A Mathematical Analysis of Air Traffic Priority Rules

Anthony Narkawicz [*]   César A. Muñoz [*]   Jeffrey M. Maddalon [*]

This paper analyzes priority rules, such as those in Part 91.113 of the Federal Aviation Regulations. Such rules determine which of two aircraft should maneuver in a given conflict scenario. While the rules in 91.113 are well accepted, other concepts of operation for NextGen, such as self separation, may allow for different priority rules. A mathematical framework is presented that can be used to analyze a general set of priority rules and enables proofs of important properties. Specific properties considered in this paper include safety, effectiveness, and stability. A set of rules is said to be safe if it ensures that it is never the case that both aircraft have priority. They are effective if exactly one aircraft has priority in every situation. Finally, a set of rules is called stable if it produces compatible results even under small changes to input data.

I.   Introduction

When the trajectories of two aircraft are in conflict, safe conflict resolution requires the maneuvers of the aircraft to be mutually compatible. If both aircraft are flying under Instrument Flight Rules (IFR), then the air traffic controller ensures this compatibility. For instance, the controller may direct one aircraft to completely resolve the conflict, or, in rare cases, may vector both aircraft simultaneously to form a compatible action. In situations where at least one aircraft is flying under Visual Flight Rules (VFR), the pilots must engage in a shared decision-making process that ensures safety. This concept is called *coordination*,[7] which comes in two types. If the maneuver decisions are communicated between the pilots, this coordination is termed *explicit*.[6] Alternatively, if the decisions are not communicated and yet remain safe, then this is called *implicit coordination*.[6] Each of these types of coordination can be divided into whether the action requires one aircraft to maneuver or both aircraft to maneuver. Single aircraft maneuvers are desired for efficiency reasons, and multiple aircraft maneuvers are often desired for safety reasons. Air traffic management concepts typically prefer single aircraft maneuvers while there is sufficient time to recover from any unexpected conditions and demand that both aircraft maneuver when such time is not available. Implicit coordination for multiple aircraft maneuvers has been extensively studied by Narkawicz and Muñoz.[4] This paper presents an analytical framework to examine key safety properties of implicit coordination for single aircraft maneuvers. The focus is on deciding which aircraft should maneuver rather than what particular maneuver should be implemented.

Implicit coordinaition for single aircraft maneuvers happens in the airspace today through the right-of-way rules in Title 14 of the Code of Federal Regulations (CFR), also known as the Federal Aviation Regulations.[2] It has been referred to in the literature as *right-of-way rules*[2] and *priority rules*.[7] Behind each of these concepts is a procedure whereby one aircraft is designated as having *priority*, and the other aircraft is designated as *burdened* and is therefore required to perform an avoidance maneuver.[1,8] The current right-of-way rules for air traffic are partially[a] captured in 14CFR91.113, for example, "(d) Converging. When aircraft of the same category are converging at approximately the same altitude (except head-on, or nearly so), the aircraft to the other's right has the right-of-way." Although these rules specifically apply in the United States, the International Civil Aviation Organization (ICAO) has similar rules in Annex 2.

The priority rules in 91.113 are very well established. The authors are not aware of any effort, anywhere in the world, to modify them. However, developing this analysis framework for priority rules has several motivations. First, new airspace concepts such as self-separation[7] and unmanned aircraft operations may require different rules in some circumstances. The acceptance of these concepts will be contingent on the priority rules. Because 91.113 is so well-established, great scrutiny will be placed on any new set of priority

---

[*]NASA Langley Research Center, Hampton, VA 23681, USA
[a]Right-of-way rules for specialized operations are contained in 14CFR91.115 and 14CFR103.13.

American Institute of Aeronautics and Astronautics

rules, and the analysis presented in this paper may alleviate concerns. Second, new surveillance technologies, such as ADS-B,[5] may allow different priority rules, which provide more efficiency in some circumstances. Finally, in a growing number of applications, automation is used to implement priority rules. Computers, by their nature, require precise specifications of priority rules. These specifications may lead to safety-critical situations that would never arise with humans implementing the rules. For instance, data uncertainty affects automated systems in very different ways than it affects humans.

An important property of the current rules in 91.113 is that the rules that are defined geometrically depend only on the current states of the aircraft. An interesting aspect of the approach presented in this paper is that such rules do not have to be applied at current states of the aircraft and could instead be applied at future states along their planned trajectories. One choice not addressed in this paper is the exact point at which the rules should be applied. The rules could be applied at the current states of the two aircraft or, if the aircraft are in conflict, the states on their respective flight plans at the last waypoint before the conflict. This latter situation is illustrated in Figure 1. Although this is a critical issue for an airspace concept, it does not factor into our analysis approach. It is not the purpose of this paper to advocate for one state being used instead of another when determining priority. Rather, this paper addresses the problem of analyzing priority rules, assuming that particular states have been chosen and both aircraft use these states.
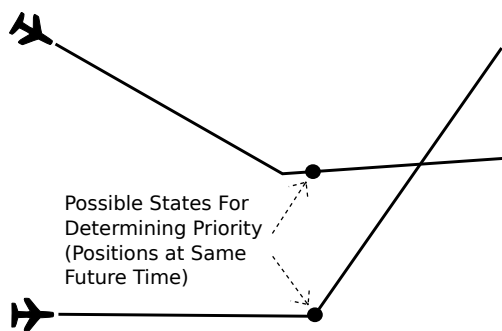

Figure 1. States at Waypoint Where Conflict is Created

Priority rules are considered from the perspective of one aircraft, called the *ownship*. The other aircraft is referred to as the *intruder*. The airspace volume is modeled using a flat-earth projection around a common projection point in a 3D rectangular system, i.e., aircraft positions and velocities are viewed as points and vectors in $\mathbb{R}^3$. The states of the ownship and the intruder can be described by position vectors $\mathbf{s}_o, \mathbf{s}_i \in \mathbb{R}^3$ and velocity vectors $\mathbf{v}_o, \mathbf{v}_i \in \mathbb{R}^3$, the subscripts of which indicate that the vector belongs to the ownship or the intruder (o for ownship and i for intruder). For $\mathbf{s} \in \mathbb{R}^3$, the notation $\mathbf{s}_{(x,y)}$ represents the 2D projection of $\mathbf{s}$, i.e., the vector in $\mathbb{R}^2$ such that $\mathbf{s}_{(x,y)} = (s_x, s_y)$. For any vector $\mathbf{v}_2$ in $\mathbb{R}^2$, $\mathbf{v}_2^\perp$ refers to the vector $(v_{2,y}, -v_{2,x})$, which is a 90-degree clockwise rotation of $\mathbf{v}_2$. Note that any other nonzero vector $\mathbf{w}$ in $\mathbb{R}^2$ is parallel to $\mathbf{v}_2$ if $\mathbf{w} \cdot \mathbf{v}_2^\perp = 0$. Finally, if β is any boolean value, equal to either true or false, then $\neg\beta$ will denote the negation of β.

