OJMO

Open Journal of Mathematical Optimization

Sonia Cafieri, Andrew R. Conn & Marcel Mongeau

The continuous quadrant penalty formulation of logical constraints

Volume 4 (2023), article no. 7 (12 pages)

https://doi.org/10.5802/ojmo.28

Article submitted on April 14, 2022, revised on July 17, 2023,
accepted on July 31, 2023.

Open Journal of Mathematical Optimization is a member of the
Centre Mersenne for Open Scientific Publishing
http://www.centre-mersenne.org/
e-ISSN: 2777-5860

CENTRE MERSENNE

The continuous quadrant penalty formulation of logical constraints

Sonia Cafieri
ENAC, Université de Toulouse, France
sonia.cafieri@enac.fr

Andrew R. Conn
IBM T.J. Watson Research Center, NY, USA

Marcel Mongeau
ENAC, Université de Toulouse, France
marcel.mongeau@enac.fr

In memory of our dearest friend Andy Conn

Abstract

Could continuous optimization address efficiently logical constraints? We propose a continuous-optimization alternative to the usual discrete-optimization (big-M and complementary) formulations of logical constraints, that can lead to effective practical methods. Based on the simple idea of guiding the search of a continuous-optimization descent method towards the parts of the domain where the logical constraint is satisfied, we introduce a smooth penalty-function formulation of logical constraints, and related theoretical results. This formulation allows a direct use of state-of-the-art continuous optimization solvers. The effectiveness of the continuous quadrant penalty formulation is demonstrated on an aircraft conflict avoidance application.

Digital Object Identifier 10.5802/ojmo.28

Keywords Logical constraints, penalty function, continuous optimization, aircraft conflicts.

1 Introduction

Logical constraints, such as an alternative constraint ($t(x) \leq 0$ or $f(x) \geq 0$) and a conditional constraint ($t(x) > 0$ implies $f(x) \geq 0$), are common in mathematical optimization models. Mathematical optimization practitioners typically reformulate a logical constraint at the expense of the introduction of one binary auxiliary variable, to get either a *big-M formulation* or a *complementary formulation.* The former involves constraints that are linear but that depend on a large-enough constant yielding potential numerical instability/inefficiency, while a complementary formulation involves nonlinear constraints. We consider whether continuous optimization could play a role in this context. We propose a novel modeling approach, the *continuous quadrant penalty* formulation, based on a continuous optimization penalty function to handle logical constraints without requiring the introduction of binary variables. Our use of penalization is also original in that it does not directly result from adding some measure of the violation of a constraint (difference between the right-hand side and the left-hand side values) in the objective function. More precisely, the penalty function we are introducing is not the usual monotonically non-decreasing function of the distance between a given point and the desirable set. We shall see that this distance is not differentiable; neither is the square of this distance. However, our penalty function can be seen as a smooth approximation of the square of the distance from the desirable set.

The remainder of this paper is structured as follows. Section 2 is a short introduction to logical constraints and penalty-function approaches. Section 3 proposes a simple piecewise-linear penalty function to model logical constraints together with some theoretical results. In order to get round the difficulties related to addressing non-differentiabilities, Section 4 then demonstrates the existence of, and constructs explicitly, a *smooth* penalty function which is piecewise quadratic. This quadrant penalty formulation of logical constraints yields non-convex continuous optimization problems. For designing a completely general, guaranteed methodology, global (continuous) optimization should thereby be invoked. Numerical experiments reported in Section 5 using a state-of-the-art *local* optimization solver on a difficult aircraft conflict avoidance application, demonstrate however the effectiveness of the proposed reformulation. Finally, conclusions are drawn in Section 6.

2 Logical constraints and penalty functions

Conditional constraints have the form

$$t(x) > 0 \quad \text{implies} \quad f(x) \geq 0, \tag{1}$$

for some real-valued functions t and f of $x \in \mathbb{R}^n$. Such logical constraints, together with feasibility constraints, alternative constraints and compound alternatives are common in operations research. They are typically modelled using mixed-integer formulations (see for instance [3]). The conditional constraint (1) is logically equivalent to the alternative constraint:

$$t(x) \leq 0 \quad \text{or} \quad f(x) \geq 0 \tag{2}$$

(at least one condition must be satisfied) since given two logical propositions P and Q, one has that $(P \Rightarrow Q)$ is logically equivalent to $(\neg P \vee Q)$, i.e., non P or Q).

The usual approach to model (2) relies on the so-called *big-M formulation*: (2) is modelled by alternative constraints, involving the introduction of a binary decision variable y, as follows:

$$\begin{aligned} t(x) - M_1 y &\leq 0, \\ -f(x) - M_2(1-y) &\leq 0, \\ y &\in \{0,1\}, \end{aligned} \tag{3}$$

where the *big-M* constants M_1 and M_2 are chosen large enough so that one has:

$$t(x) \leq M_1 \quad \text{and} \quad -f(x) \leq M_2 \quad \text{for all desirable solutions } x. \tag{4}$$

The decision $y = 0$ corresponds to imposing the constraint $t(x) \leq 0$, while $y = 1$ enforces the constraint $f(x) \geq 0$ to be satisfied. Best results in terms of computational times and numerical stability are expected when the constants M_1 and M_2 are chosen as small as possible to satisfy (4).

The other mathematical modeling trick to deal with the alternative constraint (2), called *complementary formulation*, simply yields the constraints:

$$\begin{aligned} t(x)(1-y) &\leq 0, \\ f(x)y &\geq 0, \\ y &\in \{0,1\}, \end{aligned} \tag{5}$$

where, again, y is an extra auxiliary binary decision variable.

The authors of [1] claim that this complementary formulation can be addressed in a more efficient way than the big-M formulation for certain applications involving functions t and f that are linear, as the latter formulation yields weak continuous relaxations while the former allows one to implement tighter bounds to be exploited by MINLP solvers. Remark that obtaining good continuous relaxations (studied, e.g., in [2] in the general context of disjunctive programming) is out of the scope of the present paper, which rather introduces a continuous-optimization formulation.

In the following, we propose an original continuous-optimization model to deal with the conditional constraint (1) that is based on penalization.

