



Spatio-temporal constrained zonotopes for validation of optimal control problems

Etienne Bertin, Bruno Hérissé, Julien Alexandre Dit Sandretto, Alexandre Chapoutot

► To cite this version:

Etienne Bertin, Bruno Hérissé, Julien Alexandre Dit Sandretto, Alexandre Chapoutot. Spatio-temporal constrained zonotopes for validation of optimal control problems. CDC 2021, Dec 2021, Austin, United States. hal-03197925v1

HAL Id: hal-03197925

<https://hal.science/hal-03197925v1>

Submitted on 14 Apr 2021 (v1), last revised 31 Mar 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Spatio-temporal constrained zonotopes for validation of optimal control problems*

Etienne Bertin^{1,2}, Bruno Hérissé¹, Julien Alexandre dit Sandretto², Alexandre Chapoutot²

Abstract—A controlled system subject to dynamics with unknown but bounded parameters is considered. The control is defined as the solution of an optimal control problem, which induces hybrid dynamics. A method to enclose all optimal trajectories of this system is proposed. Using interval and zonotope based validated simulation and Pontryagin’s Maximum Principle, a characterization of optimal trajectories, a conservative enclosure is constructed. The usual validated simulation framework is modified so that possible trajectories are enclosed with spatio-temporal zonotopes that simplify simulation through events. Then optimality conditions are propagated backward in time and added as constraints on the previously computed enclosure. The obtained constrained zonotopes form a thin enclosure of all optimal trajectories that is less susceptible to accumulation of error. This algorithm is applied on Goddard’s problem, an aerospace problem with a bang-bang control.

I. INTRODUCTION

Optimal control of aerospace systems is performed by modeling the considered system by dynamics depending on multiple parameters (for example the engine thrust parameter) and an initial state (initial velocity, mass,...). Pontryagin’s Maximum Principle (PMP) provides necessary optimality conditions for the resolution of optimal control problems by transforming an optimal control problem into a two point boundary value problem which can be solved accurately [1], [2]. However, some of the parameters and initial states may be subject to uncertainties, that is, their exact value is subject to perturbations and estimation errors. Since the solution of an optimal control problem can rarely be explicitly expressed in a closed loop form, the effect of those perturbations can be hard to predict. This is complicated further by the fact that optimal trajectories tend to have hybrid behaviors: they may be subject to discrete events that abruptly change dynamics and state. For instance, Goddard’s problem [1], which consists in launching a rocket to a given position while minimizing its fuel consumption, alternates between a full throttle mode and a free fall mode, and the duration of each phase depends on the parameters.

Usually, optimal control problems are solved using the nominal values of the parameters. Then the robustness of the solution is demonstrated by dispersing the parameters around nominal values with Monte Carlo simulations [3].

This method does not exclude the possibility of failure, which can be problematic on critical systems. In addition these problem-solving methods often introduce numerical approximations, which question the validity of the results.

Interval arithmetic has shown its ability to address many control problems [4], providing validated solutions while dealing with method uncertainties (numerical approximations) as well as with model uncertainties. These methods can also be used to design robust optimal control. For instance, [5] proposes algorithms to compute a control that steers a system to a desired state for any realization of a bounded noise while achieving the lowest upper bound on the cost. While intervals vectors, a.k.a. boxes, are a computationally cheap set representation, their rectangular shape induces a loss of precision known as the wrapping effect.

A more precise state representation known as zonotopes is derived from intervals [6]. Zonotopes have been used to over approximate the reachable set of dynamic systems [7], [8]. However zonotope are unfit for intersections and unions. As a consequence, simulation of hybrid systems often requires to convert to a more costly polytope representation during events [9], [10]. In recent years, a new state representation known as constrained zonotopes was proposed [11]. These sets benefit from the zonotope representation and constraints allow the representation of intersections. Thus, set valued predictor-corrector algorithms for discrete-time systems with uncertainties were designed using constrained zonotopes [12].

Our aim is to create similar algorithms for continuous-time controlled system with uncertainties when the control is defined as the solution of an optimization problem. Enclosures for such systems were theorized in a previous work [13] but proposed Interval based methods were limited to low dimension problems with no hybrid behaviors. This article presents a constrained zonotope based method to compute enclosures for a wider range of problems.

First the problem is presented and notions of optimal control and set membership methods are recalled. Then the control problem is transformed into an equivalent time-switched boundary value problem and a rough enclosure is built. The system is enclosed in spatio temporal zonotopes that allow the fixation of switch times that would normally vary with uncertainties. Afterward, boundary constraints are propagated backward in time and added as constraints on aforementioned zonotopes. The resulting algorithm as well as a simple initialization routine are showcased on Goddard’s problem.

