Low complexity constrained control using higher degree Lyapunov functions

Sarmad Munir, Morten Hovd, Sorin Olaru

Low complexity constrained control using higher degree Lyapunov functions

Sarmad Munir [a], Morten Hovd [a], Sorin Olaru [b]

[a] *Engineering Cybernetics Department, Norwegian University of Science and Technology, N-7491 Trondheim, Norway*

[b] *Laboratory of Signals and Systems (L2S), CentraleSupélec, Université Paris-Saclay, France*

Abstract

Explicit Model Predictive Control often has a complex solution in terms of the number of regions required to define the solution and the corresponding memory requirement to represent the solution in the online implementation. An alternative approach to constrained control is based on the use of controlled contractive sets. However, polytopic controlled contractive sets may themselves be relatively complex, leading to a complex explicit solution, and the polytopic structure can limit the size of the controlled contractive set. This paper develops a method to obtain a larger controlled contractive set by allowing higher order functions in the definition of the contractive set, and explores the use of such higer-order contractive sets in controller design leading to a low complexity explicit control formulation.

Key words: Contractive Sets; Constrained Control; explicit MPC ; Ellipsoidal Contractive sets

1 Introduction

The ability to capture operational constraints is of vital importance in controller design for real-life applications. It is reasonable to state that the ability to handle constraints in a transparent way is what sets the industrially very successful Model Predictive Control (MPC) [13] apart from the theoretically elegant - but less industrially successful - LQG control. Standard MPC solves an optimization problem online, but due to the computational complexity of MPC it is limited to the systems which are not safety critical (due to the use of complex and thus error prone optimization software), have sufficiently slow dynamics, and/or can afford high performance computational hardware [5]. Explicit MPC [1] to some degree resolves this problem and allows the use of low-complexity computing code in the online implementation. Unfortunately, the explicit solution to standard MPC problems often has a highly complex solution, and even in cases when the explicit solution can be found in acceptable time the implementation of the solution on the online control hardware may require excessive memory. Low complexity constrained control with modest computational complexity, small memory requirements and simple, thus verifiable code in the online implementation is therefore desired.

One approach to such low complexity constrained control is based on the use of a controlled contractive set. The complexity of the solution will then depend on the complexity of the contractive set. Therefore, obtaining a controlled contractive set of low complexity is essential for this approach to formulate low complexity explicit constrained control. A maximal polyhedral controlled contractive set with a given contraction factor can be obtained by the iterative procedure described in [3]. However, the complexity of the contractive set thus obtained may be very high. A non-iterative procedure for obtaining a contractive set of low complexity is proposed in [5]. The approach is not applicable to systems with identical modes in series (corresponding to a non-diagonalizable A-matrix in the system's state space representation). Furthermore, the contractive set obtained in [5] is of fixed complexity, which does not allow trading off the complexity against the size of the contractive set. An optimization based technique has been proposed in [7] which allows the trading off complexity versus the size of the set. A solution to the optimization problem in [7] not

⋆ The work leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement no 607957 (TEMPO).

Email addresses: sarmad.munir@itk.ntnu.no (Sarmad Munir), morten.hovd@itk.ntnu.no (Morten Hovd), sorin.olaru@centralesupelec.fr (Sorin Olaru).

only reduces the on-line computational complexity of the resulting constrained control, but also ensures significant reduction in the memory required to store the explicit solutions. However the method explained in [7] is highly non-convex, which makes it difficult to use for finding sufficiently large contractive sets for higher dimensional systems. Alternatively, ellipsoidal contractive sets with corresponding linear control laws can be computed, but the measure of these sets is limited by the linear structure of the control law and the inherent conservatism of the corresponding quadratic Lyapunov function. This paper proposes a method to obtain an enlarged contractive set by defining the contractive set using a function of variable degree (a degree which is assumed to be greater or equal to 2, thus including the quadratic forms as a particular case), and also allowing for higher order control laws. Note that the function defining the controlled contractive set can be interpreted as controlled Lyapunov function for the closed loop system.

In Section 2, the controller design using controlled contractive sets is presented, along with the formulation for finding the largest ellipsoidal controlled contractive set fulfilling state and input constraints. Section 3 describes a controller design which leads to the determination of larger contractive sets. The controller design is inspired by the results in [10], but unlike the respective work, the controller will be defined using only two regions. The method described in Section 3 is applied to illustrative examples and the results are described in Section 4, which is followed by a discussion and conclusions in Section 5.

2 Contractive Sets

Consider the constrained control of the linear discrete time system:

$$x_{k+1} = Ax_k + Bu_k \tag{1}$$

with $x_k \in \Re^{n_x}$, $u_k \in \Re^{n_u}$ representing the current state and input, respectively, while x_{k+1} is the next time step state. The system is subject to input constraints $\mathcal{U} = \{u_k | H_u u_k \leq \mathbf{1}\}$, with $H_u \in \Re^{n_{p_u} \times n_u}$.

Definition 1 *Given a function $V : \Re^{n_x} \to \Re$, the level set of $V(x)$ for a scalar α is the set $S_\alpha = \{x | V(x) \leq \alpha\}$.*

Proposition 1 *Consider a function $V(x) : \Re^{n_x} \to \Re$ satisfying the following properties:*

A1 positive definite, with $V(0) = 0$,
A2 continuous,
A3 radially unbounded, i.e., $V(x) \to \infty$ as $\|x\| \to \infty$.