The original motivation of penalty functions was to replace difficult constrained optimization problems by a sequence of easier unconstrained optimization problems for which we had good algorithms. There are close relations between penalty methods and numerous nonlinear optimization techniques (e.g., augmented Lagrangian, successive quadratic programming, interior point methods). The pioneering work of Conn in penalty methods dates back to 1971 in his doctoral thesis [7] and the introduction of a method described in [8] to deal with the non-differentiable penalty function:

$$\mu f(x) + \sum_{i \in I} \max(0, g_i(x)) + \sum_{i \in E} |g_i(x)| \tag{6}$$

(where $\mu \geq 0$ is a weighting *penalty parameter*) resulting from the method of replacement [15] of Pietrzykowski (Conn's PhD advisor) to address constraints of the form:

$$g_i(x) \leq 0, \ i \in I; \quad g_i(x) = 0, \ i \in E. \tag{7}$$

Conn concludes his 1981 NATO-conference paper on penalty function methods [9] by asking "whether penalty functions have any significant contributions to make to discrete or global optimization". In [14] Gamble, Conn and Pulleyblank specialize a non-linear penalty function method to a combinatorial algorithm for the minimum cost network flow problem.

The main contribution of the current paper is the proposition of a continuous-optimization alternative to the usual discrete-optimization (big-M (3) and complementary (5)) formulations of logical constraints, that relies on a penalty function. The intuition behind the *continuous quadrant penalty* formulation of the logical constraint we introduce is an attempt to guide the search of a *constrained* continuous-optimization method towards the parts of the domain where the logical constraint (1) is satisfied. Remark that in this context *only* the logical constraints will be penalized, as other types of constraints (such as those of the application we consider in Section 5) can readily be addressed by state-of-the-art continuous optimization methods.

Let us consider, for any given decision-variable vector value $x \in \mathbb{R}^n$, a couple, $(t(x), f(x)) \in \mathbb{R}^2$. In the sequel, we shall drop the explicit dependency upon x for simplicity of the presentation. Imposing the conditional constraint (1) is therefore equivalent to requiring that the point $p := (t, f)$ lies in the non-convex set $\mathbb{R}^2 \setminus S$, where S is the open fourth quadrant:

$$S := \{(t, f) : t > 0 \text{ and } f < 0\}, \tag{8}$$

as illustrated in the grey areas in Figure 1. Thus, one can obtain a continuous optimization formulation avoiding the introduction of the binary variable y by replacing the conditional constraint

$$t > 0 \quad \text{implies} \quad f \geq 0, \tag{9}$$

with the constraint:

$$(t, f) \in \mathbb{R}^2 \setminus S. \tag{10}$$

It is this constraint that we propose to model by the use of a penalty approach. To this aim, let us design an appropriate penalty function, $g : \mathbb{R}^2 \to \mathbb{R}$. It is desirable to consider candidate functions g that are continuous, possibly even smooth, and such that a descent method minimizing g will converge towards a point in $\mathbb{R}^2 \setminus S$.

Let us introduce a first general definition to formalize the intuitive idea of restricting our search for a penalty function to the set of functions that penalize only the set S, the "indesirable" part of the domain.

Definition 1 (S-violation dipstick)**.** *Let $\emptyset \neq S \subsetneq \mathbb{R}^n$. A function $g : \mathbb{R}^n \to \mathbb{R}$ is an* S-violation dipstick *if it satisfies:*
a. $g(p) = 0, \quad$ *if* $p \in \mathbb{R}^n \setminus S$
b. $g(p) > 0, \quad$ *if* $p \in S$.

The indicator function $\mathbf{1}_S$ (whose value is one in S, and zero elsewhere) is a special case of S-violation dipsticks. Other examples are each of the terms of the classical ℓ_1 penalty function:

$$\widehat{g}(x) := \sum_{i \in I} \max(0, g_i(x)) + \sum_{i \in E} |g_i(x)| \tag{11}$$

corresponding to the constraints (7). Indeed, given $i \in I$, the term $\max(0, g_i(x))$ is an S-violation dipstick, considering S to be here the subdomain where the constraint $g_i(x) \leq 0$ is violated (and similarly for $i \in E$ and the term $|g_i(x)|$).

The following general proposition simply states that a "reasonable" penalty function (see Definition 1) that takes zero value on a non-convex set (and only on this non-convex set) is necessarily a non-convex function.

Proposition 2. *Let $S \subset \mathbb{R}^n$ be such that $\mathbb{R}^n \setminus S$ is not convex. Then, no S-violation dipstick function can be convex.*

Proof. Let $g : \mathbb{R}^n \to \mathbb{R}$ be an S-violation dipstick function. Since the set $\mathbb{R}^n \setminus S$ is not convex, there exist necessarily two points $x, y \in \mathbb{R}^n \setminus S$, $x \neq y$, and a scalar $0 < \lambda < 1$ such that the point $\lambda x + (1 - \lambda)y$ lies outside $\mathbb{R}^n \setminus S$, i.e., in S. Thus, $g(\lambda x + (1 - \lambda)y) > 0$ by Definition 1 b). Moreover, by a), $\lambda g(x) + (1 - \lambda)g(y) = 0$, since both x and y lie in $\mathbb{R}^n \setminus S$. Thus, g is not a convex function. ◀

In particular, the above proposition says that an S-violation dipstick cannot be a linear function when $\mathbb{R}^n \setminus S$ is not convex. One must therefore search for a somewhat more complex penalty function.

3 A piecewise-linear penalty function

A first natural choice beyond linearity is that of a piecewise-linear function. The next proposition states that at least three pieces are then needed for designing such a penalty function.

Proposition 3. *Let $\emptyset \neq S \subsetneq \mathbb{R}^n$. Unless S is a half-space, there does not exist a continuous* two-piece *piecewise-linear S-violation dipstick.*

Proof. Let $g : \mathbb{R}^n \to \mathbb{R}$ be a continuous two-piece piecewise-linear S-violation dipstick. The function g must then be linear over S with value zero at the frontier of S. This is not possible unless S is a half-space or g is identically zero. ◀

Let us come back to the case of interest for us, where $n = 2$ and S is the open fourth quadrant. From the two above propositions, we know that if we want to design a continuous S-violation dipstick penalty function that is piecewise linear, it must have at least three pieces (since $\mathbb{R}^2 \setminus S$ is not convex and, a fortiori, not a half-space).

Proposition 4. *Let $S \subseteq \mathbb{R}^2$ be the open fourth quadrant. There exists a continuous piecewise-linear S-violation dipstick, noted g_α, having exactly three pieces:*

$$g_\alpha(t, f) = \begin{cases} 0, & \text{if } t \leq 0 \text{ or } f \geq 0 \\ -\dfrac{f}{\alpha}, & \text{if } -\alpha\, t \leq f \leq 0 \\ t, & \text{if } 0 \leq t \leq -\dfrac{f}{\alpha}, \end{cases} \tag{12}$$

where $\alpha > 0$ is any given positive (slope) parameter. Moreover, up to a multiplicative constant, and up to the arbitrary value of α, this function is unique.

