Formal Methods in Air Traffic Management: The Case of Unmanned Aircraft Systems (Invited Lecture)*

César A. Muñoz

NASA Langley Research Center, Hampton, Virginia 23681-2199

Abstract. As the technological and operational capabilities of unmanned aircraft systems (UAS) continue to grow, so too does the need to introduce these systems into civil airspace. Unmanned Aircraft Systems Integration in the National Airspace System is a NASA research project that addresses the integration of civil UAS into non-segregated airspace operations. One of the major challenges of this integration is the lack of an on-board pilot to comply with the legal requirement that pilots see and avoid other aircraft. The need to provide an equivalent to this requirement for UAS has motivated the development of a *detect and avoid* (DAA) capability to provide the appropriate situational awareness and maneuver guidance in avoiding and remaining well clear of traffic aircraft. Formal methods has played a fundamental role in the development of this capability. This talk reports on the formal methods work conducted under NASA's Safe Autonomous System Operations project in support of the development of DAA for UAS. This work includes specification of low-level and high-level functional requirements, formal verification of algorithms, and rigorous validation of software implementations. The talk also discusses technical challenges in formal methods research in the context of the development and safety analysis of advanced air traffic management concepts.

Extended Abstract

The unmanned aircraft industry represents a potential source of significant increase in economic developments and and safety capabilities. According to the 2013 economic report by the Association for Unmanned Vehicle Systems International (AUVSI) [6], the cumulative impact between 2015 and 2025 to the US economy resulting from the integration of Unmanned Aircraft Systems (UAS) into the National Airspace System (NAS) will be more than US $80 billions and will generate more than 100 thousand jobs. The report identifies precision agriculture and public safety as the two main potential markets for UAS in the US.

* This invited lecture reports on research conducted at NASA Langley Research Center at the Safety-Critical Avionics Systems Branch by several individuals including, in addition to the author, Anthony Narkawicz, George Hagen, Jason Upchurch, and Aaron Dutle.

As the availability and applications of UAS grow, these systems will inevitably become part of standard airspace operations. A fundamental challenge for the integration of UAS into the NAS is the lack of an on-board pilot to comply with the legal requirement identified in the US Code of Federal Regulations to see and avoid traffic aircraft. As a means of compliance with this legal requirement, the final report of the FAA-sponsored Sense and Avoid (SAA) Workshop [4] defines the concept of *sense and avoid* for remote pilots as "the capability of a UAS to remain well clear from and avoid collisions with other airborne traffic."

NASA's Unmanned Aircraft Systems Integration in the National Airspace System project aims to develop key capabilities to enable routine and safe access for public and civil use of UAS in non-segregated airspace operations. As part of this project, NASA has developed a *detect and avoid* (DAA) concept for UAS [1] that implements the sense and avoid concept outlined by the SAA Workshop. The NASA DAA concept defines a volume representing a well-clear boundary where aircraft inside this volume are considered to be in well-clear violation. This volume is intended to be large enough to avoid safety concerns for controllers and see-and-avoid pilots. It shall also be small enough to avoid disruptions to traffic flow. Formally, this volume is defined by a boolean predicate on the states of two aircraft, i.e., their position and velocity vectors at current time. The predicate states that two aircraft are *well clear* of each other if appropriate distance and time variables determined by the relative aircraft states remain outside a set of predefined threshold values. These distance and time variables are closely related to variables used in the Resolution Advisory (RA) logic of the Traffic Alerting and Collision Avoidance System (TCAS).

TCAS is a family of airborne devices that are designed to reduce the risk of mid-air collisions between aircraft equipped with operating transponders. TCAS II [16], the current generation of TCAS devices, is mandated in the US for aircraft with greater than 30 seats or a maximum takeoff weight greater than 33,000 pounds. Although it is not required, TCAS II is also installed on many turbine-powered general aviation aircraft. An important characteristic of the well-clear violation volume is that it conservatively extends the volume defined by TCAS II, i.e., for an appropriate choice of threshold values, the TCAS II RA volume is strictly contained within the well-clear violation volume [10]. Hence, aircraft are declared to be in a well-clear violation before an RA is issued. This relation between the well-clear violation volume and the TCAS II volume guarantees that software capabilities supporting the DAA concept safely interact well with standard collision avoidance systems for commercial aircraft.

The well-clear definition proposed by NASA satisfies several geometric and operational properties [11]. For example, it is *symmetric*, i.e., in a pair-wise scenario, both aircraft make the same determination of being well-clear or not. Furthermore, the well-clear violation volume is *locally convex*, i.e., in a non-maneuvering pair-wise scenario, there is at most one time interval in which the aircraft are not well clear. Symmetry and local convexity represent fundamental safety properties of the DAA concept. In particular, symmetry ensures that all aircraft are simultaneously aware of a well-clear violation. Local convexity

states that in a non-maneuvering scenario, a predicted well-clear violation is continuously alerted until it disappears. Once the alert disappears, it does not reappear unless the aircraft change their trajectories.