A set of priority rules can be viewed as a function on $\mathbf{s}_o$, $\mathbf{v}_o$, $\mathbf{s}_i$, and $\mathbf{v}_i$ that returns true or false. If this function returns true then the ownship has priority, and if it returns false then the ownship is burdened. The assumption is that both aircraft will use the same priority function. Using this notation we define the abstract interface to a priority rule as:

Definition 1. *A priority rule is a function* `Priority` *that takes* $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i,$ *and* $\mathbf{v}_i$ *as inputs and returns a Boolean value, either* `true` *or* `false`*:*

$$\texttt{Priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \in \{\texttt{true}, \texttt{false}\}.$$

In section II of this paper, some key properties of priority rules are presented. Section III presents an extended discussion of the stability property of priority rules. Section IV examines the priority rules in 91.113 by first formalizing key elements of the rules and then analyzing their important properties. Finally, section V analyzes a different set of priority rules intended for a self-separation application.

American Institute of Aeronautics and Astronautics

II. Important Properties of Priority Rules

This paper considers several important properties of priority rules that should be considered for any new set of rules. Some of these properties are enumerated below.

Property 1 (Safety). *It is never the case that both aircraft have priority.*

If both aircraft have priority, then neither aircraft will maneuver, so this is the fundamental safety property for priority rules. Formally, a priority rule is said to be *safe* if it is never the case that under the same inputs both aircraft have priority. This property is defined mathematically with the predicate `safe?`, given as follows.

Definition 2.

$$\texttt{safe?}(\textit{Priority}) \equiv \textit{For All } \mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i \in \mathbb{R}^3 : \textit{Priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \implies \neg \textit{Priority}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o).$$

Property 2 (Effectiveness). *Exactly one aircraft has priority.*

A trivial way to satisfy the safety property `safe?` is to always burden both aircraft, which would defeat the purpose of priority rules. To ensure that priority rules are useful, it should be true that, in most cases, only one aircraft has priority. This property is mathematically expressed as

$$\text{For All } \mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i : \; [\texttt{Priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \wedge \neg\texttt{Priority}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o)] \vee$$
$$[\texttt{Priority}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o) \wedge \neg\texttt{Priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)].$$

The term *effective* can also be used for only a subset of possible input states. For example, one could say that a particular priority rule is effective for all states that are not in a head-on conflict.

Property 3 (Stability). *Even with small changes to the inputs, `Priority` produces compatible outputs.*

If both aircraft use the same safe and effective priority function and identical input data, they will, by definition, provide compatible outputs. Although it is reasonable to assume that both aircraft will use the same priority function, with data dropouts and communication errors it is unrealistic to assume that both aircraft will use the same input data. The communication necessary to ensure both aircraft use the same data is just as complex as explicit coordination. This property ensures that `Priority` produces compatible outputs in the real-world situation of slight data inconsistencies without the complexity equivalent to explicit coordination.

Stability depends on the definition of *small* changes to the input data. In the definition below, $\mathbf{s}, \mathbf{v}$ are the vectors that represent the aircraft's perception of its own state at one instant in time and $\mathbf{s}', \mathbf{v}'$ represent that aircraft state's as perceived by the other aircraft. The values $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}$ are positive real numbers defining a small change in the aircraft's state vectors. The relationship between these two views of the state of an aircraft is defined as follows.

Definition 3.

$$\begin{aligned}
\texttt{perturb?}(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}, \mathbf{v}, \mathbf{s}', \mathbf{v}') \equiv\; & \|\mathbf{s}_{(x,y)} - \mathbf{s}'_{(x,y)}\| < \epsilon_{s,xy} \wedge \\
& \|\mathbf{v}_{(x,y)} - \mathbf{v}'_{(x,y)}\| < \epsilon_{v,xy} \wedge \\
& |s_z - s'_z| < \epsilon_{s,z} \wedge \\
& |v_z - v'_z| < \epsilon_{v,z}.
\end{aligned}$$

It is important to note that there are properties of priority rules that are not listed above but are still very important to consider. For instance, any new set of priority rules should be compared with the current rules as outlined in 91.113. Another important property is whether sufficient data is available from surveillance system (i.e., pilot's eyes or ADS-B) to determine priority by the rule. Finally, a rule should be geometrically comprehensive and therefore include no gaps or singularities. This last property is true for any priority function, such as those presented above, because if `Priority`$(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ is computable, then every input produces some output, so the rule has no gaps.

American Institute of Aeronautics and Astronautics

III. Stability of Priority Rules

The previous section included a qualitative discussion of the notion of stability. This section explores stability through several detailed mathematical definitions of stability. Each of the definitions for stability given below involve a specific, formal definition of *compatible* in the stability property.

The first definition of stability states that if the data used by the intruder to determine priority is a small perturbation of the data used by the ownship, then it is still the case that exactly one of them has priority and the other does not. This definition is essentially the same as property 2 incorporating data error. This is formalized in the following mathematical statement.

