arXiv:2305.01222v1 [math.OC] 2 May 2023

# 控制李雅普诺夫函数与屏障函数的 SOS 构造

**Michael Schneeberger** * **Florian D¨orfler** ** **Silvia Mastellone** *

*FHNW 电气工程研究所，瑞士温迪施（电子邮件：*
*michael.schneeberger@fhnw.ch，silvia.mastellone@fhnw.ch*）
**瑞士苏黎世苏黎世联邦理工学院信息技术与电气工程系（电子邮件：
*dorfler@control.ee.ethz.ch*）

**摘要：**
我们提出一种新方法，通过结合控制李雅普诺夫函数（CLF）和控制屏障函数（CBF），来认证受约束多项式系统的闭环稳定性与安全性。对于关于控制输入呈仿射形式的多项式系统，这两类函数均可通过平方和（SOS）规划构造。利用正定理的两个版本，我们推导出一种平方和表述，以寻找一个有理控制器；若该表述可行，则所得控制器将与控制李雅普诺夫函数及多个控制屏障函数相容。

## 1. 引言

对于具有状态约束的系统，设计一个既能确保稳定性又能满足约束的控制器至关重要。在大多数情况下，反馈控制设计主要着眼于实现稳定性，而保护函数只有在约束被违反时才会起作用。遗憾的是，这种方法会导致停机，并需要调查故障。相比之下，兼具稳定性与安全性的控制器可以牺牲控制性能，以防止系统进入不安全状态。

本文着重于构造兼容的控制李雅普诺夫函数和控制屏障函数，以证明控制系统的稳定性和安全性。控制李雅普诺夫函数为给定控制系统中稳定化控制器的存在性确立条件。类似地，控制屏障函数保证存在一个能够使控制系统保持安全的控制器。根据 Wieland 和 Allg¨ower（2007）的定义，如果从安全集中的任意状态出发的状态轨迹始终位于由状态约束定义的允许区域内，则该系统被视为安全。对于输入仿射系统，满足控制李雅普诺夫函数和控制屏障函数条件的控制器可通过在线求解二次规划（QP）来实现，参见 Ames 等人（2019）。然而，对于某些状态，这些条件可能相互冲突。若要使两者对所有状态同时成立，还需要满足控制共享性质，参见 Grammatico 等人（2013）；Xu（2016）。寻找此类兼容的控制李雅普诺夫函数和控制屏障函数通常较为困难。然而，对于多项式系统，可以通过构造 SOS 约束并利用半定规划（SDP）求解来实现这一目标。

本文的贡献有两方面：首先，我们利用两个版本的正定理，推导出关于兼容控制李雅普诺夫函数和多个控制屏障函数的 SOS 约束；其次，我们提出了一种算法，通过最大化安全集体积的替代指标，高效地求解这些 SOS 约束。Isaly 等人（2022）以及 Tan 和 Dimarogonas（2022）对兼容控制屏障函数的条件进行了表述，但未给出构造这些函数的方法。Clark（2021）所描述的构造性方法基于引入额外的 SOS 约束来强制实现兼容性。本文揭示了由控制李雅普诺夫函数条件（分别为控制屏障函数条件）推导出的 SOS 约束，与存在一个能够使闭环系统稳定（分别为安全）的有理控制器之间的对应关系。通过将这类控制器限制为相同的控制器，我们推导出了一组新的 SOS 约束，无需引入额外的 SOS 约束，即可保证一个控制李雅普诺夫函数与多个控制屏障函数之间的兼容性。

这些 SOS 约束包含双线性项，因此无法直接转换为半定规划（SDP）。为此，我们提出一种交替算法，通过反复求解两个半定规划，同时搜索控制李雅普诺夫函数和多个控制屏障函数。具体而言，该算法旨在最大化安全集的体积；该安全集由各个控制屏障函数所定义的不变集的交集构成。当单个控制屏障函数不足以满足需求时，多个控制屏障函数可以提供额外的灵活性，从而扩大安全集的体积。Anghel 等人（2013）、Kundu 等人（2019）和 Wang 等人（2022）提出了类似方法，但这些方法要么只搜索单个控制李雅普诺夫函数，要么只搜索单个控制屏障函数。Korda 等人（2014）则提出了另一种颇具启发性的方法，通过求解一个无限维线性规划问题来确定安全集。我们通过一个电力变换器控制示例展示了该方法的实用性，在这个示例中，安全性至关重要。

本文结构如下：第 2 节介绍本文采用的记号并回顾一些预备知识。随后，第 3 节回顾 aCLF 和控制屏障函数的定义，并推导出一个有理控制器。第 4 节通过引入控制共享性质，将控制李雅普诺夫函数和控制屏障函数结合起来。第 5 节定义编码控制李雅普诺夫函数和控制屏障函数条件的 SOS 程序，第 6 节介绍求解该 SOS 程序的算法。第 7 节给出数值仿真。最后，第 8 节总结全文并讨论未来工作。

## 2. 预备知识与符号

### *2.1* 符号

在本文中，我们采用以下记号。简写 $[t] := \{1, 2, ..., t\}$ 用于表示一个数值范围。若标量函数 $V\colon \mathbb{R}^n \to \mathbb{R}$ 满足 $V(x^*) = 0$ 且对 $x \neq x^*$ 有 $V(x) > 0$，则称其关于 $x^*$ 正定。$R[x]$ 表示变量 $x \in \mathbb{R}^n$ 的标量多项式集合，$\Sigma[x]$ 表示 $x$ 中的标量 SOS 多项式集合。若多项式 $p(x)$ 可写为 $p(x) = \sum_{i=1}^{k} g_i(x)^2$，其中 $g_i(x) \in R[x]$ 且 $i \in [k]$，则称其为 SOS 多项式。若 $p(x) \in \Sigma[x]$，则可将其展开为

$$p(x) = Z(x)^T Q Z(x),$$

其中，$Z(x)$ 是 $x$ 中的单项式向量，$Q$ 是一个方形半正定矩阵。下文中，我们考虑如下描述的多项式控制系统：

$$\dot{x} = \mathcal{F}_c(x, u) = f(x) + G(x)u_c, \tag{1}$$

其中，$f(x) \in (R[x])^n$ 和 $G(x) \in (R[x])^{n\times m}$ 是多项式矩阵，$u_c \in \mathbb{R}^m$ 是控制输入向量。系统 (1) 采用多项式状态反馈控制策略 $u(x) \in (R[x])^m$ 后，得到如下形式的闭环多项式系统：