The values of g_α over its three pieces are displayed on Figure 1. Remark that g_α can also be expressed as the one-line formula: $g_\alpha(t, f) = \max(0, \min(t, -f/\alpha))$.

Figure 1 Values of the unique continuous three-piece piecewise-linear S-violation dipstick function g_α (left) and its gradients (right).

Proof. Regarding the existence part, one can easily verify that g_α is a continuous three-piece piecewise-linear S-violation dipstick function.

As for the unicity of g_α, first observe that S is necessarily partitioned into two pieces, over each of which g_α is linear, and the common frontier of these two pieces must be a *straight-line* segment (otherwise, g_α could not possibly be linear over each of theses two pieces). Moreover, this straight-line segment frontier must go through the origin. Indeed, if this were not the case, as in the proof of Proposition 3, g_α would be zero over one of the two S pieces which, in turn, would yield an identically-null function g_α. Without loss of generality, let $f = -\alpha t$ be the equation of the corresponding straight line, where $\alpha > 0$. Further, let g_{α_1} and g_{α_2} be respectively the restriction of g_α over each of the two S pieces. One can show (see [11, Proposition 1]) that for g_α to be continuous, the difference of gradients,$\nabla g_{\alpha_1} - \nabla g_{\alpha_2}$, is necessarily a (non-zero) multiple of $(\alpha, 1)^T$ (the gradient of the *ridge*: $\{(t, f) : f = -\alpha t)\}$). This, together with the fact that g_{α_1} and g_{α_2} are linear functions satisfying $g_{\alpha_1}(0, f) = 0$ for all $f < 0$ and $g_{\alpha_2}(t, 0) = 0$ for all $t > 0$ yields the continuous piecewise linear function whose gradients are displayed on Figure 1 (right), up to some positive constant. The only such piecewise linear function is, up to this multiplicative constant, the proposed function g_α. ◀

We introduce the next definition in order to restrict our search to the set of penalty functions that, roughly speaking, are likely to guide a descent method away from the "undesirable" part (the set S) of the feasible domain.

Definition 5 (function leaning outwards S)**.** *Let $\emptyset \neq S \subsetneq \mathbb{R}^n$. A function $g : \mathbb{R}^n \to \mathbb{R}$ is said to* lean outwards *S if for any given point $\bar{p} \in S$, and for any descent direction $\bar{d}$ for g at $\bar{p}$, there exists a threshold step size $\bar{\gamma} > 0$ such that $\bar{p} + \gamma\bar{d} \notin S$, for all $\gamma \geq \bar{\gamma}$.*

Let us consider again the example of the classical ℓ_1 penalty function (11), this time in the simple special case of the following 2D quadratic constraint:

$$c(x) := x_1^2 + 4x_2^2 - 1 \leq 0,$$

i.e., $\widetilde{g}(x) = \max(0, c(x))$. This function $\widetilde{g}$ does *not* lean outwards the subdomain, $S = \{x : c(x) > 0\}$, where the constraint is violated. To see this, consider the point $\bar{p} = (\frac{3}{2}, \frac{\sqrt{7}}{4})$. The point $\bar{p}$ is in S, the steepest-descent direction for $\widetilde{g}$ at $\bar{p}$ is $\bar{d} = -(3, 2\sqrt{7})$, and for whatever step size, $\gamma > 0$, chosen one can show that $\bar{p} + \gamma\bar{d} \in S$. An even simpler manner to see this here is to observe that there cannot be any penalty function that can lean outwards a set S when the complement of this set is bounded.

It seems therefore rather demanding to require a penalty function to lean outwards the infeasible domain. However, the piecewise-linear penalty function g_α introduced in Proposition 4 does lean outwards the forbidden area S defined by (8), as this will be demonstrated below (Proposition 7).

We first introduce a definition that combines three properties that are desirable for the penalty function we wish to design.

Definition 6 (S-repulsive penalty function)**.** *Let $\emptyset \neq S \subsetneq \mathbb{R}^n$. A function $g : \mathbb{R}^n \to \mathbb{R}$ is an S-*repulsive penalty function *if it satisfies the following three requirements:*

i. *g is an S-violation dipstick*
ii. *g leans outwards S*
iii. *g is continuous*

Proposition 7. *Let $S \subseteq \mathbb{R}^2$ be defined by* (8)*. The piecewise-linear function g_α defined by* (12) *is an S-repulsive penalty function.*

Proof. Parts i and iii were shown above. In order to show that g_α leans outwards S, let $\bar{p} = (\bar{t}, \bar{f}) \in S$, and let $\bar{d} = (\bar{d}_1, \bar{d}_2) \in \mathbb{R}^2$ be a descent direction for g_α at $\bar{p}$.

Consider first the case where $\bar{p}$ is *not* on the ridge where $\bar{f} = -\alpha\bar{t}$ (i.e., g_α is differentiable at $\bar{p}$). By definition of descent direction, then $\bar{d}^T \nabla g_\alpha(\bar{p}) < 0$:

a. If $\bar{f} < -\alpha\bar{t}$, then $\nabla g_\alpha(\bar{p}) = (1, 0)^T$ (Figure 1, right), and it follows: $\bar{d}^T(1, 0) < 0$, or $\bar{d}_1 < 0$ (since $\bar{t} > 0$). Setting $\bar{\gamma} = \frac{-\bar{t}}{\bar{d}_1}$, the t-component of $\bar{p} + \gamma\bar{d}$ is $\bar{t} + \gamma\bar{d}_1$, which is clearly non-positive for all $\gamma \geq \bar{\gamma}$. Thus, $\bar{p} + \gamma\bar{d} \notin S$, for all $\gamma \geq \bar{\gamma}$.
b. In the case where $\bar{p}$ lies in the other subset of S (where $\bar{f} > -\alpha\bar{t}$), one obtains similarly: $\nabla g_\alpha(\bar{p}) = (0, \frac{-1}{\alpha})^T$, and $\bar{d}_2 > 0$ (since $\bar{f} < 0$), so that, setting this time $\bar{\gamma} = \frac{\bar{f}}{\bar{d}_2}$, the f-component of $\bar{p} + \gamma\bar{d}$ will be non-negative for all $\gamma \geq \bar{\gamma}$.

