Air Force Institute of Technology

AFIT Scholar

7-6-2017

Implementing Conditional Inequality Constraints for Optimal Collision Avoidance

Nathan E. Smith

Christopher D. Arendt
Air Force Office of Scientific Research

Richard G. Cobb
Air Force Institute of Technology

Jonah A. Reeger

Implementing Conditional Inequality Constraints for Optimal Collision Avoidance

Smith NE[1]*, Arendt CD[2], Cobb RG[3] and Reeger JA[4]

[1]Air Force Seek Eagle Office, Eglin AFB, FL, 32542, USA
[2]Air Force Office of Scientific Research, Arlington, VA, USA
[3]Department of Aeronautics and Astronautics, Air Force Institute of Technology, Ohio, USA
[4]Department of Mathematics and Statistics, Air Force Institute of Technology, Ohio, USA

Abstract

Current Federal Aviation Administration regulations require that passing aircraft must either meet a specified horizontal *or* vertical separation distance. However, solving for optimal avoidance trajectories with conditional inequality path constraints is problematic for gradient-based numerical nonlinear programming solvers since conditional constraints typically possess non-differentiable points. Further, the literature is silent on robust treatment of approximation methods to implement conditional inequality path constraints for gradient-based numerical nonlinear programming solvers. This paper proposes two efficient methods to enforce conditional inequality path constraints in the optimal control problem formulation and compares and contrasts these approaches on representative airborne avoidance scenarios. The first approach is based on a minimum area enclosing superellipse function and the second is based on use of sigmoid functions. These proposed methods are not only robust, but also conservative, that is, their construction is such that the approximate feasible region is a subset of the true feasible region. Further, these methods admit analytically derived bounds for the over-estimation of the infeasible region with a demonstrated maximum error of no greater than 0.3% using the superellipse method, which is less than the resolution of typical sensors used to calculate aircraft position or altitude. However, the superellipse method is not practical in all cases to enforce conditional inequality path constraints that may arise in the nonlinear airborne collision avoidance problem. Therefore, this paper also highlights by example when the use of sigmoid functions are more appropriate.

Keywords: Aviation; Collision; Inequality

Nomenclature

χ: Heading angle (radians); Δxy: Horizontal separation distance (ft); Δz: Vertical separation distance (ft); δ: Overestimation error; ∈: Sigmoid exponential; γ: Flight path angle (radians); λ: Lagrange multiplier; g: Inequality constraint; **u:** Control; x: State trajectory; μ: Bank angle (radians); ϕ: Terminal cost function; ψ: Equality constraint; τ: Time transformation; g: Gravity constant (ft/sec²); H: Hamiltonian; h: Horizontal constraint (ft); J: Performance measure; L: Cost function; N: Superellipse exponential; Nz: Normal acceleration; S: Sigmoid values; s: Sigmoid stiffness; T_{CPA}: Time to closest point of approach (CPA) (sec); V: Ground speed (ft/sec); v: Vertical constraint (ft)

Introduction

Under the Federal Aviation Administration (FAA) Modernization and Reform Act of 2012, the United States Congress tasked the FAA to "provide for the safe integration of civil unmanned aircraft systems into the national airspace system" [faa_reform_act]. A means to meet this integration mandate is through the use of algorithms that autonomously generate optimal collision avoidance trajectories to satisfy current FAA regulations that mandate passing aircraft meet either a minimum horizontal *or* vertical separation distance. A number of works have looked at trajectory planning and optimization for air vehicles using optimal control problem formulations Raghunathan [1], Eele [2], Horn [3] and some, such as Geiger [4] have even demonstrated this method in flight on a small-size unmanned vehicle. However, a potential limitation of this method is enforcing conditional inequality constraints such as maintaining either a minimum horizontal or vertical separation distance from an approaching aircraft or complying with FAA right of way (ROW) rules. For example, according to FAR 91.113 if two aircraft are approaching nearly head on, then "each aircraft shall alter course to the right." Optimal control problems are often solved using gradient-

based numerical nonlinear programming (NLP) solvers which require smooth differentiable constraints; however, conditional constraints are not always differentiable, and thus can cause gradient-based numerical solvers to fail. This paper proposes and analyzes two different methods to address the issue of non-differentiable conditional inequality path constraints. The first approach is based on a minimum area enclosing superellipse (MAES) function and the second is based on the use of sigmoid functions. Both of these approaches are differentiable, allowing the NLP solver to calculate gradients and find an optimal solution.

Standard methods for implementing conditional inequality constraints can be classified as indicator methods, including Big M Borrelli [5], Winston [6] and active set Hintermuller [7] methods, and mixed-norm methods Sadovsky [8]; however, these methods are not everywhere-differentiable, and therefore, they often cause gradient-based NLP solvers to fail to generate an optimal solution. For instance, Big M methods implement "either-or constraints" Winston [6] using a binary indicator variable along with a sufficiently large constraint variable (M); thus, constraints become non-differentiable with respect to the binary indicator variable. Similarly, active set methods use the conditional constraints to define sub-sets of feasible solutions and then optimize over each sub-set when it is indicated as the active set. However,

*Corresponding author: Smith NE, Colonel, USAF, Ph.D, Air Force Seek Eagle Office, Eglin AFB, FL, 32542, USA, Tel: 661-816-4455; E-mail: nathan.smith@us.af.mil

these methods are not well suited for dynamic conditional constraints such as the time varying airborne collision avoidance problem since the continuous-time constraints implicitly define an uncountable number of feasible sub-sets, while discretizing the constraints introduces an implicit or explicit binary indicator variable equivalent to those in Big M methods [7]. In addition, mixed-norm methods typically formulate a set of conditional constraints as a single constraint involving the maximum of a set of norms from each conditional constraint Sadovsky [8]; thus, the constraint's derivative at a point is a function of the derivative of the norm that obtains the maximum value at that point. Therefore, if the mixed set of norms do not have identical derivatives at points where the maximum norm changes from one norm to another in the set, the mixed-norm formulation will not be everywhere-differentiable. In the context of collision avoidance, Raghunathan [1] devised a novel approach for enforcing a conditional inequality constraint of maintaining either a minimum horizontal or vertical separation distance from an approaching aircraft; however, their approach did not address situations with more than two conditional constraints and required the introduction of an additional control variable appended to the objective function. An approach that is similar to the methods in this paper is known as artificial potential fields or functions, or APF. While APF methods are differentiable Ren [9], Paul [10] they do not truly enforce conditional inequality constraints. Instead, APF methods treat path constraints as "soft" obstacles and incorporate them as weighted penalties in the cost function which may result in generating infeasible trajectories [9]. However, the methods proposed in this paper provide conservative and differentiable approximations for indicator methods as well as mixed-norm methods, thus ensuring differentiability for the gradient-based NLP solver while maintaining feasibility for the optimal control problem.

The overview of the paper is as follows: First we introduce and develops the MAES and sigmoid conditional constraint approximation methods. The next sections describe and then analyze the simulation results from the two example problems in this paper. The final section summarizes the results.

Conditional Inequality Constraint Approximation Methods

This section begins by introducing the first of the two example problems in this paper to properly motivate the development of the conditional constraint formulation methods presented herein. In the first example problem, the objective is for the ownship to minimize deviations from a 3D flight path corridor while maintaining either a horizontal separation distance (Δxy) of at least 2460 ft *or* a vertical separation distance (Δz) of at least 820 ft from an intruder aircraft where[1]:

$$\Delta xy = \sqrt{(x_{intruder} - x_{ownship})^2 + (y_{intruder} - y_{ownship})^2} \qquad (1)$$

$$\Delta z = | z_{intruder} - z_{ownship} | \qquad (2)$$

Using logic *'if statements'*, this inequality constraint formulation appears algorithmically as:

for each collocation node $i= 1$ to n

if $\Delta xy (i) > 2460$

$h_{sep}(i)=1$

---

end

if $\Delta z(i)>820$

$v_{sep}(i)=1$

end

end

$$1 - [h_{sep}+v_{sep}] \leq 0 \quad \text{inequality path constraint}, \in \mathbb{R}^n \qquad (3)$$

This definition for the conditional inequality path constraint in equation (3) was evaluated using two different gradient-based NLP solvers, IPOPT and SNOPT; however, as expected, due to the non-differentiable conditional constraint caused by the logic *'if statements'*, both solvers failed to converge to a solution since they could not determine a gradient direction to search in order to find an extremal point. Therefore, an alternate approach is needed to enforce a conditional constraint without the use of logic *'if statements'*. The two approaches analyzed in this paper are (1) MAES and (2) sigmoid functions to approximate inequality path constraints for the optimal control problem. The next sections describe these two approaches.