$$\dot{x} = \mathcal{F}_a(x) = f(x) + G(x)u(x) \tag{2}$$

若 $\mathcal{F}_a(x^*) = 0$，则称状态 $x^*$ 为 (2) 的平衡点。

若对于每个 $x_0 \in \mathcal{X}$，都有 $x(t) \in \mathcal{X}$ 对所有 $t \in \mathbb{R}_+$ 均成立，则称集合 $\mathcal{X}$ 关于 (2) 是正向不变的 (Khalil, 2002)。如果对于允许状态集合 $\mathcal{X}_a \subseteq \mathbb{R}^n$ 和安全状态 $\mathcal{X}_s \subseteq \mathbb{R}^n$，$\mathcal{X}_s$ 是正向不变的且 $\mathcal{X}_s \subseteq \mathcal{X}_a$，则称系统 (2) 是安全的 (Wieland 和 Allg¨ower, 2007)。

### *2.2* 平方和规划

将二次代价函数最小化并满足 SOS 约束的 SOS 规划定义如下：

$$\begin{aligned} &\underset{v}{\text{minimize}} \quad v^T P v + c^T v \\ &\text{subject to } p_i(x, v) \in \Sigma[x] \;\; i = 1, ..., N. \end{aligned}$$

其中，$P \in \mathbb{R}^{l\times l}$ 是半正定矩阵，$c \in \mathbb{R}^l$是向量，而 $p_i(x, v) \in \Sigma[x]$ 是关于 $x \in \mathbb{R}^n$ 的 SOS 多项式，由参数 $v \in \mathbb{R}^l$ 确定，即它可以展开为

$$p_i(x, v) = Z_i(x)^T Q_i(v) Z_i(x), \tag{3}$$

其中，$Q_i(v) = Q_{0,\,i} + v_1 Q_{1,\,i} + ... + v_l Q_{1,\,l}$ 是一个方阵，其元素为半正定矩阵，且由 $v$ 线性参数化，用于编码多项式的系数。

在以下小节推导 SOS 约束时，为简便起见省略参数 $v$，并将多项式简写为 $p_i(x)$。

### *2.3* 正定性定理

下面我们介绍一种正定理的形式，可将其视为弱正定理的特例。该形式将在后续分析中发挥作用。

定理 *1.* 给定多项式 $f_1(x)$、$f_2(x)$、$g_1(x)$、$h_1(x)$、...、$h_{m+1}(x) \in R[x]$，使得

$$\begin{aligned} \Big\{x \in \mathbb{R}^n \,|& f_1(x) \geq 0, f_2(x) \geq 0, g_1(x) \neq 0, \\ & h_1(x) = 0, ..., h_{m+1}(x) = 0\Big\} = \emptyset, \end{aligned} \tag{4}$$

那么存在多项式

- $f_{\text{cone}}(x) = s_1(x)f_1(x) + s_2(x)f_2(x) + s_3(x)f_1(x)f_2(x)$
- $h_{\text{ideal}}(x) = p_1(x)h_1(x) + ... + p_{m+1}(x)h_{m+1}(x)$

使得

$$-f_{\text{cone}}(x) - h_{\text{ideal}}(x) - g_1(x)^{2k} \in \Sigma[x] \tag{5}$$

对于所有 $x \in \mathbb{R}^n$，其中 $p_1(x) ..., p_{m+1}(x) \in R[x]$ 是多项式，$s_1(x)$ $s_2(x)$ $s_3(x) \in \Sigma[x]$ 是平方和多项式，并且$k \in \mathbb{N}_+$。

**证明**。使用（Bochnak 等人，2013，定理 4.4.2），我们注意到，锥 $P$ 由 $f_1(x)$ 和$f_2(x)$ 生成，并包含于 $\Big\{s_0(x) + s_1(x)f_1(x) + s_2(x)f_2(x) + s_3(x)f_1(x)f_2(x) \mid s_0,\ s_1,\ s_2,\ s_3 \in \Sigma[x]\Big\}$。 ■

如果集合 (4) 为空，正定理可确保存在多项式 $s_1(x)$ $s_2(x) ..., p_{m+1}(x)$，但不指定它们的次数。计算求得这些多项式时，需要逐步提高多项式的次数，直至能够为 (5) 找到解。然而，提高多项式的次数会使计算时间变长。不过，对于许多值得关注的实际示例，低次多项式就足够了。

以下定理给出了正定理的一种形式，可视为 Putinar 正定理的特例。

定理 *2*。给定多项式 $f_1(x)$、$f_2(x)$、$h_1(x)$、...、$h_{m+1}(x) \in R[x]$，使得集合 $\Big\{x \in \mathbb{R}^n \mid f_2(x) \geq 0\Big\}$ 是紧的，并且

$$\begin{aligned} \Big\{x \in \mathbb{R}^n \,|& f_1(x) \geq 0, f_2(x) \geq 0, \\ & h_1(x) = 0, ..., h_{m+1}(x) = 0\Big\} = \emptyset, \end{aligned} \tag{6}$$

那么存在多项式

- $f_{\text{cone}}(x) = f_1(x) + s_2(x)f_2(x)$
- $h_{\text{ideal}}(x) = p_1(x)h_1(x) + ... + p_{m+1}(x)h_{m+1}(x)$

使得

$$-f_{\text{cone}}(x) - h_{\text{ideal}}(x) \in \Sigma[x] \tag{7}$$

对所有 $x \in \mathbb{R}^n$，其中 $p_1(x) ..., p_{m+1}(x) \in R[x]$ 是多项式，且 $s_2(x) \in \Sigma[x]$ 是 SOS 多项式。

**证明**。考虑闭半代数集

$$\begin{aligned} K := \Big\{x \in \mathbb{R}^n \,|& f_2(x) \geq 0, \\ & h_k(x) \geq 0 \quad k = 1, ..., m+1, \\ & -h_k(x) \geq 0 \quad k = 1, ..., m+1\Big\}, \end{aligned}$$