Finally, for the case where g_α is not differentiable at $\bar{p} = (\bar{t}, \bar{f})$ (i.e., $\bar{f} = -\alpha\bar{t}$), the set of all possible descent directions is:

$$\{d \in \mathbb{R}^2 : d^T(1, 0) < 0\} \cup \left\{d \in \mathbb{R}^2 : d^T\left(0, \frac{-1}{\alpha}\right) < 0\right\}.$$

Thus, one necessarily has either $\bar{d}_1 < 0$ or $\bar{d}_2 > 0$. It suffices to set $\bar{\gamma} = -\frac{\bar{t}}{\bar{d}_1}$ in the first case, and $\bar{\gamma} = -\frac{\bar{f}}{\bar{d}_2}$ in the second case, to obtain: $\bar{p} + \gamma\bar{d} \notin S$, for all $\gamma \geq \bar{\gamma}$. ◀

Let us consider for the remaining of the paper the case where $n = 2$ and S is the open fourth quadrant, relevant for addressing the logical constraint (9).

Although the piecewise-linear function g_α has several desirable properties as a candidate for a penalty function (it is S-repulsive), it is *not smooth* along the three straight-line-segment frontiers: $t = 0$, $f = 0$ and $f = -\alpha t$. For the sake of simplification, let us focus on the case where $\alpha = 1$.

The non-differentiabilities of g_α correspond to the three points of discontinuity of the restriction of g_α (displayed on Figure 2, left) to the line $f = t - 2X$ (for some given $X > 0$): at the abscissas $\pm\sqrt{2}X$ and at 0.

This non-smoothness is a drawback when selecting a state-of-the-art optimization method for addressing a penalty-function-based model. Special-purpose optimization methods could be designed for such structured non-differentiable objective functions, as it is done for instance in [10, 11]. However, g_α is not *decomposable*

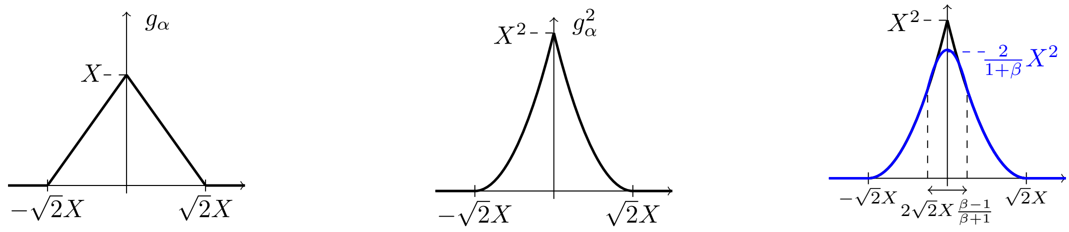

Figure 2 g_α (left) and g_α^2 (center) for $\alpha = 1$, restricted to the line (in blue on Figure 3, left) $f = t - 2X$, for some $X > 0$. Right: g_α^2 (black) and g_β (blue) when β approaches 1.

(in the sense of [11, Theorem 2]) at the point $(0, 0)$, as shown in [11, Section 6.2]. This implies that here one cannot rely on the approach proposed in [11] for optimizing general piecewise-linear functions. Moreover, we are here focusing on addressing logical constraints with standard continuous optimization approaches. Note that, contrary to the common approach of squaring the terms of the ℓ_1-penalty function (11) in order to smooth out the non-differentiabilities, here one can easily verify that along the half-line $f = -\alpha t$ with $t > 0$, the function g_α^2 is *not* differentiable (see Figure 2, center). A smoothing technique such as that proposed by Pinar and Zenios [16] that replaces a $\max(0, f)$ term with a (three-piece) piecewise-quadratic function of f is not appropriate either according to Proposition 2, as their technique can be applied only to convex functions.

4 A smooth piecewise-quadratic penalty function

Let us construct a *smooth* penalty function, so that efficient state-of-the-art methods can be applied to the resulting optimization problem. We choose here to build a piecewise-*quadratic* approximation (that we shall name g_β) of g_α^2 (our non-convex piecewise-linear penalty function, g_α, squared).

One can easily infer that a smooth S-violation dipstick penalty function that is piecewise quadratic will have *at least* four pieces. Indeed, simply consider the restriction of such a function, say g, to the line $f = t - 2X$, for some positive constant X, as displayed on Figure 3 (center): it is easy to show that there is no way to join smoothly any two quadratic polynomials p_1 and p_3 whose value and derivative are zero at the intersections of the line and the axes; a third quadratic polynomial, p_2, is needed.

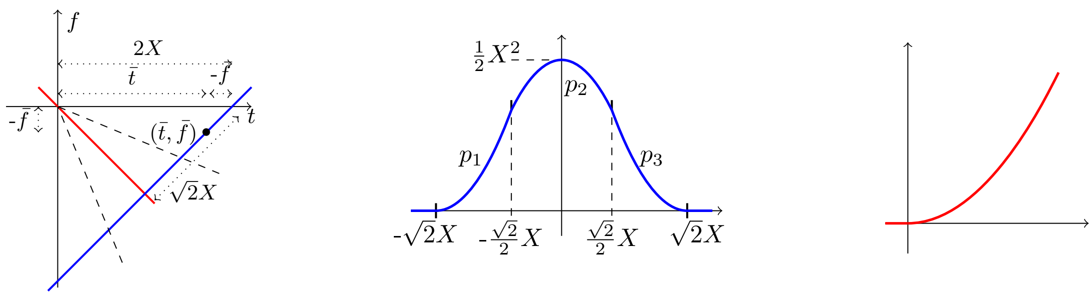

Figure 3 Graphs of g_β restricted to two lines (left): $f = t - 2X$ (center) and $f = -t$ (right), when $\beta = 3$

Three pieces are therefore required over the fourth quadrant, plus one piece where g is identically zero in the remaining quadrants.

Proposition 8 (Existence). *Let $S \subseteq \mathbb{R}^2$ be the open fourth quadrant. There exists a* smooth *S-repulsive penalty function, $g_\beta : \mathbb{R}^2 \to \mathbb{R}$, that is piecewise quadratic with exactly four pieces. A family of such functions is, for $\beta \in \mathbb{R}$, $\beta > 1$:*

$$g_\beta(t, f) = \begin{cases} 0 & \textit{if } t \le 0 \textit{ or } f \ge 0 \\ t^2 & \textit{if } 0 < t \le -\dfrac{f}{\beta} \\ \dfrac{1}{1-\beta^2}(t^2 + 2\beta\ t\ f + f^2) & \textit{if } -\dfrac{f}{\beta} < t < -\beta f \\ f^2 & \textit{if } -\dfrac{t}{\beta} \le f < 0. \end{cases} \tag{13}$$

Before proving this result, let us visualize this 2D function, g_β, for a particular value of the parameter: $\beta = 3$. It takes the value zero outside the fourth quadrant, and is composed of three quadratic pieces in the fourth quadrant. Figure 4 shows the values g_β takes over its four subdomains, and Figure 5 displays the 3D graph of g_β.