Minimum area enclosing superellipse (MAES)

The equation for a superellipse appears as:

$$\left(\frac{x}{a}\right)^N + \left(\frac{y}{b}\right)^N = 1 \qquad (4)$$

where a and b represent the semi-major and semi-minor axes of the superellipse while $N \geq 2$ is an even number Weisstein [11]. The equation for the area of a superellipse Farrell [12] appears as:

$$Area=4abc\,(N) \qquad (5)$$

where $C(N)$ is a ratio of gamma functions of N defined as described by Farrell:

$$C(N) = \frac{\left(\Gamma\left(1+\frac{1}{N}\right)\right)^2}{\Gamma\left(1+\frac{2}{N}\right)} \qquad (6)$$

Additionally, a superellipse that encloses a rectangle with length ($2h$) and width ($2v$) centered at the origin must intersect each of the 4 corners of the rectangle. That is, the minimum area enclosing superellipse must satisfy,

$$\begin{aligned}
\left(\frac{+h}{a}\right)^N + \left(\frac{+v}{b}\right)^N &= 1 \\
\left(\frac{-h}{a}\right)^N + \left(\frac{+v}{b}\right)^N &= 1 \\
\left(\frac{+h}{a}\right)^N + \left(\frac{-v}{b}\right)^N &= 1 \\
\left(\frac{-h}{a}\right)^N + \left(\frac{-v}{b}\right)^N &= 1
\end{aligned} \qquad (7)$$

Since N must always be an even number, all 4 constraint expressions in equation (7) are equivalent. Furthermore, since $4C(N)$ in equation (5) is not a function of a or b, the optimization problem to determine the semi-major and semi-minor axes values, a* and b* respectively, that minimize the area of an enclosing superellipse for any even $N \geq 2$ reduces to:

Minimize ab

such that $(\frac{h}{a})^N + (\frac{v}{b})^N = 1$, where $a, b \geq 0$

It is easy to verify that the values $a^* = 2^{1/N} h$ and $b^* = 2^{1/N} v$ satisfy the first-order KKT and second-order sufficiency conditions for optimality. Therefore, the equation of the minimum area superellipse that encloses the minimum separation rectangle appears as:

$$\left(\frac{x}{h}\right)^N + \left(\frac{y}{v}\right)^N = 2 \tag{8}$$

where h and v represent the minimum horizontal and vertical separation distance constraint, respectively.

Raising the exponential term, N, in equation (8) to higher-order even powers causes the superellipse to appear increasingly rectangular. Figure 1 graphically shows the results of increasing the exponential terms in equation (8). In this figure, the x-axis represents the horizontal separation constraint (Δxy) and the y-axis the vertical separation constraint (Δz). The red dashed lines depict the minimum separation distance of ± 2460 ft and ± 820 ft in the horizontal (h) and vertical (v), respectively.

From Figure 1, in the limit as $N \to \infty$ the superellipse approaches the rectangular conditional constraints. Substituting x and y in equation (8) with Δxy and Δz, respectively, gives the superellipse equation as:

$$\left(\frac{\Delta xy}{h}\right)^N + \left(\frac{\Delta z}{v}\right)^N = 2 \tag{9}$$

Note that,

$$\left(\frac{\Delta xy}{h}\right)^N + \left(\frac{\Delta z}{v}\right)^N = 2 \Leftrightarrow \lim_{N \to \infty} \left(\left(\frac{\Delta xy}{h}\right)^N + \left(\frac{\Delta z}{v}\right)^N\right)^{\frac{1}{N}} = \lim_{N \to \infty} 2^{\frac{1}{N}} \tag{10}$$

Furthermore Luenberger [13];

$$\lim_{N \to \infty} \left(\left(\frac{\Delta xy}{h}\right)^N + \left(\frac{\Delta z}{v}\right)^N\right)^{\frac{1}{N}} \triangleq \max\left\{\frac{\Delta xy}{h}, \frac{\Delta z}{v}\right\} \tag{11}$$

Therefore, if the mixed-norm is defined as described by Sadovsky in 2012:

$$\left\|\left(\frac{\Delta xy}{h}, \frac{\Delta z}{v}\right)\right\|_{\text{mixed}} \triangleq \max\left\{\frac{\Delta xy}{h}, \frac{\Delta z}{v}\right\} \tag{12}$$

Then in the limit as $N \to \infty$ the superellipse equation (9) is equivalent to the dashed-red rectangle in Figure 1 given by the mixed-norm equation:

$$\left\|\left(\frac{\Delta xy}{h}, \frac{\Delta z}{v}\right)\right\|_{\text{mixed}} = 1 \tag{13}$$

since,

$$\max\left\{\frac{\Delta xy}{h}, \frac{\Delta z}{v}\right\} = 1 \Leftrightarrow \lim_{N \to \infty} \left(\left(\frac{\Delta xy}{h}\right)^N + \left(\frac{\Delta z}{v}\right)^N\right)^{\frac{1}{N}} = \lim_{N \to \infty} 2^{\frac{1}{N}} = 1 \tag{14}$$

An advantage of using a MAES to approximate the conditional inequality constraint is that since the semi-major and semi-minor axes are defined as $a^* = 2^{\frac{1}{N}} h$ and $b^* = 2^{\frac{1}{N}} v$, respectively, the overestimation errors (δ_h and δ_v) for infeasible values of Δxy and Δz are
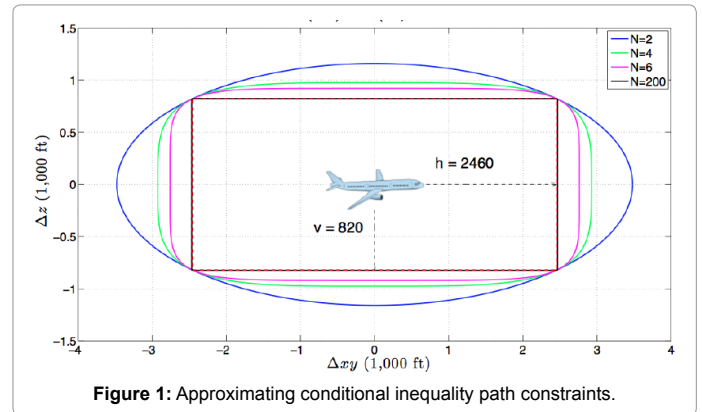

Figure 1: Approximating conditional inequality path constraints.

bounded such that:

$$\begin{aligned} 0 &\leq \delta_h \leq \left(2^{1/N} - 1\right)h \\ 0 &\leq \delta_v \leq \left(2^{1/N} - 1\right)v \end{aligned} \tag{15}$$

From equation (15), for given h and v the MAES overestimation error is strictly a function of N. For the example in this paper, the MAES method used a value of $N=200$, resulting in a maximum overestimation error of 0.3%, which is less than the typical resolution of an aircraft's onboard sensors used to calculate position or altitude. Therefore, system designers should select the appropriate value of N based on sensor resolution. The minimum value of N necessary to guarantee the overestimation error is less than the sensor tolerance, Δ_s, is given by the smallest even number that satisfies the following relationships:

$$\begin{aligned} N &\geq \frac{\ln 2}{\ln\left(\frac{\Delta_s}{h} + 1\right)} \\ N &\geq \frac{\ln 2}{\ln\left(\frac{\Delta_s}{v} + 1\right)} \end{aligned} \tag{16}$$

For example, if $\Delta_s = 12$ feet while $h = 2460$ feet and $v = 820$ feet, then:

$$\begin{aligned} N &\geq \frac{\ln 2}{\ln\left(\frac{12}{2460} + 1\right)} = 142.44 \\ N &\geq \frac{\ln 2}{\ln\left(\frac{12}{820} + 1\right)} = 47.71 \end{aligned} \tag{17}$$

Therefore, N would be set to 144.

However, for large values of N, constraints involving the equation of the superellipse become computationally difficult to evaluate. This issue of *constraint scaling* is addressed by applying the natural log on equation (8) to generate the equivalent equation of the minimizing enclosing superellipse. The adapted equation appear as:

$$\ln\left(\left(\frac{\Delta xy}{2460}\right)^N + \left(\frac{\Delta z}{820}\right)^N\right) = \ln 2 \tag{18}$$

Therefore, the standard form of the inequality path constraint for the optimal control problem appears as:

$$\ln 2 - \ln\left(\left(\frac{\Delta xy}{2460}\right)^N + \left(\frac{\Delta z}{820}\right)^N\right) \leq 0 \tag{19}$$

Equation (19) is the form of the MAES method used with equations

(1) and (2) at each collocation node to solve the first example problem in this paper. However, it may be impractical to apply the MAES method to optimal control problems with multiple, compound (or *nested*) conditional inequality constraints, represented by the second example problem. To address this limitation, the next section develops differentiable approximations of indicator methods using a sigmoid function form of the conditional inequality path constraint.

