A Sum-of-Squares-Based Procedure to Approximate the Pontryagin Difference of Semi-algebraic Sets

Andres Cotorruelo a,*, Ilya Kolmanovsky b, Emanuele Garone a

a Service d’Automatique et d’Analyse des Systèmes (SAAS), Université Libre de Bruxelles (ULB), Brussels, Belgium
b Department of Aerospace Engineering, University of Michigan, MI, U.S.A.

Abstract

The P-difference between two sets $A$ and $B$ is the set of all points, $C$, such that the addition of $B$ to any of the points in $C$ is contained in $A$. Such a set difference plays an important role in robust model predictive control and in set-theoretic control. In this paper we demonstrate that an inner approximation of the P-difference between two semi-algebraic sets can be computed using the Sums of Squares Programming, and we illustrate the procedure using several computational examples.

Key words: Pontryagin difference, Sum of Squares, Robust control

1 Introduction

The Pontryagin set difference (or simply P-difference), so named after Pontryagin who used it in the setting of game theory [1], has become an indispensable part of robust model predictive control (MPC) [2,3] and of set theoretic control [4,5]. Additionally, the P-difference has also been used in image processing applications [6] and in path planning [7]. Depending on the authors, the P-difference is sometimes referred to in the literature as a Minkowski set difference or as a set erosion. Efficient procedures to compute the P-difference or its approximations, especially in the case of non-polyhedral sets, can greatly expand the range of applications of robust MPC.

In this paper, we demonstrate that an inner approximation of the P-difference between two semi-algebraic sets can be computed using Sum of Squares Programming (SOSP). Computational examples are reported to illustrate the proposed approach.

1 Special thanks to Philip Pugeau for providing some of the case studies used in this paper. The second author acknowledges the support of the National Science Foundation grant 1931738.

Notation

The ring of polynomials in the variables $x_1, \ldots, x_n$ and with coefficients in the field $\mathbb{R}$ is denoted by $\mathbb{R}[x_1, \ldots, x_n]$. The set of all Sum of Squares polynomials in the variables $x_1, \ldots, x_n$ is denoted by $\Sigma[x_1, \ldots, x_n]$.

For a number of elements $a_1, \ldots, a_n$, $\{a_i\}_{i=1}^n$ denotes the set $\{a_1, \ldots, a_n\}$, and every operator applied to it is meant to be understood element-wise, e.g. $\{a_i\}_{i=1}^n \geq 0$ means $a_i \geq 0$, $\forall i = 1, \ldots, n$.

2 Problem statement

For two sets $A \subset \mathbb{R}^n$ and $B \subset \mathbb{R}^n$, the P-difference is defined as

$$A \ominus B = \{x \in A : x + z \in A \ \forall z \in B\},$$

where typically $0 \in B$. A simple geometrical way to interpret this operation is that the set $C = A \ominus B$ is a set so that if we select a point in $C$ and we add an uncertainty bounded by $B$, the resulting point still belongs to $A$ (see Fig. 1).

Although some algorithms able to perform this operation in the case of polyhedral $A$ and convex $B$ [8,9] exist in the literature, to the best of the authors’ knowledge there exists no systematic way to compute the Pontryagin difference for wider classes of sets. The objective of this paper is to solve the following problem...
Fig. 1. Visualization of the Pontryagin difference between sets $A$, the blue star, and $B$, the dashed-line ball. As it can be seen, $x_1$ will belong to $A \ominus B$ since every vector $z \in B$ is such that $x_1 + z \in A$; on the contrary, $x_2$ does not belong to $A \ominus B$ since there exist some $z' \in B$ such that $x_2 + z' \notin A$.

**Problem 2.1 (Pontryagin difference)** Let $A$ and $B$ be two semi-algebraic sets in the form

$$
A = \{ x : a_i(x) \geq 0, \ i = 1, \ldots, n \}, \\
B = \{ x : b_j(x) \geq 0, \ j = 1, \ldots, m \},
$$

where $a_i(x), b_j(x) \in \mathbb{R}[x]$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. Determine a systematic procedure to compute an inner approximation of $A \ominus B$.

**3 Computation of the Pontryagin Difference using Sum of Squares**

In this section we propose a way to solve Problem 2.1 for the case where the sets $A$ and $B$ are described as the intersection of polynomial inequalities. The proposed solution makes use of the Krivine – Stengle Positivstellensatz (P-satz) [10].