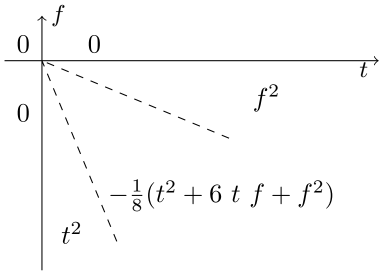

Figure 4 The four quadratic pieces of g_β, when $\beta = 3$.

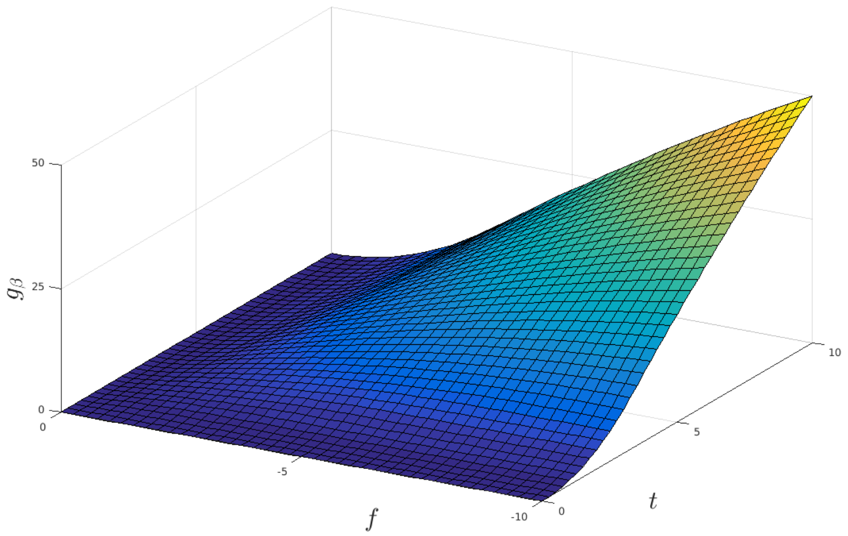

Figure 5 The 3D graph of g_β, when $\beta = 3$.

Figure 3 displays the restriction of g_β to a transverse line $f = t - 2X$, for some given value of $X > 0$ (center), and the restriction of g_β along its axis of symmetry: the line $f = -t$ (right). Observe that along any ray into S, the restriction of g_β is a convex parabola whose minimum corresponds to $(0, 0)$, as shown on Figure 3 (right).

Proof. Let $\beta > 1$. We shall first show that g_β is an S-violation dipstick, then that g_β is a smooth function, and finally that g_β leans outwards S.

In order to show that g_β is an S-violation dipstick, consider a point $p = (t, f) \in \mathbb{R}^2$. If $p \notin S$, then by definition of g_β, $g_\beta(p) = 0$. If p pertains to either subdomain where $g_\beta(t, f) = t^2$ or f^2, then one clearly has $g_\beta(p) > 0$. In order to prove that g_β is positive on the remaining subdomain, where

$$-\frac{f}{\beta} < t < -\beta f, \tag{14}$$

consider the univariate function h defined as the restriction of g_β to the line, illustrated on Figure 3 (right): $f = t - 2X$, for some given value $X > 0$ (any given point $(\bar{t}, \bar{f})$ pertains to such a line: simply take $X = \frac{\bar{t} - \bar{f}}{2}$). One has:

$$h(t) = \frac{1}{1 - \beta^2}\big(t^2 + 2\beta t(t - 2X) + (t - 2X)^2\big).$$

Observe that on that line, (14) reads $-\frac{t-2X}{\beta} < t < -\beta(t-2X)$, or $\frac{2X}{\beta+1} < t < \frac{2X\beta}{\beta+1}$. On the closed interval $\frac{2X}{\beta+1} \leq t \leq \frac{2X\beta}{\beta+1}$, the univariate concave quadratic function h is minimized either at $t = \frac{2X}{(\beta+1)}$ or at $t = \frac{2X\beta}{(\beta+1)}$. Evaluating h at these points, one obtains: $h(\frac{2X}{\beta+1}) = \frac{1}{1-\beta^2}\big((\frac{2X}{\beta+1})^2 + 2\beta\frac{2X}{\beta+1}(\frac{2X}{\beta+1} - 2X) + (\frac{2X}{\beta+1} - 2X)^2\big) = \frac{4X^2}{(\beta+1)^2} > 0$ and similarly, one finds $h(\frac{2X\beta}{\beta+1}) = \frac{4X^2}{(\beta+1)^2} > 0$. Therefore, g_β is also positive on the domain where (14) is satisfied, and g_β is therefore an S-violation dipstick.

To show that g_β is a smooth function, remark first that g_β is continuous along the four lines: $t = 0$, $f = 0$, $f = -\beta t$, and $f = -t/\beta$ that delimit the four subdomains where g_β is a quadratic function. Moreover, one clearly has:

$$\nabla g_\beta(t,f) = \begin{cases} (0,0)^T & \text{if } t < 0 \text{ or } f > 0 \\ (2t,0)^T & \text{if } 0 < t < -\dfrac{f}{\beta} \\ \dfrac{2}{1-\beta^2}(t+\beta f, f+\beta t)^T & \text{if } -\dfrac{f}{\beta} < t < -\beta f \\ (0,2f)^T & \text{if } -\dfrac{t}{\beta} < f < 0. \end{cases} \tag{15}$$

It is then straightforward to verify that: $(0,0)^T = (2t,0)^T$ along the line $t = 0$, $(0,0)^T = (0,2f)^T$ along $f = 0$, $(2t,0)^T = \frac{2}{1-\beta^2}(t+\beta f, f+\beta t)^T$ along $f = -\beta t$, and that $\frac{2}{1-\beta^2}(t+\beta f, f+\beta t)^T = (0,2f)^T$ along $f = -t/\beta$. Therefore, g_β is smooth.