Sigmoid function

Another approach to incorporate a conditional constraint is to use a sigmoid function approximation of a conditional indicator function. The earlier section described the problem that gradient-based NLP solvers have with non-differentiable conditional constraints. Like the MAES, sigmoid functions avoid this problem since they too are continuous and differentiable. Framing the development in the context of the first example problem, two unique sigmoid functions are defined to approximate the horizontal and vertical inequality path constraint indicator functions separately. The equations for the horizontal and vertical sigmoid functions, S_h and S_v, that approximate the inequality path constraint indicator functions appear as:

$$S_h(\Delta xy, s_h) = \left[1 + e^{s_h\left(1 - \frac{\Delta xy}{2460}\right)} \right]^{-1} \tag{20}$$

$$S_v(\Delta z, s_v) = \left[1 + e^{s_v\left(1 - \frac{\Delta z}{820}\right)} \right]^{-1} \tag{21}$$

where s_h and s_v are user-defined positive stiffness factors for the smoothness and orientation of their respective sigmoid. Figure 2 shows the results of plotting equations (20) and (21) for varying values of positive s_h and s_v. Negative values of s_h and s_v merely reflect the image of the sigmoid function about the critical values (2460 and 820). From Figure 2, when $\Delta xy=2460$ or $\Delta z=820$ the value of the respective sigmoid equals 0.5. When $\Delta xy < 2460$ or $\Delta z < 820$ then the value of the respective sigmoid approaches zero. Likewise, when $\Delta xy > 2460$ or $\Delta z > 820$, the value of the respective sigmoid approaches unity. Therefore, the sigmoid functions define the inequality constraint indicator function approximations which appear as:

$$S_h(\Delta xy, s_h > 0) \approx \begin{cases} 1, & \Delta xy \geq 2460 \\ 0, & \text{otherwise} \end{cases}$$

$$S_h(\Delta xy, s_h < 0) \approx \begin{cases} 0, & \Delta xy \geq 2460 \\ 1, & \text{otherwise} \end{cases} \tag{22}$$

$$S_v(\Delta z, s_v > 0) \approx \begin{cases} 1, & \Delta z \geq 820 \\ 0, & \text{otherwise} \end{cases}$$

$$S_v(\Delta z, s_v < 0) \approx \begin{cases} 0, & \Delta z \geq 820 \\ 1, & \text{otherwise} \end{cases} \tag{23}$$

This paper proposes and evaluates two different methods of using these sigmoid functions (a sum and then a product method) to approximate the conditional inequality path constraint. The first method involves summing the horizontal and vertical sigmoid values and the second involves taking the product of these two sigmoids. The following sections detail both methods where for convenience of notation, the exponential terms in equations (20) and (21) are defined as:

$$\varepsilon_h = 1 - \frac{\Delta xy}{2460} \tag{24}$$

$$\varepsilon_v = 1 - \frac{\Delta z}{820} \tag{25}$$
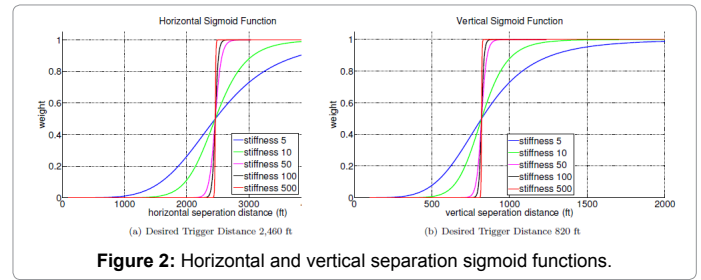

Figure 2: Horizontal and vertical separation sigmoid functions.

Sigmoid sum method: The sigmoid sum method is the more conservative of the two sigmoid methods. Given the conditional inequality path constraint of satisfying either a horizontal (*h*) or vertical (*v*) separation distance constraint, the sigmoid sum approximation of the conditional inequality path constraint appears as:

$$1 - [S_h(\Delta xy, s_h) + S_v(\Delta z, s_v)] \leq 0 \tag{26}$$

Where s_h and s_v are always positive. Therefore, if $\Delta xy < 2460$ and $\Delta z < 820$, then equation (26) is greater than zero. Thus, the sigmoid sum approximation method does not admit solutions that violate the true conditional inequality path constraint. However, if $\Delta xy > 2460$ while $\Delta z < 820$ or $\Delta xy < 2460$ while $\Delta z > 820$ then equation (26) may still be greater than zero, and thus, this method could fail to admit viable solutions that satisfy the true conditional inequality path constraint. The tolerance for when viable solutions are not admitted is a function of the user defined stiffness factor *s*. The relationships that determines if the sigmoid sum method will admit a viable solution are given by equations (27) - (29).

If $\Delta xy \geq 2460$ and $\Delta z \geq 820$ then,

$$\begin{aligned} \varepsilon_h &\leq 0 \\ \varepsilon_v &\leq 0 \end{aligned} \tag{27}$$

which implies that $S_h(\Delta xy, s_h) \geq 0.5$ and $S_v(\Delta z, s_v) \geq 0.5$ so equation (26) is correctly satisfied.

If $\Delta xy \geq 2460$ and $\Delta z < 820$ then,

$$\begin{aligned} \varepsilon_h &\leq 0 \\ \varepsilon_v &> 0 \end{aligned} \tag{28}$$

which implies that $S_h(\Delta x, s_h) \geq 0.5$ and $S_v(\Delta z, s_v) < 0.5$ so equation (26) is correctly satisfied if and only if $\varepsilon_v \leq |\varepsilon_h| \dfrac{s_h}{s_v}$.

If $\Delta xy < 2460$ and $\Delta z \geq 820$ then,

$$\begin{aligned} \varepsilon_h &> 0 \\ \varepsilon_v &\leq 0 \end{aligned} \tag{29}$$

which implies that $S_h(\Delta xy, s_h) < 0.5$ and $S_v(\Delta z, s_v) \geq 0.5$ so equation (26) is correctly satisfied if and only if $\varepsilon_h \leq |\varepsilon_v| \dfrac{s_v}{s_h}$.

Equations (27) - (29) guarantee that the sigmoid sum method will only reject feasible solutions if one constraint is violated by more than the slack of the satisfied constraint times the ratio of the two stiffness factors. Furthermore, in the worst-case when $\Delta z=0$ then $\varepsilon\_v = 1$, so the minimum value of Δxy that will satisfy the sigmoid sum approximation of the conditional inequality constraint is given by the following relation:

if $\Delta z = 0$, then

$$1 - [S_h(\Delta xy, s_h) + S_v(\Delta z, s_v)] \leq 0$$

$$\Leftarrow \Delta xy \geq (1 + \frac{s_v}{s_h})h \tag{30}$$

Similarly, in the worst-case when $\Delta xy=0$ then $\varepsilon_h=1$, so the minimum value of Δz that will satisfy the sigmoid sum approximation of the conditional inequality constraint is given by the following relation:

$$\text{if} \Delta xy = 0, \text{then}$$

$$1 - [S_h(\Delta xy, s_h) + S_v(\Delta z, s_v)] \leq 0$$

$$\Leftrightarrow \Delta z \geq (1 + \frac{s_h}{s_v})v \qquad (31)$$

Equations (30) - (31) indicate that the worst-case overestimation error can be very large. Additionally, the complexity of the sigmoid sum approximation surface indicates that NLP solvers could have difficulty estimating the gradient of the constraint (Figure 3). Therefore, a second sigmoid function method which reduces much of the overestimation error and improves differentiability is described next.

Sigmoid product method: Compared to the sigmoid sum method, the sigmoid product method allows greater precision in approximating conditional inequality constraints by reducing the maximum overestimation error that occurs when only one constraint is satisfied. Given the conditional inequality path constraint of satisfying a minimum horizontal (h) or vertical (v) separation distance constraint, the sigmoid product approximation of the conditional inequality path constraint appears as:

$$[s_h(\Delta xy, s_h) s_v(\Delta z, s_v)] - 0.25 \leq 0 \qquad (32)$$

where s_h and s_v are now both negative. The relationships given by equations (27) - (29) in the sigmoid sum method also determine if the sigmoid product method will admit a viable solution. However, for the sigmoid product method, if the horizontal constraint is satisfied but the vertical is not, that is, $\Delta xy \geq 2460$ and $\Delta z < 820$ then equation (32) is correctly satisfied if:

$$\varepsilon_h \leq \frac{1}{s_h} \ln\left(\frac{3 - e^{s_v \varepsilon_v}}{1 + e^{s_v \varepsilon_v}}\right) \qquad (33)$$

Similarly, if $\Delta xy < 2460$ and $\Delta z \geq 820$, then equation (32) is correctly satisfied if:

$$\varepsilon_v \leq \frac{1}{s_v} \ln\left(\frac{3 - e^{s_h \varepsilon_h}}{1 + e^{s_h \varepsilon_h}}\right) \qquad (34)$$

In the worst-case when $\Delta z=0$, then the minimum value of Δxy that will satisfy equation (32) is given by the following relationship:

Figure 3: 3D constraint contour comparison of sigmoid sum and sigmoid product methods.

$$\Delta xy \geq \left(1 + \frac{1}{|s_h|} \ln\left(\frac{3 - e^{s_v}}{1 + e^{s_v}}\right)\right) h \qquad (35)$$

However, $e^{s_v} \to 0$ as $s_v \to -\infty$, so equation (35) can be approximated conservatively as:

$$\Delta xy \geq \left(1 + \frac{1}{|s_h|} \ln(3)\right) h \qquad (36)$$

Likewise, when $\Delta xy=0$, then the minimum value of Δz that will satisfy equation (32) is given by the following relationship:

$$\Delta z \geq \left(1 + \frac{1}{|s_v|} \ln\left(\frac{3 - e^{s_h}}{1 + e^{s_h}}\right)\right) v \qquad (37)$$

which can also be approximated conservatively as:

$$\Delta z \geq \left(1 + \frac{1}{|s_v|} \ln(3)\right) v \qquad (38)$$

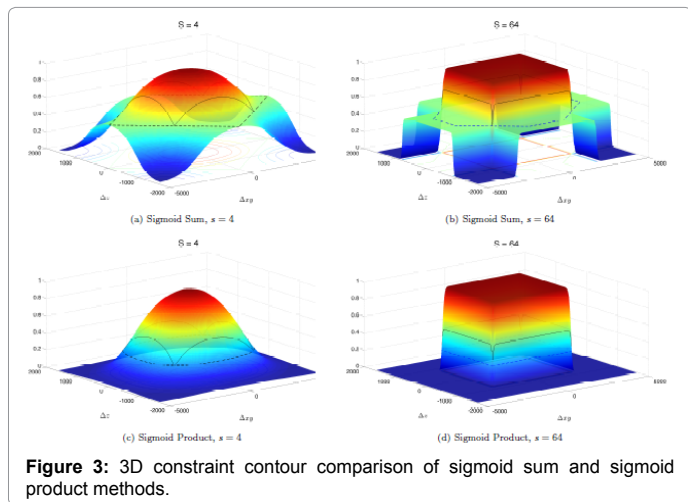Therefore, the sigmoid product method bounds the overestimation errors (δ_h and δ_v) for infeasible values of Δxy and Δz such that:

$$0 \leq \quad \delta_h \leq \frac{h}{|s_h|} \ln(3)$$

$$0 \leq \quad \delta_v \leq \frac{v}{|s_v|} \ln(3) \qquad (39)$$

From equation (39), the maximum overestimation error is strictly a function of $|s_h|$ and $|s_v|$. Due to the computational difficulty resulting from large exponential powers, with 40 fixed collocation nodes the largest absolute values for the sigmoid product parameters that were achieved were $|s_h| = 240$ and $|s_v| = 80$. With these values the maximum overestimation error was 0.5% in the horizontal and 1.4% in the vertical. These values are much lower than the sigmoid sum method while comparable to the maximum overestimation error obtained in the MAES method. As with the MAES method, system designers can select the appropriate value of s_h and s_v based on sensor resolution. The minimum values of $|s_h|$ and $|s_v|$ necessary to guarantee that the overestimation error is less than the sensor tolerance, Δ_s, are given by the following relationships:

$$|s_h| \geq \quad \frac{h}{\Delta_s} \ln(3)$$

$$|s_v| \geq \quad \frac{v}{\Delta_s} \ln(3) \qquad (40)$$

Since $h=2460$ and $v=820$ in the example problem, the minimum value of $|s_h|$ needed to achieve a given tolerance will be 3 × greater than the minimum value of $|s_v|$ needed to achieve the same tolerance.

Additionally, Figure 3 shows the normalized 3D constraint contour plots for the sigmoid sum and product methods for stiffness values of $s=4$ and 64 where $s=s_h=s_v$. The blue-colored area in the figure represents the feasible region where at least one constraint is satisfied and the red-colored area represents the infeasible region where neither constraint is satisfied. The thin black line on the constraint surface represents the true feasibility threshold and the thicker dashed line represents the conservative approximation to the feasibility threshold. For the sigmoid sum method the inequality constraint values, equation (26), range between ± 1 where negative values indicate at least one constraint is satisfied and values above 0.5 indicate neither constraint is satisfied. In

the sigmoid product method the inequality constraint values, equation (32), range between -0.25 and 0.75 where negative values indicate at least one constraint is satisfied and values above 0.25 indicate neither constraint is satisfied. Therefore, to compare the two sigmoid methods the constraint contours (g*) in Figure 3 are normalized using:

$$\mathbf{g}^* = \frac{\mathbf{g} - \mathbf{g}_{min}}{\mathbf{g}_{max} - \mathbf{g}_{min}} \qquad (41)$$

such that the constraint contour plots for both methods range between 0 and 1 where 0 indicates at least one constraint is satisfied and 1 indicates neither constraint is satisfied. During the optimization, the gradients of these constraint surfaces need to be calculated. Clearly, the sigmoid sum gradient is more complex due to the "stair-steps" and "sharp valleys" in the contour plots and thus is more difficult for the optimizer to establish the correct search direction. Subsequently, as shown in the analysis and confirmed by the results, the sigmoid product method is more efficient and allows higher stiffness values compared to the sigmoid sum method. Thus, the sigmoid product method was selected for use in the general case to resolve optimal control problems with multiple, compound (or *nested*) conditional inequality constraints. For problems of this type, it is necessary to use the generalized form of the sigmoid product constraint formulation given by:

$$\left(\prod_{k=1}^{K}\left[1+e^{s_k\left(\frac{h_k-g_k}{\max\{g_k\}-h_k}\right)}\right]^{-1}\right)-2^{-K} \leq 0 \qquad (42)$$

where K is the total number of conditional constraints being evaluated, $g_k \leq \max\{g_k\}$ is a bounded constraint function such that $h_k - g_k \leq 0$ if and only if condition k is satisfied and $h_k - g_k > 0$ if and only if the condition is not satisfied, and $s_k < 0$ is the stiffness factor. The overestimation error for each constraint in the generalized sigmoid product method is bounded similarly to the two-sigmoid case such that:

$$0 \leq \quad \delta_k \leq \left(\frac{\max\{g_k\}-h_k}{|s_k|}\right)\ln(2^K - 1) \qquad (43)$$

where δ_k is the overestimation error for values that violate conditional constraint k. Equation (43) also indicates system designers can select the appropriate value of s_k based on desired precision. The minimum value of $|s_k|$ necessary to guarantee that the overestimation error is less than the precision tolerance, Δ_k, is given by the following relationships:

$$|s_k| \geq \left(\frac{\max\{g_k\}-h_k}{\Delta_k}\right)\ln(2^K - 1) \qquad (44)$$

Thus, equations (39) and (40) were used to determine the sigmoid product method parameters for the first example problem, while equations (43) and (44) were used to determine the sigmoid product method parameters for the second example problem.

Description of Example Problems

The following section describes the example problems in this paper. The objective of the first example problem, as described earlier, is for the ownship to minimize deviations from a 3D flight path corridor while maintaining either a horizontal separation distance (Δxy) of at least 2460 ft *or* a vertical separation distance (Δz) of at least 820 ft from an intruder. The objective of the second example problem is identical to the first but requires the ownship to also adhere to FAA right of way (ROW) rules in addition to maintaining the intruder separation

distances above. In this problem, the turn direction is conditioned on time and range from the intruder. The setup conditions for both example problems are identical. For both example problems, the collocation is performed at Legendre-Gauss-Radau quadrature points as described by Patterson in 2013.

Constraints

Equation (45) represents the dynamic constraints for the ownship that the optimization algorithm must satisfy when generating the optimal collision avoidance trajectory [musica:ddd]:

$$\dot{x} = \mathbf{f}\big(x(t),\mathbf{u}(t),t\big) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \dot{\gamma}(t) \\ \dot{\chi}(t) \end{bmatrix} = \begin{bmatrix} V\cos\gamma(t)\cos\chi(t) \\ V\cos\gamma(t)\sin\chi(t) \\ V\sin\gamma(t) \\ \dfrac{N_z g\cos\mu(t) - g\cos\gamma(t)}{V} \\ \dfrac{N_z g\sin\mu(t)}{V\cos\gamma(t)} \end{bmatrix} \qquad (45)$$

The states, $x(t)$, in equation (45) consist of the cartesian directions (x,y,z), flight path angle (γ), and heading angle (χ). The controls, $\mathbf{u}(t)$, are bank angle (μ) for horizontal control and normal acceleration (N_z) for longitudinal or z-axis control where N_z is defined in the velocity-axis frame [musica:ddd]. The remaining variables in equation (45) are ground speed (V) and gravitational acceleration (g). An assumption for this model is that the aircraft's flight control system will keep the vehicle speed constant throughout the avoidance maneuver.