Then

(1) All level sets S_α exist and are bounded for all $0 \leq \alpha < \infty$.
(2) If $\beta < \alpha$, $S_\beta \subset S_\alpha$.

Proof:
From A1 it follows that the level sets $S_\alpha = \emptyset$ if $\alpha < 0$. Claim (1) follows directly from *A1*, *A2* and *A3*. For claim (2) we note that $S_\beta \subseteq S_\alpha$ is a consequence of Definition 1. Next, consider two points x_1 and x_2 with $V(x_1) = V(x_2) + \delta$ for some $\delta > 0$. Then by applying the Mean Value Theorem, continuity of $V(x)$ implies that the points x_1 and x_2 must be separated by some nonzero distance. Hence, we get strict inclusion, $S_\beta \subset S_\alpha$ if $\beta < \alpha$. □

Definition 2 *Consider a continuous and radially unbounded function $V : \Re^{n_x} \to \Re_{\geq 0}$. A level set S_α is controlled γ-contractive with respect to (1) for a given $\gamma \in (0, 1)$, if $\forall x_k \in S_\alpha, \exists u_k \in \mathcal{U}$ such that $x_{k+1} \in S_{\gamma\alpha}$.*

The functions $V(x)$ fulfilling the assumptions of Proposition 1 are natural ingredients in control designs enforcing contractiveness properties, as for example in the low complexity optimization based formulation

$$\min_{u_k, x_{k+1}} \frac{1}{2} x_{k+1}^T Q x_{k+1} + \frac{1}{2} u_k^T R u_k \tag{2a}$$

subject to

$$x_{k+1} = Ax_k + Bu_k \tag{2b}$$
$$H_u u_k \leq \mathbf{1} \tag{2c}$$
$$V(x_{k+1}) \leq \gamma V(x_k) \tag{2d}$$

where Q and R represent the state and input weights.

Consider next the bounded state constraints $x_k \in \mathcal{X}$ with $\mathcal{X} = \{x_k | H_x x_k \leq \mathbf{1}\}$ where $H_x \in \Re^{p_x \times n_x}$.

Proposition 2 *Let $V(x)$ be a function fulfilling assumptions $A1 - A3$ of Proposition 1, and let $V(x) = \alpha$, Then, if*

(1) the corresponding level set S_α is controlled γ-contractive, and
(2) $S_\alpha \subseteq \mathcal{X}$

the control action obtained as a solution of (2) guarantees an exponentially stability of the closed loop which in addition fulfills input and state constraints over S_α.

Proof:
Follows directly from Proposition 1 and Definition 2. □

As a result of Proposition 2, the function $V(x)$ is a Lyapunov function for the system (1) inside the set $S_{\bar{\alpha}}$, where $\bar{\alpha} = \max_\alpha$ such that $S_\alpha \subseteq \mathcal{X}$.

In [5] and [7], a controller based on (2) with polytopic controlled contractive sets $S = \{x_k | Fx_k \leq 1\}$ were studied based on a piecewise linear function

$$V(x_k) = max\{Fx_k\} \tag{3}$$

Using the function specified as in (3), the optimization (2) becomes a standard quadratic program, which may be solved parametrically with x_k and $V(x)$ as parameters. This is done by imposing a virtual parameter $\alpha_k = V(x_k)$ before solving the optimization in (2) at time k. The constraint in (2d) then simply becomes $F(Ax_k + Bu_k) \leq \gamma\alpha_k$. As the total number of constraints and the number of degrees of freedom are typically quite modest in (2) compared to a classical MPC problem utilizing a longer prediction horizon, the parametric solution is also of modest complexity. However, this approach suffers from the drawbacks described in the Introduction, and this paper therefore focuses on allowing more general types of function $V(x)$, to obtain a larger operating region with modest online computational complexity and memory requirement for the control.

In the developments below, two ellipsoidal controlled contractive sets will be important as terms of comparison:

- The set $\Omega = \{x \in \Re^{n_x} | x^T P^{-1} x \leq 1\}$, the largest controlled γ-contractive set that can be obtained using linear state feedback.
- The set $\Omega_{uc} = \{x \in \Re^{n_x} | x^T P_{uc}^{-1} x \leq 1\}$, the ellipsoidal set where γ-contractiveness is achieved with the linear state feedback $u_k = K_{uc} x_k$.

Constraints in both states and inputs are accounted for in the calculation of both Ω and Ω_{uc}. These sets can be calculated using well known techniques based on Linear Matrix Inequalities, see, *e.g.*, [2] or [8] for details.

While the set Ω_{uc} can be found for any given controller [1] K_{uc}, for the subsequent use in this paper it will be considered to be the unconstrained solution to (2), see the inverse optimality arguments in [9] for the choice of weights Q and R. When ignoring the input and contractivity constraints, (2) yields the controller

$$u_k = \underbrace{-(R + B^T Q B)^{-1} B^T Q A}_{K_{uc}} x_k$$

For notational convenience in the following, we will define $P_1 = P^{-1}$ and $P_0 = P_{uc}^{-1}$.

[1] For subsequent developments to make sense, the controller K_{uc} should clearly be designed such that the unconstrained closed loop system is γ-contractive, *i.e.*, such that $max\ |eig(A + BK)| \leq \sqrt{\gamma}$.