以及二次模

$$M(f_2(x), h_1(x), -h_1(x), ..., -h_{m+1}(x)),$$

那么，根据（Laurent，2009，定理 3.17），利用 $\{x \in \mathbb{R}^n \mid f_2(x) \geq 0\}$ 是紧集这一事实， $M$ 是阿基米德的。空集条件 (6) 等价于 $-f_1(x) > 0$ 在 K 上，因此，$K$ 等价于

$$-f_1(x) = s_0(x) + s_2(x)f_2(x) + \sum_{k=1}^{m+1} \left(s_{h,k}^p(x) - s_{h,k}^n(x)\right) h_k(x), \tag{8}$$

其中 $s_0(x)$ $s_2(x)$ $s_{h,1}^p(x)$ ……， $s_{h,m+1}^n(x) \in \Sigma[x]$ (Laurent, 2009, 定理 3.20)。注意， $s_{h,k}^p(x) - s_{h,k}^n(x)$可以替换为多项式 $p_k(x)$。

■

## 3. 闭环稳定性与安全性

本节首先回顾控制李雅普诺夫函数和控制屏障函数的概念。对于给定的多项式系统 (1)，我们随后提出一种方法，通过控制李雅普诺夫函数和控制屏障函数条件的平方和表述构造有理控制器。我们证明，在采用此控制器时，所得闭环系统 (2) 具有全局渐近稳定性和安全性。

### *3.1* 控制李雅普诺夫函数

要使控制系统 (1) 在平衡点 $x^*$附近可稳定化，我们要求在每个状态 $x \in \mathbb{R}^n$ 处都存在一个控制输入 $u_c \in \mathbb{R}^m$ ，使标量多项式 $V(x) \in R[x]$ 的次水平集正向不变。这引出了控制李雅普诺夫函数的如下定义：

定义 *1.* (Isidori（1995）) 考虑一个可微函数 $V$ ： $\mathbb{R}^n \to \mathbb{R}$ ，使得对所有 $x \neq x^*$， $V(x^*) = 0$ 且 $V(x) > 0$ 。若满足以下条件，则称该标量函数为控制系统 (1) 的控制李雅普诺夫函数 (CLF)：

$$\nabla V(x)^T f(x) < 0 \tag{9}$$

$$\text{for all } x \in \left\{x \in \mathbb{R}^n \mid \nabla V(x)^T G(x) = 0, x \neq x^*\right\}.$$

不等式 (9) 也可以表述为空集条件

$$\left\{x \in \mathbb{R}^n \mid \nabla V(x)^T f(x) \geq 0, \nabla V(x)^T G(x) = 0, x \neq x^*\right\} = \emptyset. \tag{10}$$

将条件 (10) 限定在多项式 控制李雅普诺夫函数上时，可以通过平方和规划求解。因此，在本文余下部分中，我们将注意力限制在多项式标量函数 $V \in R[x]$ 上。

接下来，我们用一个不等式约束 $x = (x_1, \ldots, x_n) \neq x^*$ 替换 (10) 中的不等式约束 $l(x) \neq 0$ （参见 Tan 和 Packard (2004)），其中 $l(x^*) = 0$ ，而 $l(x) \neq 0$ 在其他情况下。单个不等式约束的优点是，它可以转化为更简单的 SOS 约束。由此得到的与 (10) 等价的空集条件如下：

$$\left\{x \in \mathbb{R}^n \mid \nabla V(x)^T f(x) \geq 0, \nabla V(x)^T G(x) = 0, l(x) \neq 0\right\} = \emptyset. \tag{11}$$

根据定理 1，通过选择 $k = 1$，空集条件 (11) 变为

$$-s_1(x)\nabla V(x)^T f(x) - \nabla V(x)^T G(x)p(x) - l(x)^2 \in \Sigma[x], \tag{12}$$

其中， $s_1(x) \in \Sigma[x]$ 是一个 SOS 多项式，而 $p(x) = [p_1(x) \ \ldots \ p_m(x)]^T \in R[x]^m$ 是一个多项式向量。

如果 $s_1(x)$ 关于 $x^*$ 严格为正，则由 SOS 约束 (12) 可自然得到以下控制器，其形式为有理函数：

$$u_{CLF}(x) := p(x)/s_1(x). \tag{13}$$

注 *1.* 将式 (1) 中的控制器固定为 $u_c = u_{CLF}(x)$，可得到闭环系统 (2)。因此，李雅普诺夫函数 $V(x)$ 的存在性保证了稳定性。不过，为便于阅读，我们仍将 $V(x)$ 称为控制李雅普诺夫函数。

引理*1.* 给定控制系统 (1) 的 CLF $V(x)$，且式 (12) 中的 $s_1(x)$ 严格为正，则闭环系统 (2) 使用式 (13) 中定义的控制器 $u(x) = u_{CLF}(x)$ 时全局渐近稳定 (GAS)。

**证明**。 多项式控制李雅普诺夫函数 $V(x)$ 关于 $x^*$ 严格正定。因此， $V(x)$ 是径向无界的。由 (12)，可得 $s_1(x)\nabla V(x)^T f(x) + \nabla V(x)^T G(x)p(x) < 0$对所有 $x \neq x^*$ 均成立。由于 $s_1(x)$ 关于 $x^*$ 严格为正，除以 $s_1(x)$ 后得到 $\nabla V(x)^T(f(x) + G(x)u_{CLF}(x)) < 0$ 对所有 $x \neq x^*$ 均成立。由 Khalil (2002) 中的定理4.2可知系统全局渐近稳定。■

对 $s_1(x)$ 的严格正性条件可以通过以下 SOS 约束来强制实现

$$s_1(x) - \epsilon_{s1} \in \Sigma[x], \tag{14}$$

对于某个 $\epsilon_{s1} > 0$。

### *3.2* 控制障碍函数

与稳定性论证类似，可以通过存在一个标量函数 $B_1$ ： $\mathbb{R}^n \to \mathbb{R}$来断言安全性。具体而言，若安全状态集合

$$\mathcal{X}_{s,1} = \left\{x \in \mathbb{R}^n \mid B_1(x) \leq 0\right\}, \tag{15}$$

对于每个状态$x \in \mathbb{R}^n$ ，若存在控制输入 $u_c \in \mathbb{R}^m$ 使得 $\mathcal{X}_{s,1}$ 是正向不变的，则该控制系统 (1) 相对于该集合是安全的。这引出了以下控制屏障函数的定义（参见 Wang 等人 (2022)）：