Finally, in order to show that g_β leans outwards S, let $\bar{p} = (\bar{t}, \bar{f}) \in S$, and let $\bar{d} = (\bar{d}_1, \bar{d}_2) \in \mathbb{R}^2$ be a descent direction for g_β at $\bar{p}$. By definition of descent direction, then $\bar{d}^T \nabla g_\beta(\bar{p}) < 0$. According to the location of $\bar{p}$, we have three cases to consider. If $0 < \bar{t} \leq -\bar{f}/\beta$, then $\nabla g_\beta(\bar{p}) = (2\bar{t}, 0)^T$, which implies $2\bar{t}\bar{d}_1 < 0$, or $\bar{d}_1 < 0$. The conclusion follows exactly as in part i) of the proof of Proposition 7. The case where $-\frac{\bar{t}}{\beta} < \bar{f} < 0$ is analogous. For the last case, where $-\frac{\bar{f}}{\beta} < \bar{t} < -\beta\bar{f}$, one obtains:

$$\bar{d}^T \nabla g_\beta(\bar{p}) = \frac{2}{1-\beta^2}(\bar{t}+\beta\bar{f}, \bar{f}+\beta\bar{t})^T \bar{d} < 0.$$

Since $\frac{2}{1-\beta^2} < 0$, $\bar{t}+\beta\bar{f} < 0$ and $\bar{f}+\beta\bar{t} > 0$, one cannot have both $\bar{d}_1 \geq 0$ and $\bar{d}_2 \leq 0$. Therefore, either $\bar{d}_1 < 0$ or $\bar{d}_2 > 0$, and it suffices to set $\bar{\gamma} = -\frac{\bar{t}}{\bar{d}_1}$ in the first case, and $\bar{\gamma} = -\frac{\bar{f}}{\bar{d}_2}$ in the second case, to obtain: $\bar{p} + \gamma\bar{d} \notin S$, for all $\gamma \geq \bar{\gamma}$. ◀

▶ Remark.

1. The piecewise quadratic function, g_β, is not unique. It is defined up to a multiplicative constant, and changing the units of t and f (scaling) amounts to changing the slope of its axis of symmetry (here set to $f = -t$).
2. As for classical penalty functions, g_α (in the case where $\alpha = 1$) is a monotonically non-decreasing *function of the (ℓ_2) distance* between a given point and the desirable set, in our case the non-convex set $\mathbb{R}^2 \setminus S$. Let us denote this distance by $d_{\mathbb{R}^2\setminus S}$. In contrast, it is easy to show that the penalty function g_β (with $\beta > 1$) is *not* a monotonically non-decreasing function of $d_{\mathbb{R}^2\setminus S}$. This is illustrated on Figure 6 where, for a small-enough positive step size $\gamma > 0$: $g_\beta(\bar{x}) < g_\beta(\bar{x} + \gamma d)$ while $d_{\mathbb{R}^2\setminus S}(\bar{x}) > d_{\mathbb{R}^2\setminus S}(\bar{x} + \gamma d)$.

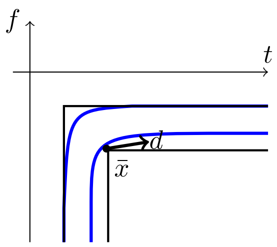

Figure 6 Level curves of the functions g_α and $d_{\mathbb{R}^2\setminus S}$ (which are identical, in black), and those of g_β (blue) in the case where $\alpha = 1$ and $\beta = 3$.

3. When $\alpha = 1$ and β takes a value near 1, the angle of the sector $-\frac{f}{\beta} < t < -\beta f$ (over which g_β takes the values of the concave quadratic function $\frac{1}{1-\beta^2}(t^2 + 2\beta\ t\ f + f^2)$, see Figure 4) becomes small, and g_β approaches the *three*-piece piecewise quadratic function g_α^2 (obtained by squaring the piecewise linear function g_α). The

function g_β can therefore be seen as a smooth approximation of g_α^2 (for the symmetric case where $\alpha = 1$). In fact, both functions coincide, except on the central sector, as illustrated on Figure 2 that plots the restrictions of g_α^2 (for $\alpha = 1$, center) and of g_β (right) along the transversal line $f = t - 2X$ $(X > 0)$ for a value of β near 1.

4. In order to guide a user to choose a particular value of β, let us consider again the restriction, h^X, of g_β to the line $f = t - 2X$, for some given $X > 0$. This univariate piecewise-defined function h^X is composed of five pieces: one straight-line (identically zero), three univariate quadratic polynomials p_1^X, p_2^X, p_3^X, and again one straight-line (identically zero). The case where each of the four *breakpoints* (discontinuities in the second derivative of h^X) together with the point corresponding to the axis of symmetry of h^X are equally spread over the domain where h^X is positive, corresponds to the special case where $\beta = 3$, and is illustrated on Figure 3.

5 Computational assessment

The objective of this section is to demonstrate the usefulness of the quadrant penalty formulation of logical constraints for real-world optimization problems. To this aim, we address in particular a difficult problem arising in aeronautics, and discuss some numerical results obtained solving a problem formulation based on g_β and presented in [5], in which the interested reader will find a more thorough presentation of the model and extensive results. We consider the aircraft conflict avoidance problem, which involves several logical constraints, as many as there are possible pairs of aircraft at stake. The aim is to keep aircraft pairwise separated by a given *separation distance*, d, (otherwise, there is a *conflict*). The inherent combinatorics in such *conflict avoidance problems* leads naturally to conditional constraints: if two vehicles are converging spatially, then their inter-distance must be bounded below at all time. The usual big-M or complementary formulation approaches to address these logical constraints require introducing as many binary variables as there are pairs of aircraft. We show that our quadrant penalty formulation of logical constraints allows continuous optimization to make a significant contribution to solving such problems that are typically addressed by combinatorial optimization approaches (see [4] and references therein). To that aim, we consider here more specifically the continuous optimization formulation of this problem, presented in [5].

In the variant of the aircraft conflict avoidance problem that is the closest to air transportation operational constraints, both the speed and the heading angle of each aircraft can be changed to *separate* the trajectories (see e.g., [12]). Let $A = \{1, 2, \ldots, n\}$ be an index set corresponding to aircraft flying at a same given altitude, on straight-line segment trajectories. The following input data are given for each aircraft $i \in A$ at a certain time of interest $t = 0$: initial position, $(x_i^0, y_i^0) \in \mathbb{R}^2$; initial speed, v_i; and initial heading angle, ϕ_i. The aim is to decide speed and heading-angle variations for each aircraft i, simultaneously, to guarantee that aircraft remain pairwise separated. In other words, considering a particular pair of aircraft $i, j \in A$ $(i < j)$, the following *separation constraint* must be satisfied:

$$\|\mathbf{x}_{ij}(t)\| \geq d \qquad \text{for all} \quad t \geq 0, \tag{16}$$

where the vector $\mathbf{x}_{ij}(t) = \mathbf{x}_i(t) - \mathbf{x}_j(t) \in \mathbb{R}^2$ gives the position of aircraft i relative to that of aircraft j, and where $\|\cdot\|$ stands for the Euclidean norm. This constraint not only induces in general a non-convex feasible set, but it must moreover hold at *all* time $t \geq 0$.

