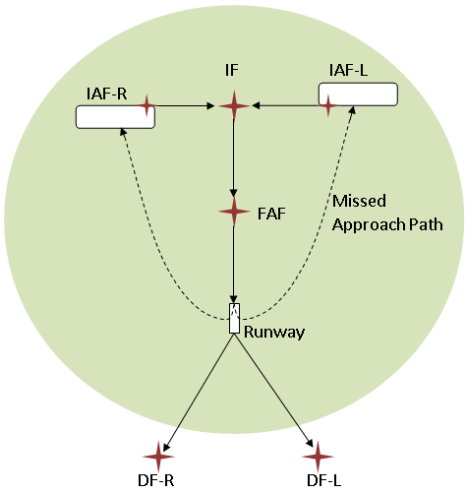

Fig. 1: Top view of the SCA [13]

The fixes are initial arrival fixes (IAFs), intermediate fix (IF), final approach fix (FAF) and departure fixes (DFs), as shown in Fig. 1. The IAFs serve two purposes, i.e., holding fix, when an aircraft enters the SCA, and missed approach holding fix (MAHF), which is required when an aircraft misses landing, and flies back to the IAF via missed approach path.

There are two types of entries into the SCA: vertical entry and lateral entry [9,25], as depicted in Fig. 2. Vertical entry is always made from the 3000 feet holding fix at the left (above IAF-L) or right (above IAF-R). Thereafter, the aircraft descends to the respective 2000 feet holding fix when it becomes available. Next, under certain conditions, the aircraft moves to the base segment (IAF to IF). On the other hand, in a lateral entry, the aircraft flies from the point of entry to the base segment directly or through the 2000 feet holding fix. Once the aircraft is in the base segment or 2000 feet holding fix, there is no dependency on its type of entry. After base segment, the aircraft goes through the IF, FAF, and finally reaches the runway. This procedure is primarily composed of a series of transitions through different segments of the SCA that are conducted by the aircraft if sufficient separation from the other aircraft is available and all conditions for the given transitions hold. If an aircraft misses its landing, due to any reason, it has to follow the missed approach path to move to the IAF corresponding to its MAHF assignment, as shown in Fig. 1.

The AMM has the responsibility to grant permissions to the aircraft for entering the SCA [7,31]. While granting the permission, the AMM assigns a landing sequence and a MAHF to the aircraft. These landing sequence numbers encode the leader information and also identify whether an aircraft is the first aircraft in a specific zone of SCA. The aircraft entering later thus follows the leader during the transitions. The MAHF assignment is in terms of 'side', which

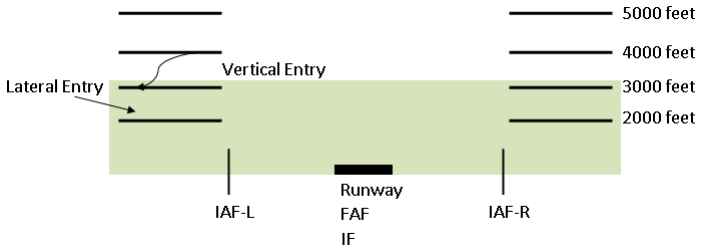

Fig. 2: Side view of the SCA [13]

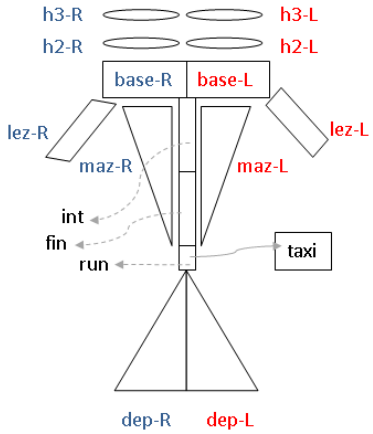

Fig. 3: Zones of the SCA [13]

can assume values of right or left. If the entering aircraft is the first one in sequence, then its MAHF will be in the same side from which it is entering. Whereas, the next aircraft, with sequence other than 1, will have the MAHF that is opposite to that of its leader.