定义 *2.* 考虑一个可微函数 $B_1$： $\mathbb{R}^n \to \mathbb{R}$ ，使得 $\mathcal{X}_{s,1}$ 非空。若满足以下条件，则称该标量函数为控制系统 (1) 的控制屏障函数（控制屏障函数）：

$$\nabla B_1(x)^T f(x) < 0 \tag{16}$$

$$\text{for all } x \in \left\{x \in \mathbb{R}^n \mid \nabla B_1(x)^T G(x) = 0, B_1(x) = 0\right\}.$$

控制屏障函数（CBF）的另一种定义（参见 Ames 等人（2019））涉及上确界和 K类函数。然而，该定义并不适用于多项式优化，因为由此得到的函数无法直接转化为多项式不等式。与控制李雅普诺夫函数（CLF）类似，本文余下部分将重点讨论多项式 控制屏障函数 $B_1(x) \in R[x]$。

不等式（16）也可以表述为空集条件

$$\Big\{x \in \mathbb{R}^n \,|\nabla B_1(x)^T f(x) \geq 0, \nabla B_1(x)^T G(x) = 0, \quad B_1(x) = 0\Big\} = \emptyset. \tag{17}$$

假设 *1*。集合 $\{x \in \mathbb{R} \mid B_1(x) = 0\}$ 是紧集。

根据定理 2 和假设 1，空集条件 (17) 等价于 SOS 约束

$$-s_1(x)\nabla B_1(x)^T f(x) - \nabla B_1(x)^T G(x)p(x) - p_{m+1}(x)B_1(x) \in \Sigma[x], \tag{18}$$

其中，$s_1(x) \in \Sigma[x]$是 SOS 多项式，$p(x) = [p_1(x) \ \dots \ p_m(x)]^T \in (R[x])^m$ 是多项式向量，而 $p_{m+1}(x) \in R[x]$ 是标量多项式。

若 (18) 中的 $s_1(x)$ 关于 $x^*$ 严格为正，则存在一个由有理函数定义的控制器

$$u_{\mathrm{CBF},1}(x) := p(x)/s_1(x). \tag{19}$$

注 *2*。与注 1 中一样，我们将 $B(x)$ 记为控制系统和闭环系统的控制屏障函数。

引理 *2*。给定控制系统 (1) 的控制屏障函数 $B_1(x)$，且 (18) 中的 $s_1(x)$ 严格为正，则 (15) 中的集合 $\mathcal{X}_{s,\,1}$ 若为紧集（参见假设 1），则在系统 (2) 下，使用由 (19) 定义的控制器 $u(x) = u_{\mathrm{CBF},\,1}(x)$ 时，该集合正向不变。

证明。若满足条件 $\nabla B_1(x)^T(f(x) + G(x)u(x)) < 0$ 对所有 $x \in \{x \in \mathbb{R}^n \mid B_1(x) = 0\}$（Blanchini，1999年，Nagumo 定理 $\nabla B_1\ x\ ^T G\ x\ p\ x\ < 0 \quad x \in \{x \in \mathbb{R}^n \mid B_1\ x\ = 0\}$ $3\ 1 \quad 18 \quad s_1\ x\ \nabla B_1\ x\ ^T f\ x\ +.$），则标量多项式$B_1(x)$的紧零次水平集相对于控制为 $u(x)$ 的闭环系统 (2) 正向不变。由 ( ) 可得 ( ) ( ) ( ) ( ) ( )，对所有 ( ) 均成立。由于 $s_1(x)$ 相对于 $x^*$ 严格为正，除以 $s_1(x)$ 后得到 $s_1(x)\nabla B_1(x)^T(f(x) + G(x)u_{\mathrm{CBF},1}(x)) < 0$，对所有 $x \in \{x \in \mathbb{R}^n \mid B_1(x) = 0\}$均成立。

■

## 4. 兼容控制李雅普诺夫函数与多个控制屏障函数的 SOS 构造

本节旨在寻找彼此兼容的控制李雅普诺夫函数和多个控制屏障函数。为此，我们推导一组新的 SOS 约束，这些约束是通过要求有理控制器 (13) 和 (19) 相同而得到的。

多个控制屏障函数提供了额外的灵活性，可增大安全集 $\mathcal{X}_s$ 的体积。该安全集定义为所有控制屏障函数的零次水平集的交集。直观而言，这是合理的，因为令所有控制屏障函数相等，即可恢复单个控制屏障函数的解。

### *4.1* 多个控制屏障函数

考虑一组控制屏障函数 $\{B_i(x)\}_{i\in[t]}$，其中某个 $t \in \mathbb{N}_+$ 定义了一个正向不变集

$$\mathcal{X}_{s,i} = \Big\{x \in \mathbb{R}^n \mid B_i(x) \leq 0\Big\}. \tag{20}$$

如果多个控制屏障函数具有控制共享性质（参见 Grammatico 等人（2013）），则它们彼此兼容。

定义 *3*。考虑带有反馈控制器 $u(x)$ 的闭环系统（2）。一组控制屏障函数 $\{B_i(x)\}_{i\in[t]}$ 及其相关控制器 $u_{\mathrm{CBF},\,i}(x)$ 使（20）保持不变，若对于所有 $i \in [t]$，$u(x) = u_{\mathrm{CBF},\,i}(x)$，则称其具有控制共享性质。

兼容的控制屏障函数集合 $\{B_i(x)\}_{i\in[t]}$ 构成一个新的正向不变集 $\mathcal{X}_s$，称为安全集，其定义为各个 $B_i(x)$ 对应的不变集的交集：

$$\mathcal{X}_s := \bigcap_{i\in[t]} \mathcal{X}_{s,i} = \Big\{x \in \mathbb{R}^n \mid B_i(x) \leq 0 \ \forall i \in [t]\Big\}. \tag{21}$$

考虑如下形式的允许状态集

$$\mathcal{X}_a = \Big\{x \in \mathbb{R}^n \mid w_i(x) \leq 0 \ \forall i \in [t]\Big\}, \tag{22}$$