*This work was partially supported by the “Chair Complex Systems Engineering – Ecole polytechnique, Thales, DGA, FX, Dassault Aviation, Naval Group Research, ENSTA Paris, Télécom Paris, and Fondation ParisTech”

¹DTIS, ONERA, Université Paris Saclay, F-91123 Palaiseau, France. firstname.lastname@onera.fr

²U2IS, ENSTA Paris, Institut Polytechnique de Paris, Palaiseau, France. firstname.lastname@ensta-paris.fr

II. PROBLEM FORMULATION

Consider the following Optimal Control Problem (OCP):

$$\begin{aligned} \min_{u(\cdot)} & \int_0^{t_f} \ell(y(t), u(t), \xi_{nom} + \delta\xi(t)) dt \\ \text{s.t.} & \begin{cases} \dot{y}(t) = f(y(t), u(t), \xi_{nom} + \delta\xi(t)), \\ y(0) = y_0, \quad y(t_f) \in \mathcal{Y}_f, \quad t_f \text{ is free.} \end{cases} \end{aligned} \quad (1)$$

Interpretation: the trajectory of a controlled system is optimized. The system is characterized by the following data:

- A state $y(t) \in \mathbb{R}^d$. The time function $y(\cdot) : [0, \Theta] \rightarrow \mathbb{R}^d$ is the trajectory.
- A control input $u(\cdot) : [0, \Theta] \rightarrow \mathcal{U}$.
- Parameters $\xi_{nom} + \delta\xi(t) \in [\xi]$. They are equal to a fixed nominal value ξ_{nom} subject to time dependent disturbances $\delta\xi(\cdot)$, e.g. estimation errors, material fatigue.
- Dynamics $f : \mathbb{R}^d \times \mathcal{U} \times [\xi] \rightarrow \mathbb{R}^d$.
- An instantaneous cost $\ell : \mathbb{R}^d \times \mathcal{U} \times [\xi] \rightarrow \mathbb{R}$.
- An initial state $y_0 \in \mathbb{R}^d$.
- Final constraints $y(t_f) \in \mathcal{Y}_f$ and eventual state constraints.

The exact value of the initial state y_0 is unknown but within an interval enclosure: $y_0 \in [y_0]$.

It is assumed that the disturbances depend on time but are bounded. It follows that there exists a box $[\xi]$ s.t. $\forall t, \xi_{nom} + \delta\xi(t) \in [\xi]$. The time dependency establishes a general framework that covers more scenarios than the approach considering that the parameters are unknown constant values, as in [3]. It is worth noting that if one expects a given parameter to be constant, it is possible to add this parameter as a part of the system's state with a null derivative. This usually yields more precise enclosures as the algorithm collects dependencies between the possible states and the parameter value. To simplify notations, $\xi(\cdot) : [0, \Theta] \rightarrow [\xi]$ will denote $\xi_{nom} + \delta\xi(\cdot)$.

III. PRELIMINARIES

A. Characterization of optimal trajectories

OCP (1) is considered. A co-state $p(\cdot) : [0, \Theta] \rightarrow \mathbb{R}^d$ (also known as adjoint vector) is introduced to characterize the optimal solution with Pontryagin's Maximum Principle (2). A reference theorem can be found in [2], [1], [14], we propose a version with no abnormal optimums in which parameters are highlighted.

a) Pontryagin's Maximum Principle (PMP): If $(y(\cdot), u(\cdot))$ are a normal optimum, then there exists a non trivial co-state $p(\cdot)$ such that $(y(\cdot), p(\cdot), u(\cdot))$ satisfy the following equations:

$$\begin{aligned} H(y, p, u, \xi) &= \ell(y, u) + p \cdot f(y, u, \xi) \\ \dot{p}(t) &= -\frac{\partial H}{\partial y}(y(t), p(t), u(t), \xi(t)) \\ \forall t \in [0, t_f], u(t) &\in \arg \min_{v \in \mathcal{U}} H(y(t), p(t), v, \xi(t)), \end{aligned} \quad (2)$$

where H is the pre-Hamiltonian. The optimal control $u(\cdot)$ is defined implicitly as minimizing the pre-Hamiltonian at every time.

Considering the full state composed of time, state and co-state, Equations (1) and (2) can be combined to obtain uncontrolled System (3). The time coordinate simplifies notations and is used by algorithms in Section IV.

$$\begin{aligned} \dot{x}(t) &= g(x(t), \xi(t)) \\ x &= \begin{pmatrix} t \\ y \\ p \end{pmatrix}, g = \begin{pmatrix} 1 \\ f(y, \arg \min H, \xi) \\ -\frac{\partial H}{\partial y}(y, p, \arg \min H, \xi) \end{pmatrix} \end{aligned} \quad (3)$$

This system is subject to a transversality condition at final time t_f depending on final constraints and final costs (see [1], [2], [14] for detail). We write this condition as a constraint $C(x(t_f)) = 0$ with function $C : \mathbb{R}^{1+2d} \rightarrow \mathbb{R}^{1+d}$.

Even when the controlled system is not hybrid, the pre-Hamiltonian minimization in System (3) creates hybrid behaviors, notably bang-bang controls. Furthermore, constrained systems often undergo a co-state translation jump when a state constraint activates [14]. Nevertheless, examination of the problem often yields the optimal sequence of modes as well as switching functions¹ $C_n : \mathbb{R}^{2d+1} \mapsto \mathbb{R}$ that cause a transition from mode n to mode $n+1$ when they reach 0. It is common practice to add transition times Θ_n as variables and $C_n(x(\Theta_n)) = 0$ as constraints to the problem as it turns the hybrid system into a simpler time switched system.