Departure fixes are outside the SCA and under the ATC control. An aircraft ready to depart requests ATC for clearance. After clearance, the departure operation starts at the runway and it moves to the departure fix corresponding to its MAHF assignment. A safe distance of 10 or 5 nautical miles has to be maintained from the aircraft flying to the same or opposite departure fixes, respectively [13].

The SCA can be divided into different zones, illustrated in Fig. 3 and presented in Table 1. These zones represent the state of the aircraft. The complete information about the aircraft will thus include the sequence and MAHF assigned by AMM and the current location/zone of aircraft. The safety verification is based on the number of aircraft in a zone and their separation from other aircraft in other zones [23].

Table 1: Zones of SCA [13]

Zone	Symbol	Description
1	h3-R	Holding at 3000 feet at right side
2	h3-L	Holding at 3000 feet at left side
3	h2-R	Holding at 2000 feet at right side
4	h2-L	Holding at 2000 feet at left side
5	lez-R	Lateral entry zone at right side
6	lez-L	Lateral entry zone at left side
7	base-R	Right segment of base (IAF-R to IF)
8	base-L	Left segment of base (IAF-L to IF)
9	int	Intermediate segment (IF to FAF)
10	fin	Final segment (FAF to runway)
11	run	Runway
12	maz-R	Missed approach zone at right of base
13	maz-L	Missed approach zone at left of base
14	taxi	Taxi
15	dep-R	Right departure path towards right departure fix
16	dep-L	Departure path towards left departure fix

3 Formal Modeling of SATS as a DTMC in PRISM

In this section, we first describe our refinements to the SATS ConOps. Then the main challenges encountered in modeling the system in PRISM are presented. This is followed by the description of how these challenges were tackled in our model.

3.1 Refinements to original SATS

The proposed model of the SATS ConOps in the PRISM language overcomes some of the limitations of the non-deterministic, asynchronous transition system presented by Dowek et. al [13]. Before presenting the details of our model, we find it appropriate to point out the discrepancies in the existing algorithm and our proposed solution.

1. In a non-deterministic model, if two or more rules are enabled simultaneously, any one of them is allowed to be executed. In other words, only one non-deterministic action happens at a time. This means that in such a model, at each time step, *only one* aircraft will move to the next zone while all other aircraft hold in the same zone, even if the conditions are satisfied for all aircraft to move to their respective next zones. Thus, one aircraft could change zones several times while another remains idle [13]. Hence, such a model is unrealistic [23], as it fails to depict the real scenario.
2. The lowest available altitude determination (Rule 12) [13] is a simultaneous transition, potentially involving 2 aircraft, when the holding pattern at 3000 feet is occupied but 2000 feet is available. In this case, the transition

determines 3000 feet as the lowest available altitude and forces the aircraft holding at 3000 feet to descend to the holding pattern at 2000 feet. This is a weakness of the model because simultaneous transition is not possible in a fully non-deterministic model.

Our proposed solution for both the above limitations is to build a fully synchronous model that allows simultaneously moving aircraft. Hence, at each time step, all aircraft satisfying conditions to move to their respective next zones are allowed to proceed concurrently. Moreover, this model also facilitates the simultaneous transition in the lowest available altitude determination.

3.2 Modelling Challenges of SATS in PRISM

Parallel Composition of Modules
Parallel composition of modules in PRISM may seem to be the best option for developing the interleaved model of concurrency of aircraft in the SCA, where each module represents an aircraft. However, there are critical limitations in such a model, as discussed in Section 3.1. When multiple commands (belonging to any of the modules) are enabled at the same time, the choice between which command is executed by PRISM is *non-deterministic* in case of Markov decision process (MDP) and *probabilistic* in case of DTMC [2]. Specifically in the case of a DTMC, PRISM selects the command for execution uniformly at random. For instance, if there are 4 aircraft in the SCA and guards are satisfied for one command in each module, then there is a probability of 0.25 for each aircraft to move forward to the next zone. But only one of them is selected to move at a time.