3 Controller Design for Higher Order Contractive Set

3.1 Approximate optimization problem solution

Despite the computational advantages of the ellipsoidal sets recalled above, the associated linear feedback and the quadratic structure of the Lyapunov function limit the volume of the contractive set. Larger contractive sets can be obtained by increasing the complexity of the Lyapunov function. A natural approach is to focus on polynomial forms of higher order. By relaxing the control structure to allow the input to have higher order dependency on the state, further relaxations on the contractive set can be obtained.

Using (2b) to eliminate x_{k+1} from the optimization formulation, (2) may be reformulated as Problem $\mathcal{P}_u$:

$$\min_{u_k} \frac{1}{2} u_k^T H u_k + x_k^T F u_k \tag{4a}$$

subject to

$$H_u u_k \leq \mathbf{1} \tag{4b}$$

$$V(x_{k+1}) \leq \gamma V(x_k) \tag{4c}$$

where $H = (B^T Q B + R)$, $F = A^T Q B$.

The Lagrangian function for Problem $\mathcal{P}_u$ is

$$\begin{aligned} \mathcal{L}(u_k) = \frac{1}{2} u_k^T H u_k + x_k^T F u_k \\ + \lambda_u^T (H_u u_k - \mathbf{1}) + \lambda_q (V(x_{k+1}) - \gamma V(x_k)) \end{aligned} \tag{5}$$

The corresponding KKT conditions are

$$H u_k + F^T x_k + H_u^T \lambda_u + V_{k+1}^1(u_k) \lambda_q = 0 \tag{6a}$$

$$H_u u_k - \mathbf{1} \leq 0 \tag{6b}$$

$$V(x_{k+1}) - \gamma V(x_k) \leq 0 \tag{6c}$$

$$\lambda_u \geq 0 \tag{6d}$$

$$\lambda_q \geq 0 \tag{6e}$$

$$\lambda_u^T (H_u u_k - \mathbf{1}) + \lambda_q (V(x_{k+1}) - \gamma V(x_k)) = 0 \tag{6f}$$

where $V_{k+1}^1(u_k)$ denotes the first derivative of $V(x_{k+1})$ with respect to u_k.

Next, consider the optimizaton problem $\mathcal{P}_c$:

$$\min_{c, u_k, \lambda_u, \lambda_q} c \tag{7a}$$

subject to (6a) - (6e) and

$$-\lambda_u^T (H_u u_k - \mathbf{1}) - \lambda_q (V(x_{k+1}) - \gamma V(x_k)) \leq c \tag{7b}$$

Denote the solution to (7) by $c^*, u_k^*, \lambda_u^*, \lambda_q^*$. Observe that if $c^* = 0$, u_k^* is also the optimal solution to $\mathcal{P}_u$. Clearly, $c^* < 0$ is not possible. The constraints of Problem $\mathcal{P}_c$ includes the constraints of Problem $\mathcal{P}_u$, and hence a feasible solution to $\mathcal{P}_c$ is also a feasible solution to $\mathcal{P}_u$ with u_k^* (obviously fulfilling the constraints of problem $\mathcal{P}_c$) being a suboptimal solution to $\mathcal{P}_u$.

Define by $J(u_k^*(c)) = \frac{1}{2}(u_k^*)^T H u_k^* + x_k^T F u_k^*$ the value of the objective function of Problem $\mathcal{P}_u$ evaluated with the input u_k^* from the solution to Problem $\mathcal{P}_c$. Correspondingly, let $J(u_k^*(0))$ denote the (optimal) value function for Problem $\mathcal{P}_u$.

Lemma 3 *Consider an optimal solution $(c^*, u_k^*, \lambda_u^*, \lambda_q^*)$ to $\mathcal{P}_c$, with $c^* > 0$. Then u_k^* is a suboptimal solution to $\mathcal{P}_u$, with $J(u_k^*(c)) - J(u_k^*(0)) < c$.*

Proof:
This proof follows the approach in [10]. For any feasible u_k

$$\begin{aligned}\frac{1}{2}u_k^T H u_k + x_k^T F u_k \geq \frac{1}{2}u_k^T H u_k + x_k^T F u_k \\ + \begin{bmatrix}\lambda_u^* \\ \lambda_q^*\end{bmatrix}^T \begin{bmatrix} H_u u_k - \mathbf{1} \\ V(x_{k+1}) - \gamma V(x_k)\end{bmatrix} = M(u_k)\end{aligned} \tag{8}$$

Next, minimize both sides subject to constraints (6b) and (6c). Thus

$$J(u_k^*(0)) \geq \min_{(6b),(6c)} M(u_k) \geq \min_{u_k \in \Re^{n_u}} M(u_k) \tag{9}$$