To sum up, System (3) can often be expressed as Switched System (4), which has a sequence of N modes and translation jumps.

$$\begin{cases} \dot{x}(t) = g_n(x(t), \xi(t)), & \forall t \in [\Theta_{n-1}^+, \Theta_n^-] \\ x(\Theta_n^+) = x(\Theta_n^-) + z_n, & \forall n \in 0..N-1 \\ x(0) = x_0 \end{cases} \quad (4)$$

where $0 = \Theta_0 < \Theta_1 < \dots < \Theta_N = t_f$ are transition times and $z_n \in \mathbb{R}^{1+2d}$ are jump vectors. Switching and optimality conditions are compiled as Constraints (5).

$$C_n(x(\Theta_n^-)) = 0, \forall n \in 1..N, \quad (5)$$

where $C_N(x(\Theta_N^-)) = 0$ is the previously mentioned transversality condition $C(x(t_f)) = 0$. Uncertainties nudge the state which changes the time at which constraints are satisfied. Hence Θ_n vary depending on uncertainties.

B. Flow and resolvent of an Ordinary Differential Equation (ODE)

Consider an ODE with initial time τ in the form:

$$\begin{cases} \dot{x}(t) = g(x(t), \xi(t)) \\ x(\tau) = x_\tau \end{cases}$$

A solution can be approximated by numerical methods, such as Euler's method, or more efficient Runge-Kutta methods [8]. Integrating the dynamics between two boundary times τ and T can be seen as a flow function $\Phi_{\tau, T}$. Function $\Phi_{\tau, T}$ takes an initial state, simulates it from τ to T and

¹It is common in the literature to have events triggered by the system entering guard sets. This formalism is equivalent to switching functions that reach zero on guard sets' boundaries.

returns the final state. $\Phi_{\tau,T}$ depends on parameters $\xi(\cdot)$ but since parameters are data, this argument will be omitted to simplify notations. If g is twice differentiable, the gradient of $\Phi_{\tau,T}$ exists and is the resolvent $R_{\tau,T}(x_\tau)$ of the linearized system [15].

The resolvent $R_{\tau,T}(x_\tau)$ may be computed by integrating ODE (6).

$$\begin{cases} \dot{x}(t) &= g(x(t), \xi(t)) \\ \dot{R}_{\tau,t}(x_\tau) &= \frac{\partial g}{\partial x}(x(t), \xi(t)) \cdot R_{\tau,t}(x_\tau) \\ x(\tau) &= x_\tau \\ R_{\tau,\tau}(x_\tau) &= I_n. \end{cases} \quad (6)$$

Chain Rule (7) is derived.

$$R_{\tau,T} = R_{t,T} \cdot R_{\tau,t}, \forall t, \tau, T \quad (7)$$

A global flow $\Phi_{\tau,T}(x, z)$ of Switched System (4) is constructed by composing flows in between transitions and jumps. This flow depends on jump vector $z = (z_0, \dots, z_{N-1}) \in \mathbb{R}^{N(2d+1)}$. Because jumps are translation by a vector, the gradient of the flow of Switched System (4) may also be expressed as a resolvent that satisfies Chain Rule (7).

C. Interval arithmetic

An interval $[x]$ is a convex subset of \mathbb{R} that contains all reals between a lower bound \underline{x} and an upper bound \bar{x} . With $\underline{x}, \bar{x} \in \mathbb{R} \cup \{-\infty, +\infty\}$. Interval arithmetic can be used as an alternative to floating-point arithmetic to obtain an enclosure of the solution of a problem, rather than an approximation [16].

Let a real valued function $g : x \mapsto g(x)$. An associated inclusion function $[g] : [x] \mapsto [g]([x])$ is a set valued function satisfying Inclusion (8).

$$[g]([x]) \supseteq \{g(x) | \forall x \in [x]\}. \quad (8)$$

We note $[g]$ any set-valued function that satisfies (8).

Interval vectors, or boxes, are an axis-aligned rectangular set in a finite dimensional space. They are an inexpensive representation of a high dimensional set (compared to polytopes) but their shape induces a wrapping effect during computations [4].

D. Zonotopes and Constrained Zonotopes

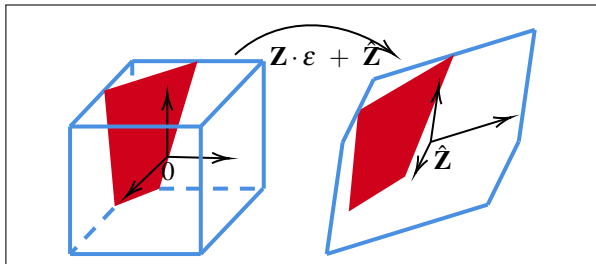


Fig. 1. Zonotope and constrained zonotope. Left: The unit box $[-1, 1]^{d_e}$ (blue) and the intersection of the unit box with constrained set $A \cdot \varepsilon + \hat{A} = 0$ (red). Right: The d dimensional deformation of these sets by $Z \cdot \varepsilon + \hat{Z}$ with $d = 2$. Zonotope (blue) and constrained zonotope (red).

Zonotopes and constrained zonotopes are subsets of \mathbb{R}^d . A zonotope $\mathbb{Z} = [Z, \hat{Z}]$ is the affine deformation of a unit box defined by $\{Z \cdot \varepsilon + \hat{Z} : \varepsilon \in [-1, 1]^{d_e}\}$, where Z is the generator sparse matrix, of dimension $d \times d_e$ and \hat{Z} is the center vector of dimension $d \times 1$. A zonotope is represented in blue right of Figure 1. Each coordinate of ε is called a noise symbol. New noise symbols are added during each computation to enclose errors and nonlinear phenomena, hence d_e is greater than d and increases with computations. Operations on zonotope, known as affine arithmetic, satisfies principle (8). Standard operations such as sum (precisely Minkowski sum) and multiplication may be found in [6].