Synchronization
PRISM supports synchronized transitions using synchronization labels. In this case, commands can be labelled with actions, which can be used to force two or more modules to make transitions simultaneously. By default, all modules are combined using the standard CSP parallel composition, i.e., modules synchronize over all their common actions [2]. However, in SATS application, the aircraft can be in any of the 16 zones and thus only a *specific scenario* can be modelled using synchronization labels. For instance, if there are two aircraft and the command for the first aircraft to be in the third zone is synchronized with the command for the second aircraft to be in the first zone, then they will make the transition simultaneously, if available, but it models a special case out of the many possibilities. They will no longer be synchronized in some future time step when the first aircraft is, for instance, in the seventh zone while the second aircraft is in the first zone.

Global variables with Synchronization
Global variables seem useful in modelling the state of the aircraft in the SCA as, unlike local variables, they are modifiable from any module. However, an

important restriction on the use of global variables in PRISM is the fact that global variables cannot be updated on a synchronized command [2]. PRISM detects this and reports an error if an attempt is made to do so.

Probabilistic Updates
In order to correctly model the semantics of the communication between aircraft and AMM, both aircraft and AMM should have separate modules in PRISM. Unfortunately, there is no direct way of changing a variable in a different module for *only one* probabilistic update of a command in the *same* time step. However, such probabilistic updates are frequently required. For instance, when an aircraft is in the final zone and it can move to the runway or missed approach path with certain probabilities. In case a pilot chooses the missed approach path, a new sequence number is to be assigned to the aircraft by the AMM while in case of transition to runway, there is no change in the sequence number. A possible solution could be to change the model such that the relevant variable is part of the same module as the probabilistic update but it will not represent the actual scenario of the communication between aircraft and the AMM.

Therefore, the challenge is to achieve a synchronization such that all aircraft move together whenever the guard conditions are satisfied, while incorporating probabilistic updates from the AMM in the model.

3.3 Modeling SATS in PRISM

In our formal model [28], we formulate the SATS ConOps as a DTMC in the PRISM model checker using an abstract timing model. Both sides of the approach are symmetric [13,29] and there can be at most two aircraft on each side of the SCA [13,23]. Therefore, we have assumed two aircraft in the right side of the SCA in this work for the purpose of simplicity. Our model ensures that after a landing aircraft has landed safely, it unloads passengers of the current flight in the taxi state. Then, it loads passengers of the next flight and is ready for departure. After departure, it reaches its destination and the next time it becomes a landing aircraft for the SCA. Hence, the process of landing and departure continues.

Model of Concurrency
In order to cope with the challenges, described in Section 3.2, we modeled the SATS ConOps as fully synchronously parallel automata, as in [17], where each transition is labeled with the same synchronization label, and therefore at each time step, at least one transition of each module is active. Hence, in such a fully synchronous model, both aircraft move concurrently to the next respective zones whenever the conditions are satisfied. In order to use the same synchronization label `t` with all commands in all modules, we ensure that *at least* one condition is true for each module for each reachable state in our model.

Model of SATS Transition Rules and Procedures
The modules `aircraft1` and `aircraft2` in our formal model [28], corresponding to each aircraft, implement the rules of ConOps, i.e., under what conditions the aircraft moves from one zone to the next. The modules are symmetric except that priority is assigned to `aircraft1` in case of simultaneous entry. Due to our proposed fully synchronous model, aircraft can enter inside the SCA individually or simultaneously with another aircraft. The state variables `zone1` and `zone2` represent the current zone of `aircraft1` and `aircraft2`, respectively. They are modelled as integer variables with values in the range 0 - 16, and the encoding is listed in Table 1. One additional zone is to be included into the model, which is the 'fly zone', for an aircraft outside the SCA. We encode it with a value of zero. In our model, we used formulas for compact representation of the conditions and to avoid repetition. For instance, `z1_total` represents the total number of aircraft in zone 1 and `z7_total_R` represents number of aircraft in zone 7 with an MAHF assignment of right, as shown in the following lines of the code in PRISM language:

$$formula\ z1\_total = (zone1 = 1?1 : 0) + (zone2 = 1?1 : 0);$$

$$formula\ z7\_total\_R = (zone1 = 7\ \&\ mahf1 = true?1 : 0) + (zone2 = 7\ \&\ mahf2 = true?1 : 0);$$