其中 $w_1(x) \dots, \ w_t \in R[x]$ 为多项式。注意，对于定义允许集 $\mathcal{X}_a$ 的每个多项式 $w_i(x)$，都可以为其分配一个相应的控制屏障函数。

假设 *2*。集合 $\{x \in \mathbb{R} \mid w_i(x) \leq 0\}_{i\in[t]}$ 是紧集。

为了得出安全性，所得不变集 $\mathcal{X}_s$ 需要包含于允许集内，即 $\mathcal{X}_s \subseteq \mathcal{X}_a$。这可以通过将每个控制屏障函数的不变集限制为相应 $w_i(x)$ 的零次水平集来实现，具体如以下引理所述。

引理 *3*。考虑一组控制屏障函数 $\{B_i(x)\}_{i\in[t]}$，其关于输入 $u(x)$ 具有控制共享性质。如果对于每个 $i \in [t]$

$$\mathcal{X}_{s,i} \subseteq \Big\{x \in \mathbb{R}^n \mid w_i(x) \leq 0\Big\}, \tag{23}$$

那么使用控制器 $u(x)$ 的闭环系统 (2) 关于安全集 $\mathcal{X}_s$ (21) 和允许集 $\mathcal{X}_a$ (22) 是安全的。

**证明**。对于闭环系统而言，要保证其安全性，我们需要证明 $\mathcal{X}_s$ 是 (i) 正向不变的，以及 (ii) $\mathcal{X}_s \subseteq \mathcal{X}_a$。注意，由于 (23)，集合 $\mathcal{X}_{s,i}$, $i \in [t]$ 是紧集。根据引理 2，每个 $\mathcal{X}_{s,i}$, $i \in [t]$ 相对于闭环系统都是正向不变的，而正向不变集的交集也正向不变。这证明了条件 (i)。条件 (ii) 可由以下内容看出

$$\mathcal{X}_s = \bigcap_{i\in[t]} \mathcal{X}_{s,i} \subseteq \bigcap_{i\in[t]} \Big\{x \in \mathbb{R}^n \mid w_i(x) \leq 0\Big\} = \mathcal{X}_a.$$

■

最后，我们将 (23) 形式化为空集条件

$$\Big\{x \in \mathbb{R}^n \mid w_i(x) \geq 0, -B_i(x) \geq 0\Big\} = \emptyset \tag{24}$$

对所有 $i \in [t]$ 均成立。在定理 2 和假设 2 下，条件 (24) 等价于以下 SOS 条件：

$$B_i(x) - s_{4,i}(x)w_i(x) \in \Sigma[x] \tag{25}$$

其中，对于所有 $i \in [t]$，$s_{4,i} \in \Sigma[x]$ 是一个 SOS 多项式。

### *4.2 Compatible CLF and CBFs*

通过以下控制共享性质，控制李雅普诺夫函数与多个控制屏障函数彼此兼容；二者同时存在，即可统一稳定性与安全性的概念。

定义 *4.* 考虑具有反馈控制器 $u(x)$ 的闭环系统 (2)。控制李雅普诺夫函数及其相关控制器

$u_{\text{CLF}}(x)$ 将系统稳定到 $x^*$，以及一组控制屏障函数 $\{B_i(x)\}_{i\in[t]}$ 及其相关控制器 $u_{\text{CBF},\,i}(x)$ 使 (20) 保持不变。若对所有 $i \in [t]$，$u(x) = u_{\text{CLF}}(x)$ 且 $u(x) = u_{\text{CBF},\,i}(x)$，则它们具有控制共享性质。

有了定义4，我们现在可以陈述主要命题：

命题 *4.* 给定一个控制李雅普诺夫函数 $V(x)$ 和一组控制屏障函数$\{B_i(x)\}_{i\in[t]}$，它们与输入 $u(x)$ 具有控制共享性质。如果条件 (23) 对每个 $i \in [t]$都成立，则输入 $u(x)$ 的闭环系统 (2) 是渐近稳定的，并且相对于允许集 $\mathcal{X}_a$ (22) 是安全的。

**证明.** 引理1和引理 3证明了使用输入 $u(x)$ 的闭环系统的稳定性和安全性。

■

根据条件 (12)、(14)、(18) 和 (25)，引理4中的稳定性与安全性条件可改写为以下 SOS 约束：

$$\begin{aligned}
&-\nabla V(x)^T\left(s_1(x)f(x)+G(x)p(x)\right)-l(x)^2 \\
&\quad\in\Sigma[x], \\
&-\nabla B_i(x)^T\left(s_1(x)f(x)+G(x)p(x)\right) \\
&\quad -p_{m+1,i}(x)B_i(x)\in\Sigma[x], && \forall i\in[t] \\
&B_i(x)-s_{4,i}(x)w_i(x)\in\Sigma[x], && \forall i\in[t] \\
&s_1(x)-\epsilon_{s1}\in\Sigma[x],
\end{aligned} \tag{26}$$

其中，$s_1(x)s_{4,i}(x) \in \Sigma[x]$ 是平方和多项式，$p(x) = [p_1(x)\ \ldots\ p_m(x)]^T \in (R[x])^m$ 是多项式向量，而 $p_{m+1,i} \in R[x]$ 是标量多项式。

由此得到的控制输入 $u(x) = p(x)/s_1(x)$ 使系统在 $\mathcal{X}_a$ 上渐近稳定，且相对于允许集 $\mathcal{X}_a$ 和安全集 $\mathcal{X}_s$ 保持安全。

## 5. SOS规划问题

在本节中，我们陈述用于为控制系统 (1) 求解一个控制李雅普诺夫函数和多个控制屏障函数的 SOS 程序。该 SOS 程序在满足 SOS 条件 (26) 的约束下，最大化 $\mathcal{X}_s$体积的替代指标。

### *5.1* 代价函数

理想情况下，SOS 程序将在安全集 $\mathcal{X}_s$ 的体积上优化控制李雅普诺夫函数和控制屏障函数：

$$\begin{aligned}
\underset{v}{\text{minimize}} \quad & -vol\left(\left\{x\in\mathbb{R}^n \mid B_i(x,v)\le 0 \text{ for } i\in[t]\right\}\right) \\
\text{subject to} \quad & \text{SOS constraints (26)}
\end{aligned} \tag{27}$$