Definition 4 (Strong Stability). *For all* $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i \in \mathbb{R}^3$, *there exists* $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z} > 0$ *such that for all* $\mathbf{s}_o', \mathbf{v}_o', \mathbf{s}_i', \mathbf{v}_i' \in \mathbb{R}^3$,

$$perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_o', \mathbf{v}_o') \wedge perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_i', \mathbf{v}_i', \mathbf{s}_i, \mathbf{v}_i) \implies$$
$$[Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \ and \ \neg Priority(\mathbf{s}_i', \mathbf{v}_i', \mathbf{s}_o', \mathbf{v}_o')] \ or$$
$$[Priority(\mathbf{s}_i', \mathbf{v}_i', \mathbf{s}_o', \mathbf{v}_o') \ and \ \neg Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)]$$

The problem with this definition is that it is too strong. Intuitively, if the aircraft are near the point where the ownship switches from burdened to priority, then this stability property allows both aircraft to perceive the other aircraft as burdened and therefore violate the safety property. Mathematically, this is expressed in the following theorem.

Theorem 1. *It is impossible to define priority rules that are safe (property 1), effective (property 2), and stable (as represented by definition 4).*

A sketch of the proof of this theorem is to observe that `Priority` must return exactly the same result if the inputs are changed a small amount. However, this implies that `Priority` is a continuous function. If `false` is represented by 0 and `true` is represented by 1, then the Intermediate Value Theorem states that such a function must be constant. A constant true function is unsafe and a constant false function is not effective. A priority function that satisfies such a strong stability condition is not useful.

Clearly, a less restrictive definition of stability is needed. A solution to this problem is to allow cases where both aircraft are burdened when the input data is near a point where priority would change from one aircraft to another. In this way, the rules are effective as long at the data is away from a priority-swapping point. This paper considers two alternate forms of stability, referred to as *basic stability* and *uniform stability*. Both of them allow for cases where neither aircraft has priority.

The first alternate definition of stability is referred as *basic stability*. The only time when safety of the priority function `Priority` becomes a problem is when both aircraft have priority at the same time. One way to mitigate this situation is to ensure that when one aircraft has priority, it will continue to have priority under slight data perturbations. This is an informal way to define basic stability; a formal mathematical definition is given below.

Definition 5 (Basic Stability).

$$basic\_stability?(Priority) \equiv$$

For All $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i \in \mathbb{R}^3$: *There Exists* $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z} > 0$: *For All* $\mathbf{s}_o', \mathbf{v}_o', \mathbf{s}_i', \mathbf{v}_i' \in \mathbb{R}^3$:

$$perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_o', \mathbf{v}_o') \wedge perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_i', \mathbf{v}_i') \wedge$$
$$Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \implies Priority(\mathbf{s}_o', \mathbf{v}_o', \mathbf{s}_i', \mathbf{v}_i').$$

This definition would be trivially satisfied if the function `Priority` always returned `true`, which would be unsafe. However, if `safe?(Priority)` and `basic_stability?(Priority)` both hold, then for any aircraft that has priority, even under slight perturbations of the data, it will still have priority and the other aircraft will not.

As it turns out, there is a close relationship between priority functions that satisfy the predicates `safe?` and `basic_stability?` and continuous functions on $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i$ that return a real number, which is stated in the following theorem.

American Institute of Aeronautics and Astronautics

Theorem 2. *If f is a continuous function that takes $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i$ as inputs and returns a real number and also satisfies the property that*

$$f(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = -f(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o)$$

for all possible inputs $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i$, then the priority function $Priority_f$ defined by

$$Priority_f(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv \; (f(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) > 0)$$

satisfies $basic\_stability?(Priority_f)$ and $safe?(Priority_f)$.

The more interesting property is that the reverse implication is true, a result stated in the next theorem.

Theorem 3. *If $basic\_stability?(Priority)$ and $safe?(Priority)$, then there exists a continuous function f as defined in Theorem 2. That is, $Priority = Priority_f$ and f has the property that $f(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = -f(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o)$ for all inputs.*

The problem with basic stability is that, even though an aircraft with priority continues to have priority even under perturbations, the size of these allowed perturbations may be smaller or larger depending on the input data. For some inputs, the size of the allowed perturbations can very small. Thus, a notion of stability is also needed that, for a fixed bounded range of the input data (e.g., both aircraft are within 100 km of each other traveling at no more than 1000 kts), gives a fixed size for perturbations such that all perturbations less than that size are allowed. However, an allowed perturbation in this context does not mean that the aircraft with priority will continue to have priority, but rather that if one aircraft has priority, then the other aircraft will not have priority, even using the perturbed data. This is notion referred to as *uniform stability* in this paper.

Definition 6 (Uniform Stability). *Given bounds on the input data, there exists a certain size for perturbations such that, if an aircraft has priority, then for any perturbation less than this size, the other aircraft will not have priority. This is formalized in the following predicate.*

$uniform\_stability?(Priority) \equiv$

> *There Exists* $M_{s,xy}, M_{v,xy}, M_{s,z}, M_{v,z} > 0:$ *For All* $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i \in \mathbb{R}^3 :$
>
> *There Exists* $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z} > 0:$ *For All* $\mathbf{s}'_o, \mathbf{v}'_o, \mathbf{s}'_i, \mathbf{v}'_i \in \mathbb{R}^3 :$
>
> $\|\mathbf{s}_{o(x,y)}\| < M_{s,xy} \; \wedge \; \|\mathbf{s}_{i(x,y)}\| < M_{s,xy} \; \wedge \; \|\mathbf{v}_{o(x,y)}\| < M_{v,xy} \; \wedge \; \|\mathbf{v}_{i(x,y)}\| < M_{v,xy} \; \wedge$
>
> $|s_{oz}| < M_{s,z} \; \wedge \; |s_{iz}| < M_{s,z} \; \wedge \; |v_{oz}| < M_{v,z} \; \wedge \; |v_{iz}| < M_{v,z} \wedge$
>
> $perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_o, \mathbf{v}_o, \mathbf{s}'_o, \mathbf{v}'_o) \wedge perturb?(\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}, \mathbf{s}_i, \mathbf{v}_i, \mathbf{s}'_i, \mathbf{v}'_i) \wedge$
>
> $Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \implies \neg Priority(\mathbf{s}'_i, \mathbf{v}'_i, \mathbf{s}'_o, \mathbf{v}'_o)$

In this definition, the positive real numbers $M_{s,xy}, M_{v,xy}, M_{s,z}, M_{v,z}$ are bounds on the input data, and the parameters $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}$ are bounds on the size of the perturbations allowed in the data, which will change as the bounds $M_{s,xy}, M_{v,xy}, M_{s,z}, M_{v,z}$ change. That is, as the values of $M_{s,xy}, M_{v,xy}, M_{s,z}, M_{v,z}$ get larger, more possible input states will be required to satisfy the stability property, so the maximum perturbation sizes $\epsilon_{s,xy}, \epsilon_{v,xy}, \epsilon_{s,z}, \epsilon_{v,z}$ will often decrease.

