Modeling and Verification of an Air Traffic Concept of Operations

César A. Muñoz
National Institute of Aerospace
144 Research Drive
Hampton, VA 23666, USA
munoz@nianet.org

Gilles Dowek
Laboratoire d'Informatique
Ecole polytechnique
91128 Palaiseau, France

Víctor Carreño
NASA
Langley Research Center
Hampton, VA 23681, USA

ABSTRACT

A high level model of the concept of operations of NASA's Small Aircraft Transportation System for Higher Volume Operations (SATS-HVO) is presented. The model is a non-deterministic, asynchronous transition system. It provides a robust notion of safety that relies on the logic of the concept rather than on physical constraints such as aircraft performances. Several safety properties were established on this model. The modeling and verification effort resulted in the identification of 9 issues, including one major flaw, in the original concept. Ten recommendations were made to the SATS-HVO concept development working group. All the recommendations were accepted and incorporated into the current concept of operations. The model was written in PVS. The verification is performed using an explicit state exploration algorithm written and proven correct in PVS.

Categories and Subject Descriptors

F.4 [**Theory of Computation**]: Logics and Meanings of Programs; I.1 [**Computing Methodologies**]: Symbolic and Algebraic Manipulations

General Terms

Verification, Reliability

Keywords

Air traffic management systems, Theorem proving, Model checking

1. INTRODUCTION

The *Small Aircraft Transportation System (SATS)* program [9], led by NASA and in partnership with the Federal Aviation Administration, industry, and state and local aviation and airport authorities, aims to increase access to small and medium size airports. The great majority of these airports in the United States are underutilized for various reasons including limited use of general aviation aircraft, minimal or no commercial transport services, lack of facilities, etc. Airports lacking radar coverage and control tower facilities rely on procedural separation for access during Instrument Meteorological Conditions. Procedural separation uses a method of one-in/one-out. That is, only one aircraft is given access to the airport airspace at a given time. This method guarantees a highly safe airspace but it also results in a significant reduction in airport throughput. The objective of the SATS Higher Volume Operations concept (SATS-HVO) is to increase access and number of operations at these airports during Instrument Meteorological Conditions with a minimum of infrastructure and at a low cost, while maintaining the safety standard of the current system.

The SATS-HVO concept is a significant departure from typical Instrument Flight Rules operations where separation assurance services are provided by Air Traffic Control. In a SATS environment, pilots accept responsibility for separation inside a constrained airspace called the SCA (Self Controlled Area) (Figure 1). Separation inside the SCA is supported by operational rules, flight procedures, and on board navigation tools.

Aircraft separation represents a major safety concern for aviation regulatory agencies. Showing that the operational rules and flight procedures are safe is a top priority for the SATS-HVO development group. The task of verifying that the concept of operations is correct, that is, aircraft flying nominal SATS-HVO scenarios are safely separated, will be accomplished by formal mathematical analysis. This paper

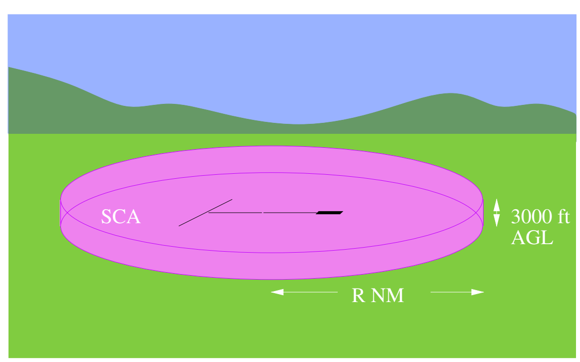

Figure 1: Self Controlled Area (SCA)

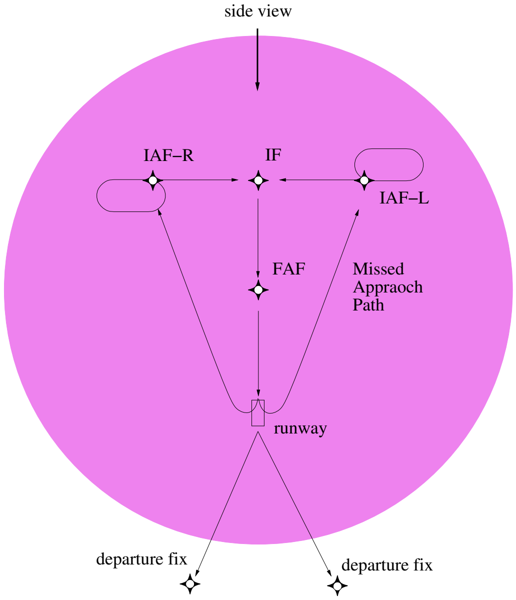

Figure 2: Top view of SCA

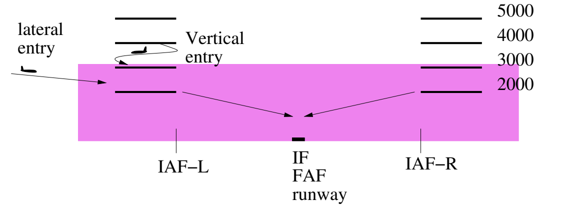

Figure 3: Side view of SCA

reports on the first findings of this formal modeling and verification effort.