其中，$B_i(x,\ v) = Z_{B,\ i}(x)^T Q_{B,\ i}(v) Z_{B,\ i}(x)$（参见式（3））。为了使式（27）在计算上具有可处理性，采用编码 $B_i(x)$ 的二次矩阵的迹作为 $\mathcal{X}_s$ 体积的替代指标。由此，代价变为：

$$c_1(v) = \sum_{i=1}^{t} tr(Q_{B,i}(v)). \tag{28}$$

此外，我们通过实验观察到，将控制屏障函数限制在预先定义的中心点上可以改善结果。其理由在于，只有当 $B(x) = 0$ 时，控制屏障函数条件才有效因此，当 $B(x) \neq 0$ 时，并不会限制控制屏障函数取任意大的值。

我们为每个控制屏障函数 $B_i(x)$ $i \in [t]$ 定义一个中心点 $(x_{c,\ i},\ B_{c,\ i}) \in (\mathbb{R}^n,\ \mathbb{R})$。然后，我们根据与该点的偏差定义代价函数：

$$c_2(v) = \sum_{i=1}^{t} (B_i(x_{c,i}) - B_{c,i})^2 \tag{29}$$

SOS 程序的代价函数设为

$$c(v) = c_1(v) + c_2(v). \tag{30}$$

### *5.2 SOS* 程序

给定代价 (30) 和 SOS 约束 (26)，我们将 SOS 程序概括如下：

$$\begin{aligned}
\text{find} \quad & V, B_i, s_1, s_{4,i}, p_{m+1,i}\in R[x], \\
& p\in(\mathbb{R}[x])^m \\
\text{minimize} \quad & \sum_{i=1}^{t} tr(Q_{B,i}) + (B_i(x_{c,i}) - B_{c,i})^2 \\
\text{subject to} \quad & -\nabla V^T(s_1 f + Gp) - l^2 \in \Sigma[x] \\
& -\nabla B_i^T(s_1 f + Gp) - p_{m+1,i}B_i \\
& \quad \in \Sigma[x] && \forall i\in[t] \\
& B_i - s_{4,i}w_i \in \Sigma[x] && \forall i\in[t] \\
& s_1 - \epsilon_{s1} \in \Sigma[x] \\
& s_{4,i} \in \Sigma[x] && \forall i\in[t] \\
& V\in\Sigma[x], V(x^*) = 0
\end{aligned}$$

其中，$(x_{c,\ i},\ B_{c,\ i}) \in (\mathbb{R}^n,\ \mathbb{R})$ 是为每个控制屏障函数预先定义的中心点。

## 6. ALGORITHM

由于决策变量中存在双线性项，第 5.2 节中的 SOS 问题无法直接转化为半定规划问题。然而，交替算法可以为这个非凸问题找到足够好的解。

### *6.1* 交替算法

考虑一个抽象优化问题，其包含由 $f_1(v_1,\ v_2) \in (R[x])^{n_c}$定义的 $n_c$ 个 SOS 不等式约束，这些约束关于决策变量 $v_1 \in \mathbb{R}^{n_1}$ 和 $v_2 \in \mathbb{R}^{n_2}$ 是双线性的，且代价函数 $f_0(v_1) \in \mathbb{R}$ 仅依赖于 $v_1$：

$$\begin{aligned}
\underset{v_1,v_2}{\text{minimize}} \quad & f_0(v_1) \\
\text{subject to} \quad & f_1(v_1,v_2)\in(\Sigma[x])^{n_c}
\end{aligned} \tag{31}$$

优化问题 (31) 无法直接通过 SOS 程序求解。为此，我们提出一种方法：固定一个变量，同时对另一个变量进行搜索，并在二者之间交替进行。

从一次迭代 $v_1^{(0)}$ 和 $v_2^{(0)}$ 开始，第 $k$ 次迭代的算法定义如下：

- Step 1: Substitute $v_2 = v_2^{(k-1)}$ and solve for $v_1$

$$\begin{aligned}
\underset{v_1}{\text{minimize}} \quad & f_0(v_1) \\
\text{subject to} \quad & f_1(v_1, v_2^{(k-1)})\in(\Sigma[x])^{n_c}
\end{aligned} \tag{32}$$

- 步骤 2：代入 $v_1 = v_1{}^k$，并求解 $\epsilon$ 和 $v_2$

$$\begin{aligned}
\underset{\epsilon,v_2}{\text{minimize}} \quad & \epsilon \\
\text{subject to} \quad & f_1(v_1^{(k)}, v_2) + s(v_1^{(k)})\epsilon\in(\Sigma[x])^{n_c}
\end{aligned} \tag{33}$$

当 $|f_0(v_1^{(k)}) - f_0(v_1^{(k-1)})|$ 低于给定阈值时，算法终止。

引理 *5.* 如果初始对 $(v_1^{(0)}, v_2^{(0)})$ 满足 (31) 的可行性，那么在交替算法的过程中，可行性始终得以保持，并且 (32) 中的代价 $f_0(v_1^{(k)})$ 随迭代 $k$ 非增。

**证明**。首先，我们证明：如果迭代 $k$ 中的步骤 1 对 (32) 可行，那么迭代 $k$ 中的步骤 2 对 (33) 也必然可行。为满足可行性，我们只需找到一个点 $(v_2, \epsilon)$，使步骤 2 中的约束成立。由步骤 1 可知，$f_1(v_1^{(k)}, v_2^{(k-1)})$ 是一个 SOS 多项式。因此，$(v_2, \epsilon) = (v_2^{(k-1)}, 0)$ 是 (33) 的一个可行点。因此，$\epsilon$ 必须小于或等于零，即 $\epsilon \leq 0$。

接下来，我们证明迭代 $k$ 的解是迭代 $k+1$中步骤 1的一个可行点。首先，注意到 $-s(v_1^{(k)})\epsilon$是一个 SOS 多项式向量，因为 $\epsilon \leq 0$。但于是$f_1(v_1^{(k)}, v_2^{(k)})$也是一个 SOS 多项式。因此，$(v_1^{(k)}, v_2^{(k)})$是迭代 $k$中步骤 1的一个可行点。综上，我们证明了 $(v_1^{(k)}, v_2^{(k)})$对 (31) 仍然可行。因此，$f_0(v_1^{(k+1)})$小于或等于 $f_0(v_1^{(k)})$。