Model of the AMM
The AMM is the sequencer of the SCA. It typically resides at airport ground and communicates with the aircraft via a data link [8]. We model AMM as a separate module `AMM` in PRISM to represent this communication with the aircraft. It has two state variables, i.e., `seq` and `mahf` for each aircraft. For a landing aircraft, `seq` represents the relative landing sequence number, such that the aircraft with landing sequence n is the leader of the aircraft with landing sequence *n+1*, i.e., an aircraft with sequence number 1 is leader of the aircraft with sequence number 2. It is modelled as an integer variable with values in the range 0 - 10. When an aircraft enters the SCA, `seq` is assigned a new value calculated by the formula `nextseq`. This value is calculated based on the number of the aircraft already in the landing zones of the SCA. In case of simultaneous entry by both aircraft, different sequence numbers are assigned to both the aircraft, with priority to `aircraft1`. A new sequence number is also assigned when an aircraft initiates a missed approach path and the sequence numbers of all other aircraft in the landing zones of the SCA are decremented by one. Moreover, when an aircraft enters runway, the sequence numbers of all other aircraft in the SCA are again decremented by one. When an aircraft moves to the taxi state, its sequence number becomes 0. For a departing aircraft, `seq` represents the distance of the aircraft from runway in nautical miles. It is incremented by one in each time step when it is in one of the departure zones, until it becomes 10, where it is assumed to have left the SCA. The MAHF of an aircraft, represented by `mahf`, is a boolean variable with `true` representing right MAHF, and `false` representing left MAHF. It is assigned whenever an aircraft enters the SCA. Moreover, it

is re-assigned when an aircraft executes a missed path approach. We consider MAHF of only right side for simplicity of the model in this paper.

Timing Model
We use an abstract timing model in our formalization of the SATS ConOps. We assume that each aircraft stays in a zone for at least one time step. So, an aircraft must transition to the next zone after one time unit if the conditions for transition are satisfied. When the guard conditions are not fulfilled, it stays in the zone until the conditions become true.

Randomness in Model
Since there is no direct way of changing a variable in a different module for only one probabilistic update of a command in the *same* time step, we introduce an additional chooser module for each probabilistic decision. For instance, consider an aircraft in the final zone. Now it can either choose the missed approach path with a probability `p_map` or it can continue landing and transit to the runway with probability `1-p_map`. In case of the missed approach path, a new sequence number and MAHF is to be assigned to the aircraft. However, there is no change in its sequence number and MAHF if it proceeds to runway. We propose to use the chooser module, `choose_p_map` which contains a single state variable `p_map_state` of type integer and with two possible values: 0 and 1. When the probability `p_map` is selected, `p_map_state` is set to 1, otherwise it is 0. This is achieved by using the following command in PRISM:

$$[t]\ Guard \rightarrow p\_map : (p\_map\_state' = 1) + (1 - p\_map) : (p\_map\_state' = 0);$$

It is important to note that instead of setting `true` as a guard, we use the conditions of transition to final zone, i.e., one step back condition as the guard [28]. This way, the command does not execute on each time step. `p_map_state` is updated when the aircraft enters the final zone and is ready to be used when checking conditions for the next transition to runway or missed approach zone in the next time step.