Following [6], one can reformulate this *semi-infinite* constraint (defined over a *continuous* time interval) by first re-writing it as the univariate quadratic constraint:

$$f_{ij}(t) = \|\mathbf{v}_{ij}\|^2 t^2 + 2\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij} t + \|\mathbf{x}_{ij}^0\|^2 - d^2 \geq 0, \qquad \text{for all} \quad t \geq 0, \tag{17}$$

where $\mathbf{v}_{ij}$ is the velocity vector of aircraft i relative to that of aircraft j, and $\mathbf{x}_{ij}^0$ is the vector of their relative initial positions. Then, the separation constraint (16) is shown to be equivalent to the logical constraint:

$$t_{ij}^m > 0 \quad \text{implies} \quad f_{ij}^m \;\geq\; 0, \tag{18}$$

where $t_{ij}^m = \frac{-\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij}}{\|\mathbf{v}_{ij}\|^2}$ and $f_{ij}^m = \|\mathbf{v}_{ij}\|^2(\|\mathbf{x}_{ij}^0\|^2 - d^2) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2$ are respectively the minimizer (known as the time of *closest point of approach* in aeronautics) and the minimal value of f_{ij}.

The aircraft conflict avoidance problem boils down to a *feasibility* problem that aims at deciding speed variation, q_i, and heading angle variation, θ_i maneuvers for each aircraft $i \in A$ so that the separation constraints are satisfied. More precisely, once these decisions are made, the new speed of aircraft i is $q_i v_i$, and its new heading angle is $\phi_i + \theta_i$, where v_i is the aircraft original speed, and ϕ_i is its original angle. Following [5], for the angle decision, we rather consider the heading-angle maneuver variables, ω_i, π_i, that are related to both the original angle decision variable, θ_i, and to the speed decision variable, q_i, as follows:

$$\omega_i := \cos(\phi_i + \theta_i) q_i v_i \tag{19}$$
$$\pi_i := \sin(\phi_i + \theta_i) q_i v_i. \tag{20}$$

The constraints to be satisfied are:

$$t^m_{ij} > 0 \quad \text{implies} \quad f^m_{ij} \geq 0 \qquad i,j \in A : i < j \tag{21}$$
$$f^m_{ij} \|\mathbf{v}_{ij}\|^2 = \|\mathbf{v}_{ij}\|^2 (\|\mathbf{x}^0_{ij}\|^2 - d^2) - (\mathbf{x}^0_{ij} \cdot \mathbf{v}_{ij})^2 \qquad i,j \in A : i < j \tag{22}$$
$$t^m_{ij} \|\mathbf{v}_{ij}\|^2 = -\mathbf{x}^0_{ij} \cdot \mathbf{v}_{ij} \qquad i,j \in A : i < j \tag{23}$$
$$\mathbf{v}_{ij} = \begin{pmatrix} \omega_i - \omega_j \\ \pi_i - \pi_j \end{pmatrix} \qquad i,j \in A : i < j \tag{24}$$
$$\omega_i^2 + \pi_i^2 = (q_i v_i)^2 \qquad i \in A \tag{25}$$
$$\underline{q_i} \leq q_i \leq \overline{q_i} \qquad i \in A \tag{26}$$
$$\underline{\omega_i} \leq \omega_i \leq \overline{\omega_i} \qquad i \in A \tag{27}$$
$$\underline{\pi_i} \leq \pi_i \leq \overline{\pi_i} \qquad i \in A, \tag{28}$$

where $\mathbf{v}_{ij}$, t^m_{ij}, and f^m_{ij} are auxiliary decision variables that can be straightforwardly computed explicitly in terms of the decision variables: q_i, ω_i, π_i and q_j, ω_j, π_j and of the input data, and where $\underline{q_i}, \overline{q_i}, \underline{\omega_i}, \overline{\omega_i}, \underline{\pi_i}, \overline{\pi_i}$ are given lower and upper bounds on the main decision variables.

Modeling the logical constraints (21) through the continuous quadrant penalty function, g_β, introduced in Proposition 8, yields an optimization problem whose objective function, to be minimized, reads:

$$\sum_{i,j \in A : i < j} g_\beta(t^m_{ij}, f^m_{ij}) \tag{29}$$

subject to constraints (22)–(28). This allows the direct use of state-of-the-art continuous optimization solvers.

Remark that the objective function of this optimization problem is not a convex function (by Proposition 2), and the feasible domain (22)–(28) is not a convex set, which can be seen simply by considering constraints (25). A mathematical optimization method for continuous optimization may thereby converge to a local minimum that does not satisfy the original (penalized) logical constraints (non-zero optimal value), yielding a so-called *local infeasibility*.

The instances considered are realistic benchmark problems known as Random Circle Problems (RCP). They are available from [12] and depend on the number, n, of aircraft. We consider three different values for n, namely $n = 10, 20$ and 30, and for each of these values we perform tests on 10, 10 and 15 instances, respectively.

We set the quadrant-penalty parameter to $\beta = 3$ (corresponding to equally-spreaded breakpoints for h_X, as mentioned above and illustrated on Figure 3.) To handle the issue of potential *local infeasibilities* mentioned above, we implement a simple multistart heuristic, from different (randomly-generated) starting guesses.

The optimization model is implemented in `AMPL` [13] and the instances are solved using the `IPOPT` solver for nonlinear optimization problems [17], version 3.12, with the feasibility tolerance set to 10^{-5}. A 2.66 GHz Intel Xeon (octo core) processor with 32 GB of RAM under Linux is used to perform our numerical tests.

Conducting computational tests on the 35 instances yields the following observations.

First, a solution satisfying the separation constraints (zero-value penalty function) is always obtained. At most 2 runs of the multistart heuristic are needed, and this happens only for 2 instances out of the 35 instances studied (one with $n = 20$ and one with $n = 30$). All the computed solutions are globally optimal with respect to the penalty objective function (29), as its lower bound zero is attained.

Second, the CPU times (reported in Table 1) reveal that we benefit from the efficiency of continuous optimization methods to compute solutions satisfying the difficult separation constraints, this computation being generally demanding for large-scale instances when such logical constraints are addressed by combinatorial optimization approaches.

Computing times do not exceed 3.09 seconds for instances involving $n = 20$ aircraft sharing the same airspace, and this number of aircraft is already larger than what is commonly observed in practice when air traffic

Table 1 Mean value, standard deviation, minimum and maximum values of the total CPU times (in seconds), for the instances involving n aircraft

	CPU time (seconds)			
n	mean	st.dev.	min	max
10	0.23	0.28	0.00	0.94
20	1.05	0.99	0.19	3.09
30	3.68	4.79	0.90	20.06

controllers deconflict cruise flights. Even for the larger instances involving $n = 30$ aircraft (requiring 3.68 seconds, on average), our tests tend to show that the proposed approach can be useful in the real-world application context considered. The maximum computing time (20.06 seconds) results from the sum of the CPU times of two runs required by the multistart heuristic. The 14 remaining instances involving $n = 30$ aircraft do not require multistart, and solutions are computed in less than 6 seconds. The distribution of the values of the CPU times for all the considered instances with $n = 10, 20$ and 30 is displayed in Figure 7.