In general, the priority functions that satisfy the predicate $basic\_stability?$ are not the same as those that satisfy the predicate $uniform\_stability?$, even when they also satisfy $safe?$. For instance, the function $Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = (s_{oz} > s_{iz})$ satisfies $safe?$ and $basic\_stability?$ but not $uniform\_stability?$. The function $Priority(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = (\mathbf{s}_o = \mathbf{s}_i + (10, 10, 0) \; \wedge \; \mathbf{v}_o = \mathbf{v}_i + (100, 100, 0))$ satisfies $safe?$ and $uniform\_stability?$ but not $basic\_stability?$. In Section V, a specific priority function is proposed that satisfies all three predicates: $safe?$, $basic\_stability?$, and $uniform\_stability?$. Theorem 3 implies that this priority function is therefore equal to $Priority_f$ for a continuous function f, which is explicitly defined in Section V.

IV.   Analysis of 91.113 Priority Rules

The right-of-way rules for air traffic given in Title 14 of the Code of Federal Regulations, Part 91.113,[2] can be formalized with a priority function as follows. This is a relatively simple formalization of the rules in 91.113 and is intended to illustrate the usefulness of the formal framework for analyzing priority rules

American Institute of Aeronautics and Astronautics

described in the previous sections. While many of the rules in 91.113 are operational, such as the rule that a hot air balloon always has priority over any other category of aircraft, the rules for aircraft of the same category flying at the same altitude are based on the states of the aircraft. In particular, rules D, E, and F in 91.113 state the following.

- (D) When aircraft of the same category are converging at approximately the same altitude (except head-on, or nearly so), the aircraft to the other's right has the right-of-way.

- (E) When aircraft are approaching each other head-on, or nearly so, each pilot of each aircraft shall alter course to the right.

- (F) Each aircraft that is being overtaken has the right-of-way.

These statements can be formalized as follows. Horizontal convergence between the aircraft is equivalent to the following condition.

Definition 7. $\texttt{horiz\_converging?}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv (\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot (\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)}) < 0$

If this condition is true, then the aircraft are getting closer at the states represented by $\mathbf{s}_o$, $\mathbf{v}_o$ and $\mathbf{s}_i$, $\mathbf{v}_i$. Whether the intruder is on the right of the ownship can be formalized as:

Definition 8. $\texttt{on\_right?}(\mathbf{s}_o, \mathbf{v}_o)(\mathbf{s}_i, \mathbf{v}_i) \equiv (\mathbf{s}_{i(x,y)} - \mathbf{s}_{o(x,y)}) \cdot \mathbf{v}_{o(x,y)}^{\perp} > 0$

If the aircraft are approaching, either head-on or with one overtaking the other, then the horizontal relative position vector between the aircraft, namely $\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}$ is parallel to the both $\mathbf{v}_{o(x,y)}$ and $\mathbf{v}_{i(x,y)}$, which are therefore parallel to each other. The condition that distinguishes between the head-on case and the overtaking case is whether $\mathbf{v}_{o(x,y)}$ and $\mathbf{v}_{i(x,y)}$ point in the same direction or in opposite directions. Conditions defining the head-on and overtaking cases are given as follows.

Definition 9.
$$\begin{aligned}
\texttt{head\_on?}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \;\equiv\;& (\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot \mathbf{v}_{o(x,y)}^{\perp} = 0 \;\wedge \\
& (\mathbf{s}_{i(x,y)} - \mathbf{s}_{o(x,y)}) \cdot \mathbf{v}_{i(x,y)}^{\perp} = 0 \;\wedge \\
& (\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot \mathbf{v}_{o(x,y)} < 0 \;\wedge \\
& (\mathbf{s}_{i(x,y)} - \mathbf{s}_{o(x,y)}) \cdot \mathbf{v}_{i(x,y)} < 0
\end{aligned}$$

Definition 10.
$$\begin{aligned}
\texttt{overtaking?}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \;\equiv\;& (\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot \mathbf{v}_{o(x,y)}^{\perp} = 0 \;\wedge \\
& (\mathbf{s}_{i(x,y)} - \mathbf{s}_{o(x,y)}) \cdot \mathbf{v}_{i(x,y)}^{\perp} = 0 \;\wedge \\
& (\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot \mathbf{v}_{o(x,y)} < 0 \;\wedge \\
& (\mathbf{s}_{i(x,y)} - \mathbf{s}_{o(x,y)}) \cdot \mathbf{v}_{i(x,y)} > 0
\end{aligned}$$

Note that the two conditions defined above are identical except for the last inequality symbol. The condition given by `overtaking?` states that the ownship is overtaking the intruder. One way to formally interpret rules D, E, and F in 91.113 is to say that the ownship has priority precisely when it is to the right of, or being overtaken by, the intruder. That is, we could define

$$\texttt{rules\_91.113\_first\_try}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv \texttt{on\_right?}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o) \;\vee\; \texttt{overtaking?}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o).$$

This certainly seems like a reasonable interpretation of 91.113. However, there is a problem:

Theorem 4. *The priority function* `rules_91.113_first_try` *allows cases where both aircraft have priority. That is,* `safe?(rules_91.113_first_try)` *does not hold.*

A specific case where both aircraft have priority when using this priority function is given in Figure 2. In this figure, each aircraft is to the right of the other aircraft, and would therefore have priority. It is interesting that it is unclear from the language in 91.113 which aircraft would have priority in this case.