To simplify the problem, the first step is to note that the set $A$ can be represented as

$$
A = \bigcap_{i=1}^{n} A_i
$$

where $A_i = \{ x : a_i(x) \geq 0 \}, \ i = 1, \ldots, n$. Since

$$
A \ominus B = \bigcap_{i=1}^{n} A_i \ominus B,
$$

we can focus on a single set $A_i$ at a time without any loss of generality. Consider the P-difference $A_i \ominus B$

$$
A_i \ominus B = \{ x : a_i(x + z) \geq 0, \ z \in B \}.
$$

A possible way to approximate $A_i \ominus B$ is by means of a set $\mathcal{C}_i = \{ x : c_i(x) \geq 0 \} \subseteq A_i \ominus B$, where the function $c_i$ must be such that

$$
c_i(x) \geq \min_{z \in B} a_i(x + z). \tag{1}
$$

Note that whenever (1) is an equality, $\mathcal{C}_i = A_i \ominus B$.

Condition (1) is equivalent to the following set emptiness condition

$$
\{ x, z : c_i(x) - a_i(x + z) < 0, \ z \in B \} = \emptyset. \tag{2}
$$

Since in the Krivine – Stengle P-satz, the set required to be empty is described in terms of equal-to, greater-than-or-equal-to, and not-equal-to operators, the set (2) is rewritten in terms of these operators as

$$
\{ x, z : c_i(x) - a_i(x + z) \geq 0, \\
c_i(x) - a_i(x + z) \neq 0, \ (b_j(z))_{j=1}^{m} \geq 0 \} = \emptyset. \tag{3}
$$

At this point the Krivine – Stengle P-satz states that (3) is satisfied if and only if there exist two polynomials $p(x, z)$ and $q(x, z)$ such that

$$
p(x, z) + q^2(x, z) = 0,
$$

where

$$
p \in \text{Cone} \left( \{ c_i(x) - a_i(x + z), b_1(z), \ldots, b_m(z) \} \right), \\
q \in \text{Monoid} \left( c_i(x) - a_i(x + z) \right).
$$

Performing standard algebraic manipulations this allows to obtain the sufficient condition

$$
(c_i(x) - a_i(x + z))^2 + s_0(x, z) (c_i(x) - a_i(x + z)) + \sum_{j=1}^{m} s_j(x, z) b_j(z) (c_i(x) - a_i(x + z)) = 0
$$

where $s_i(x, z) \in \Sigma[x, z], \ i = 0, \ldots, m$. This equation can be further simplified by cancelling $c_i(x) - a_i(x + z)$

$$
c_i(x) - a_i(x + z) + s_0(x, z) + \sum_{j=1}^{m} s_j(x, z) b_j(z) = 0.
$$

Since $s_0 \in \Sigma[x, z]$, it follows that the latter is equivalent to

$$
P_i(x, z) = a_i(x + z) - c_i(x) - \sum_{j=1}^{m} s_j(x, z) b_j(z) \in \Sigma[x, z]. \tag{4}
$$
Finally, since we are interested in the largest inner approximation of \( \mathcal{A} \ominus \mathcal{B} \), using (4) we can define the problem of finding \( c_i(x) \) as the following Sum of Squares Programming (SOSP) problem

\[
\begin{align*}
\max & \quad \int_{\mathcal{R}} c_i(x) \, dx \\
\text{s.t.} & \quad P_i(x, z) \in \Sigma[x, z] \\
& \quad \{s_j(x, z)\}^m_{j=1} \in \Sigma[x, z],
\end{align*}
\]

where \( \mathcal{R} \) is a normal domain \([11]\) that contains \( \mathcal{A} \). As is well known \([12]\), optimization problem (5) can in turn be cast into a Semi-Definite Programming (SDP) optimization problem that can be solved efficiently using existing SDP solvers.

**Remark 3.1** Note that whenever \( \mathcal{R} \) is a normal domain described by polynomials, \( \int_{\mathcal{R}} c_i(x) \, dx \) can be computed in closed form and is polynomial \([13]\), which implies that the objective function of (5) is linear in the coefficients of \( c_i(x) \). If one prefers to not use a normal set \( \mathcal{R} \), a practical approach is to randomly select a (possibly large) number of points \( r_1, \ldots, r_N \in \mathcal{R} \), and use the objective function

\[
\frac{1}{N} \sum_{j=1}^{N} c_i(r_j). \tag{6}
\]

Note that for a sufficiently large \( N \), optimizing over (6) is equivalent to optimizing over \( \int_{\mathcal{R}} c_i(x) \, dx \).

### 4 Examples

In this section we apply the proposed methodology to a number of 2 and 3-dimensional sets to illustrate its effectiveness. All of the showcased examples depict \( \mathcal{C} \approx \mathcal{A} \ominus \mathcal{B} \), where \( \mathcal{A} = \{x : a(x) \geq 0\} \) and \( \mathcal{B} = \{x : b(x) \geq 0\} \) with varying \( a(x) \) and \( b(x) \) depending on the example. Table 1 reports the expressions of \( a(x) \) and \( b(x) \) as well as the chosen degrees of \( c(x) \) and the \( s_i(x, z) \), and the elapsed time to compute the approximation. Lastly, Figs. 2–6 depict \( \mathcal{A} \) as a solid blue set, \( \mathcal{B} \) as a solid green set, and \( \mathcal{C} \) as a solid orange set. For space reasons, the expressions of \( c(x) \) have been omitted in this paper, but they can be found in the addendum [http://www.gprix.it/SoSPontryagin.pdf](http://www.gprix.it/SoSPontryagin.pdf). All optimization problems were solved using MATLAB R2019b and YALMIP \([14]\), running on an Intel Core i7-7500 at 2.7 GHz with 16 GB of RAM.