A constrained zonotope $\mathbb{Z}^A = [Z, \hat{Z}, A, \hat{A}]$ is the affine deformation of a unit box intersected with a plane $\{Z \cdot \varepsilon + \hat{Z} : \varepsilon \in [-1, 1]^{d_e}, A \cdot \varepsilon + \hat{A} = 0\}$, where A the constraint matrix and \hat{A} the constraint vector. See red in Figure 1.

A constrained zonotopes may be *lifted*, that is represented as a regular zonotope of higher state dimension d :

$$\mathbb{Z}^A = [Z, \hat{Z}, A, \hat{A}] \rightarrow \tilde{\mathbb{Z}} = \left[\begin{pmatrix} Z \\ A \end{pmatrix}, \begin{pmatrix} \hat{Z} \\ \hat{A} \end{pmatrix} \right]$$

$\tilde{\mathbb{Z}}$ may be manipulated by zonotope-based algorithms then converted back into a constrained zonotope [11]. Hence the constraint part of \mathbb{Z}^A may be treated as a constraint zonotope $\mathbb{A} = [A, \hat{A}]$.

E. Validated simulation

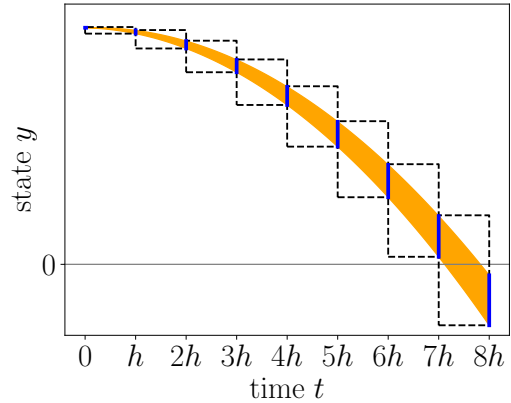


Fig. 2. Possible trajectories of a falling ball with unknown initial velocity (orange) enclosed with validated simulation library Dynlbex [8]. The propagation of the uncertainty on the initial velocity creates a tube of possible trajectories. Validated simulation encloses trajectories in zonotopes (blue, plain) at instants t_j , and by Picard boxes (black, dashed) on time ranges $[t_j, t_{j+1}]$. Here, time step is constant: $t_j = jh$.

An uncontrolled set membership ODE is considered:

$$\begin{cases} \dot{x}(t) &\in [g](x, [\xi]) \\ x(\tau) &\in \mathbb{X}, \end{cases}$$

Validated simulation encloses every solution of this system in a sequence of boxes and zonotopes.

To that end, the time range $[\tau, T]$ is discretized in $(t_j)_{j=0..J}$, $t_0 = \tau, t_J = T$. Starting with an enclosure \mathbb{X}_j of the systems at time t_j , a Picard box $[\tilde{x}]$ that encloses the system on time

range $[t_j, t_{j+1}]$ is built. Then a zonotope enclosure \mathbb{X}_{j+1} of $x(t_{j+1})$ is computed. We use the validated simulation library Dynlbex [8], which utilizes validated Runge-Kutta methods coupled with inflating terms that enclose all sources of error. The output of validated simulation is shown in Figure 2.

Validated simulation is similar to simulating multiple systems on a common duration. Hence variable time horizons like Θ_n in System (4) that differ from one system to another are challenging.

IV. ENCLOSURES OF ALL OPTIMAL TRAJECTORIES

In this section, an algorithm that encloses optimal trajectories is constructed. First variable mode durations are synchronized using spatio temporal zonotopes. Then optimality conditions are added to these zonotopes as constraints. Lastly, a simple method to initialize the algorithm is proposed.

A. Event time fixation with Spatio-Temporal Zonotopes

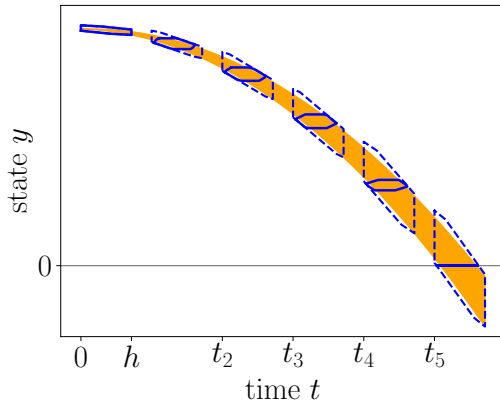


Fig. 3. Enclosure of a falling ball. Spatio temporal zonotopes (dashed) enclose the system on a time ranges $[t_j, t_j + h]$ with h such that the last zonotope encloses $[\underline{\Theta}, \bar{\Theta}]$ the entire duration during which the system intersects with the set where $C(x) = 0$ (black line). Constrained zonotopes (plain) are obtained by propagating constraint $C(x(\Theta)) = 0$ backward, they enclose the system at time $\Theta - \tau_j$ with $\tau_j = t_5 - t_j$.

Optimal mode durations $\theta_n = \Theta_n - \Theta_{n-1}$ of System (4) depends on the uncertainties. It follows that there is a set $[\theta_n, \bar{\theta}_n]$ of possible mode durations.