In this problem the intruder aircraft maintains a constant speed of 300 ft/sec, a constant heading of 180° and a constant altitude of 6,000 feet. Although the optimal control problem formulation can easily accommodate multiple intruders and more complex and even stochastic models Smith [14] in this example problem we intentionally limited the problem to a single intruder and kept the intruder dynamic constraints simple in order to focus on the methodology for enforcing the conditional inequality path constraints. Thus, the intruder dynamic constraints appear as:

$$\dot{x}_{int} = \mathbf{f}\big(x_{int}(t),t\big) = \begin{bmatrix} \dot{x}_{int}(t) \\ \dot{y}_{int}(t) \\ \dot{z}_{int}(t) \end{bmatrix} = \begin{bmatrix} 300\cos(180°) \\ 300\sin(180°) \\ 0 \end{bmatrix} = \begin{bmatrix} -300 \\ 0 \\ 0 \end{bmatrix} \qquad (46)$$

The equality boundary constraints are time initial (t_0), time final (t_f), ownship initial position (x_0, y_0, z_0), and intruder initial position $\left(x_{0_{int}}, y_{0_{int}}, z_{0_{int}}\right)$. These boundary constraints appear as:

$$\begin{aligned} t_0 &= 0 \quad \text{sec} \\ t_f &= 60 \quad \text{sec} \\ (x_0, y_0, z_0) &= [0, 0, 6000]' \quad \text{ft} \\ \left(x_{0_{int}}, y_{0_{int}}, z_{0_{int}}\right) &= [20000, 2000, 6000]' \quad \text{ft} \end{aligned} \qquad (47)$$

The inequality path constraint for the collision avoidance problem is the ownship must maintain at least 2460 feet separation distance horizontally or 820 feet vertically at all time from the intruder. To approximate this conditionally inequality constraint, the MAES, sigmoid sum, and sigmoid product methods are evaluated as described by equations (19), (26), and (32), respectively. In addition, the inequality control constraints, $\mathbf{u}(t)$, appear as:

$$\begin{aligned} -45° &\leq \mu(t) \leq 45° \\ 0.59 &\leq N_z(t) \leq 1.41 \end{aligned} \qquad (48)$$

The symmetric control bounds on $N_z(t)$ are reasonable maneuver limits for commercial transport or large remotely piloted aircraft. In addition, the upper bound on $N_z(t)$ corresponds to the upper bound on $\mu(t)$ such that when both controls are at their maximum value the aircraft performs a level turn.

Performance measure

The performance measure for this problem is to minimize overall deviation distance (d) from the specified 3D flight path corridor centerline. In Figure 4 adapted from [wolfram: 3D_distance], x_1 and x_2 identify two consecutive waypoints, $x(t)$ the current ownship position, and (d) the deviation distance such that [wolfram: 3D_distance]:

$$J = \int_{t_0}^{t_f} d(t)dt \qquad (49)$$

$$\text{where } d(t) = \frac{|(x(t)-x_1) \times (x(t)-x_2)|}{|x_2 - x_1|} \qquad (50)$$

In this problem, x_1 and x_2 are defined in feet as:

$$\begin{aligned} x_1 &= [0, 2000, 6000] \\ x_2 &= [25000, 2000, 6000] \end{aligned} \qquad (51)$$

Note that the ownship is trying to fly along the line through x_1 and x_2, but there is no time specified with either point.

Example Problem 1

In this section we first analyze the results of using the MAES to approximate the inequality path constraint and then examine the results of using the sigmoid methods for the first example problem. For comparison, in each case we used a global polynomial with 40 fixed collocation nodes and used IPOPT as the NLP solver. Simulations in this paper used Matlab version 2012 b on a laptop computer operating with OS X version 10.9 operating system and a 2.3 GHz Intel Core *I* 5 processor with 16 GB 1333 MHz DDR3 memory. The performance measure for this scenario was to minimize path deviation, as in equation (49). Since the minimum horizontal separation distance was 3× greater than the minimum vertical separation distance, intuitively the minimum deviation trajectory was for the ownship to intercept the 3D flight path corridor and change altitude only when required to meet the conditional separation constraint. The simulation results confirmed this intuition. The primary differences in these two approaches was the accuracy of the approximation and the time required for the optimizer to achieve a solution. To standardize the results we provided the NLP solver with the same conservative initial guess for each simulation run which consisted of the ownship flying level at the initial condition heading of 0 degrees. The minimum path deviation trajectory for all three methods appeared similar. Although Figure 5 is a time-series quad chart of only the MAES simulation results, the minimum path deviation trajectories from the sigmoid methods appeared the same as the results in this figure. In this figure, the ownship is depicted in blue, the intruder in red, and the desired 3D flight path corridor in black. Both aircraft started the scenario at the same altitude of 6,000 feet MSL with the ownship 2,000 feet south (positive y-axis) of the 3D corridor while the intruder started and remained on the corridor.

As seen in panel (a) of Figure 5, at the start of the scenario the ownship began an immediate left turn to intercept the 3D flight path corridor and then continued on the corridor at the specified corridor altitude of 6,000 feet as shown in panel (b). In panel (c), the ownship then deviated from the corridor altitude by climbing only when required to meet the conditional separation inequality constraint. After satisfying this conditional constraint, the ownship then descended and maintained the desired 3D flight path corridor for the duration of the time horizon shown in panel (d).

MAES simulation results

Table 1 summarizes the effects of increasing the exponential terms in equation (19) on the normalized cost (J), number of NLP inequality constraint evaluations, CPU time in NLP evaluations, and the vertical and horizontal separation distances from the intruder at the closest point of approach (CPA).

The normalized cost (J) in Table 1 represents the ratio of the cost of intercepting the 3D corridor with an avoidance maneuver normalized by the cost of intercepting the 3D corridor without an avoidance maneuver. Note that these results are for 40 collocation nodes. While the number of nodes will affect the results, increasing the number of nodes will not necessarily cause the generated trajectory at the CPA to achieve the minimum feasible separation distances. In fact, due to the interaction between the aircraft dynamics and cost function, equations (45) and (49), the minimum vertical and horizontal separation distances at the CPA may overshoot the minimum feasible separation distances of the active constraint for any number of collocation nodes.

Figure 6 graphically displays the results for $N=200$. The blue asterisks in the plots show the horizontal (Δxy) and vertical (Δz) separation distances between the ownship and the intruder aircraft respectively at each collocation node for the 60 second time-horizon. The redline in each plot depicts the minimum horizontal (2460 ft) or vertical (820 ft) separation distance. Figure 6 shows that at approximately 12 seconds, the ownship began a climb so that as the horizontal separation decreased to below 2460 ft, at approximately 26 seconds, the ownship achieved the required vertical separation of at least 820 ft. In this plot, the ownship climbed above the minimum altitude of 820 ft and peaked at an altitude of approximately 870 ft. The results for both sigmoid methods appeared similar to the results in Figure 6.

Sigmoid simulation results

Sigmoid sum results: Table 2 summarizes the results of increasing the stiffness factors (s_h and s_v) in equation (26) on the cost (J), number of inequality constraint evaluations, CPU time in NLP evaluations, maximum vertical separation distance from the intruder, and the minimum horizontal separation distance from the intruder.

Sigmoid product results: Table 3 summarizes the results for the sigmoid product method of increasing the stiffness factors (s_h and s_v) in equation (32) on the cost (J), number of inequality constraint evaluations, CPU time in NLP evaluations, maximum vertical separation distance from the intruder, and the minimum horizontal separation distance from the intruder.

Sensor tolerance evaluation results
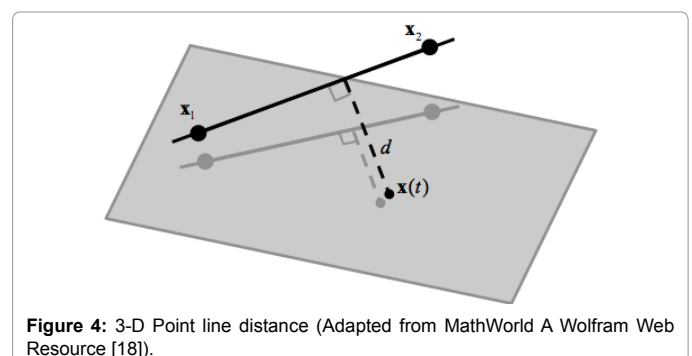
The sigmoid sum method had the largest computational time