The rest of the paper is organized as follows. The next section briefly describes the SATS-HVO concept of operations as it is defined in [1]. An abstract mathematical model of the concept, its properties, and its verification are described in Section 3. Section 4 discusses modeling issues and limitations. Section 5 summarizes this work and presents areas of future research. For quick reference, the appendix lists all the acronyms used in this paper.

2. SATS-HVO CONCEPT OF OPERATIONS

The concept of operations is a collection of rules and procedures which, when followed, will support separation assurance during transition to the SCA, approach, missed approach, landing, takeoff, departure, and transition out of the SCA. On board navigation tools will provide advisories to aid pilots in following these procedures. The logical components of the concept are the Self Controlled Area (SCA) and the Airport Management Module (AMM).

2.1 The Self Controlled Area (SCA)

The SCA is an airspace volume surrounding the airport facility. The design of the SCA is similar to a GPS T approach [7], were pilots are required to fly by latitude/longitude points in the space, called *fixes*, in order to perform a landing approach or a departure. In a T approach, the fixes are geographically arranged as a T. Some of these fixes are *holding fixes*. Under particular circumstances, an aircraft is allowed to fly around a holding fix waiting for another aircraft to go first in a landing approach. A *missed approach holding fix* is a holding fix to which an aircraft will proceed in case it executes a missed approach.

Figures 2 and 3 show top and side views, respectively, of a nominal SCA design. The top view shows the fixes and segments of the arrival and departure paths. The fixes are the right and left initial arrival fixes (IAF-R, IAF-L), intermediate fix (IF), final approach fix (FAF), and right and left departure fixes (DF-R, DF-L).[1] The side view shows the holding altitudes above the initial arrival fixes. The holding fixes at 2000 and 3000 feet above ground level are inside the SCA. Holding fixes at 4000 and above are outside the SCA. Right and left initial approach fixes also serve as right and left missed approach holding fixes (MAHF-R, MAHF-L), respectively.

There are two types of entry into the SCA: *vertical entry* and *lateral entry*. In a vertical entry, an aircraft flies to the IAF at an altitude above the SCA. The aircraft holds at the IAF above the SCA until entry is granted by the AMM. The aircraft then descends over the IAF flying a race track trajectory through 4000 to the 3000 feet holding fix. A lateral entry is possible when there are no aircraft at or assigned to the IAF. In this case, the aircraft proceeds to the IAF in a flight trajectory to arrive at the IAF at or above 2000 feet. When an entry is granted by the AMM, the aircraft receives a *follow notification* and a *missed approach holding fix assignment*. The follow notification is either *none*, if it is the first aircraft in the landing sequence, or the identification of a *lead* aircraft. An aircraft should proceed from the IAF to the IF, and from there to the FAF and, finally, to the runway threshold, soon after some spacing criteria with respect to the lead aircraft are satisfied. In case of a missed approach, the aircraft flies to its assigned missed approach holding fix at the lowest available altitude (2000 or 3000 feet). Then, it re-initiates the approach and either follows a normal landing procedure or leaves the SCA.

Departure fixes are outside the SCA. Hence, prior to a departure, aircraft must request clearance to Air Traffic Control. After clearance is granted and the aircraft is ready for departure, the departing aircraft monitors the arrival stream for a departure slot. In case of multiple departure operations, a separation of 10 nautical miles is required for aircraft flying to the same departure fix. For aircraft flying to opposite departure fixes, a minimum separation of 3 nautical miles is required. On board navigation tools will assist the pilot in identifying a departure slot.

2.2 The Airport Management Module (AMM)

The AMM is an automated system which will typically reside at the airport grounds. It serves as an arbiter and sequencer of the SCA. It receives state information from aircraft in the vicinity of the airport and communicates with aircraft via data link. The AMM is not intended to replace traditional air traffic control services. It minimally supports

[1]As it is usually depicted, right and left are relative to the pilot facing the runway, i.e., opposite from the reader point of view.

flight operations by implementing the entry rules, providing follow notifications, and assigning missed approach holding fixes.

AMM rules form vertical entries and reassignments after a missed approach will serve to illustrate the formal model presented in the next section. They are described in [1] as follows.