The value of `p_map_state` is now used in such a way that the guard condition of `p_map_state=1` checks whether `p_map` is selected. For instance, in the `AMM` module, the following command ensures that `seq1` and `mahf1` are updated as soon as it makes the transition to zone 12:

$$[t]\ Guard\ \&\ p\_map\_state = 1 \rightarrow (seq1' = nextseq)\ \&\ (mahf1' = nextmahf1);$$

4 Verification Results

4.1 Safety Properties

Based on our model, explained in Section 3, safe separation is not maintained when two aircraft reside simultaneously in the specific zones. These zones include

the approach, final approach, missed approach, runway and departure zones. Hence, we label this state `danger` as follows:

$$\begin{aligned} label\ \text{“}danger\text{”} = ((zone1 = 7 \& zone2 = 7) \mid (zone1 = 9 \& zone2 = 9) \\ \mid (zone1 = 10 \& zone2 = 10) \mid (zone1 = 11 \& zone2 = 11) \\ \mid (zone1 = 12 \& zone2 = 12) \mid (zone1 = 15 \& zone2 = 15)); \end{aligned}$$

Safety in all Paths: $P =?\ [F\ \text{“}danger\text{”}]$;

We analyze safety in our model using the above property, which computes the value of the probability that `danger` is satisfied in the future by the paths from the initial state. PRISM shows a result of 0, which confirms that no path leads to a collision from the initial state.

Safety in all Reachable States: $filter\ (forall, P <= 0\ [F\ \text{“}danger\text{”}])$;

In order to confirm that the probability of occurrence of `danger` remains 0 for all *reachable* states, we formalize the property using filters as above. The property verifies to be true in PRISM and thus guarantees the safety in our model.

4.2 Analysis of Landing and Departure Operations

Expected Time for Landing: $R =?\ [F\ \text{“}landings1\text{”}]$;

We utilize the *reachability* reward [2] in PRISM to find the *expected* time taken for the landing of an aircraft in our model. In this case, a reward of unity is awarded to each state of the model and the rewards are accumulated along a path until a certain point is reached. We define this point as the state in which the aircraft is in the taxi state, for instance, for `aircraft1`:

$$label\ \text{“}landings1\text{”} = (zone1 = 14);$$

Since very limited information is available on the probability of executing a missed approach path `p_map` for SATS, we leverage upon the PRISM's parametric model checking functionality to perform the sensitivity analysis on the values of `p_map` from 0.001 to 0.9. The results are shown in Fig. 4, which depict the exponential increase in the expected time taken for landing with `p_map`. Since `aircraft1` is assigned priority in case of simultaneous entry, the values for this aircraft are slightly smaller as compared to those of `aircraft2`. The overall expected time for any aircraft to land is also shown.

Expected Number of Departures in a Fixed Time: $R =?\ [C\ <= T]$;

We leverage upon the *cumulative* reward properties [2] to find the *expected* number of departures of the aircraft in a fixed time in our model. In this case,