Figure 4: 3-D Point line distance (Adapted from MathWorld A Wolfram Web Resource [18]).
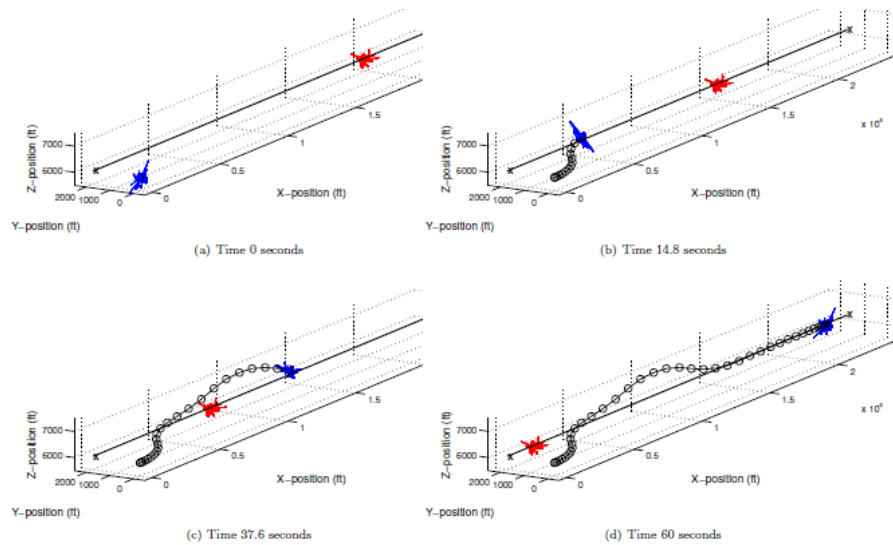
Figure 5: Time series of optimal trajectory for own ship (Blue) avoiding the intruder aircraft (Red) while minimizing path deviation (MAES, Sigmoid results similar).

MAES Order (N)	Normalized Cost (J)	Inequality Constraint Evaluations	CPU time in NLP Evaluations (sec)	Separation at CPA	
				Vertical (ft)	Horizontal (ft)
2	1.666	130	26.41	1129	795
4	1.46	59	23.32	974	775
100	1.344	64	22.8	875	772
200	1.341	69	27.26	872	761

Table 1: IPOPT simulation results for MAES approximation of conditional inequality constraint.

Stiffness factor (S_h, S_v)	Normalized Cost (J)	Inequality Constraint Evaluations	CPU time in NLP Evaluations (sec)	Separation at CPA	
				Vertical (ft)	Horizontal (ft)
(50, 50)	1.656	583	228.70	1122	790
(100, 100)	1.458	1174	365.56	971	776
(125, 125)	1.423	1189	400.82	941	777

Table 2: IPOPT simulation results for sigmoid sum approximation of conditional inequality constraint.

Stiffness factor (S_h, S_v)	Normalized Cost (J)	Inequality Constraint Evaluations	CPU time in NLP Evaluations (sec)	Separation at CPA	
				Vertical (ft)	Horizontal (ft)
(180, 60)	1.443	121	50.81	825	989
(210, 70)	1.422	128	52.9	824	968
(240, 80)	1.403	80	38.69	824	915

Table 3: IPOPT simulation results for sigmoid product approximation of conditional inequality constraint.

and was the most conservative of the three methods. Therefore, the following sensor tolerance evaluation focuses on the performance of the MAES and sigmoid product methods with parameters chosen to guarantee maximum overestimation errors less than a given sensor tolerance. For simplicity, this evaluation assumes the horizontal and vertical sensor tolerances are equal. From the previous example, for a sensor tolerance of ± 12 feet, the minimum value of N for the MAES method to guarantee the maximum overestimation error is less than the sensor tolerance is $N=144$. Similarly, from equation (40) for the sigmoid product method the minimum values of $|s_h|$ and $|s_v|$ required to guarantee the maximum overestimation error is less than the sensor tolerance are $|s_h|= 226$ and $|s_v|= 76$. Table 4 shows the results for the MAES and sigmoid product methods with parameter values that guarantee the maximum overestimation error is less than sensor tolerances of 12, 25 and 50 feet.

The previous results used only 40 collocation nodes and these nodes spanned the entire trajectory as a single "global" interpolating

polynomial. However, to increase the fidelity of the solution especially near the constraint activation boundaries, the results in Table 4 divided the trajectory into 20 equal-spaced segments with 10 collocation nodes per segment as shown by Huntington in 2007. Although not reflected in the table, an alternate formulation applied an adaptive mesh refinement strategy by Patterson in 2013, which adaptively increased the number and placement of collocation nodes to achieve a user-defined level of accuracy. However, since the execution times for the adaptive node placement strategy varied significantly based on the number of mesh refinements, for standardization and comparison of results, a fixed number of collocation nodes was preferred for this analysis. Furthermore, due to the longer execution times of an adaptive node placement strategy, any eventual implementation of a real-time airborne collision avoidance algorithm would likely use fixed collocation nodes.

To better gauge the changes in the ownship trajectory as a function of the change in the sensor tolerance, the results in Table 4 replaced

Sensor Tolerance (ft)	Method	Normalized Cost (J)	Constraint Activation times (sec)	CPU time in NLP (sec)	Separation at CPA	
					Vertical (ft)	Horizontal (ft)
12	MAES	1.399	t_1 = 25.69	254.88	919	116
	N = 144		t_2 = 32.73			
	Sigmoid Product	1.394	t_1 = 25.81	346.13	929	102
	sh = −226		t_2 = 32.86			
	sv = −76					
25	MAES	1.406	t_1 = 25.81	236.54	918	110
	N = 70		t_2 = 32.86			
	Sigmoid Product	1.396	t_1 = 25.81	285.51	931	102
	sh = −109		t_2 = 32.85			
	sv = −37					
50	MAES	1.401	t_1 = 25.80	147.1	936	107
	N = 36		t_2 = 32.85			
	Sigmoid Product	1.399	t_1 = 25.81	162.75	934	107
	sh = −55		t_2 = 32.86			
	sv = −19					

Table 4: Comparison of methods for achieving error less than sensor tolerance.

the column showing the number of inequality constraints evaluations in Tables 1-3 with a new column that showed the constraint activation times. As seen in Figure 6, the intersection of the red-dashed line and blue-asterisk indicate the constraint activation times. For the sensor tolerance evaluation, changes in the constraint activation times can provide additional insight into the sensitivity of the aircraft dynamics to the sensor tolerance. For instance, the performance measure (J) in this problem should force the optimal trajectory towards the "corner" of the constraint boundary where the horizontal and vertical constraints are active since in both MAES and sigmoid methods, these constraint corners are where the overestimation error is zero. This fact is particularly evident in Figure 1 where the formulation of the MAES optimization problem in equation (7) minimized the overestimation error at the corners of the rectangular constraint area.

In general the normalized cost (J) in Table 4 increased slightly as the predicted overestimation error increased. However, the constraint activation times remained consistent with the constant ground speed assumption, and the minimum separation distances at the CPA did not noticeably change as the sensor tolerance increased. These results indicate that the aircraft dynamics and trajectory optimization process were not sensitive to the range of sensor tolerances in the table. Since the overestimation error achieved its minimum value at the constraint corner, the optimizer forced the trajectory to intersect this corner as seen by the consistent constraint activation times. Even at a sensor tolerance of 50 feet, the ownship dynamic constraints were still what drove the optimal trajectory to start a climb away from the desired flight path in order to intersect the constraint corner; that is, the ownship climbed to reach an altitude of 820 feet above the intruder at the exact moment the horizontal separation distance decreased to less than 2460 feet. Likewise, after the two aircraft passed, the ownship then descended below 820 feet above the intruder at the exact moment when the horizontal separation distance again increased to greater than 2460. Thus, the trajectory was insensitive to sensor tolerances up to 50 feet. This was because the ownship's dynamics forced the aircraft to climb to intersect the constraint corner rather than to avoid the worst-case overestimation region corresponding to sensor tolerances up to 50 feet. As a result, based on the intercept geometry of this example problem,

the optimal trajectory should not change significantly until the sensor tolerance is significantly greater than the aircraft's dynamic constraints required to intersect the constraint corner. For example, the earlier MAES results with N=2 correspond to sensor tolerances of greater than 330 feet which was reflected in the fact that at the CPA the altitude separation was approximately 360 feet greater than the minimum required separation distance. Therefore, based on the intercept geometry the aircraft dynamics may be more important in determining the precision of the approximation rather than the sensor tolerance since the optimal trajectory may remain unchanged for varying values of realistic sensor tolerances and scaling may not be required.

Nonetheless, Table 4 confirmed the methods presented in the paper and provides users a means to implement conditional inequality path constraints with a gradient-based numerical solver to the desired level of precision. Additionally, the results confirm that if the problem involves only two simple constraints, then the MAES method is the superior approximation method.