### 5 Concluding remarks

In this paper we proposed a systematic approach based on SOSP for the computation of an inner approximation of the Pontryagin difference between two semi-algebraic sets. We subsequently showcased the capabilities of this methodology by applying it to several different examples in two and three dimensions. Possible applications for this methodology include the analytical determination of an inner approximation of constrained sets in robust...
Table 1
Polynomials used in the Figures.

| Fig. | \( a(x) \) | \( b(x) \) | \( \partial c(x) \) | \( \partial s_j(x, z) \) | \( t \) |
|------|-------------|-------------|-----------------|------------------|-----|
| 2    | \( 0.1 - x_1^4 - x_2^4 + 10x_1^2 - x_2^2 \) | \( 1 - x_1^2 - x_2^2 \) | 14              | 6               | 11.93 s |
| 3    | \( x_1^2 - (x_1 - 0.5)^3 - (x_1 - 0.5)^4 \) | \( 0.1 - 2x_1^4 - 16x_2^2 \) | 10              | 6               | 2.63 s  |
| 4    | \( 4 - x_1^2 - x_2^2 \) | \( 0.1 - 25x_1^2x_2^2 \) | 10              | 2               | 1.02 s  |
| 5    | \(- (x_1^2 + x_2^2 + x_3)^3 + 3(x_1^2 + x_2^2 + x_3)^2 \) | \(-9(x_1^2 + x_2^2 + 3) + 16(x_1^3 - 3x_1x_2^2 + 2x_3^2) \) | 10              | 4               | 52.11 s |
| 6    | \( 1 - x_1^2 - x_2^5 - x_3^6 + 5x_1^2x_2x_3 - 3x_1^2x_2^2 \) | \(-10x_1^2x_2^3x_3 - 3x_1^2x_2^4 + x_2^5x_3 \) | \( 10^{-4} - x_1^6 - x_2^6 - x_3^6 \) | 10 | 4 | 29 min |

Fig. 5. Result of subtracting an ellipsoid from the 3-dimensional 2-torus.

Fig. 6. Result of subtracting the 6-norm ball from the rotated 5-pointed star algebraic cylinder.

control.

References

[1] L. S. Pontryagin, Linear differential games. i, ii, in: Doklady Akademii Nauk, Vol. 175, Russian Academy of Sciences, 1967, pp. 764–766.
[2] B. Kouvaritakis, M. Cannon, Model predictive control, Switzerland: Springer International Publishing (2016).
[3] J. B. Rawlings, D. Q. Mayne, M. Diehl, Model predictive control: theory, computation, and design, Vol. 2, Nob Hill Publishing Madison, WI, 2017.
[4] F. Blanchini, S. Miani, Set-theoretic methods in control, Springer, 2008.
[5] I. Kolmanovsky, E. G. Gilbert, Theory and computation of disturbance invariant sets for discrete-time linear systems, Mathematical problems in engineering 4 (1998).
[6] H. J. Heijmans, Mathematical morphology: A modern approach in image processing based on algebra and geometry, SIAM review 37 (1) (1995) 1–36.
[7] Y. Luo, P. Cai, A. Bera, D. Hsu, W. S. Lee, D. Manocha, Porca: Modeling and planning for autonomous driving among many pedestrians, IEEE Robotics and Automation Letters 3 (4) (2018) 3418–3425.
[8] M. Althoff, On computing the minkowski difference of zonotopes, arXiv preprint arXiv:1512.02794 (2015).
[9] H. Barki, F. Denis, F. Dupont, A new algorithm for the computation of the minkowski difference of convex polyhedra, in: 2010 Shape Modeling International Conference, IEEE, 2010, pp. 206–210.
[10] G. Stengle, A nullstellensatz and a positivstellensatz in semialgebraic geometry, Mathematische Annalen 207 (2) (1974) 87–97.
[11] J. Stewart, Calculus, Brooks Cole, 2007.
[12] P. A. Parrilo, Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization, Ph.D. thesis, California Institute of Technology (2000).
[13] S. Lang, Algebra, Vol. 211, Springer-Verlag New York, 2002.
[14] J. Löfberg, Yalmip : A toolbox for modeling and optimization in matlab, in: In Proceedings of the CACSD Conference, Taipei, Taiwan, 2004.