- **Vertical entries**: "*The AMM rules that determine if a normal (vertical) entry into the SCA is permitted are: (1) There are less than 2 aircraft either at that fix or assigned to the fix, (i.e., as a missed approach holding fix), and (2) no aircraft assigned to that fix as a missed approach holding fix on the approach*". Moreover, "*Alternating missed approach holding fixes are given (by the AMM) to sequential aircraft*".
- **Reassignments after a missed approach**: "*... once the aircraft gets within the proximity of the MAHF, the aircraft is reassigned (by the AMM) for another approach*".

3. ABSTRACT MODEL

The high level model of the SATS-HVO concept of operations is a transition system that captures nominal flight scenarios inside the SCA. The model was written in the Prototype Verification System (PVS) [10]. It was verified using an explicit exploration algorithm written and proven correct in PVS. A representative set of elements of the model is informally described in this paper. A detailed description of the model, including PVS sources, has been published as a NASA technical report [6].

Roughly speaking, the state of the system is composed by the state of the AMM and the state of the SCA, which includes the aircraft states. Non-deterministic and asynchronous transitions represent nominal procedures such as holding, approach initiation, missed approach, landing, and take-off.

3.1 Set of States

Entries into the SCA are granted in the order they are requested. Therefore, follow notifications provided by the AMM can be modeled as landing sequences assigned to arriving aircraft: 1 to the *first* aircraft, 2 to the aircraft following 1, and so forth. Landing sequences vary over time. When an aircraft lands or initiates a missed approach procedure, landing sequences change such that the aircraft with landing sequence 2 becomes the first aircraft, and the aircraft with landing sequence $n+1$ gets the landing sequence n. Landing sequences are artifacts of the abstract model rather than an integral part of the SATS-HVO concept. Knowing each aircraft follow notification is sufficient to describe aircraft landing sequences and vice-versa: the aircraft assigned to the landing sequence n is the *leader* of the aircraft assigned to $n+1$. Moreover, the aircraft assigned to 1 follows *none*.

The state of the AMM is represented by the next available landing sequence and the next alternating missed approach holding fix. Landing sequences are natural numbers (starting from 1), missed approach holding fixes are either `right` or `left`.

To discretize the position of an aircraft, the SCA is logically divided into 15 zones (Figure 4):

- `holding3(right)` and `holding3(left)`: Holding patterns at 3000 feet, right and left, respectively.

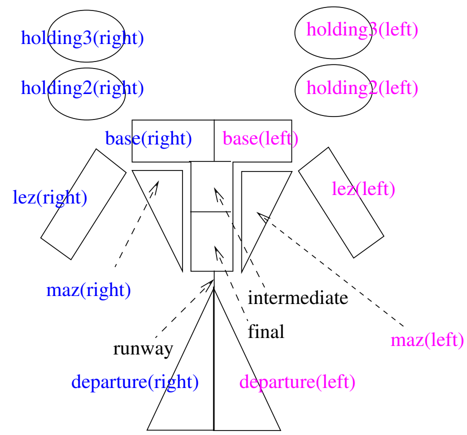

Figure 4: SCA zones

- `holding2(right)` and `holding2(left)`: Holding patterns at 2000 feet, right and left, respectively.
- `lez(right)` and `lez(left)`: Lateral entry zones, right and left, respectively.
- `base(right)` and `base(left)`: Base segments, right and left, respectively.
- `intermediate`, `final`, and `runway`: Intermediate segment, final segment, and runway.
- `maz(right)` and `maz(left)`: Missed approach zones, right and left, respectively.
- `departure(right)` and `departure(left)`: Departure zones, right and left, respectively.

Geographically, these zones are not disjoint. Indeed, lateral entry and missed approach zones may overlap. One of the goals of this verification effort is to show that the concept of operations prevent hazardous situations such as one aircraft on a missed approach operation at a given IAF while another aircraft is flying a lateral entry procedure at the same fix.

The state of each one of these zones is described by a list of aircraft states. The state of an aircraft is a record with 2 fields: landing sequence and MAHF assignment. Aircraft identifications are implicit in this model. The lists of aircraft states define time/space relations between aircraft. In particular, the order of aircraft in a list is the order of arrival to the zone.

For departure operations, the MAHF field of the aircraft state encodes the departure fix. Moreover, the landing sequence encodes a discrete separation with the previous departing aircraft (or *none* if it is 0).

Due to the use of natural numbers and unbounded lists, the number of states in this model is potentially infinite.