The function $M(u_k)$ can be recognized as the Lagrangian function or $\mathcal{P}_u$, with fixed $\lambda_u = \lambda_u^*$ and $\lambda_q = \lambda_q^*$. The unconstrained minimization of $M(u_k)$ yields (6a), again with $\lambda_u = \lambda_u^*$ and $\lambda_q = \lambda_q^*$. One therefore finds that the optimal value for the unconstrained minimization of $M(u_k)$ yields $u_k = u_k^*$. Thus,

$$J(u_k^*(0)) \geq M(u_k^*)$$

Multiply the inequality above with -1 and add $J(u_k^*(c))$ to both sides to obtain

$$J(u_k^*(c)) - J(u_k^*(0)) \leq c^* \tag{10}$$

□

3.2 Problem reformulation

In the following, the approximate solution to the optimization problem $\mathcal{P}_u$ is sought as a function of the present state vector x_k. However, for u_k expressed as a polynomial function of x_k, $V(x_{k+1})$ will have a higher order dependence on the polynomial coefficients of $u_k(x_k)$, which will cause problems in the resulting controller design formulation. To circumvent this problem, attention is refocused on the equivalent formulation (2), the KKT conditions of which are

$$Ru_k - B^T\lambda_e + H_u^T\lambda_u = 0 \tag{11a}$$
$$Qx_{k+1} + \lambda_e + \lambda_q \nabla V(x_{k+1}) = 0 \tag{11b}$$
$$x_{k+1} - Ax_k - Bu_k = 0 \tag{11c}$$
$$H_u u_k - \mathbf{1} \leq 0 \tag{11d}$$
$$V(x_{k+1}) - \gamma V(x_k) \leq 0 \tag{11e}$$
$$\lambda_u \geq 0 \tag{11f}$$
$$\lambda_q \geq 0 \tag{11g}$$
$$-\lambda_u^T(H_u u_k - \mathbf{1}) - \lambda_q(V(x_{k+1}) - \gamma V(x_k)) = 0 \tag{11h}$$

where the operator $\nabla \triangleq \frac{d}{dx_{k+1}}(.)$. As before, the complementarity constraints are relaxed, and solutions with a commensurate relaxation are sought, in which the input is expressed as a function of the present state x_k. However, as a novelty of the present approach, instead of considering the model equations as a separate constraint, this will be added as an extra term in all other constraints involving both x_k and x_{k+1}. This yields the formulation

$$\min c \tag{12a}$$

subject to constraints (11d), (11f), (11g), and

$$Ru_k - B^T\lambda_e + H_u^T\lambda_u + \mu_1^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{12b}$$
$$Qx_{k+1} + \lambda_e + \lambda_q \nabla V(x_{k+1}) + \mu_2^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{12c}$$
$$V(x_{k+1}) - \gamma V(x_k) + \mu_3^T(x_{k+1} - Ax_k - Bu_k) \leq 0 \tag{12d}$$
$$\begin{aligned}-\lambda_u^T(H_u u_k - \mathbf{1}) - \lambda_q(V(x_{k+1}) - \gamma V(x_k)) \\ + \mu_4^T(x_{k+1} - Ax_k - Bu_k) \leq c\end{aligned} \tag{12e}$$

The multipliers μ_1, μ_2, μ_3 and μ_4 as well as λ_e are polynomial functions of x_k and x_{k+1} with no positivity constraint. The multipliers λ_q and λ_u are also polynomials in x_k and x_{k+1}, but have the positivity constraints as explicitly stated above. We retain from the control framework that it is important to evaluate the input based on information available online, and hence u_k can only be designed as a function of x_k. The degrees of freedom in this optimization are the polynomial coefficients of the multipliers, the coefficients of the input considered as a polynomial feedback function, and finally the coefficients of the polynomial Lyapunov function $V(x_k)$. This reformulation can be interpreted as lifting the original problem formulation to a higher dimensional space with both x_k and x_{k+1} as independent variables, while the terms involving the μ_i ensure that the equations in

which they are inserted hold on the manifold of system trajectories (where the term inside the parentheses is identically zero). The system equations (1) are fulfilled by the physics of the system - and thus need not be enforced by the control. In contrast, the controller design has to ensure that equalities (11a) and (11b) are fulfilled along the trajectories of the system, in order to ensure an approximately optimal control.

The controller design attempts to find a feasible approximately optimal solution to (2) for the set

$$S = \{x | V(x) \leq 1\}$$

The design procedure is initialized with $S = \Omega$ and $V(x) = x^T P_1 x$, and thereafter the set S is iteratively enlarged while allowing for higher order $V(x)$. The optimal controller inside Ω_{uc} is already known to be K_{uc}. Note that $\Omega_{uc} \subseteq S$. The controller design can therefore be divided in two parts, one is the unconstrained controller for Ω_{uc} and the other controller is designed for a region inside S but outside the ellipsoid Ω_{uc}. This region can be defined by the set

$$S_C = \{x_k | p(x_k) > 0\} \tag{13a}$$

where

$$p(x_k) = -(1 - V(x_k))(1 - x_k^T P_0 x_k) \tag{13b}$$

Using the S-procedure, the constraints defined above can be enforced in the region where $p(x_k)$ is positive. In addition, it is natural to allow for a larger relaxation of the complementarity constraints (and thus larger absolute distance to optimum) when the state is far from the origin. Therefore, the complementarity constraints are relaxed by a factor $cx_k^T x_k$ instead of relaxing only by c. This yields the optimization formulation

$$\min c \tag{14a}$$

$$Ru_k - B^T\lambda_e + H_u^T\lambda_u + \mu_1^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{14b}$$

$$Qx_{k+1} + \lambda_e + \lambda_q \nabla V(x_{k+1}) + \mu_2^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{14c}$$