Example Problem 2

Unlike the previous example problem of satisfying a minimum horizontal or vertical separation distance where the MAES method performed well, an optimal control problem formulation may include multiple, compound (or *nested*) conditional constraints that do not lend themselves practically to the MAES formulation. An example of this type of complication is adhering to FAA right of way (ROW) rules, which state that if two aircraft are approaching nearly head on, then "each aircraft shall alter course to the right." Since air traffic control procedures prefer horizontal over vertical maneuvers to maintain safe separation, in addition to implementing this conditional ROW constraint, this example problem also uses a new weighted cost function that separately penalizes ownship horizontal and vertical deviations from a desired 3D flight path corridor. This new cost function appears as,

$$J = \int_{t_0}^{t_f} [(\frac{d_{xy}(t)}{3038})^2 + (\frac{d_z(t)}{300})^2] dt \qquad (52)$$

Where $d_{xy}(t)$ is the horizontal deviation from the 3D corridor

centerline and $d_z(t)$ is the vertical deviation. The quadratic penalty in equation (52) is based on an assumed 3D corridor defined as ± 3038 feet (half a nautical mile) horizontally and ±300 feet vertically from centerline. Because this new cost function will likely cause the ownship to maneuver horizontally instead of vertically to maintain safe separation, the ownship will need to comply with the horizontal ROW constraint and alter course to the right since their approach will be nearly head on in this example problem.

Right of way formulation

In formulating the conditional ROW constraint, the sigmoid product method is used to implement a set of conditional logic *'if statements.'* The feasibility region for the conditional ROW constraint is described by the following set of compounded logic *OR* conditions: If the separation distance between the ownship and intruder is greater than or equal to 2× the horizontal keep out radius of 2460 feet *OR* the time to CPA (T_{CPA}) is greater than or equal to 30 seconds *OR* time from CPA is less than or equal to -5 seconds *OR* the relative azimuth angle (θ) between the ownship and the intruder is greater than or equal to zero (so the ownship will pass to the right of the intruder) then the solution is feasible; otherwise, the trajectory is not feasible. This inequality constraint formulation appears algorithmically as follows:

if $\left(\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \geq 2 \times 2460 \text{feet}\right)$

feasible

else if $(T_{CPA} \leq \text{-5 seconds})$

feasible

else if $(T_{CPA} \geq 30 \text{ Seconds})$

feasible

else if $(\theta \geq 0)$

feasible

else

infeasible

end

Each of the four conditional constraints in the ROW formulation are approximated using unique sigmoid functions. The range separation indicator function approximation at each point appears as:

$$S_r(\Delta x, \Delta y, \Delta z, s_r) = \left[1 + e^{-s_r(1 - \frac{\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}}{2 \times 2460})}\right]^{-1} \qquad (53)$$

Thus, when $\left(\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} < 2 \times 2460\right)$, the range indicator function approximation is active. By assuming a constant velocity and solving for the time that minimizes the instantaneous separation distance, the time to CPA (T_{CPA}) appears as:

$$T_{CPA} = -\frac{[\Delta x\ \Delta y\ \Delta z][\Delta \upsilon_x\ \Delta \upsilon_y\ \Delta \upsilon_z]^T}{[\Delta \upsilon_x^2 + \Delta \upsilon_y^2 + \Delta \upsilon_z^2]} \qquad (54)$$

where negative values indicate the two aircraft have passed or their velocity vectors are on non-convergent paths. Thus, the T_{CPA} indicator function approximations appear as:

$$S_{t_{entry}}(T_{CPA}, s_t) = \left[1 + e^{s_t(\frac{T_{cpa} - T_{high}}{T_{high} - T_{low}})}\right]^{-1}$$

$$S_{t_{exit}}(T_{CPA}, -s_t) = \left[1 + e^{s_t(\frac{T_{cpa} - T_{low}}{T_{high} - T_{low}})}\right]^{-1} \qquad (55)$$

where T_{low}=-5 seconds and T_{high}=30 seconds, and when (T_{CPA}> -5) OR (T_{CPA}< 30) the T_{CPA} indicator function approximation is active. Finally, the turn direction constraint (S_θ) is formulated based on relative azimuth angle (θ) where,

$$\theta = \tan^{-1}[\frac{\Delta y}{\Delta x}] \qquad (56)$$

and is approximated at each instance in time using the following sigmoid function,

$$S_\theta(\theta, s_\theta) = \left[1 + e^{s_\theta(1 - \frac{\tilde{\theta}}{\pi})}\right]^{-1} \qquad (57)$$

where $\tilde{\theta} = \theta + \pi$. Thus, when (θ>0) the "right turn" constraint approximated by equation (57) is satisfied. Therefore, based on equation (42) with K=4, the approximation of the ROW conditional inequality path constraint appears as:

$$[S_r(\Delta x, \Delta y, \Delta z, s_r)S_{t_{entry}}(T_{CPA}, s_t)S_{t_{exit}}$$
$$(T_{CPA}, -s_t)S_\theta(\theta, s_\theta)] - 0.0625 \leq 0 \qquad (58)$$

where $S_r(\Delta x, \Delta y, \Delta z, s_r), S_{t_{entry}}(T_{CPA}, s_t), S_{t_{exit}}(T_{CPA}, s_t)$ and $S_\theta(\theta, s_\theta)$ are defined in equations (53), (55), and (57), respectively.

Simulation results

As described earlier, the setup for this second example problem is identical to the first problem; however, the cost function in equation (49) is now replaced by the weighted cost function in equation (52). In addition, the ownship must now not only satisfy the conditional inequality path constraint in equation (19) formulated using the MAES method (N=200), but also satisfy the conditional inequality ROW constraint in equation (58) formulated using the sigmoid product method (s_t=s_r=s_θ=200). Like the sensor tolerance evaluation in the first example problem, this example divided the trajectory into 20 equal-spaced segments with 10 collocation nodes per segment. Figure 7 shows the simulation results. As in the first example problem, at the start of the scenario the ownship immediately maneuvered north (positive y axis) to minimize the path deviation from the 3D flight path corridor. However, due to the weighted cost function the ownship now maneuvered horizontally instead of vertically to keep out of the minimum separation distance from the intruder and correctly altered course to the right to comply with the conditional horizontal ROW constraint.

This example problem demonstrated that the sigmoid product method can effectively resolve multiple conditional constraints, to include constraints that are not naturally bounded (such as conditions that involve time or variables that are unrestricted in sign), and offers a robust alternative for problems where the MAES method is not suitable. Further, even with four conditional constraints as in this example problem, the error bounds for the sigmoid product method are valid. For example, based on equation (43) with s_t=s_r=200, the
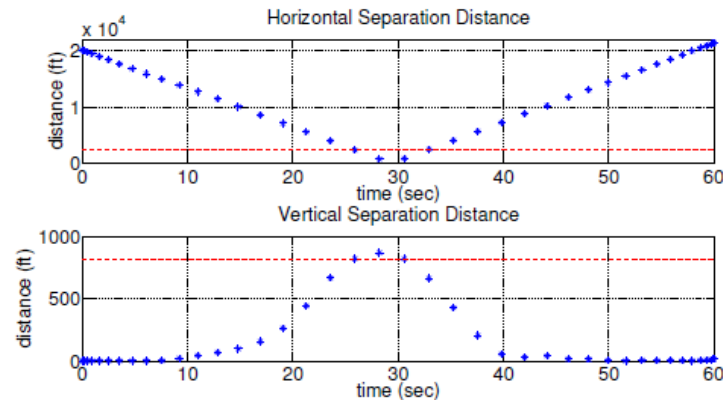
Figure 6: Simulation results using 200th order MAE's approximation for inequality path constraint.
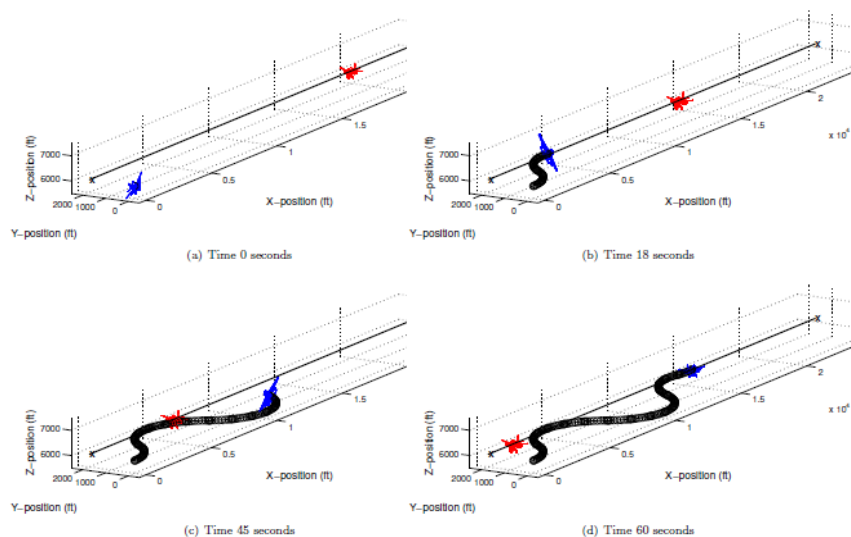