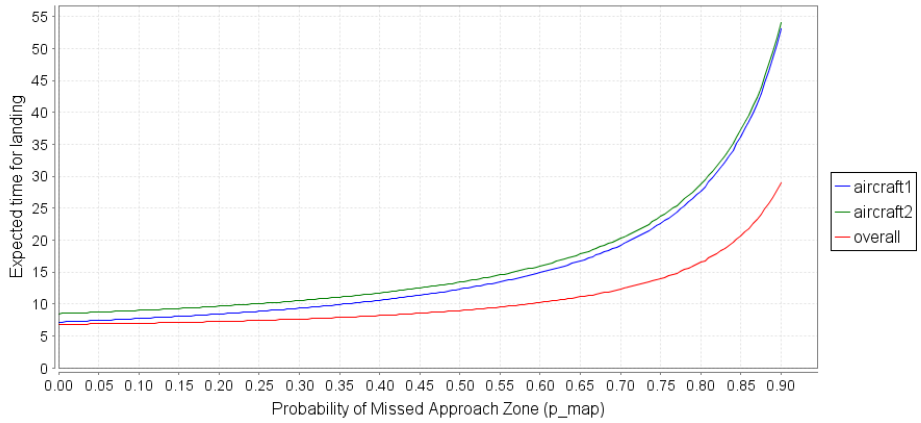

Fig. 4: Expected time for landing vs. Probability of the Missed Approach Zone

a reward of unity is awarded to each transition of departure and the rewards are accumulated until `T` time steps have elapsed. Fig. 5 shows the results of an experiment with `T` set to 10,00,000 which is large enough for the purpose of comparative analysis. Since `aircraft1` is assigned priority in case of simultaneous departure, the expected number of departures for this aircraft are slightly larger as compared to those of `aircraft2`.

Comparison of SATS and SATS-SMA: Reproduction of the corresponding non-deterministic model [13] in PRISM shows that the expected number of landing or departure operations are much greater in our proposed SATS-SMA than the corresponding non-deterministic model. For instance, with no aircraft executing a missed approach path, i.e., `p_map` of 0, the *expected* operations in the original non-deterministic asynchronous model and our refined SATS-SMA are 51280 and 81081, respectively, i.e., around 1.6 times greater throughput. The reason is that original SATS allows only one aircraft to move at a time while we allow all aircraft satisfying the conditions to move simultaneously to the respective next zones.

The key advantages of this work include the increase in the throughput, while maintaining aircraft safety, through simultaneous operations. The work also provides important quantitative landing and departure insights of the SATS ConOps. Our PRISM code and properties file is available for download [28], and thus can be benefited by researchers and verification engineers for further developments and analysis of the SATS ConOps.

5 Conclusion

Given the random and unpredictable nature of entry of aircraft into the SCA and transitions between the zones, we propose to use a probabilistic model checker,

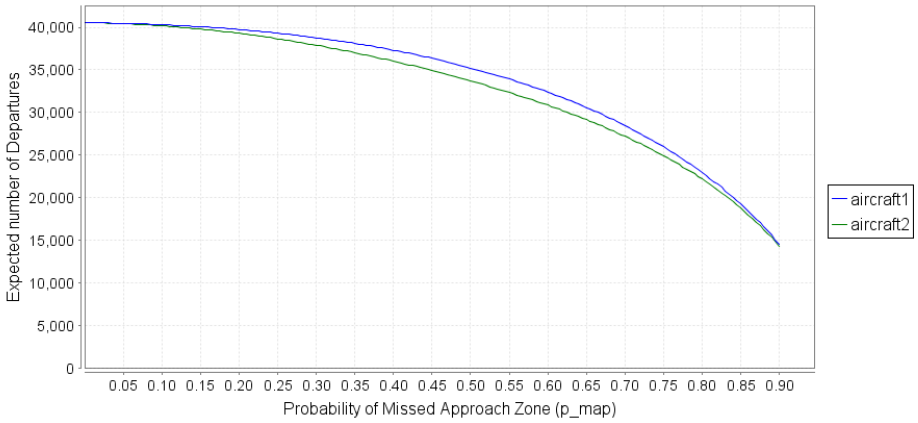

Fig. 5: Expected departures vs. Probability of the Missed Approach Transition

PRISM, to analyze the SATS ConOps in this paper. A fully synchronous DTMC model of SATS is proposed and is verified to increase the expected throughput of the airport as compared to the traditional non-deterministic, asynchronous model. Moreover, the successful modeling and verification of the transition procedures for two aircraft moving concurrently, has verified the safety of aircraft in terms of safe separation in all zones including take-off and landing. The landing and departure operations of SATS are analyzed with respect to the probability associated with the missed approach transition.