The NASA DAA concept also includes self-separation and alerting algorithms intended to provide remote pilots appropriate situational awareness of proximity to other aircraft in the airspace. These algorithms are implemented in a software library called DAIDALUS (Detect & Avoid Alerting Logic for Unmanned Systems) [12]. DAIDALUS consists of algorithms for determining the current well-clear status between two aircraft and for predicting a well-clear violation within a lookahead time, assuming non-maneuvering trajectories. In the case of a predicted well-clear violation, DAIDALUS also provides an algorithm that computes the time interval of well-clear violation. Furthermore, DAIDALUS implements algorithms for computing prevention bands, assuming a simple kinematic trajectory model. Prevention bands are ranges of track, ground speed, and vertical speed maneuvers that are predicted to be in well-clear violation within a given lookahead time. These bands provide awareness information to remote pilots and assist them in avoiding certain areas in the airspace. When aircraft are not well clear, or when a well-clear violation is unavoidable, the DAIDALUS algorithms compute well-clear recovery bands. Recovery bands are ranges of horizontal and vertical maneuvers that assist pilots in regaining well-clear status within the minimum possible time. Recovery bands are designed so that they do not conflict with resolution advisory maneuvers generated by systems such as TCAS II. DAIDALUS implements two alternative alerting schemas. One schema is based on the prediction of well-clear violations for different sets of increasingly conservative threshold values. The second schema is based on the types of bands, which can be either preventive or corrective, computed for a single set of threshold values. A band is preventive if it does not include the current trajectory. Otherwise, it is corrective. Recovery bands, by definition, are always corrective. In general, both schemas yield alert levels that increase in severity as a potential pair-wise conflict scenario evolves. The DAIDALUS library is written in both C++ and Java and the code is available under NASA's Open Source Agreement. DAIDALUS is currently under consideration for inclusion as DAA reference implementation of the RTCA Special Committee 228 Minimum Operational Performance Standards (MOPS) for Unmanned Aircraft Systems.

Given the safety-critical nature of the UAS in the NAS project, formal methods research has been conducted under NASA's Safe Autonomous System Operations project in support of the development of the DAA concept for UAS. The use of formal methods includes a formal definition of the well-clear violation volume, formal proofs of its properties, formal specification and verification of all DAIDALUS algorithms, and the rigorous validation of the software implementation of DAIDALUS algorithms against their formal specifications. All formal specifications and proofs supporting this work are written and mechanically verified in the Prototype Verification System (PVS) [15]. The tool PVSio [8] is used to animate PVS functional specifications.

The application of formal methods to the safety analysis of air traffic management systems faces technical challenges common to complex cyber-physical systems (CPS). Chief among those challenges is the interaction of CPS with the physical environment that yields mathematical models with both continuous and discrete behaviors. Formally proving properties involving continuous mathematics, and in particular, non-linear arithmetic is a well-known problem in automated deduction. As part of this research effort, several automated decision and semi-decision procedures for dealing with different kinds of non-linear real arithmetic problems have been developed [2,7,9,13,14]. Most of these procedures are formally verified and are available as proof-producing automated strategies in the PVS theorem prover.

The formal verification of software implementations of a CPS is a major endeavor even when the algorithms that are implemented have been formally verified. First, there is a large semantic gap between modern programming languages and the functional notation used in formal tools such as PVS. However, the main difficulty arises from the fact that modern programming languages utilize floating point arithmetic while formal verification is usually performed over the real numbers. An idea for lifting functional correctness properties from algorithms that use real numbers to algorithms that use floating-point numbers is discussed in [5]. However, this research area is still in an early stage. In [3], a practical approach to the validation of numerical software is proposed. The approach, which is called *model animation*, compares computations performed in the software implementations against those symbolically evaluated to an arbitrary precision on the corresponding formal models. While model animation does not provide an absolute guarantee that the software is correct, it increases the confidence that the formal models are faithfully implemented in code. Model animation has been used to validate in a rigorous way the software implementation of DAIDALUS algorithms against their formal specifications.

Finally, air traffic management systems are unique in some aspects. For instance, these systems involve human and automated elements and these elements are often subject to strict operational (and sometimes legal) requirements. These requirements restrict the design space of operational concepts, such as detect and avoid for UAS. More importantly, new concepts and algorithms have to support an incremental evolution of the air space system at a global scale. All these requirements and restrictions may result in solutions that are non-optimal from a theoretical point of view or that have complex verification issues due to legacy systems.