$$H_u u_k - \mathbf{1} + s_1(x_k)p(x_k) \leq 0 \tag{14d}$$

$$V(x_{k+1}) - \gamma V(x_k) + \mu_3^T(x_{k+1} - Ax_k - Bu_k) + s_2(x_k, x_{k+1})p(x_k) \leq 0 \tag{14e}$$

$$\lambda_u - s_3(x_k)p(x_k) \geq 0 \tag{14f}$$

$$\lambda_q - s_4(x_k)p(x_k) \geq 0 \tag{14g}$$

$$-\lambda_u^T(H_u u_k - \mathbf{1}) - \lambda_q(V(x_{k+1}) - \gamma V(x_k)) + \mu_4^T(x_{k+1} - Ax_k - Bu_k) + s_5(x_k, x_{k+1})p(x_k) \leq cx_k^T x_k \tag{14h}$$

Here $s_1(x_k)$, $s_2(x_k, x_{k+1})$,$s_3(x_k)$, $s_4(x_k)$ and $s_5(x_k, x_{k+1})$ are SOS polynomials.

It was explained above that the set S is increased iteratively. Note that $S_{j-1} \subseteq S_j$ provided:

$$(1 - V_j(x_k)) - s_6(x_k)(1 - V_{j-1}(x_k)) \geq 0 \tag{15}$$

It is not yet guaranteed that the set S fulfills the state constraints. The state constraints are given by $H_x x_k \leq \mathbf{1}$ and thus may be regarded as the intersection of the n_c unbounded sets $C_r = \{x_k | H_{x,r} x_k \leq 1\}$, where $H_{x,r}$ is the $r'th$ row of H_x. Hence the state constraints become:

$$(1 - H_{x,r}x_k) - \sigma_r(x_k)(1 - V_j(x_k)) \geq 0, r = 1, \ldots p_x \tag{16}$$

where $s_6(x_k)$ and $\sigma_r(x_k)$ are SOS polynomials. Therefore, an approximate solution to the optimization problem described in (2) can be found by solving

$$\min c \tag{17a}$$

subject to

$$Ru_k - B^T\lambda_e + H_u^T\lambda_u + \mu_1^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{17b}$$

$$Qx_{k+1} + \lambda_e + \lambda_q \nabla V(x_{k+1}) + \mu_2^T(x_{k+1} - Ax_k - Bu_k) = 0 \tag{17c}$$

$$H_u u_k - \mathbf{1} + s_1(x_k)p(x_k) \leq 0 \tag{17d}$$

$$V(x_{k+1}) - \gamma V(x_k) + \mu_3^T(x_{k+1} - Ax_k - Bu_k) + s_2(x_k, x_{k+1})p(x_k) \leq 0 \tag{17e}$$

$$\lambda_u - s_3(x_k)p(x_k) \geq 0 \tag{17f}$$

$$\lambda_q - s_4(x_k)p(x_k) \geq 0 \tag{17g}$$

$$-\lambda_u^T(H_u u_k - \mathbf{1}) - \lambda_q(V(x_{k+1}) - \gamma V(x_k)) + \mu_4^T(x_{k+1} - Ax_k - Bu_k) + s_5(x_k, x_{k+1})p(x_k) \leq cx_k^T x_k \tag{17h}$$

$$(1 - V_j(x_k)) - s_6(x_k)(1 - V_{j-1}(x_k)) \geq 0 \tag{17i}$$

$$(1 - H_{x,r}x_k) - \sigma_r(x_k)(1 - V_j(x_k)) \geq 0, r = 1, \ldots n_c \tag{17j}$$

3.3 Solving the Sums-of-Squares problem

The optimization problem described in (17) can be solved by using sum-of-squares (SOS) programming. The SOS method has received a lot of attention since the PhD thesis of Parrilo [12]. The SOS technique generalizes the known algorithmic tool in Linear Matrix Inequalities (LMIs) for which there exist many efficient solvers. The SOS method employs the similar techniques

as LMI problems, but all problems are formulated in terms of the polynomials or polynomial matrices [11].

In the problem described in (17), all constraints are of SOS type except the constraints (17b) and (17c), which gives linear constraints on the parameters of the polynomials $u_k(x_k)$, $\lambda_e(x_k, x_{k+1})$, $\lambda_q(x_k, x_{k+1})$ and $\lambda_u(x_k)$. The polynomials s_1,s_2,s_3,s_4,s_5, s_6 and σ_r are all SOS polynomials in x_k, as is also $V(x_k)$. It can be noticed that the problem is bilinear in the coefficients of s_i, σ_r and $V(x_k)$. It is also bilinear with respect to coefficients of λ_u and u_k. Instead of attempting to solve the bilinear problem formulation directly, this paper instead uses the common approach of iteratively solving linear subproblems:

- First λ_u, λ_q, λ_e, s_i and σ_r are optimized with given $V(x_k)$ and u_k.
- Then $V(x_k)$ and u_k are optimized, with λ_u, λ_q, s_i and σ_r from the step above.

The problem (17) makes sure that the set $S_{j-1} \subseteq S_j$ but it does not guarantee that $S_{j-1} \subset S_j$ and it may result in $S_{j-1} = S_j$ for any number of iterations. To enforce that the set S_j is larger than the set S_{j-1}, we select the points outside S_{j-1} and ensure that they are included in S_j, which results in a larger set S_j. After a certain number of iterations we may not be able to find a set S_j greater than S_{j-1}. Increasing the degree of $V(x_k)$ and/or $u(x_k)$, one may then be attempted in order to increase the size of S_j further.