An important direction of future work is to improve the timing model by incorporating zone distances and abstract aircraft kinematics [25]. A more detailed analysis can be carried out by removing the simplifying assumptions of 2 aircraft and right side MAHF. Similarly, detailed comparison of non-SATS (one-in/one-out), SATS and SATS-SMA is an interesting direction for future research. Furthermore, we also plan to conduct the probabilistic analysis of the SATS ConOps under off-nominal conditions [24,6,12], such as equipment malfunction and emergency situations, using the parametric model checking functionality of PRISM, like it was utilized for the analysis of probability of missed approach in this paper. Moreover, Continuous-Time Markov Chains (CTMCs) of the SATS ConOps can also be developed to verify some time-related properties, where Erlang distribution can be used to model discrete time delays [18].

Acknowledgments. We would like to express our profound gratitude and heartfelt thanks to Dr. Cesar A. Munoz from NASA Langley Research Center for the valuable insights related to SATS and their model. We are also enormously pleased to precise our intense gratefulness and deepest gratitude to Dr. Matthias Gudemann for his helpful tips on modeling the system in PRISM. K. A. Hoque and T. T. Johnson are supported in part by the National Science Foundation (NSF) via grant number CNS 1464311, the Air Force Research

Laboratory (AFRL) via contract number FA8750-15-1-0105, and the Air Force Office of Scientific Research (AFOSR) via contract number FA9550-15-1-0258. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of AFRL, AFOSR, or NSF.

References

1. Instrument Procedures Handbook. U.S. Department of Transportation, Federal Aviation Administration (2015)
2. PRISM - Probabilistic Symbolic Model Checker. `http://www.prismmodelchecker.org` (2016)
3. Arons, T., Pnueli, A., Ruah, S., Xu, Y., Zuck, L.: Parameterized verification with automatically computed inductive assertions? In: Computer Aided Verification, vol. 2102, pp. 221–234. Springer (2001)
4. Bai, C., Zhang, X.: Aircraft landing scheduling in the small aircraft transportation system. In: International Conference on Computational and Information Sciences. pp. 1019–1022. IEEE (2011)
5. Baier, C., Katoen, J.P., et al.: Principles of model checking, vol. 26202649. MIT Press (2008)
6. Baxley, B., Williams, D., Consiglio, M., Adams, C., Abbott, T.: The small aircraft transportation system (SATS), higher volume operations (HVO) off-nominal operations. In: Aviation, Technology, Integration, and Operations Conference. American Institute of Aeronautics and Astronautics (2005)
7. Baxley, B., Williams, D., Consiglio, M., Adams, C., Abbott, T.: Small aircraft transportation system, higher volume operations concept and research summary. Journal of Aircraft 45(6), 1825–1834 (2008)
8. Carreño, V.: Concept for multiple operations at non-tower non-radar airports during instrument meteorological conditions. In: Digital Avionics Systems Conference. vol. 1, pp. 5.B.1–5.1–9. IEEE (2003)
9. Carreño, V., Muñoz, C.: Safety verification of the small aircraft transportation system concept of operations. In: Aviation, Technology, Integration, and Operations Conference. American Institute of Aeronautics and Astronautics (2005)
10. Cheng, A., Niktab, H., Walston, M.: Timing analysis of small aircraft transportation system (SATS). In: Conference on Embedded and Real-Time Computing Systems and Applications. pp. 58–67. IEEE (2012)
11. Clarke, Jr., E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press (1999)
12. Consiglio, M., Conway, S., Adams, C., Syed, H.: SATS HVO procedures for priority landings and mixed VFR/IFR operations. In: Digital Avionics Systems Conference. vol. 2, pp. 13.B.2–1–13.B.2–8. IEEE (2005)
13. Dowek, G., Munoz, C., Carreño, V.A.: Abstract model of the SATS concept of operations: Initial results and recommendations. Tech. Rep. NASA/TM-2004-213006, NASA Langley Research Center (2004)
14. Fedeli, A., Fummi, F., Pravadelli, G.: Properties incompleteness evaluation by functional verification. IEEE Transactions on Computers 56(4), 528–544 (2007)
15. Gariel, M., Spieser, K., Frazzoli, E.: On the statistics and predictability of go-arounds. In: Conference on Intelligent Data Understanding (2011)
16. Greco, A., Magyarits, S., Doucett, S.: Air traffic control studies of small aircraft transportation system operations. In: Digital Avionics Systems Conference. vol. 2, pp. 13.A.4–1–13.A.4–12. IEEE (2005)