One way to fix this problem is to only give the ownship priority when it is either being overtaken or is to the intruder's right and does not have the intruder to its right. The problem in that case is that it is not effective for states where both aircraft are on the other's right. Thus, there is a large set of states where neither aircraft has priority.
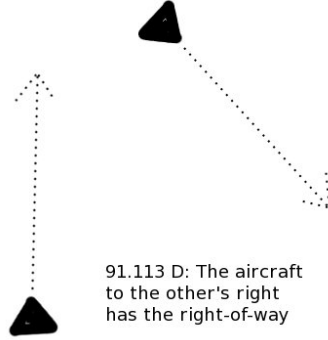
American Institute of Aeronautics and Astronautics

Figure 2. Each Aircraft is On the Other's Right

Another way to fix the effectiveness and safety issues inherent in the formalizations of 91.113 provided above is to project the states of the aircraft backward in time to when exactly one aircraft is on the other's right and apply the rules at that time. This is possible because in the cases where each aircraft is to the other's right, the intersection point of their trajectories has already occurred. Thus, projecting the states backward to a time before the aircraft crossed that intersection point will give a past state at which exactly one aircraft was on the other's right. The rules can then be applied at that state, where exactly one aircraft will have priority. It can be seen from standard geometric reasoning that the following function determines whether the intruder was on the ownship's right at this past state.

Definition 11. $\mathtt{was\_on\_right?}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv \mathbf{v}_{o(x,y)} \cdot \mathbf{v}_{i(x,y)}^{\perp} > 0$

Using this condition instead of the `on_right?` condition in the formalization of the 91.113 rules gives the following priority function.

$$\mathtt{rules\_91.113}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv \mathtt{was\_on\_right?}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o) \ \lor \ \mathtt{overtaking?}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o).$$

It is easy to see that $\mathtt{was\_on\_right?}(\mathbf{s}_i, \mathbf{v}_i, \mathbf{s}_o, \mathbf{v}_o)$ and $\mathtt{was\_on\_right?}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ cannot both hold at the same time, which partly implies the following theorem.

Theorem 5. *The priority function* **rules_91.113** *never allows cases where both aircraft have priority. That is, the condition* **safe?**(**rules_91.113**) *holds.*

The function `rules_91.113` is also effective for all but a few cases.

Theorem 6. *The priority function* **rules_91.113** *is effective for inputs states where the aircraft are converging, not in a head-on conflict, and not on trajectories that are both parallel and non-intersecting.*

Interestingly, the function `rules_91.113` does not satisfy the stability condition `basic_stability?` (`rules_91.113`). However, the failure of stability only happens in the overtaking case. That is, if the intruder is overtaking the ownship, and hence the ownship has priority, a tiny perturbation of the data can produce states where the condition `was_on_right?` holds, meaning that the intruder is (or *was*) to the right of the ownship, giving the intruder priority in those states.

V. A Proposed Priority Function

The previous section showed that the priority rules in Part 91 of the Federal Aviation Regulations can indeed be formalized mathematically, and that important properties can be verified about them. This section shows that those rules are not the only useful rules that have geometric intuition behind them. Another priority function is presented that satisfies the predicates `safe?`, `basic_stability?`, and `uniform_stability?`, and it agrees with the current rules in 91.113 in some important scenarios. This is accomplished in two stages: (1) An informal description of the rules, and (2) a precise mathematical definition of the priority function.

American Institute of Aeronautics and Astronautics

A. An Informal Description of the Rules

The rules are presented below, in an informal and non-mathematical description of the rules. These rules are intended for a self-separation application[7] and may be applied at a much longer time horizon than 91.113. For this reason, these rules project the future state of the aircraft just before the encounter.

- If the two aircraft have different flight modes (ascending/descending/cruising), and neither of the aircraft is near a vertical speed that would change this fact, then the aircraft with the lower vertical speed has priority.

- If either aircraft is near a vertical speed where it would change flight modes and either the current or the new mode is possibly equal to the mode of the other aircraft, then neither aircraft has priority.

- Pick a future time just before the encounter, and consider the states of each aircraft at that time. At these future states, calculate the projections of each aircraft's horizontal velocity along the path from its position toward the position of the other aircraft. If the lengths (speeds) of these projections are close, then neither aircraft has priority. Otherwise, the aircraft whose projection has the smallest speed has priority. See Figure 3.
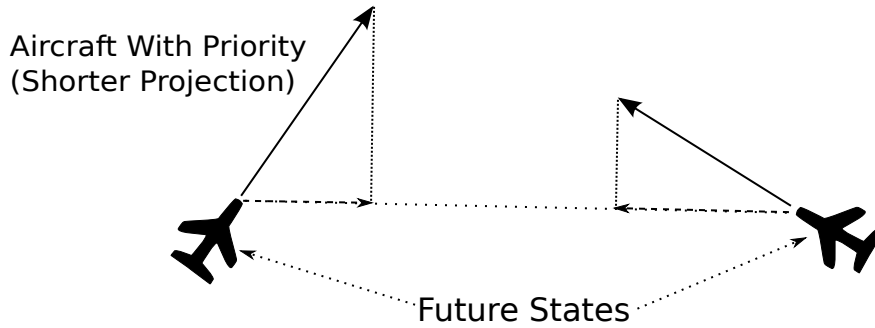

Figure 3. **Determining Priority Based on Projections**

B. Flight Modes

As indicated by the informal rules above, the term *flight mode* in this paper refers to either descending, ascending, or cruise. The definition of cruising is that the absolute value of the vertical speed is no greater than a fixed value (e.g., 150 ft/minute), treated here as a parameter VCB, which is short for *vertical cruise buffer*. If the state of the ownship aircraft is given by the position and velocity vectors $\mathbf{s}_o, \mathbf{v}_o$, then this means that the ownship is descending if $v_{oz} < -\text{VCB}$, ascending if $v_{oz} > \text{VCB}$, and cruising if $|v_{oz}| \leq \text{VCB}$. However, due to perturbations in the data, when computing priority the ownship and intruder may be using slightly different data. Thus, if v_{oz} is just slightly greater than $-\text{VCB}$, the ownship may be in cruise mode, while the data used by the intruder implies that the vertical speed of the ownship is less than $-\text{VCB}$, in which case the intruder thinks that the ownship is descending. If this change has an impact on which aircraft has priority, then uniform stability of the rules may be violated. Thus, the concept of being *near* this decision point (cruising/descending) is introduced, and a new parameter VSafetyBuffer is introduced to define what nearness to this decision point means mathematically. The parameter VSafetyBuffer is a positive real number (VSafetyBuffer > 0) such that VSafetyBuffer $<$ VCB.