3.2 Transitions

The dynamics of the SCA environment is modeled as a set of non-deterministic asynchronous transitions over the global state of the SCA. In this transition system, zones behave as first-in first-out data structures: aircraft are removed from the head of one zone and added to the tail of the next zone. Among other things, the transitions guarantee that an aircraft is in at most one zone at a time, it does not overtake its leader, and it orderly goes trough the zones `holding3`, `holding2`, `base`,`intermediate`, and `final`.

Twenty four transitions were identified. Each one of them corresponds to a well-defined phase of an arrival or departure procedure:

- Vertical entry (right, left).
- Lateral entry (right, left).
- Descend from 3000 to 2000 feet (right, left).
- Approach initiation for vertical entry (right, left).
- Approach initiation for lateral entry (right, left).
- Transition from base segment to intermediate segment (right, left).
- Transition from intermediate segment to final segment.
- Landing.
- Taxiing.
- Missed approach initiation.
- Determination of lowest available altitude (right, left).
- Emergency departure from SCA.
- Departure initiation (right, left).
- Takeoff.
- Departing from SCA (right,left).

Transitions for vertical entry and missed approach initiation are given below. These transitions are rigorous descriptions of the corresponding AMM rules presented in Section 2.2.

- **Vertical entry**. For *side* $\in \{\texttt{right}, \texttt{left}\}$, a vertical entry transition may take place at the *side* IAF, only if all the following conditions hold:

  1. $|\texttt{holding3(}side\texttt{)}| + |\texttt{holding2(}side\texttt{)}| + |\texttt{maz(}side\texttt{)}| + |\texttt{lez(}side\texttt{)}| + r < 2$, where r is the number of aircraft in the opposite zones assigned to the *side* MAHF.
  2. No aircraft assigned to the *side* MAHF on `base`, `intermediate`, or `final`.
  3. No aircraft on $\texttt{maz(}side\texttt{)}$, $\texttt{lez(}side\texttt{)}$, or $\texttt{holding3(}side\texttt{)}$.

  If the transition takes place, an aircraft is added to the tail of $\texttt{holding3(}side\texttt{)}$. It gets the next landing sequence from the AMM state. If the new aircraft is *first*, it is assigned to the *side* MAHF. Otherwise, it is assigned to the next alternating missed approach fix. The state of the AMM is updated accordingly.

- **Missed approach initiation**. A missed approach initiation transition may take place only if there is an aircraft on `final`. In the new state of the SCA, an aircraft is removed from the head of `final` and added to the tail of $\texttt{maz(}side\texttt{)}$, where *side* is the MAHF assignment of the aircraft. The aircraft gets the next landing sequence from the AMM state. If it becomes the *first* aircraft, it keeps its MAHF assignment. Otherwise, it is reassigned to the next alternating missed approach fix. The state of the AMM and the landing sequence of the remaining aircraft are updated accordingly.

A curious reader may have noticed that the condition (3) of the vertical entry transition does not appear in the vertical entry rule of the SATS-HVO concept (Section 2.2). Moreover, the MAHF reassignment in the missed approach initiation transition occurs immediately after an aircraft goes miss rather than in the vicinity of the MAHF as required by the reassignment rule of Section 2.2. These and other modifications were introduced during the modeling and verification process. They were in total 10 recommendations, all of which were carefully reviewed by the SATS-HVO concept development working group and considered for a revised version of the concept. Some of the recommendations where implicit omissions in the specification while others where more serious in nature. For example, condition (3) of the vertical entry transition was omitted in the rule but it was implicitly assumed in the document. On the other hand, the original MAHF reassignment rule allowed for scenarios which were intended to be precluded by the operational concept. The proposed modification in the missed approach initiation transition corrects the problem but yields a major logical change of the concept that is explained in detail in [6].

3.3 Properties

An important design hypothesis of the SATS-HVO concept is that there is always an altitude available at a missed approach holding fix for an aircraft on the arrival approach. Since there are only two MAHFs and two possible altitudes (2000 and 3000 feet), the SATS-HVO concept shall satisfy an upper bound of four simultaneous arrival operations. This property and several other safety properties were identified and verified in the abstract model. For instance, at any time and for *side* $\in \{\texttt{right}, \texttt{left}\}$:

- There are no more than two aircraft assigned to the *side* MAHF.
- The number of aircraft on *side* is at most 2, i.e., $|\texttt{holding3(}side\texttt{)}| + |\texttt{holding2(}side\texttt{)}| + |\texttt{maz(}side\texttt{)}| + |\texttt{lez(}side\texttt{)}| \leq 2$.
- There is at most one aircraft on $\texttt{holding3(}side\texttt{)}$ and at most one aircraft on $\texttt{holding2(}side\texttt{)}$.
- There are no more than 2 aircraft on $\texttt{maz(}side\texttt{)}$.
- If there is an aircraft in $\texttt{lez(}side\texttt{)}$, then $\texttt{holding3(}side\texttt{)}$, $\texttt{holding2(}side\texttt{)}$, and $\texttt{maz(}side\texttt{)}$ are empty.