3.4 Algorithm

This section provides a comprehensive procedure to obtain a controlled contractive set. We start with the linear system and obtain the largest ellipsoidal contractive set Ω and the largest unconstrained ellipsoidal contractive Ω_{uc} set for that system as discussed in section 2. These sets always exist in the linear system framework. We want to find a contractive set S which is larger than Ω. Of interest is only the part of this contractive set where the function $p(x_k)$ is positive. This controller will ensure that all the system trajectories contract towards the origin at least by factor γ. Once the trajectories are inside Ω_{uc}, the controller K_{uc} will take over (representing in fact a switched control with a state-based switching rule).

We can choose the degree x_{deg} of $V(x_k)$ (which also defines the set S) and the degree u_{deg} of the control law u_k as per system requirements. Select *num* number of uniformly distributed points on the boundary of the ellipsoid Ω and check which point undergoes most contraction. In this way we select a point where the contraction constraint is farthest from being active. The idea is to push this point outwards and ensure that S contains the new scaled point, in this way S will be larger than Ω. As discussed in the section 3.3, obtaining S is an iterative procedure. In the first iteration, the problem formulated in (17) is solved by keeping $S = \Omega$ and u_k as the corresponding control law for Ω. By solving the problem, the optimized values for λ_u, λ_q, λ_e, μ, s_i, and σ_r can be obtained. These are now kept fixed and(17) can be updated by solving for $V(x_k)$ and u_k. An additional constraint is added to (17), which if a valid solution is found ensures that the point outside Ω are included in the set S. Now we can take the new set S as our starting set and repeat the whole procedure to obtain even larger set. This process should continue until the contractive set cannot be enlarged any further. In that case the iteration should stop, or either x_{deg} or u_{deg} (or both) should be increased.

Algorithm 1 Algorithm to obtain a larger contractive set

Input: A contractive ellipsoid Ω with control law u_k. Maximum allowed degree (x_{deg}) for the Lyapunov function, maximum allowed degree (u_{deg}) for the control law, and maximum acceptable measure of sub-optimality (c_{max}).

Output: A large contractive set of degree $\leq x_{deg}$ with control law of degree $\leq u_{deg}$.

1: Set $j = 0$, $S_j = \Omega$ such that $S_j = \{x_k | V(x_k) \leq 1\}$ and set solution = feasible.
LOOP Process
2: **while** solution is feasible **do**
3: Set $j = j + 1$
4: Find boundary points of the set S_{j-1} by solving for $V(x_k) == 1$ along rays in directions defined by vectors from the origin to points uniformly distributed on the n_x-dimensional unit sphere.
5: Check which point contracts most by applying control law u_k. Select that point *point*.
6: Solve the SOS problem (17) by keeping $V(x_k)$ and u_k fixed. Optimize for λ_u, λ_q, λ_e, s_i and σ_r.
7: Set a small positive number *increment*, which will be used to specify a point outside S_{j-1}.
8: Find a point outside S_{j-1} by adding $increment \times point$ to *point* i.e. $point_{new} = point + increment \times point$.
9: Keep λ_u, λ_q, s_i and σ_r fixed, optimize (17) using $V(x_k)$ and u_k, while ensuring that the point $point_{new}$ is included in S_j.
10: **if** solution is feasible & $c \leq c_{max}$ **then**
11: Update $V(x_k)$, u_k, λ_u, λ_q and s_i.
12: **end if**
13: **end while**

Let the controller obtained from algorithm 1 be denoted by $u_k = c_u(x_k)$, then the explicit controller for the overall problem becomes:

$$u_k = \begin{cases} K_{uc} x_k, \text{ if } x_k \in \Omega_{uc} \\ c_u(x_k), \text{ if } x_k \in S_C \end{cases} \tag{18}$$

4 Examples

The application of the method described in section 3 is illustrated in this section. The SOS problems are formulated and solved using YALMIP [6].

4.1 Example 1

Consider a linear system whose state representation is given as:

$$x_{k+1} = \begin{bmatrix} 1 & 0.2 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0.22 \\ 0.2 \end{bmatrix} u_k \tag{19}$$

The input constraints are given as $-2 \leq u_k \leq 2$ and each element j of the state vector should fulfill the constraints $-5 \leq x_{k,j} \leq 5$. The specified contraction factor is 0.90. The contractive sets obtained are shown in Figure 1. The set in front is obtained by the Dorea-Hennet procedure [3], it is comprised of 250 hyperplanes. The black set represents the largest ellipsoid obtained with the linear feedback, light and dark gray sets behind black set are obtained using the controller design described in section 3. The light gray set is a level set of the Lyapunov function of degree 8 and the dark gray one is a level set of the Lyapunov function of degree 10. Both of these sets use a linear structure for the control law.