■

### *6.2* 主算法

主算法通过引入一个运行区域 $\mathcal{X}_{op}$来改善可行性。在假设 2下，存在一个标量多项式 $r(x)$，其零次水平集是紧集，且严格包含允许集 $\mathcal{X}_a$，即

$$\mathcal{X}_s \subseteq \mathcal{X}_a \subseteq \mathcal{X}_{op} := \Big\{ x \in \mathbb{R}^n \mid r(x) \leq 0 \Big\}. \tag{34}$$

请注意，我们感兴趣的所有状态轨迹都包含在运行区域 $\mathcal{X}_{op}$内。我们将利用这一事实，对前几章中提出的空集条件及其对应的 SOS 约束进行细微调整。

给定一个由 $w_i(x)$ 定义的允许集 $\mathcal{X}_a$，对于 $i \in [t]$ (22) 的正定 SOS 多项式 $l(x) \in \Sigma[x]$、一个正标量 $\epsilon_{s1} > 0$、每个 $i \in [t]$对应的中心点 $(x_{c,\,i},\ B_{c,\,i}) \in (\mathbb{R}^n,\ \mathbb{R})$、由 $r(x)$ (34) 定义的运行区域，以及初始值 $s_1(x)$、$p(x)$ 和 $p_{m+1,\,i}(x)$（针对 $i \in [t]$），主 SOS 算法总结如下。

- Step 1: Given a controller $u = p/s_1$ find a CLF and multiple CBFs.

$$\begin{array}{lll}
\text{查找} & V, B_i, s_2, s_{3,i}, s_{4,i} \in R[x] & \\
\text{保持 fixed} & s_1, p_{m+1,i} \in R[x], p \in (\mathbb{R}[x])^m & \\
\text{minimize} & \sum_{i=1}^{t} tr(Q_{B,i}) + (B_i(x_{c,i}) - B_{c,i})^2 & \\
\text{subject to} & -\nabla V^T (s_1 f + Gp) - l^2 + s_2 r \in \Sigma[x] & \\
& -\nabla B_i^T (s_1 f + Gp) - p_{m+1,i} B_i & \\
& \quad + s_{3,i} r \in \Sigma[x] & \forall i \in [t] \\
& B_i - s_{4,i} w_i \in \Sigma[x] & \forall i \in [t] \\
& s_2 \in \Sigma[x] & \\
& s_{3,i} \in \Sigma[x] & \forall i \in [t] \\
& s_{4,i} \in \Sigma[x] & \forall i \in [t] \\
& V \in \Sigma[x], V(x^*) = 0 &
\end{array}$$

- 步骤 2：给定一个控制李雅普诺夫函数和多个控制屏障函数，求一个控制器 $u = p/s_1$。

$$\begin{array}{lll}
\text{find} & \epsilon_i \in \mathbb{R}, s_1, s_2, s_{3,i}, p_{m+1,i} \in R[x], & \\
& p \in (\mathbb{R}[x])^m & \\
\text{hold fixed} & V, B_i \in R[x] & \\
\text{minimize} & \sum_{i=1}^{t} \epsilon_i & \\
\text{subject to} & -\nabla V^T (s_1 f + Gp) - l^2 + s_2 r \in \Sigma[x] & \\
& -\nabla B_i^T (s_1 f + Gp) - p_{m+1,i} B_i & \\
& \quad + s_{3,i} r + \epsilon_i \in \Sigma[x] & \forall i \in [t] \\
& s_1 - \epsilon_{s1} \in \Sigma[x] & \\
& s_2 \in \Sigma[x] & \\
& s_{3,i} \in \Sigma[x] & \forall i \in [t]
\end{array}$$

## 7. 仿真

第 6.2 节中的交替算法被应用于一个直流/交流电力变换器模型的三维向量场实现。对于这一应用而言，能够避免不安全状态的反馈控制器至关重要，因为所有电气变量（例如电压和电流）都需要始终受到约束。以直流电压和交流电流作为状态，控制系统 (1) 定义为

$$f(x) = \begin{bmatrix} -0.05x_1 - 57.9x_2 + 0.00919x_3 \\ 1710x_1 + 314x_3 \\ -0.271x_1 - 314x_2 \end{bmatrix} \tag{35}$$

and

$$G(x) = \begin{bmatrix} 0.05 - 57.9x_2 & -57.9x_3 \\ 1710 + 1710x_1 & 0 \\ 0 & 1710 + 1710x_1 \end{bmatrix}. \tag{36}$$

允许集 $\mathcal{X}_a$ 由两个多项式 $w_1(x)$ 和 $w_2(x)$ 定义，分别表示电流和电压的状态约束：

$$\begin{aligned} w_1(x) &= (x_1 + 0.3)^2 + (x_2/20)^2 + (x_3/20)^2 - 0.5^2 \\ w_2(x) &= (x_1/20)^2 + x_2^2 + x_3^2 - 1.2^2. \end{aligned} \tag{37}$$

式 (34) 中的运行区域 $\mathcal{X}_{op}$ 可参数化为：

$$r(x) = (x_1/0.8)^2 + (x_2/1.2)^2 + (x_3/1.2)^2 - 1.8. \tag{38}$$

更贴合允许集的 $r(x)$ 可使交替算法更快收敛。不过，为了便于说明，我们选择了一个次优运行区域。控制屏障函数的中心点为：$x_{c,\,i} = [-0.3\ 0\ 0]^T$ 和 $B_{c,\,i} = -10$ （$i \in [2]$）。算法中涉及的多项式次数总结如下：

| $V(x)$ | $B_i(x)$ | $s_1(x)$ | $s_2(x)$ | $s_{3,\,i}(x)$ | $s_{4,\,i}(x)$ | $p(x)$ | $p_{m+1,\,i}(x)$ |
|---|---|---|---|---|---|---|---|
| 4 | 4 | 2 | 6 | 6 | 3 | 4 | 2 |

对于 $V(x)$ 和 $B_i(x)$，奇数次系数设为零：

- $deg\,(V(x)) = (2, 4)$
- $deg\,(B_1(x - x_{c,1})) = (0, 2, 4)$
- $deg\,(B_2(x - x_{c,2})) = (0, 2, 4)$.