Figure 7: Time series of optimal trajectory for ownership (Blue) avoiding the intruder aircraft (Red) by adhering to right of way while minimizing path deviation.

maximum overestimation error for the time and range conditional constraints was only 2.7%. Nevertheless, the use of this approach requires an understanding of the potential limitations. For instance, a well-known and often-stated critique of gradient-based NLP search methods is they produce local optimal solutions, which may or may not be global solutions. The formulation and testing of the ROW formulation highlighted the potential applicability of this critique in the context of airborne collision avoidance. For instance, given identical initial conditions, to enforce a "left turn" constraint required an initial trajectory guess to the left in order for the optimizer to locate the global vice the local optimal solution. A follow-on research effort explores potential methods such as those listed by Raghunathan in 2004, for appropriately choosing "smart" initial guesses for complex compounded conditional constraints. Another important consideration is the number of collocation nodes and stiffness of the sigmoid function. For instance, if the nodes are too sparse then the sigmoid appears as a binary switching function causing the NLP to fail since the conditional constraint approximation is no longer differentiable. For example, Figure 8 shows the NLP approximation of the conditional constraint ($T_{CPA} \leq -5$ seconds) for 200 fixed collocation nodes; panel (a) shows the results for ($-s_t=200$) where the NLP successfully converged and panel (b) shows the results for ($-s_t=300$) where the NLP failed to coverage. Thus, the number and location of collocation nodes along with the sigmoid stiffness plays an important role in determining differentiability of the conditional constraint. Besides increasing collocation nodes and/or decreasing the sigmoid stiffness factor, an additional remedy to this situation is to use an adaptive mesh refinement strategy described by Patterson in 2013, which adaptively increases the number and placement of collocation nodes to help maintain differentiability of the conditional constraints approximated by the sigmoid functions. A final consideration when using this method is the potential for long convergence times. With 200 collocation nodes the NLP took 184.5 seconds to converge to a solution; however, in this paper the simulation algorithms were not necessarily optimized for speed but were coded for robust post-processing analysis. For real-time implementation these convergence times will need to be improved using techniques such as parallel processing or a more efficient programing language.

Conclusion

This paper motivated the application of conditional inequality path constraints in the nonlinear airborne collision avoidance optimal control problem. This paper then developed and demonstrated two different methods to enforce conditional inequality path constraints using numerical gradient-based solvers by approximating the mixed-norm and indicator function classes of constraint formulations. In addition, this paper analytically derived the maximum overestimation

Figure 8: Differentiability of sigmoid indicator function approximation.
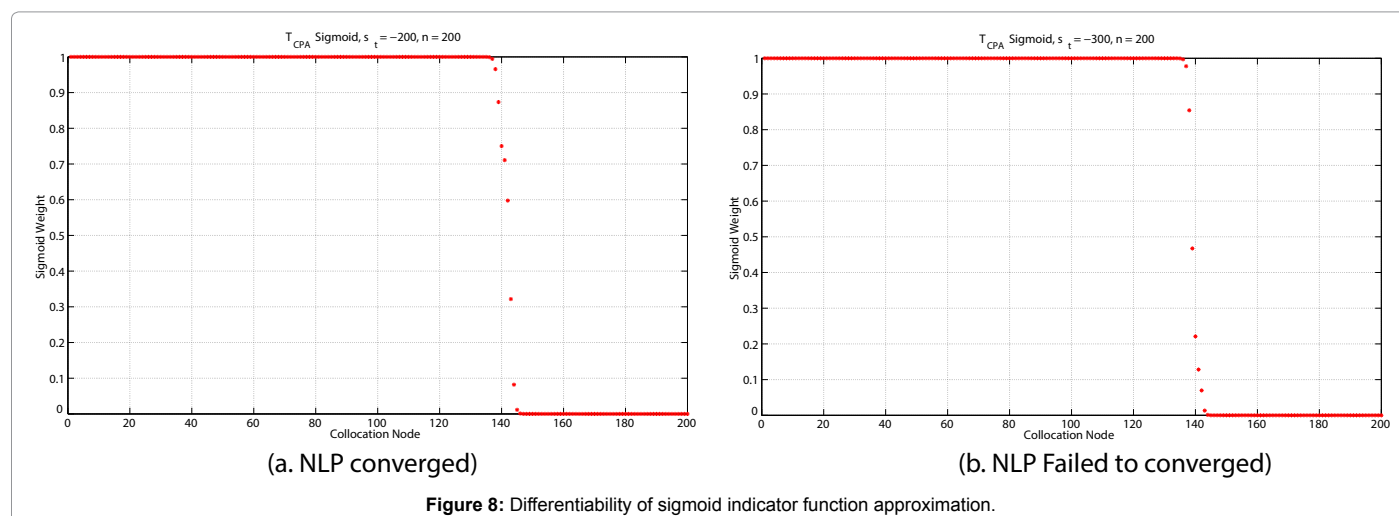
error bounds associated with these different approximation methods and also provided designers a means to determine the minimum computational complexity needed to achieve desired results based on sensor performance. Using realistic collision avoidance scenarios, this paper demonstrated the performance of these methods and confirmed the validity of the error bounds. Furthermore, both the minimum area enclosing superellipse (MAES) and sigmoid product methods yielded good results; however, due to the geometric intuition and faster computation times the MAES method may be more advantageous for normalized and non-complex constraints. However, the MAES method is not well-suited if the conditional constraints are not continuous or if the constraints are compounded. In these cases, the sigmoid product method provides a robust means to satisfy conditional constraints and has good error bounds.

Acknowledgements

References

1. Raghunathan AU, Gopal V, Subramanian D, Biegler LT, Samad T (2004) Dynamic optimization strategies for three-dimensional conflict resolution of multiple aircraft. J Guidance Control and Dynamics 27: 586-594.

2. Eele AJ, Richards A (2007) Path-planning with avoidance using nonlinear branchand-bound optimization. J Guidance Control and Dynamics 32: 384-394.

3. Horn JF, Schmidt EM, Geiger BR, DeAngelo MP (2012) Neural network-based trajectory optimization for unmanned aerial vehicles. J Guidance, Control and Dynamics 35: 548-562.

4. Geiger BR, Horn JF, Sinsley GL, Ross JA, Long LN (2008) Flight testing a real-time direct collocation path planner. J Guidance Control and Dynamics. 31: 1575-1586.

5. Borrelli F, Subramanian D, Raghunathan AU, Biegler LT (2006) MILP and NLP techniques for centralized trajectory planning of multiple unmanned air vehicles. American Control Conference IEEE, USA.

6. Winston W, Goldberg J (2004) Operations research: Applications and algorithms, Thomson Brooks/Cole, USA.

7. Hintermuller M, Ito K, Kunisch K (2013) The Primal-dual active set strategy as a semismooth newton method. SIAM Journal on Optimization 13: 865-888.

8. Sadovsky A, Davis D, Isaacson D (2012) Optimal routing and control of multiple agents moving in a transportation network and subject to an arrival schedule and separation constraints p. 33.

9. Ren J, McIsaac KA, Patel RV, Peters TM (2007) A Potential field model using generalized sigmoid functions. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions. 37: 477-484.

10. Paul T, Krogstad TR, Gravdahl JT (2008) Modelling of UAV formation flight using 3D potential field. Simulation Modelling Practice and Theory 16: 1453-1462.

11. Weisstein EW (2002) CRC concise encyclopedia of mathematics, (2nd edn). Chapman and Hall, USA.

12. Farrell OJ, Ross B (2013) Solved problems in analysis: as applied to gamma, beta, legendre and bessel functions (Dover Books on Mathematics), Dover Publications, USA. 11: 30-34.

13. Luenberger DG (1968) Optimization by vector space methods. John Wiley & Sons.

14. Smith NE, Cobb RG, Pierce SJ, Raska VM (2014) Optimal collision avoidance trajectories via direct orthogonal collocation for unmanned/remotely piloted aircraft sense and avoid operations. AIAA SciTech 2014 Conference. National Harbor, Maryland, USA.

15. (2012) Conference report to accompany HR 658, FAA modernization and reform act of 2012, 112th Congress 2d Session, House of Representatives, Report 112-381 US Government Printing Office, Washington DC.

16. Patterson MA, Rao AV (2013) GPOPS- II: A MATLAB software for solving multiple-phase optimal control problems using hp–adaptive Gaussian quadrature collocation methods and sparse nonlinear programming. ACM Transactions on Mathematical Software.

17. (2010) Design description document (DDD) for multi-sensor integrated conflict avoidance (MuSICA). Northrop Grumman Aerospace Systems, California, USA.

18. Weisstein EW (2013) Point-line distance–3-dimensional. MathWorld A Wolfram Web Resource.

19. Huntington GT (2007) Advancement and analysis of a gauss pseudo-spectral transcription for optimal control problems. Citeseer, American Institute of Aeronautics and Astronautics, Reston, VA, USA.