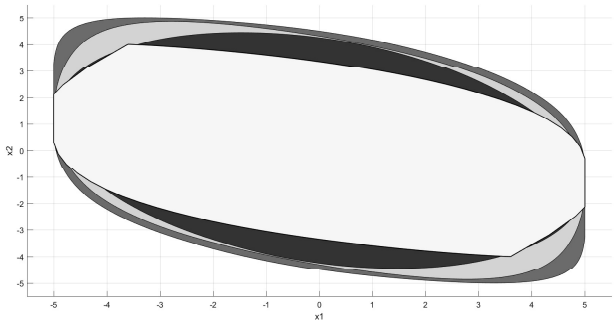

Fig. 1. Contractive Sets of variable degree on V(x) with linear input for Example 1

The state trajectories for the dark gray set are shown in figure 2. The method described in section 3 is used to design the controller for the region (dark gray), which is outside the ellipsoidal (light gray) set, whereas inside the ellipsoidal set the unconstrained controller $K_u c$ is used. The ellipsoidal set is the set corresponding to the unconstrained controller, and the unconstrained (linear) control law is applied once state is inside the ellipsoidal set. The value of c, the measuring distance from the optimum comes out to be $4.54 \cdot 10^{-12}$.

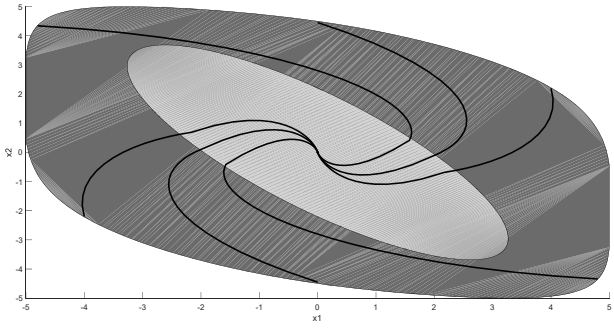

Fig. 2. State Trajectories for Example 1

In Figure 3, contractive sets with variable input degree are shown. All sets use a Lyapunov function of degree 8. The white set is found with a controller of degree 3, while the dark and light gray sets correspond to the controllers of degree 2 and 1 respectively.

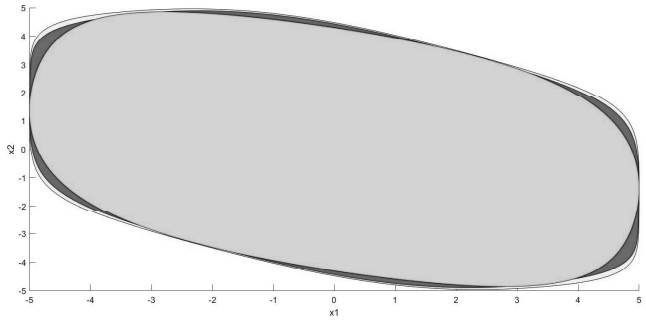

Fig. 3. Contractive Sets with varaible input degree for Example 1

4.2 Example 2

Consider the following system with oscillatory modes (corresponding to a complex conjugate pair of eigenvalues).

$$x_{k+1} = \begin{bmatrix} 0.3 & -0.2 \\ 0.1 & 0.1 \end{bmatrix} x_k + \begin{bmatrix} 0.2 & 0.4 \\ 0.4 & 0.5 \end{bmatrix} u_k \tag{20}$$

The input and the state constraints are given as $-2 \leq u_{k,i} \leq 2, i = 1, 2$ and $-5 \leq x_{k,j} \leq 5, j = 1, 2$ respec-

Table 1
Contractive sets method versus volume

Method	Degree or Hyperplanes	Volume
Dorea-Hennet	8	75.7677
Ellipsoidal	2	78.5398
VdLF	8	87.2972

tively. The contraction factor is selected to be 0.20.

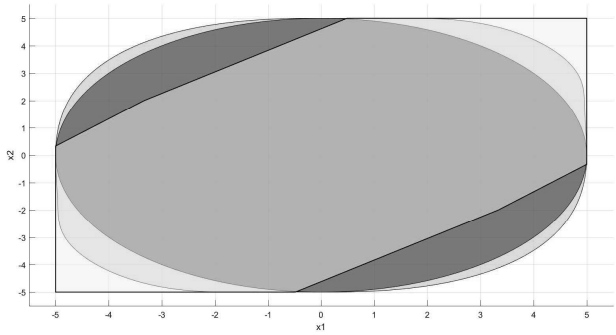

Fig. 4. Polyhedral, Ellipsoidal and pre-specified degree Contractive Sets for Example 2

The set shown in front of the Figure 4 is the largest polyhedral contractive set fulfilling these constraints, obtained by the method described in [3]. The dark gray set is the largest ellipsoid and the light gray set at the back is the contractive set obtained by the method described in this paper. The control law for the light gray set is linear and is given as:

$$u_k = \begin{bmatrix} -0.0870 & -0.0592 \\ -0.0988 & 0.0653 \end{bmatrix} x_k$$

Table 1 compares the results of the contractive sets obtained by different methods. Dorea-Hennet is the method described in [3] (the set in front in the figure (4)), ellipsoidal is the method which gives the largest ellipsoid fulfilling the constraints (the dark gray set in the figure (4))and VdLF is a variable degree Lyapunov function method developed in this paper. The value of c is $1.22 \cdot 10^{-10}$.