If the ownship's vertical speed v_{oz} is within VSafetyBuffer of $-\text{VCB}$, then it is near the vertical speed where it will change between cruising and descending. This motivates the definitions of the following predicates.

$$\texttt{descending}(v_{oz}) \equiv (v_{oz} < -\text{VCB} - \text{VSafetyBuffer})$$

$$\texttt{cruising}(v_{oz}) \equiv (|v_{oz}| < \text{VCB} - \text{VSafetyBuffer})$$

$$\texttt{ascending}(v_{oz}) \equiv (v_{oz} > \text{VCB} + \text{VSafetyBuffer})$$

The number VSafetyBuffer basically gives the amount of uncertainty that is allowed in the vertical speed of an aircraft. That is, if the vertical speed varies no more than VSafetyBuffer, then it is known, for instance,

American Institute of Aeronautics and Astronautics

that an aircraft is definitely descending if its vertical speed is less than $-\texttt{VCB} - \texttt{VSafetyBuffer}$. On the other hand, to know with certainty that an aircraft is not descending, its vertical speed should be greater than $-\texttt{VCB} + \texttt{VSafetyBuffer}$. Similar reasoning can be used to determine conditions under which it is known that an aircraft is not ascending or not cruising. These conditions are all specified by the following *negative* predicates.

$$\texttt{not\_descending}(v_{oz}) \equiv (v_{oz} > -\texttt{VCB} + \texttt{VSafetyBuffer})$$
$$\texttt{not\_cruising}(v_{oz}) \equiv (v_{oz} > \texttt{VCB} + \texttt{VSafetyBuffer} \ \lor \ v_{oz} < -\texttt{VCB} - \texttt{VSafetyBuffer})$$
$$\texttt{not\_ascending}(v_{oz}) \equiv (v_{oz} < \texttt{VCB} - \texttt{VSafetyBuffer})$$

If it is known that the ownship and the intruder are currently in different flight modes, for instance that the ownship is ascending and the intruder is descending, then the aircraft with the lower vertical speed will have priority. This is consistent with rule (G) of part 91.113, which states that when an aircraft is landing, it has priority over other aircraft in flight.

Thus, the vertical rules will apply if it can be determined using the predicates defined above that the aircraft have different modes. This motivates the definition of the following predicate, which returns true precisely when it is known that the vertical rules will apply.

$$\begin{aligned}
\texttt{vertical\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i) \equiv &(\texttt{descending}(v_{oz}) \ \land \ \texttt{not\_descending}(v_{iz})) \lor \\
&(\texttt{descending}(v_{iz}) \ \land \ \texttt{not\_descending}(v_{oz})) \lor \\
&(\texttt{cruising}(v_{oz}) \ \land \ \texttt{not\_cruising}(v_{iz})) \lor \\
&(\texttt{cruising}(v_{iz}) \ \land \ \texttt{not\_cruising}(v_{oz})) \lor \\
&(\texttt{ascending}(v_{oz}) \ \land \ \texttt{not\_ascending}(v_{iz})) \lor \\
&(\texttt{ascending}(v_{iz}) \ \land \ \texttt{not\_ascending}(v_{oz}))
\end{aligned}$$

Alternatively, if it is known, given uncertainties, that both aircraft have the same mode, then a horizontal rule can apply. The following predicate determines when this is the case.

$$\begin{aligned}
\texttt{horizontal\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i) \equiv &(\texttt{descending}(v_{oz}) \ \land \ \texttt{descending}(v_{iz})) \lor \\
&(\texttt{cruising}(v_{oz}) \ \land \ \texttt{cruising}(v_{iz})) \lor \\
&(\texttt{ascending}(v_{oz}) \ \land \ \texttt{ascending}(v_{iz}))
\end{aligned}$$

It is also possible that neither $\texttt{vertical\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i)$ nor $\texttt{horizontal\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i)$ holds, in which case one aircraft has a vertical speed within $\texttt{VSafetyBuffer}$ of either $\texttt{VCB}$ or $-\texttt{VCB}$, its mode is therefore uncertain, and it may or may not have the same mode as the other aircraft. The aircraft in such a situation are said to be in a *vertical deadband* in this paper.

C.   The Mathematical Definition of the Priority Function

This section uses the flight modes presented above to build a priority function, P. As described in the first bullet of section A, the aircraft with the smaller vertical speed has priority, if the vertical rules apply. If the horizontal rules apply then more formulas are needed, which are presented in this section.

If $\texttt{horizontal\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i)$ holds, then a rule is needed to determine priority that is based on the horizontal geometry of the situation. The idea is to pick future states of the aircraft that capture the geometry of the encounter. At these future states, the projections of each aircraft's horizontal velocity along the path from its position to the position of the other aircraft are calculated. If the lengths (speeds) of these projections are close, then neither aircraft has priority. Otherwise, the aircraft whose projection has the smallest speed has priority. See Figure 3.

If $\mathbf{s}_o, \mathbf{v}_o$ and $\mathbf{s}_i, \mathbf{v}_i$ represent these future states of the aircraft, then, in most cases, the ownship has priority precisely when

$$\texttt{velcomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) > 0, \tag{1}$$

where $\texttt{velcomp}$ is the function defined by

$$\texttt{velcomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = \begin{cases} \frac{\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}}{\|\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}\|} \cdot \left(\mathbf{v}_{o(x,y)} + \mathbf{v}_{i(x,y)}\right) & \text{if } \mathbf{s}_o \neq \mathbf{s}_i \\ 0 & \text{if } \mathbf{s}_o = \mathbf{s}_i. \end{cases}$$

American Institute of Aeronautics and Astronautics

Formula (1) is equivalent to the condition that the projection of the intruder's horizontal velocity, $\mathbf{v}_{i(x,y)}$, toward the ownship has greater length than that of the ownship's velocity, $\mathbf{v}_{o(x,y)}$, toward the intruder (see Figure 3). In fact, the quantity $\mathtt{velcomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ is equal to the projection of the intruder's horizontal velocity minus that of the ownship.

The future states of the aircraft that will be used in the function $\mathtt{velcomp}$ to determine priority can be the future states at any future time, assuming that the ownship and intruder agree on the states. The goal in picking these future states is that they will be representative of the geometry of the encounter and can be used to determine priority through geometric reasoning.