Furthermore:

- The leader of an aircraft on base is either on the final approach or the first aircraft on the opposite base segment.
- Aircraft land in order according to the landing sequences.
- There is at most one aircraft on the runway at any time.
- Consecutive departure operations are separated.

- Aircraft eventually land or depart the SCA.
- There are no operational deadlocks.

3.4 Verification

The transition system that models the SATS-HVO concept of operation was written in PVS [10]. The PVS model is described in detail in [6]. The SCA state is a record type containing the zones (`holding3`, `holding2`, `lez`, `maz`, `base`, `departure`, `intermediate`, `final`, and `runway`), the next missed approach holding fix assignment (`nextmahf`) and the next landing sequence (`nextseq`).

```
SCA : TYPE = [#
  holding3,     % Holding Pattern 3kft
  holding2,     % Holding Pattern 2kft
  lez,          % Lateral Entry Zone
  maz,          % Missed Approach Zone
  base,         % Base segment
  departure    :% Departure zone
  [Side→Zone],
  intermediate,% Intermediate segment
  final,        % Final segment
  runway       :% Runway
  Zone,
  nextmahf     : Side, % Next missed approach holding fix
  nextseq      : nat,  % Next sequence number
#]
```

In this model, `Side` is the enumeration type {`right`, `left`} and `Zone` is a list of aircraft state, where an aircraft state is defined by the record type:

```
Aircraft : TYPE = [#
  seq : nat, % Sequence number
  mahf: Side % Missed approach holding fix assignment.
#]
```

The model fully exploits the symmetry of right and left sides of the SCA by parameterizing the zones `holding3`, `holding2`, `lez`, `maz`, `base` and `departure` with the side. For instance, the missed approach zone at the right is written `maz(right)`.

In total, there are 17 state variables in this PVS model:

- 15 unbounded lists of aircraft states, each one of the aircraft states composed of a natural number (landing sequence) and a 2-valued variable (missed approach holding fix assignment).
- 2 state variables of the AMM, i.e., a natural number (next landing sequence) and a 2-valued variable (next alternating missed approach holding fix).

Non-deterministic transition rules are modeled as functions from `SCA` to list of `SCA`. For instance,

```
VerticalEntry(side:Side)(state:SCA):list[SCA] = ...
LateralEntry(side:Side)(state:SCA):list[SCA] = ...
...
Landing(state:SCA):list[SCA] = ...
Taxiing(state:SCA):list[SCA] = ...
```

As in the case of the symmetric zones, symmetric transitions are parameterized by the side, e.g., the right vertical entry transition is written `VerticalEntry(right)`.

The global SCA transition, called `Next`, is the asynchronous composition of all the transition rules:

```
Next(state:SCA):list[SCA] =
  append(VerticalEntry(right)(state),
  append(VerticalEntry(left)(state),
  append(LateralEntry(right)(state),
  append(LateralEntry(left)(state),
  ...
  append(Landing(state),Taxiing(state))))))
```

Safety properties are specified as predicates over `SCA`. For instance, the property that states that there are at most 4 arrival aircraft is written:

```
four_arrivals(state:SCA):bool =
  total_arrivals(state) ≤ 4
```

The safety invariant of the SCA is the conjunction of all the safety properties described in Section 3.3.

```
Invariant(state):bool =
  four_arrivals(state) AND
  well_assigned(state) AND
  ...
  non_incursion(state)
```

As noted before, the set of states is potentially infinite. However, using a depth-first exploration algorithm, written in PVS, it was discovered that, from an empty configuration of the SCA, just 2811 states were reachable. The safety invariant was verified on all of these states.

The depth-first exploration algorithm was proven correct in PVS. It is a generic algorithm that inputs a state type `State`, a set of initial states `init`, a transition relation `next` (expressed as a function from `State` to list of `State`), a safety property `prop?`, and a maximum number of states to be explored `k`. The algorithm outputs:

- `explored`: a list of reachable states that satisfy `prop?`.
- `counterex`: a list of counterexamples.
- `deadlocks`: a list of deadlocks.
- `unexplored`: a list of unexplored nodes.

It has been formally proven in PVS that if `unexplored` and `counterex` are both null, then `explored` is exactly the set of reachable states from `init` for the transition relation `next`. Furthermore, if `counterex` is not null, it contains a reachable state that does not satisfy `prop?`. Finally, the field `deadlocks` is a list of reachable states from which the system does not progress any longer.