4.3 Example 3

Consider the following linear system,

$$x_{k+1} = \begin{bmatrix} 1 & 0.98 & 0.1 \\ 0 & 1 & 0.98 \\ 0 & 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 0.8 \\ 0.3 \\ 0.2 \end{bmatrix} u_k \qquad (21)$$

where $-5 \leq x_{k,j} \leq 5, j = 1, 2, 3$ and $-2 \leq u_k \leq 2$. The contraction factor is selected to be 0.9. The polyhedral contractive set obtained by the method described in [3] is shown in the figure 5 and its volume is 11.9. The largest ellipsoidal set Ω has volume 71.13, and is shown in red in Figure 5. The volume of the set S obtained by the method proposed in the paper, comes out to be 124.36 (green set in figure 5). The set S is the level set of a Lyapunov function of degree 6 and it is obtained using a linear control law.

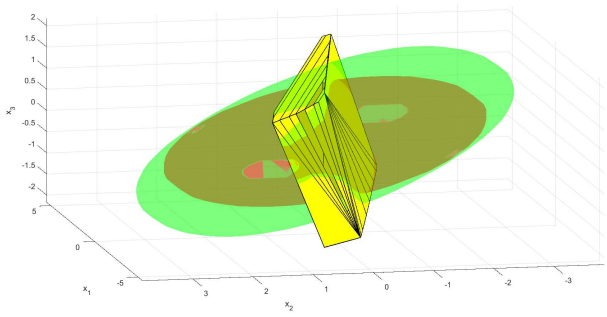

Fig. 5. Controlled contractive sets for example 3.

5 Discussion and Conclusion

This paper presents a method to obtain a large controlled contractive set of specified degree with the corresponding control law. It was shown that the contractive set can be enlarged by increasing the degree of the contractive set or the degree of the control law. The examples show a significant increase in the size of the contractive set, and hence a corresponding increase in the size of the region where the controller is defined. The resulting controllers are explicit and characterized by low complexity, with a modest memory footprint for the online implementation, and require only very simple computer code.

Currently, the size of SOS programming problems that can be handled is limited, and this will limit both the number of states and the polynomial order that can be handled by the method described in this paper. However, research aiming at increasing the size of SOS problems that can be handled is very active and is making progress, as was documented at the 2017 CDC conference (see, *e.g.*, [14] and [4]).

The results in this paper handle nominal linear dynamics only. Conceptually, they are straight forward to generalize to linear systems with polytopic uncertainty, by replicating the design criteria for each extreme point in the polytopic uncertainty set, while keeping the Lyapunov function and the feedback control identical for all extreme dynamics. Simple interpolation can then be used to show that the design criteria also hold for internal points in the uncertainty polytope. However, this will quickly lead to very large SOS problems.

References

[1] Bemporad, A., Morari, M., Dua, V., and Pistikopoulos,E.N. (2002). *The explicit linear quadratic regulator for constrained systems.* Automatica Volume 38, Issue 1, January, Pages 3–20.

[2] Blanchini, F., and Miani, S. *Set Theoretic Methods in Control - Second Edition.*

[3] Dorea, C., and Hennet, J.C. (1999). *(a,b)-invariant polyhedral sets for linear discrete-time systems.* Journal of Optimization Theory and Applications, 103(3):521-542.

[4] Hall, G. (2017) *Iterative LP and SOCP-Based Approximations to Semidefinite Programs, 56th IEEE Conference on Decision and Control*, Melbourne, Australia.

[5] Hovd, M., Olaru, S., and Bitsoris, G. (2014). *Low Complexity Constraint Control Using Contractive Sets. Preprints of the 19th World Congress.*The International Federation of Automatic Control, Cape Town, South Africa, August 24-29.

[6] Löfberg, J. *YALMIP : A Toolbox for Modeling and Optimization in MATLAB, Proceedings of the CACSD Conference*, Taipei, Taiwan.

[7] Munir, S., Hovd, M., Sandou, G., and Olaru, S. (2016). *Controlled contractive sets for low-complexity constrained control.* IEEE Multi-conference on Systems and Control, Buenos Aires, September.

[8] Ngyuyen, H.N. (2012) *Constrained control of uncertain systems: An interpolation baszed approach.* PhD thesis, Supelec, Gif-sur-Yvette, France.

[9] Nguyen, N. A., Olaru, S., Rodriguez-Ayerbe, P., and Hovd, M. (2014). *An inverse optimality argument to improve robustness in constrained control, Preprints of the 19th World Congress, The International Federation of Automatic Control*, Cape Town, South Africa. August 24-29.

[10] Oishi, Y. (2012). *Direct Design of a Polynomial Model Predictive Controller, Proceedings of the 7th IFAC Symposium on Robust Control Design.* The International Federation of Automatic Control Aalborg, Denmark, June 20-22, 2012

[11] Papachristodoulou, A. and Prajna, S. (2005). *A Tutorial on Sum of Squares Techniques for Systems Analysis.* American Control Conference June 8-10, Portland, OR, USA.

[12] Parrilo, P.A. (2000). *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization.* PhD thesis, California Institute of Technology, Pasadena, CA.

[13] Qin, S. J., and Badgwell, T. A. (2003). *A Survey of Industrial Model Predictive Control Technology.* Control Engineering Practice, 11(7):733-764.

[14] Zheng, Y. and Papachristodoulou, A. (2017) *Exploiting Sparsity in the Coefficient Matching Conditions in SOS Programming, 56th IEEE Conference on Decision and Control*, Melbourne, Australia. December.