As stated in Sections III-E, validated simulation between two varying time stamps is difficult. To bypass this difficulty, the system is transformed into an equivalent system with fixed mode durations equal to $\underline{\theta}_n$ through the addition of jumps. The idea is to add width to zonotopes along the time axis so that a single zonotope may enclose the entire duration $[\Theta_n, \bar{\Theta}_n]$.

First, bounds on the mode durations are identified. These bounds may be computed with validated simulation. For instance, in Figure 2, the enclosure of the height at time $7h$ is entirely above 0 while it is entirely below zero at time $8h$. Suppose an event is triggered by reaching 0, then $\underline{\theta}_1 = 7h$ and $\bar{\theta}_1 = 8h$. Alternatively these bounds may be computed with an inflate and contract method as proposed in Section IV-D.

Then, spatio-temporal zonotopes are introduced. Unlike zonotopes in Figure 2 which enclose the state at instants,

spatio temporal zonotopes enclose the system on a time range (see Figure 3). To construct those zonotopes, we proceed as follows. Let \mathbb{X} a zonotope enclosure of $x(\Theta_n^-)$. \mathbb{X} is either the initial state or the output of previous simulations. System (4) induces that $x(\Theta_n^+) \in \mathbb{X} + z_n$. Then a validated Taylor interpolation [16] may be used. For instance, an order zero interpolation yields Enclosure (9).

$$\forall t \in [\Theta_n^+, \Theta_n + h], x(t) \in \mathbb{X} + z_n + t \times [g_{n+1}]([\tilde{x}]), \quad (9)$$

where $[\tilde{x}]$ is a Picard box computed with validated simulation that encloses time, state and co-states on time range $[\Theta_{n-1}^+, \Theta_{n-1} + h]$.

Take $h = \theta_{n+1} - \theta_n$. By replacing the time variable in Equation (9) by a time zonotope \mathbb{T} , Formula (10) is obtained.

$$\forall t \in [\Theta_n, \Theta_n + h], x(t) \in \mathbb{X} + z_n + \mathbb{T} \times [g_{n+1}]([\tilde{x}]) \in \mathbb{X} + \mathbb{Z}_n \quad (10)$$

where $\mathbb{T} = \frac{h}{2} + \frac{h}{2}\epsilon^n$ is a zonotope that encloses time range $[0, h]$ and \mathbb{Z}_n is a zonotope enclosure of $z_n + \mathbb{T}_n \times [g]([\tilde{x}])$. z_n may be enclosed like the initial co-state (see Section IV-D).

$\mathbb{X} + \mathbb{Z}_n$ is a zonotope enclosure of the state over the time range $[\Theta_n^+, \Theta_n + \theta_{n+1} - \theta_n]$. Hence simulating $\mathbb{X} + \mathbb{Z}_n$ using dynamics g_{n+1} with fixed time horizon θ_{n+1} yields an enclosure of the system on time range $[\Theta_n + \theta_{n+1}, \Theta_n + \theta_{n+1}] = [\Theta_{n+1}^-]$. Hence it encloses the system during the transition from mode n to mode $n+1$.

By applying this procedure at each transition, System (11) is obtained.

$$\begin{cases} \dot{x}(t) \in [g_n](x(t), [\xi]), & \forall t \in [\underline{\Theta}_{n-1}^+, \underline{\Theta}_n^-] \\ x(\underline{\Theta}_n^+) \in x(\underline{\Theta}_n^-) + \mathbb{Z}_n, & \forall n \in 0..N-1 \\ x(0) \in \mathbb{X}_0, \end{cases} \quad (11)$$

where $\underline{\Theta}_n = \sum_{k=1}^n \underline{\theta}_k$. System (11) encloses the trajectories of System (4) and has the same structure, but all mode durations are fixed, hence it is easier to use validated simulation. The jump that System (11) undergoes at each event corresponds to simulating System (4) over a period $h \in [0, \bar{\theta}_{n+1} - \underline{\theta}_{n+1}]$ (see Figure 3). It follows that for all n and for all $\tau \in [0, \underline{\theta}_n]$, an enclosure of System (11) at time $\underline{\Theta}_n - \tau$ encloses System (4) on time range $[\underline{\Theta}_n - \tau, \underline{\Theta}_n - \tau]$. In particular it encloses System (11) at time $\underline{\Theta}_n - \tau$, which may be further characterized by optimality conditions.

B. Propagate optimality conditions

Using backward propagation of boundary constraints, we define a constraint that characterizes optimal solutions.

Proposition 1: Consider $n \in 1..N$, $\tau \in [0, \underline{\theta}_n]$ and two zonotopes \mathbb{X}, \mathbb{Z} .