The time when the aircraft achieve minimum horizontal separation is known as the time of closest horizontal approach.[3] The proposed priority rules will apply the function $\mathtt{velcomp}$ at a time slightly before the time of closest horizontal approach. The states of the aircraft just before this time are geometrically representative of the severity of the encounter. The amount of time before the time of closest horizontal approach, when these states will be chosen, will be referred to by a parameter $\mathtt{tc}$. It can be set to any positive real time.

If the aircraft are converging and flying at constant velocities with initial positions of $\mathbf{s}_o$ and $\mathbf{s}_i$, then the time of closest horizontal approach is equal to $(1/\|(\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)})\|^2)(\mathbf{s}_o - \mathbf{s}_i) \cdot (\mathbf{v}_o - \mathbf{v}_i)$. If they are not converging, it is equal to 0. This motivates the definition of the function $\mathtt{tca}$, which computes the time of closest horizontal approach, given by

$$\mathtt{tca}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = \max(0, \frac{-(\mathbf{s}_{o(x,y)} - \mathbf{s}_{i(x,y)}) \cdot (\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)})}{\|\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)}\|^2})$$

when $\mathbf{v}_o \neq \mathbf{v}_i$ and given by $\mathtt{tca}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = 0$ when $\mathbf{v}_{o(x,y)} = \mathbf{v}_{i(x,y)}$. It is at the time $\mathtt{tca}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) - \mathtt{tc}$ when the state of each aircraft will be chosen for determining priority, where $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i$ represent the states of the aircraft at time zero (the current time). However, if the horizontal components $\mathbf{v}_{o(x,y)}$ and $\mathbf{v}_{i(x,y)}$ of the aircraft's velocities are almost equal, in which case the aircraft are flying with nearly the same direction and speed, then the time $\mathtt{tca}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ is not stable. In this case, neither aircraft should have priority, and therefore both should maneuver.

This motivates the definition of a parameter $\mathtt{LRelDB}$ (for "lateral relative deadband"), representing a *relative speed deadband* on the relative horizontal speed between the aircraft. If the horizontal rules apply, and if $\|\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)}\| < \mathtt{LRelDB}$, then neither aircraft will have priority. In general, priority when the horizontal rules apply will be determined by the following function.

$$\mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) = \begin{cases} 0 & \text{if } \|\mathbf{v}_{o(x,y)} - \mathbf{v}_{i(x,y)}\| \leq \mathtt{LRelDB} \\ \mathtt{velcomp}(\mathbf{s}_o + t^*\mathbf{v}_o, \mathbf{v}_o, \mathbf{s}_i + t^*\mathbf{v}_i, \mathbf{v}_i) & \text{otherwise} \end{cases}$$

where $t^* = \max(0, \mathtt{tca}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) - \mathtt{tc})$. In the second part of this equation, the states of each aircraft are projected ahead to the time t^*.

If $\mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ is positive, then it will typically be the case that the ownship has priority. However, if it is very closed to zero, then priority may be close to changing between one aircraft and another, so to ensure uniform stability, both aircraft should be burdened. This is accomplished through another parameter $\mathtt{HorizDB}$, representing a *horizontal deadband*. Basically, if the absolute value of $\mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)$ is no greater than $\mathtt{HorizDB}$, then the aircraft are in a horizontal deadband, and neither aircraft has priority. This motivates the definition of the following lateral priority function.

$$\mathtt{lateral\_priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv \mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) > 0 \ \wedge \ |\mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)| > \mathtt{HorizDB}$$

The latter part of this definition (after the *and* sign, $\wedge$) implies that neither aircraft has priority in the horizontal case when $|\mathtt{entrycomp}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)| \leq \mathtt{HorizDB}$.

The function $\mathtt{P}$, defined below, gives a mathematical definition of a proposed priority function. When it is known that both aircraft have the same flight mode, priority is determined by the function $\mathtt{lateral\_priority}$.

$$\begin{aligned} \mathtt{P}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i) \equiv &(\mathtt{vertical\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i) \wedge \ v_{oz} < v_{iz}) \\ &\vee \ (\mathtt{horizontal\_rules\_apply}(\mathbf{v}_o, \mathbf{v}_i) \ \wedge \ \mathtt{lateral\_priority}(\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i)) \end{aligned}$$

Below, specific examples are given of priority between aircraft in some common encounter situations. It is important to note that if neither the horizontal rules apply nor the vertical rules apply, then this function returns false. In such a case, it is unclear, based on perturbations in the data, which rule should apply, so neither aircraft has priority.

American Institute of Aeronautics and Astronautics

D.   Properties of the Proposed Priority Function

There are several important properties listed in Section II for a priority function. These properties are considered here for the priority function P, which is defined above. The following theorem implies that the function P never allows a case where both aircraft have priority.

Theorem 7. *The priority function P satisfies* `safe?`*(P).*

Even with small changes to the inputs, the priority function P produces compatible outputs:

Theorem 8. *The priority function P satisfies* `basic_stability?`*(P) and* `uniform_stability?`*(P).*

As noted in Theorem 1 in Section III, it is impossible for any function to satisfy the conditions `safe?`, `stable?`, and be effective in the sense that exactly one aircraft is burdened. Thus, while the priority function P was constructed with this limitation in mind, the next theorem follows from Theorem 1.