The exploration algorithm was instantiated with the SCA model as follows:

- `State` is `SCA`.
- `init` is an empty configuration of the SCA.
- `next` is `Next`.
- `prop?` is `Invariant`.
- `k` is 2811.

Finally, it was executed in the PVS ground evaluator, which is a PVS tool that produces efficient executable Lisp code from a PVS functional specification [11]. After the concept was modified incorporating the verification team recommendations, no counterexamples or deadlocks were reported by the algorithm.

The exploration algorithm and its correctness proof are freely available as a PVS package from `http://research.nianet.org/~munoz/Besc`.

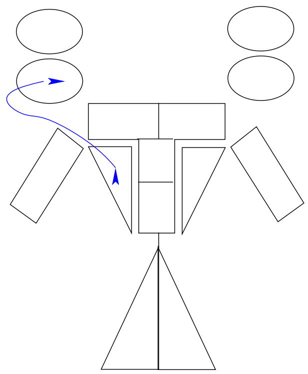

Figure 5: 3000 and 2000 feet available

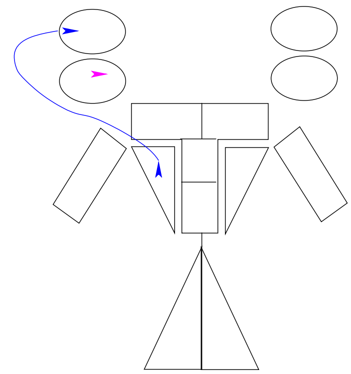

Figure 6: 3000 feet available, 2000 feet occupied

4. MODELING ISSUES

4.1 Non-deterministic Asynchronous Model

In order to accommodate all possible scenarios, due, for example, to different aircraft performances, the transition system is non-deterministic: for a given state of the SCA, all possible transitions are considered.

If two aircraft are holding at 3000 feet at opposite holding fixes and both of them are preparing for descend, determining which one of them will effectively be the first aircraft to descend in a real situation depends on several factors including aircraft performances and pilot preferences. The operational concept precludes aircraft from hovering in a holding pattern once a lower altitude becomes available. However, the exact time when this will occur is not defined. Such uncertainty is modeled by a non-deterministic set of asynchronous transitions where either one of the possible scenarios can potentially occur. The model includes conditions that are not physically possible; for example, one aircraft could change zones several times while another remains idle. This characteristic of the model does not adversely impact the safety verification since the safety conditions are shown for both realistic and unrealistic conditions.

4.2 Simultaneous Transitions

When an aircraft in a missed approach goes to its MAHF, it proceeds to the lowest available altitude. According to the SATS-HVO concept, the lowest available altitude is 2000 feet when the IAF is empty (Figures 5). Otherwise, it is 3000 feet when the holding pattern at 2000 feet is occupied (Figures 6). For completeness, the *lowest available altitude* transition also considers the case when 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 (Figures 7).

The fact that the *lowest available altitude determination* is a simultaneous transition, potentially involving 2 aircraft, can be considered a weakness of the model. However, the operational concept precludes an aircraft from hovering at a given altitude when a lower altitude is available. Therefore, the transition just reflects the fact that the aircraft holding at 3000 feet has enough time to descend to 2000 feet, before the aircraft in a missed approach enter its MAHF.

4.3 The Idle Effect

In the physical world aircraft do not remain idle (except, of course, aircraft on the ground). A *natural* implication of this physical constraint would be that it is always possible to move aircraft from a non-empty zone. However, this is not the case in the proposed model. Aircraft in the holding zones and aircraft in the base segments may in some circumstances stay in these zones for all possible transitions. That is, some aircraft may remain idle.

The fact that aircraft on the holding zones remain idle does not defy the laws of physics. Since each holding pattern is modeled as one atomic zone, from an abstract point of view, holding aircraft *do* remain idle.

The case of base segment is different. The transition from base segment to intermediate segment moves an aircraft from the base segment to the final approach only if its leader is already there. That means that an aircraft in the base segment remains idle waiting for its leader to go first on the final approach.

Intuitively, the fact that the condition of the merging rule is always true and that, from a practical point of view, aircraft do not remain idle on the base segment is a consequence of the operational concept. Under the SATS-HVO concept, an aircraft that initiates an approach is safely spaced from its leader. Therefore, the lead aircraft always goes first on the final approach.

As in the case of the simultaneous transition, the non-idle

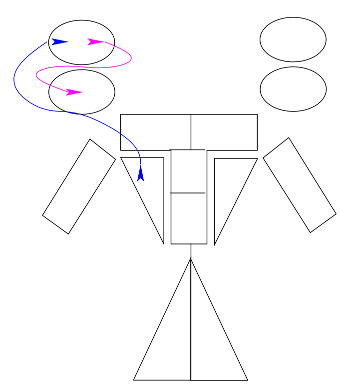

Figure 7: 3000 feet occupied, 2000 feet available

effect in the base segment has to be established outside of the model in a more accurate time-space model.