Solutions $(x, z) \in \mathbb{X} \times \mathbb{Z}$ of boundary value problem with System (11) and Constraints (5) at time $\underline{\Theta}_n - \tau$ satisfy the following set membership constraints for all $m \geq n$:

$$0 \in [C_m]([\Phi]_{\underline{\Theta}_n - \tau, \underline{\Theta}_m}(\hat{\mathbf{X}}, \hat{\mathbf{Z}})) + [\nabla C_m]([\Phi]_{\underline{\Theta}_n - \tau, \underline{\Theta}_m}(\mathbb{X}, \mathbb{Z})) \cdot \mathbb{W}_m,$$

where \mathbb{W}_m are defined sequentially by:

$$\begin{aligned} \mathbb{W}_n &= [R]_{\underline{\Theta}_n - \tau, \underline{\Theta}_n} \cdot (x - \hat{\mathbf{X}}) \\ \mathbb{W}_m &= [R]_{\underline{\Theta}_{m-1}, \underline{\Theta}_m} \cdot (\mathbb{W}_{m-1} + z - \hat{\mathbf{Z}}), \forall m \in n+1..N. \end{aligned} \quad (12)$$

Proof: Using the flow of System (11), boundary Constraints (5) are propagated backward in time, which yields Constraints (13).

$$C_m(\Phi_{\Theta_n - \tau, \Theta_m}(x, z)) = 0, \forall m \in n..N, \quad (13)$$

Because $(x, z) \in \mathbb{X} \times \mathbb{Z}$, the following first order centered validated enclosure² holds.

$$C_m(\Phi_{\Theta_n - \tau, \Theta_m}(x, z)) \in [C_m](\Phi_{\Theta_n - \tau, \Theta_m}(\hat{\mathbf{X}}, \hat{\mathbf{Z}})) + [\nabla C_m] \cdot [\nabla \Phi_{\Theta_n - \tau, \Theta_m}] \cdot (x - \hat{\mathbf{X}}, z - \hat{\mathbf{Z}})$$

Let $\mathbb{W}_m = [\nabla \Phi_{\Theta_n - \tau, \Theta_m}] \cdot (x - \hat{\mathbf{X}}, z - \hat{\mathbf{Z}})$. Because gradients of the global flow are resolvents, \mathbb{W}_m may be expressed as follows:

$$\mathbb{W}_m = [R]_{\Theta_n - \tau, \Theta_m} \cdot (x - \hat{\mathbf{X}}) + \sum_{k=n}^{m-1} [R]_{\Theta_k, \Theta_m} \cdot (z_k - \hat{\mathbf{Z}}_k)$$

Applying Chain Rule (7) yields Equations (12). ■

Propagation of constraints is showcased in Figure 3.

Flows and resolvents may be enclosed with finite time validated simulation as durations are fixed equal to τ or θ_n .

C. Method: Enclosing optimal solutions

Algorithm 1 uses previously presented techniques to compute enclosures $\mathbb{X}_j^\mathbb{A}$ at instants $\Theta_n - \tau_j^n$. $(\tau_j^n)_{j=0..J}$ and number J are chosen by the user. The n exponent specifies the phase to which a given τ_j^n belongs. $\mathbb{X}_j^\mathbb{A}$ are self contained as they have a time coordinate, hence they do not need this specification. Lines 6 to 16 compute the constraint created in Section IV-B.

Having a single constraint zonotope \mathbb{A} for all zonotopes \mathbb{X}_j yields more precise enclosures. Indeed, an optimal trajectory satisfies optimality conditions at all times. It follows that the optimal state can be characterized by both applying optimality conditions at its current time and applying optimality conditions at another time and propagating them to the current time. Having a single set of constraints does the later, as constraints computed on \mathbb{X}_j are applied to all other enclosures of the system.

Dynlbex allows us to keep zonotopes \mathbb{A} , \mathbb{X}_j and \mathbb{Z} separate in Algorithm 1. Indeed, noise symbols in Dynlbex are global and not confined to a single zonotope, hence correlations between zonotopes are conserved. When using a zonotope library that does not have global noise symbols, zonotopes \mathbb{X}_j , \mathbb{A} and \mathbb{Z} need to be concatenated in a single zonotope in a manner similar to the lift in Section III-D.

D. Simple inflate and contract algorithm to enclose variables

Algorithm 1 requires an enclosure \mathbb{P}_0 of all possible initial co-states. When no enclosure is known, a box enclosures $[p_0]$ may be computed using contractor theory [4]. Indeed, let $[K]$ the function that takes a box $[p_0]$, transforms it into a zonotope \mathbb{P}_0 , inputs it in Algorithm 1 and returns the bounding box of $\mathbb{P}_0^\mathbb{A}$, where $\mathbb{P}_0^\mathbb{A}$ are the co-state coordinates of the initial enclosure $\mathbb{X}_0^\mathbb{A}$.

Since Algorithm 1 is such that $\mathbb{P}_0^\mathbb{A}$ encloses all optimal co-states in $[p_0]$, we have the following properties

²similar to first order Interval Newton method [16]

Algorithm 1: Enclose all optimal trajectories

Result: Constrained zonotopes $\mathbb{X}_j^\mathbb{A}$ that enclose optimal trajectories