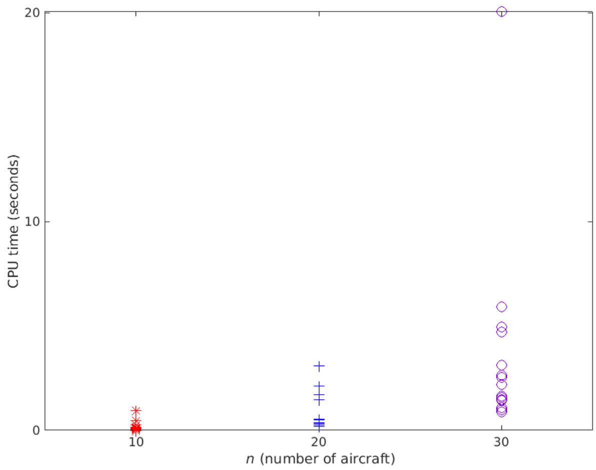

Figure 7 CPU times (seconds) for the instances involving $n = 10, 20, 30$ aircraft

To summarize, the approach that we propose in this paper is not directly comparable to the traditional global optimization approach that relies on introducing binary variables, since we rather compute locally-optimal solutions of a continuous optimization problem which is not convex. However, our alternative formulation of the logical constraints arising in the aircraft separation requirement allows one to determine, efficiently, good-quality (upper, in a minimization problem) bounds. These upper bounds are shown in [5] to have a significant impact on an exact global resolution by branch-and-bound. More precisely, the exact mixed-integer nonlinear optimization approach (relying on the classical complementary formulation of the logical constraints) introduced in [5] cannot solve within 600 seconds difficult instances, among the largest ones, of the problem considered here. However, when combining the strength of the upper bounds obtained by our continuous quadrant penalty formulation with this mixed-integer nonlinear programming approach within a dedicated 3-phase approach proposed in [5], the most difficult instances are solved with a proof of global optimality.

6 Conclusion

Logical constraints are ubiquitous in practical optimization. This paper proposes a continuous-optimization alternative to the two classical, big-M and complementary, mixed-integer formulations of logical constraints: the *continuous quadrant penalty* formulation. We emphasize some theoretical properties of the 2D smooth non-convex piecewise-quadratic penalty function upon which is based the new approach. We demonstrate the usefulness of our approach on a real-world application, the aircraft conflict avoidance problem, which is commonly addressed by combinatorial optimization, requiring as many extra binary variables as there are possible pairs of vehicles with the classical ways of reformulating logical separation constraints.

This work is in the spirit of the questioning of Conn as to whether penalty functions can have a significant impact in discrete or global optimization. Future work will address other problems involving logical constraints that would incur too numerous extra binary variables. Addressing *optimization* problems will involve managing a weighting penalty parameter to take into account the original problem's objective.

Acknowledgments

S. Cafieri and M. Mongeau thank Barbara Conn for proofreading the manuscript.

References

1 Pietro Belotti, Pierre Bonami, Matteo Fischetti, Andrea Lodi, Michele Monaci, Amaya Nogales-Gómez, and Domenico Salvagnin. On handling indicator constraints in mixed integer programming. *Comput. Optim. Appl.*, 65(3):545–566, 2016.

2 Pierre Bonami, Andrea Lodi, Andrea Tramontani, and Sven Wiese. On Mathematical Programming with indicator constraints. *Math. Program.*, 151:191–223, 2015.

3 Stephen P. Bradley, Arnoldo C. Hax, and Thomas L. Magnanti. *Applied Mathematical Programming.* Addison-Wesley Publishing Group, 1977. Available on line: `http://web.mit.edu/15.053/www/AMP.htm`.

4 Sonia Cafieri. MINLP in Air Traffic Management: Aircraft conflict avoidance. In Tamás Terlaky, Miguel F. Anjos, and Shabbir Ahmed, editors, *Advances and Trends in Optimization with Engineering Applications*, MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2017.

5 Sonia Cafieri, Andrew R. Conn, and Marcel Mongeau. Mixed-integer nonlinear and continuous optimization formulations for aircraft conflict avoidance via heading and speed deviations. *Eur. J. Oper. Res.*, 310(2):670–679, 2023.

6 Sonia Cafieri and Nicolas Durand. Aircraft deconfliction with speed regulation: New models from mixed-integer optimization. *J. Glob. Optim.*, 58(4):613–629, 2014.

7 Andrew R. Conn. *A gradient type method of locating constrained minima.* PhD thesis, University of Waterloo, Ontario, Canada, 1971.

8 Andrew R. Conn. Constrained Optimization Using a Nondifferentiable Penalty Function. *SIAM J. Numer. Anal.*, 10:760–784, 1973.

9 Andrew R. Conn. Penalty function methods. In Michael J. D. Powell, editor, *Nonlinear Optimization 1981*, pages 235–242. NATO Conference Series, 1982.

10 Andrew R. Conn and Yuying Li. A Structure Exploiting Algorithm for Nonlinear Minimax Problems. *SIAM J. Optim.*, 2(2):242–263, 1992.

11 Andrew R. Conn and Marcel Mongeau. Discontinuous piecewise linear optimization. *Math. Program.*, 80(3):315–380, 1998.

12 Fernando H. C. Dias, Hassan L. Hijazi, and David Rey. Disjunctive linear separation conditions and mixed-integer formulations for aircraft conflict resolution. *Eur. J. Oper. Res.*, 296(2):520–538, 2022.

13 Robert Fourer, David M. Gay, and Brian W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming.* Brooks/Cole, 2nd edition, 2002.

14 A. Bruce Gamble, Andrew R. Conn, and William R. Pulleyblank. A Network Penalty Method. *Math. Program.*, 50(1):53–73, 1991.

15 Tomasz Pietrzykowski. The potential method for conditional maxima in the locally compact metric spaces. *SIAM J. Numer. Anal.*, 14:325–329, 1970.

16 Mustafa C. Pinar and Stavros A. Zenios. On smoothing exact penalty functions for convex constrained optimization. *SIAM J. Optim.*, 4:486–511, 1994.

17 Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1):25–57, 2006.