4.4 Theorem Proving vs. Model Checking

The abstract model does not assume any bounds on the state variables. Hence, the set of reachable states is potentially infinite. A further analysis has revealed that the set of reachable states is finite and, furthermore, quite small for model checking standards: 2811 states.

Theoretically, the number of arrival aircraft inside the SCA is bounded to 4. If the number of departure aircraft is also bounded to 4, the abstract model presented in this paper could be translated into a *finite transition system* of 213 *Boolean* variables as follows:

- 183 = (15 [zones] $*4$ [maximum length] $+1$ [AMM]) $*$ 3 [Boolean variables],
- 30 = 15 [zones] $*\,2$ [encoding of zone length],

where sequence numbers are encoded with 2 Boolean variables, and missed approach holding fixes are encoded with 1 Boolean variable.

A system of 213 variables is at the limit of the capabilities of BDD based symbolic model checking [4], although leading edges technologies such as bounded model checking [3] can handle systems containing several hundred variables and even larger systems [12]. Explicit state model checking, such as SPIN [8], can be efficient on systems with large number of variables provided that the number of reachable states is small.

Model checkers usually provide a specification language that includes basic finite types and temporal logic operators. Data types, parametrization, and arithmetic computations are either not included or they are minimally supported. SATS-HVO characteristics that may be cumbersome to specify in a model checker are:

- The representation of each state is a complex data structure. In this case, the SCA state is a record containing lists of aircraft states. Aircraft states are also represented by a record data type.
- The transition rules are described by arbitrary algorithms. For instance, conditions to be checked involve arithmetic computations related to the number of aircraft in some zones or assigned to a determined missed approach holding fix.
- The predicates to be checked are expressed by arbitrary algorithms. Some of these predicates are inductively defined over lists.
- The structure of the SCA is highly symmetric. Transition rules for left and right zones are completely symmetric.

Hence, the use of a limited language makes the specification of the model error prone and, once it has been developed, difficult to maintain.

In contrast to model checkers, theorem provers provide very expressive specification languages. In particular, PVS is based on a higher order logic enhanced with a powerful type system. It also provides simple readable notations for data structures such as records and unbounded lists. As it has been shown in Section 3.4, the higher order logic allows for parameterized left–right transition rules and zones that enormously simplify the specification.

For all these reasons, the model was written in PVS. Furthermore, the verification was automated via an in-house state exploration algorithm written and proven correct in PVS. Nevertheless, as part of this research, the high level model was also written in the recently released SAL system [2, 5]. SAL provides several tools for the analysis of systems specified as transition relations and a language that is syntactically and semantically similar to PVS. Unfortunately, none of the SAL' symbolic model checkers (including a bounded model checker) was able to handle the SATS-HVO model.

5. CONCLUSION

The SATS-HVO operational concept describes nominal arrival and departure operations inside the SCA. One of the key safety hypotheses of the concept's design is that aircraft flying nominal operations are always separated. The formal validation of this hypothesis involves answering several kind of questions. Some of these questions are on structural issues such as "Do the rules cover all possible scenarios?", while other questions are more of physical nature such as "How does flight performance affect the overall safety of the concept?".

The mathematical model presented in this paper addresses the first kind of issues: the structural and logical properties of the concept. Time and space dimensions are discretized to enable mechanical exploration of nominal scenarios. Moreover, by allowing non-deterministic behaviors, aircraft and pilot performances are abstracted away. For these reasons, the model is conservative, i.e., the set of scenarios described by the model is a superset of the set of nominal scenarios. Hence, this model yields a robust notion of safety, i.e., a notion that relies only on the logic of the concept and not on space-time properties such as the geometry of the SCA, physical constraints such as aircraft performances, or human factors such as pilot preferences.

The model, a non-deterministic, asynchronous transition system, was written in PVS and verified using a state exploration algorithm written and proven correct in PVS. In total, 2811 nominal scenarios were identified. Checking by hand each one of them is clearly not an alternative. The exploration algorithm checked in a few minutes a handful set of safety properties. The algorithm also checked that the model does not have *deadlocks*, i.e., scenarios where the global SCA state cannot progress any longer. It has also been checked that when entries are systematically denied, the SCA eventually evolves into an empty state. From a practical point of view, this property means that the SCA can be effectively and properly sterilized if needed (for example, when an aircraft declares an emergency and the air traffic service provider takes control of the SCA).