Data: Enclosures of event durations $[\theta_n, \bar{\theta}_n]_{n \in 1..N}$, enclosure of the initial state \mathbb{Y}_0 and initial co-state \mathbb{P}_0 , time stamps τ_j^n such that $\tau_j^n < \theta_n$

```

1  $\mathbb{X}_0 \leftarrow (0, \mathbb{Y}_0, \mathbb{P}_0)$ ;
2 Apply Section IV-A to compute jump vectors
   $\mathbb{Z} = (\mathbb{Z}_0, \dots, \mathbb{Z}_{N-1})$ ;
3  $\mathbb{A} \leftarrow$  zonotope with 0 lines;
4 foreach  $\tau_j^n$  do
5    $\mathbb{X}_j \leftarrow$  validated simulation of System (11) from
    time 0 to time  $\Theta_n - \tau_j^n$  with initial state  $\mathbb{X}_0$ 
    (adding jumps  $\mathbb{Z}_k$  at intermediate events);
6    $[R]_{\Theta_n - \tau, \Theta_n} \leftarrow$  validated simulation of System (6)
    from time  $\Theta_n - \tau$  to time  $\Theta_n$ ;
7    $\mathbb{W}_n \leftarrow [R]_{\Theta_n - \tau, \Theta_n} \cdot (\mathbb{X}_j - \hat{\mathbf{X}}_j)$ ;
8   foreach  $m \in n+1..N$  do
9      $[R]_{\Theta_{m-1}, \Theta_m} \leftarrow$  validated simulation of
      System (6) from time  $\Theta_{m-1}$  to time  $\Theta_m$ ;
10     $\mathbb{W}_m \leftarrow [R]_{\Theta_{m-1}, \Theta_m} \cdot (\mathbb{W}_{m-1} + \mathbb{Z} - \hat{\mathbf{Z}})$ ;
11  end
12  foreach  $m \in n..N$  do
13     $[\Phi]_{\Theta_n - \tau_j^n, \Theta_m}(\hat{\mathbf{X}}_j, \hat{\mathbf{Z}}) \leftarrow$  validated simulation of
      System (11) from time  $\Theta_n - \tau_j^n$  to time  $\Theta_m$ 
      with initial state  $\hat{\mathbf{X}}_j$ 
      (adding jumps  $\hat{\mathbf{Z}}_k$  at intermediate events);
14     $\mathbb{A}_{m,j} \leftarrow$ 
       $[C_m](\Phi_{\Theta_n - \tau_j^n, \Theta_m}(\hat{\mathbf{X}}_j, \hat{\mathbf{Z}})) + [\nabla C_m] \cdot \mathbb{W}_m$  ;
15    Add line  $\mathbb{A}_{m,j}$  to  $\mathbb{A}$ ;
16  end
17 end
18 end

```

- $[K]([p_0]) \subset [p_0]$,
- $[K]([p_0])$ contains all optimal co-states in $[p_0]$,
- If $[K]([p_0]) \subset \text{Int}([p_0])$, the interior of $[p_0]$, then $[K]([p_0])$ contains all optimal co-states, or at least all connected local optimums.

A classic inflate and contract algorithm is used to enclose initial co-states: i) take a box $[p_0]$ computed by solving the optimal control problem for some parameter values, ii) inflate $[p_0]$ until $[K]([p_0]) \subset \text{Int}([p_0])$, iii) contract it with fixed point iteration $[p_0] \leftarrow [K]([p_0])$.

$[K]$ may be extended to enclose jump vectors z_n and mode duration θ_n as well.

V. APPLICATION: GODDARD'S PROBLEM

A vertical take-off problem is considered:

$$\begin{aligned}
& \min_{u(\cdot)} \int_0^{t_f} |u| dt \\
& \text{s.t. } \begin{cases} \dot{r}(t) = v(t), & r(0) = r_0, & r(t_f) = r_f, \\ \dot{v}(t) = -\frac{G}{r^2} + \frac{C}{m} u, & v(0) = v_0, & v(t_f) \text{ free}, \\ \dot{m}(t) = -b|u(t)|, & m(0) = m_0, & m(t_f) \text{ free}. \end{cases}
\end{aligned} \quad (14)$$

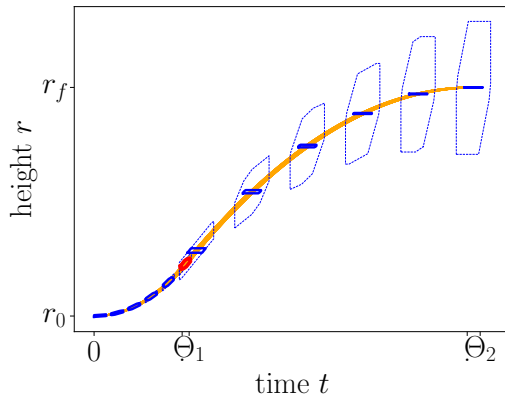


Fig. 4. Results on Goddard's Problem (14). Orange: simulations with random parameters. Θ_1 has two graduations that indicate bounds $\underline{\Theta}_1$ and $\bar{\Theta}_1$, likewise for $t_f = \Theta_2$. The generated constrained zonotopes (blue, plain) are a much more precise enclosure of optimal trajectories than the unconstrained hull (blue, dashed) which accumulates error. The red zonotope encloses the system during the transition from full throttle to free fall.