Theorem 9. *The priority function P is not effective. That is, in some cases, neither aircraft has priority.*

One way that neither aircraft has priority is if the horizontal projections of each aircraft's velocity toward the other aircraft, at the states where priority is determined, are equal or nearly equal. In such a case, the aircraft are in a horizontal deadband. For such a case, see Figure 4. The function P attempts to minimize the number of such cases by reducing the number of places where deadbands are used.
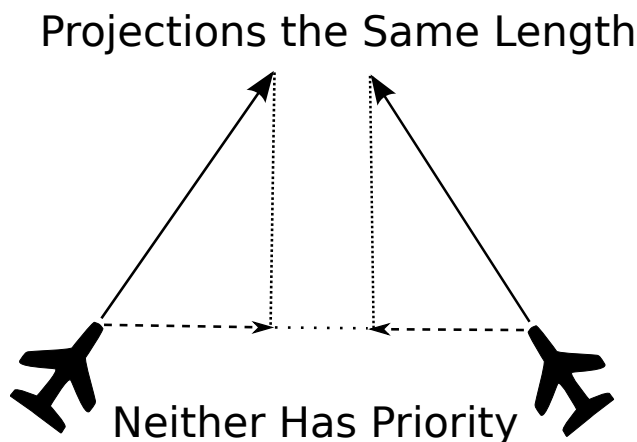

Figure 4. A Horizontal Deadband

The function P can also be compared with the current rules as outlined in 91.113. In particular, it can be compared with the geometric rules in 91.113 that are listed at the beginning of Section IV. When aircraft of the same category are converging at approximately the same altitude (except head-on, or nearly so), rule D in 91.113 gives the priority to the aircraft to the other's right. The priority function P does not agree with this rule. In fact, either aircraft could have priority in this case if the function P is used.

When aircraft are approaching each other head-on, rule E in 91.113 gives neither aircraft priority. However, the function P sometimes allows one aircraft to have priority in this case. When aircraft are approaching head on, the slower aircraft usually has priority, except when the speeds of the aircraft are close, in which case the function agrees with rule E and neither aircraft has priority.

When one aircraft is overtaking another, rule F in 91.113 gives priority to the aircraft that is being overtaken. In this case, the function P agrees with the current rule, as indicated in Figure 5.

VI.   Concluding Remarks

This paper analyzes priority rules, such as those in Title 14 of the Code of Federal Regulations, Part 91.113[2] that determine which of two aircraft on conflicting trajectories should maneuver. The analysis enables new concepts of operation for NextGen, such as self-separation, which may involve different priority rules than the current, well-accepted rules found in 91.113. A mathematical framework is presented for analyzing such rules, with specific properties in mind, such as safety, effectiveness, and stability. A theoretical result
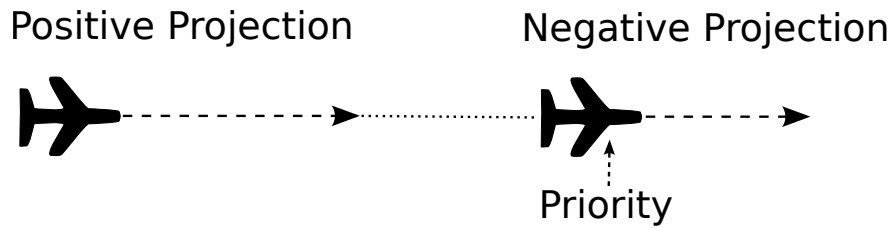
American Institute of Aeronautics and Astronautics

Positive Projection   Negative Projection

Priority

Figure 5. An Aircraft Being Overtaken

is presented that states that it is impossible for a priority function to be safe (never do both aircraft have priority), stable (compatible results occur even under slight perturbations to input data), and effective (exactly one aircraft has priority in every case).

This paper analyzes, using the mathematical framework developed, the current right-of-way rules for air traffic given in 91.113. The first formalization of these rules, presented in Section IV, did not immediately satisfy the key safety property of priority rules, which states that at least one aircraft has priority in every situation. That is, it is possible that two aircraft are converging, not in a head-on situation, and that each aircraft is on the other's right. In this case, it is unclear from the rules in 91.113 which aircraft should have priority. Another formalization is given for those rules that fixes this problem, by interpreting rule D in 91.113 in a particular way. Thus, a specific interpretation of the subjective rules in 91.113 is required to ensure that there are no cases where both aircraft have priority.

Finally, a new priority rule function is presented that is both safe and stable, which, by the theorem relating these two properties to effectiveness, automatically allows some cases where neither aircraft has priority. The development of this function shows that the rules in 91.113 are not the only rules that satisfy important operational and safety properties. The proposed priority function essentially works as follows. If the two aircraft have different flight modes (ascending/descending/cruising), and neither of the aircraft is near a vertical speed that would change this fact, then the aircraft with the lower vertical speed has priority. If either aircraft is near a vertical speed where it would change flight modes and either the current or the new mode is possibly equal to the mode of the other aircraft, then neither aircraft has priority. Otherwise, pick a future time just before the encounter, and consider the states of each aircraft at that time. At these future states, calculate the projections of each aircraft's horizontal velocity along the path from its position toward the position of the other aircraft. If the lengths (speeds) of these projections are close, then neither aircraft has priority. Otherwise, the aircraft whose projection has the smallest speed has priority.

The analysis of rules in 91.113 and the proposed priority function show that the mathematical framework is general enough to verify an arbitrary set of priority rules. Future work will focus on comparing the proposed priority function with 91.113, through simulation, to determine the effects that their differences have on airspace operations.

References

[1]Mark G. Ballin, Vivek Sharma, Robert A. Vivona, Edward J. Johnson, and Ermin Ramiscal. A flight deck decision support tool for autonomous airborne operations. In *Proceedings of the AIAA Guidance Navigation, and Control Conference*, 2002.

[2]United States Government. General operating and flight rules, 14CFR91.113, 2012.

[3]César Muñoz and Anthony Narkawicz. Time of closest approach in three-dimensional airspace. Technical Memorandum NASA/TM-2010-216857, NASA, Langley Research Center, Hampton VA 23681-2199, USA, October 2010.

[4]Anthony Narkawicz and César Muñoz. State-based implicit coordination and applications. Technical Publication NASA/TP-2011-217067, NASA, Langley Research Center, Hampton VA 23681-2199, USA, March 2011.

[5]RTCA, DO-242A. Minimum aviation system performance standards for Automatic Dependent Surveillance Broadcast (ADS-B), June 2002.

[6]RTCA, DO-263. Application of airborne conflict management: Detection, prevention, and resolution, December 2000.

[7]David J. Wing and William B. Cotton. Autonomous flight rules: A concept for self-separation in U.S. domestic airspace. Technical Publication NASA/TP-2011-217174, NASA, Langley Research Center, Hampton VA 23681-2199, USA, November 2011.

[8]David J. Wing, Robert A. Vivona, and David A. Roscoe. Airborne tactical intent-based conflict resolution capability. In *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Hilton Head, South Carolina, USA, September 2009.