Valid questions on this work are (1) whether or not the model accurately captures the *real* operational concept, and (2) whether or not the properties that the model satisfies *completely* cover all the safety requirements of the SATS-HVO concept. The first question, of course, cannot be formally answered. For this reason, the formal model was extensively discussed with the SATS-HVO concept development group. The validation flowed in both directions. Indeed, as result of this formal work, 9 issues related to the original concept were identified and 10 recommendations were issued by the formal methods team. All the recommendations were accepted and implemented in the current concept. They are necessary to achieve the intended safe functionality of SATS-HVO procedures.

The answer to the second question is clearly negative. The high level abstraction is not fine enough to handle safety properties that *do* require aircraft and pilot performances such as:

- Aircraft on the missed approach zone are separated.
- Aircraft on the final approach are spaced.
- An aircraft in a holding pattern has enough time to initiate the approach before a missed aircraft completes its operation (see discussion on Section 4.2).
- Aircraft have enough time to merge for the final approach (see discussion on Section 4.3).
- Aircraft are separated during the transition from one zone to the other.
- Departing aircraft are separated from landing aircraft.

Future work includes the construction of a refined model of the SATS-HVO concept, on top of the abstract one, that allows for the verification of properties that cannot be handled by the current approach.

6. ACKNOWLEDGMENTS

The authors are very thankful to the members of the SATS-HVO concept development working group at NASA Langley for supporting this investigation. They also appreciate the technical expertise on SAL provided by its main developer: Leonardo de Moura.

For the first two authors, this work was supported by the National Aeronautics and Space Administration under NASA Cooperative Agreement NCC-1-02043.

7. REFERENCES

[1] C. Adams, M. Consiglio, K. Jones, and D. Williams. SATS HVO Operational Concept: Nominal Operations. NASA Langley Research Center, 2003.

[2] S. Bensalem, V. Ganesh, Y. Lakhnech, C. Muñoz, S. Owre, H. Rueß, J. Rushby, V. Rusu, H. Saïdi, N. Shankar, E. Singerman, and A. Tiwari. An overview of SAL. Technical Report NASA/CP-2000-210100, NASA Langley Research Center, Hampton, Virginia, June 2000.

[3] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu. Symbolic model checking without BDDs. In *Tools and Algorithms for the Analysis and Construction of Systems (TACAS'99)*, volume 1579 of *Lecture Notes in Computer Science*, pages 193–207. Springer, 1999.

[4] J. R. Burch, E. M. Clarke, and K. L. McMillan. Symbolic model checking: 10^{20} states and beyond. *Information and Computation*, 98:142–170, 1992.

[5] L. de Moura, S. Owre, and N. Shankar. The SAL language manual. Technical Report SRI-CSL-01-01 (Rev. 2), SRI International, August 2003. Available at: `http://sal.csl.sri.com`.

[6] G. Dowek, C. Muñoz, and V. Carreño. Abstract model of the SATS concept of operations: Initial results and recommendations. Technical Report NASA/TM-2004-213006, NASA Langley Research Center, NASA LaRC,Hampton VA 23681-2199, USA, March 2004.

[7] *Federal Aviation Regulations/Aeronautical Information Manual*, 1999.

[8] G. J. Holzmann. *The SPIN Model Checker, Primer and Reference Manual*. Addison-Wesley, 2003.

[9] S. P. Office. Small Aircraft Transportation System Program Plan. NASA Langley Research Center, `http://sats.larc.nasa.gov/main.html`, 2001.

[10] S. Owre, J. M. Rushby, and N. Shankar. PVS: A prototype verification system. In D. Kapur, editor, *11th International Conference on Automated Deduction (CADE)*, volume 607 of *Lecture Notes in Artificial Intelligence*, pages 748–752, Saratoga, NY, June 1992. Springer-Verlag.

[11] N. Shankar. Efficiently executing PVS. Project report, Computer Science Laboratory, SRI International, Menlo Park, CA, Nov. 1999. Available at `http://www.csl.sri.com/shankar/PVSeval.ps.gz`.

[12] O. Shtrichman. Tuning SAT checkers for Bounded Model Checking. In *Computer Aided Verification, 12th International Conference, CAV 2000*, volume 1855 of *Lecture Notes in Computer Science*, pages 480–494. Springer, 2000.

APPENDIX

Acronyms used in this paper:

AMM	Airport Management Module
DF	Departure Fix
FAF	Final Approach Fix
GPS	Global Positioning System
HVO	Higher Volume Operations
IAF	Initial Arrival Fix
IF	Intermediate Fix
NASA	National Aeronautics and Space Administration
NIA	National Institute of Aerospace
MAHF	Missed Approach Holding Fix
PVS	Prototype Verification System
SATS	Small Aircraft Transportation System
SCA	Self Controlled Area