It is the one-dimensional version of Goddard's Problem as formulated in [1], without the drag force. Application of the PMP yields:

$$\begin{aligned} H(y, p, u, \xi) &= p_r v + p_v \left(-\frac{G}{r^2} + \frac{C}{m} u \right) - p_m b |u| \\ \dot{p}_r(t) &= -2p_v(t) \frac{G}{r^3} \\ \dot{p}_v(t) &= -p_r(t) \\ \dot{p}_m(t) &= p_v(t) \frac{C}{m^2} \\ u &= \begin{cases} \frac{p_v}{|p_v|} & \text{if } C|p_v| - (1 + bp_m)m > 0 \\ 0 & \text{if } C|p_v| - (1 + bp_m)m < 0 \end{cases} \end{aligned} \quad (15)$$

The expression of u leads to a hybrid automaton with a free fall mode when $C|p_v| - (1 + bp_m)m > 0$, a full throttle mode when $C|p_v| - (1 + bp_m)m < 0$ and a singular mode if $C|p_v| - (1 + bp_m)m = 0$ over a period of time. It follows that $C|p_v| - (1 + bp_m)m = 0$ at each mode transition. Further analysis of the problem shows that the optimal solution of the take off problem with no drag force has $N = 2$ phases, starting by full throttle and ending in free fall. This coupled with transversality condition leads to constraints functions (16).

$$\begin{aligned} C_1(x) &= C|p_v| - (1 + bp_m)m \\ C_2(x) &= \begin{pmatrix} r - r_f \\ p_v \\ p_m \\ v \end{pmatrix} \end{aligned} \quad (16)$$

We took an exact initial state and uncertain parameters: $r_0 = 1$, $r_f = 1.01$, $v_0 = 0$, $m_0 = 1$ and $C \in [3.4, 3.6]$, $G \in [0.99, 1.01]$ and $b \in [6.8, 7.2]$. We computed 100 optimal trajectories with parameters taken at random in their respective intervals. We then computed an enclosure of initial co-states using the method developed in Section IV-D and applied Algorithm 1. Results are displayed on Figure 4. Optimal trajectories are enclosed as expected and optimality constraints compensate for the accumulation of errors caused by uncertainties.

VI. CONCLUSIONS

We proposed a method to enclose optimal trajectories. The control problem was converted into a switched system with constraints. This system was enclosed with spatio-temporal zonotopes in a manner that fixed switch times. This simplified the use of validated simulation, which in turn enabled the propagation of constraints backward in time, thus refining the enclosure. Lastly, a simple method was proposed to initialize the Algorithm, making the method self-contained. This method successfully enclosed optimal trajectories of Goddard's problem. The presented algorithm uses constrained zonotopes which are a cheap state representation that could be used in high dimension problems.

In future works, the addition of optimality constraints will be embedded in the computation of zonotopes, rather than be a separate step. This should reduce the accumulation of errors and lead to tighter enclosures.

REFERENCES

- [1] F. Bonnans, P. Martinon, and E. Trélat, "Singular arcs in the generalized goddard's problem," *Journal of Optimization Theory and Applications*, vol. 139, no. 2, pp. 439–461, 2008.
- [2] E. Trélat, "Optimal Control and Applications to Aerospace: Some Results and Challenges," *JOTA*, vol. 154, no. 3, 2012.
- [3] E. Brendel, B. Hérisse, and E. Bourgeois, "Optimal guidance for Toss Back concepts of Reusable Launch Vehicles," in *EUCASS*, 2019.
- [4] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis: With Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer London, 2001.
- [5] A. Rauh and E. P. Hofer, "Interval methods for optimal control," in *Variational Analysis and Aerospace Engineering* (G. Buttazzo and A. Frediani, eds.), ch. 22, pp. 397–418, New York, NY: Springer New York, 2009.
- [6] L. H. de Figueiredo and J. Stolfi, *Self-Validated Numerical Methods and Applications*. Brazilian Mathematics Colloquium monographs, IMPA/CNPq, Rio de Janeiro, Brazil, 1997.
- [7] M. Althoff, *Reachability Analysis and its Application to the Safety Assessment of Autonomous Cars*. PhD thesis, Technische Universität München, 2010.
- [8] J. Alexandre dit Sandretto and A. Chapoutot, "Validated explicit and implicit Runge–Kutta methods," *Reliable Computing*, vol. 22, pp. 79–103, Jul 2016.
- [9] M. Althoff, O. Stursberg, and M. Buss, "Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 233 – 249, 2010. IFAC World Congress 2008.
- [10] J. Alexandre dit Sandretto and J. Wan, "Reachability analysis of nonlinear odes using polytopic based validated runge-kutta," in *Reachability Problems* (I. Potapov and P.-A. Reynier, eds.), (Cham), pp. 1–14, Springer International Publishing, 2018.
- [11] J. K. Scott, D. M. Raimondo, G. R. Marzaglia, and R. D. Braatz, "Constrained zonotopes: A new tool for set-based estimation and fault detection," *Automatica*, vol. 69, pp. 126 – 136, 2016.
- [12] B. S. Rego, G. V. Raffo, J. K. Scott, and D. M. Raimondo, "Guaranteed methods based on constrained zonotopes for set-valued state estimation of nonlinear discrete-time systems," *Automatica*, vol. 111, p. 108614, 2020.
- [13] E. Bertin, E. Brendel, B. Hérisse, J. Alexandre dit Sandretto, and A. Chapoutot, "Prospects on solving an optimal control problem with bounded uncertainties on parameters using interval arithmetics," *Acta Cybernetica*, Feb. 2021.
- [14] R. F. Hartl, S. P. Sethi, and R. G. Vickson, "A survey of the maximum principles for optimal control problems with state constraints," *SIAM review*, vol. 37, no. 2, pp. 181–218, 1995.
- [15] J. Coron and American Mathematical Society, *Control and Nonlinearity*. Mathematical surveys and monographs, American Mathematical Society, 2007.
- [16] R. E. Moore, *Interval analysis*, vol. 4. Prentice-Hall Englewood Cliffs, 1966.