图 7 展示了算法每次迭代计算出的安全集序列。计算得到的控制李雅普诺夫函数和控制屏障函数不仅证明了稳定反馈控制器的存在，还保证这类控制器

能使系统相对于安全集 $\mathcal{X}_s$ 保持安全。通过使用两个控制屏障函数，算法最终得到一个与允许集 $\mathcal{X}_a$ 紧密贴合的安全集。

所得半定规划中涉及的变量和约束数量总结如下：

| | 步骤 1 | 步骤 2 |
|---|---|---|
| 变量 | 337 | 372 |
| 约束 | 5188 | 4892 |

该算法在 90 秒内完成。每次迭代中，半定规划算法平均需要 33 次内迭代来求解步骤 1，平均需要 21 次内迭代来求解步骤 2。

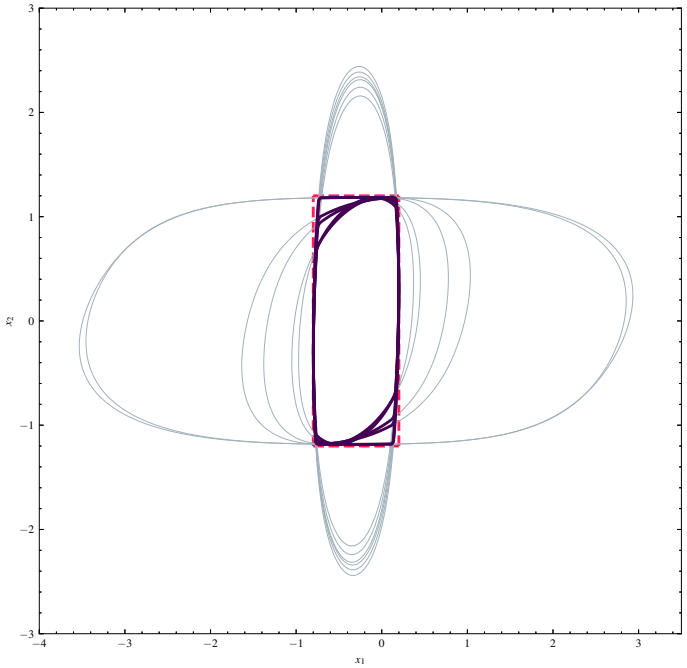

图 1. 展示了经过 6 次迭代后，安全集序列投影到 $x_1$ 和 $x_2$ 坐标上的演化。允许集 $\mathcal{X}_a$ 的边界以红色表示。控制屏障函数 $B_1(x)$ 和 $B_2(x)$ 的零水平集序列以灰色表示。

## 8. 结论与未来工作

本文提出了一个结合控制李雅普诺夫函数和多个控制屏障函数来实现稳定性与安全性条件的框架。通过采用正定理的两个版本，我们综合设计了一个控制器，用于证明控制李雅普诺夫函数与控制屏障函数之间的兼容性。随后，我们形式化了对兼容的控制李雅普诺夫函数和控制屏障函数条件进行编码的 SOS 约束。最后，我们提出了一种算法，通过迭代求解两个半定规划来求解由此得到的 SOS 程序。

未来的工作中，我们计划研究该算法的计算复杂度，并探索一个统一框架，在较弱的 CLF 和 CBF 条件下证明稳定性和安全性。此外，我们还计划将抗噪能力纳入平方和表述，正如 Kang 等人（2023）所展示的那样。

## 参考文献


Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). 控制屏障函数：理论与应用。载于 *2019*年第*18*届欧洲控制会议（*ECC*），3420–3431。IEEE。Anghel, M., Milano, F., and Papachristodoulou, A. (2013). 用于电力系统稳定性分析的李雅普诺夫函数算法构造。*IEEE* 电路与系统汇刊 *I*：常规论文，60(9)，2533–2546。Blanchini, F.（1999年）。控制中的集合不变性。*Automatica*，35(11)，1747–1767。Bochnak, J., Coste, M., and Roy, M.F. (2013). 实代数几何，第36卷。Springer Science & Business Media。Clark, A. (2021). 控制屏障函数的验证与综合。载于*2021*年第*60*届 *IEEE* 决策与控制会议（*CDC*），6105–6112。IEEE。Grammatico, S., Blanchini, F., and Caiti, A. (2013). 控制李雅普诺夫函数的共享与合并。*IEEE* 自动控制汇刊，59(1)，107–119。Isaly, A., Ghanbarpour, M., Sanfelice, R.G., and Dixon, W.E. (2022). 关于受约束微分包含中由多个控制屏障函数定义的反馈控制器的可行性与连续性。载于*2022*年美国控制会议（*ACC*），5160–5165。IEEE。Isidori, A. (1995). 非线性控制系统。Springer。Kang, S., Chen, Y., Yang, H., and Pavone, M. (2023). 鲁棒控制屏障函数的验证与综合：多层次多项式优化与半定松弛。*arXiv* 预印本 *arXiv:2303.10081*。Khalil, H. (2002). 非线性系统。Pearson Education。Prentice Hall。Korda, M., Henrion, D., and Jones, C.N. (2014). 多项式控制系统最大受控不变集的凸计算。*SIAM* 控制与优化杂志，52(5)，2944–2969。Kundu, S., Geng, S., Nandanoori, S., Hiskens, I.A., and Kalsi, K. (2019). 用于基于逆变器的微电网安全运行的分布式屏障证书。1042–1047。Laurent, M. (2009). 平方和、矩阵矩与多项式优化。代数几何的新兴应用，157–270。Tan, W. and Packard, A. (2004). 使用平方和规划搜索控制李雅普诺夫函数。*sibi*，1(1)。Tan, X. and Dimarogonas, D.V. (2022). 输入受限系统多个控制屏障函数的兼容性检查。*arXiv* 预印本 *arXiv:2209.02284*。Wang, H., Margellos, K., and Papachristodoulou, A. (2022). 输入约束系统的安全性验证与控制器综合。*arXiv* 预印本 *arXiv:2204.09386*。Wieland, P. and Allg¨ower, F. (2007). 使用控制屏障函数构造安全性。*IFAC* 论文集，40(12)，462–467。Xu, X. (2016). 控制共享屏障函数及其在受约束控制中的应用。载于*2016*年第*55*届 *IEEE* 决策与控制会议（*CDC*），4880–4885。IEEE。