References

1. María Consiglio, James Chamberlain, César Muñoz, and Keith Hoffler. Concept of integration for UAS operations in the NAS. In *Proceedings of 28th International Congress of the Aeronautical Sciences, ICAS 2012*, Brisbane, Australia, 2012.
2. William Denman and César Muñoz. Automated real proving in PVS via Meti-Tarski. In Cliff Jones, Pekka Pihlajasaari, and Jun Sun, editors, *Proceedings of*

the 19th International Symposium on Formal Methods (FM 2014), volume 8442 of *Lecture Notes in Computer Science*, pages 194–199, Singapore, May 2014. Springer.

3. Aaron Dutle, César Muñoz, Anthony Narkawicz, and Ricky Butler. Software validation via model animation. In Jasmin Blanchette and Nikolai Kosmatov, editors, *Proceedings of the 9th International Conference on Tests & Proofs (TAP 2015)*, volume 9154 of *Lecture Notes in Computer Science*, pages 92–108, L'Aquila, Italy, July 2015. Springer.
4. FAA Sponsored Sense and Avoid Workshop. Sense and avoid (SAA) for Unmanned Aircraft Systems (UAS), October 2009.
5. Alwyn Goodloe, César Muñoz, Florent Kirchner, and Loïc Correnson. Verification of numerical programs: From real numbers to floating point numbers. In Guillaume Brat, Neha Rungta, and Arnaud Venet, editors, *Proceedings of the 5th NASA Formal Methods Symposium (NFM 2013)*, volume 7871 of *Lecture Notes in Computer Science*, pages 441–446, Moffett Field, CA, May 2013. Springer.
6. Darryl Jenkins and Bijan Vasigh. The economic impact of Unmanned Aircraft Systems integration in the United States. Economic report of the Association For Unmanned Vehicle Systems International (AUVSI), March 2013.
7. Mariano Moscato, César Muñoz, and Andrew Smith. Affine arithmetic and applications to real-number proving. In Christian Urban and Xingyuan Zhang, editors, *Proceedings of the 6th International Conference on Interactive Theorem Proving (ITP 2015)*, volume 9236 of *Lecture Notes in Computer Science*, Nanjing, China, August 2015. Springer.
8. César Muñoz. Rapid prototyping in PVS. Contractor Report NASA/CR-2003-212418, NASA, Langley Research Center, Hampton VA 23681-2199, USA, May 2003.
9. César Muñoz and Anthony Narkawicz. Formalization of a representation of Bernstein polynomials and applications to global optimization. *Journal of Automated Reasoning*, 51(2):151–196, August 2013.
10. César Muñoz, Anthony Narkawicz, and James Chamberlain. A TCAS-II resolution advisory detection algorithm. In *Proceedings of the AIAA Guidance Navigation, and Control Conference and Exhibit 2013*, number AIAA-2013-4622, Boston, Massachusetts, August 2013.
11. César Muñoz, Anthony Narkawicz, James Chamberlain, María Consiglio, and Jason Upchurch. A family of well-clear boundary models for the integration of UAS in the NAS. In *Proceedings of the 14th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, number AIAA-2014-2412, Georgia, Atlanta, USA, June 2014.
12. César Muñoz, Anthony Narkawicz, George Hagen, Jason Upchurch, Aaron Dutle, and María Consiglio. DAIDALUS: Detect and Avoid Alerting Logic for Unmanned Systems. In *Proceedings of the 34th Digital Avionics Systems Conference (DASC 2015)*, Prague, Czech Republic, September 2015.
13. Anthony Narkawicz and César Muñoz. A formally verified generic branching algorithm for global optimization. In Ernie Cohen and Andrey Rybalchenko, editors, *Proceedings of the 5th International Conference on Verified Software: Theories, Tools, and Experiments (VSTTE 2013)*, volume 8164 of *Lecture Notes in Computer Science*, pages 326–343, Menlo Park, CA, US, May 2014. Springer.
14. Anthony Narkawicz, César Muñoz, and Aaron Dutle. Formally-verified decision procedures for univariate polynomial computation based on Sturm's and Tarski's theorems. *Journal of Automated Reasoning*, 54(4):285–326, 2015.

15. S. Owre, J. Rushby, and N. Shankar. PVS: A prototype verification system. In Deepak Kapur, editor, *Proceedings of the 11th International Conference on Automated Deduction*, volume 607 of *Lecture Notes in Artificial Intelligence*, pages 748–752. Springer-Verlag, June 1992.
16. RTCA SC-147. RTCA-DO-185B, Minimum operational performance standards for traffic alert and collision avoidance system II (TCAS II), July 2009.