17. Güdemann, M., Ortmeier, F.: A framework for qualitative and quantitative formal model-based safety analysis. In: Symposium on High-Assurance Systems Engineering. pp. 132–141. IEEE (2010)
18. Hoque, K.A., Mohamed, O.A., Savaria, Y.: Towards an accurate reliability, availability and maintainability analysis approach for satellite systems based on probabilistic model checking. In: Design, Automation Test in Europe Conference Exhibition. pp. 1635–1640. IEEE (2015)
19. Johnson, T.T., Mitra, S.: Parameterized verification of distributed cyber-physical systems: An aircraft landing protocol case study. In: International Conference on Cyber-Physical Systems. pp. 161–170. IEEE (2012)
20. Johnson, T.T., Mitra, S.: A small model theorem for rectangular hybrid automata networks. In: Joint International Conference on Formal Methods for Open Object-Based Distributed Systems and Formal Techniques for Networked and Distributed Systems. pp. 18–34. Springer (2012)
21. Johnson, T.T., Mitra, S.: Invariant synthesis for verification of parameterized cyber-physical systems with applications to aerospace systems. In: Infotech at Aerospace Conference. American Institute of Aeronautics and Astronautics (2013)
22. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: Verification of probabilistic real-time systems. In: Computer Aided Verification. vol. 6806, pp. 585–591. Springer (2011)
23. Muñoz, C., Dowek, G., Carreño, V.: Modeling and verification of an air traffic concept of operations. Software Engineering Notes 29(4), 175–182 (2004)
24. Muñoz, C., Carreño, V., Dowek, G.: Formal analysis of the operational concept for the small aircraft transportation system. In: Rigorous Development of Complex Fault-Tolerant Systems, vol. 4157, pp. 306–325. Springer (2006)
25. Muñoz, C., Dowek, G.: Hybrid verification of an air traffic operational concept. In: IEEE ISoLA Workshop on Leveraging Applications of Formal Methods, Verification, and Validation (2005)
26. Owre, S., Rushby, J.M., Shankar, N.: PVS: A prototype verification system. In: Conference on Automated Deduction, pp. 748–752. Springer (1992)
27. Peters, M.: Capacity analysis of the NASA Langley airport management module. In: Digital Avionics Systems Conference. vol. 1, pp. 4.D.6 – 41–12. IEEE (2005)
28. Sardar, M.U., Hoque, K.A.: Probabilistic formal verification of the SATS concept of operation. `http://save.seecs.nust.edu.pk/projects/SATS` (2016)
29. Umeno, S., Lynch, N.: Proving safety properties of an aircraft landing protocol using I/O automata and the PVS theorem prover: A case study. In: International Symposium on Formal Methods, pp. 64–80. Springer (2006), `http://dx.doi.org/10.1007/11813040_5`
30. Viken, S.A., Brooks, F.M.: Demonstration of four operating capabilities to enable a small aircraft transportation system. In: Digital Avionics Systems Conference. vol. 2, pp. 13.A.1–1–13.A.1–16. IEEE (2005)
31. Williams, D.M.: Point-to-point! validation of the small aircraft transportation system higher volume operations concept. In: International Congress of Aeronautical Sciences (2006)
32. Williams, D., Consiglio, M., Murdoch, J., Adams, C.: Flight technical error analysis of the SATS higher volume operations simulation and flight experiments. In: Digital Avionics Systems Conference. vol. 2, pp. 13.B.1–1–13.B.1–12. IEEE (2005)
33. Xu, Y., Baik, H., Trani, A.: A preliminary assessment of airport noise and emission impacts induced by small aircraft transportation system operations. In: Aviation Technology, Integration and Operations Conference. American Institute of Aeronautics and Astronautics (2006)