2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Optimal Design of Control-Lyapunov Functions by Semi-Infinite Stochastic Programming*

Wentao Tang[1] and Prodromos Daoutidis[2]

***Abstract*— Lyapunov-based control is a common method to enforce closed-loop stability of nonlinear systems, where the choice of a control-Lyapunov function has a strong impact on the resulting performance. In this paper, we propose a generic semi-infinite stochastic programming formulation for the optimal control-Lyapunov function design problem and discuss its various specializations. Specifically, the expected performance evaluated on simulated trajectories under different scenarios is optimized subject to infinite constraints on stability and performance specifications. A stochastic proximal primal-dual algorithm is introduced to find a stationary solution of such a semi-infinite stochastic programming problem. The proposed method is illustrated by a chemical reactor case study.**

I. Introduction

For the stability of nonlinear control systems, Lyapunov stability analysis [1] is the canonical approach, aiming at finding a Lyapunov function and establishing its descent property. To enforce closed-loop stability proactively with a controller, one can assign a *control-Lyapunov function* and construct a *Lyapunov-based controller* that guarantees its descent [2]. For example, Lin and Sontag [3] gave an explicit control law that guarantees the Lyapunov function descent at a specified rate. In model predictive control (MPC) [4] where inputs are determined from the recursive solution of an optimal control problem, closed-loop stability can be established with the cost function as a control-Lyapunov function. Also, the Lyapunov function descent can be directly incorporated as a constraint in MPC formulations to enforce closed-loop stability [5]. This idea has been extensively used in Lyapunov-based MPC, EMPC, and Lyapunov barrier function-based methods [6]–[8].

However, developing systematic and generic approaches to the design of control-Lyapunov functions is a fundamentally challenging problem. Typically, special forms or restrictive conditions satisfied by the system dynamics are needed to construct control-Lyapunov functions [9]. For polynomial systems, sum-of-squares (SOS) programming approaches [10] and especially Lasserre hierarchy algorithms [11], [12] have been proposed, which usually require a simple objective of control-Lyapunov function design, e.g., to maximize the volume of an estimated domain of attraction. More generally, control-Lyapunov functions can be found by solving optimal control problems in the form of Hamilton-Jacobi-Bellman (HJB) equations [13] or alternatively, according to Rantzer's duality of Lyapunov stability [14], formulated as an optimization problem over Lyapunov measures [15], [16]. Finally, as universal approximators, neural networks can be used to learn the Lyapunov functions; on the other hands, neural networks typically involve an excessive number of parameters to tune and are subject to convergence issues and often difficult to guarantee global properties [17]–[19].

In this paper, we aim at providing a *generic formulation* for the control-Lyapunov function design as a *semi-infinite stochastic programming* problem. This formulation has the following technical features and practical advantages.

1) The control-Lyapunov function, or its indirect representation by the MPC cost function, are linearly parameterized, and the parameters become the decision variables in the optimization problem. Therefore, the dimension of the decision variables is dependent on the choice of the parameterization structure (e.g., quadratic or SOS) and the procedure is in principle more *scalable* with increasing dimension of the state space (than numerical approaches for the solution of optimal control problems [16]) as fewer parameters need to be determined.
2) Requirements on the rate of the Lyapunov function descent and/or the disturbance effect in the closed-loop system, as characterized by a gain function, are considered as an infinite number of inequality constraints to be satisfied for all states on a state space region. Hence, the user has the *flexibility* to specify the gain from disturbances and the rate of the Lyapunov function descent as any desired functions (as long as feasibility is retained), including the typical exponential decay, without being restricted to specific function structures (as in SOS programming [10]).
3) The objective function is postulated as the expectation of a performance cost function evaluated under a finite or infinite set of simulation scenarios. The user may choose scenarios that reflect the typical operating conditions of the system, and therefore give a *direct and practical* assessment of the quality of the control-Lyapunov function design according to user specifications. Moreover, in contrast to the typical quadratic cost used in optimal control problems [13], such a simulation-based objective function can account

*This work was supported by the faculty startup fund for Wentao Tang at NC State University.

[1]Wentao Tang is an Assistant Professor at the Department of Chemical and Biomolecular Engineering, North Carolina State University, Raleigh, NC 27695. Email: wtang23@ncsu.edu.

[2]Prodromos Daoutidis is a College of Science and Engineering Distinguished Professor of the Department of Chemical Engineering and Materials Science, University of Minnesota, Minneapolis, MN 55455, USA. Email: daout001@umn.edu

for a variety of dynamic performance characteristics, such as state and input magnitudes, smoothness, and overshooting.

The proposed formulation is largely motivated by the chemical engineering literature on the analysis and design of flexible plants that remain operable under uncertainty while achieving optimal economic performance. A review was given by Grossmann et al. [20]. Such a conceptual connection was drawn in our previous works [21], [22].

We refer the readers to [23] and [24] for tutorials on stochastic and semi-infinite programming, respectively. In this work, to handle the stochastic, simulation-based objective function and semi-infinite constraints involved in the control-Lyapunov function design problem, we propose to use a stochastic proximal primal-dual algorithm slightly modified from [25]. The algorithm is essentially driven by the data samples drawn from the probability distributions defining the semi-infinite programming problem throughout the iterations, and will be explained in §IV and demonstrated by a case study on a chemical reactor example in §V. Next, we introduce some preliminaries of Lyapunov stability theory (§II) that are needed for the proposed semi-infinite programming formulation (§III).

II. Preliminaries

We consider a continuous-time nonlinear control system:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + b(x(t))d(t) \qquad (1)$$

in which $x(t) \in \mathbb{R}^{n_x}$, $u(t) \in \mathbb{R}^{n_u}$ and $d(t) \in \mathbb{R}^{n_d}$ represent the states, control inputs, and exogenous inputs (disturbances) respectively. We assume that f, g, b are continuously differentiable and $f(0) = 0$.

A. Lyapunov stability

In the absence of exogenous inputs, for any control policy $u = \kappa(x)$, assuming continuous, the closed-loop stability is related to the existence of a Lyapunov function [2, §4.5].

Fact 1. *Let $X \subseteq \mathbb{R}^n$ be a closed set whose interior contains the origin, and κ be a given continuous control policy on X. If there exists a continuously differentiable function, called a Lyapunov function, $V : X \to \mathbb{R}$, such that*

$$w_1(\|x\|) \leq V(\|x\|) \leq w_2(\|x\|) \qquad (2)$$

and

$$\dot{V}(x) = \nabla V(x)^\top (f(x) + g(x)\kappa(x)) \leq -\sigma(\|x\|) \qquad (3)$$

hold for some $\mathcal{K}$-class functions w_1, w_2 and σ for all $x \in X$, then $x = 0$ is asymptotically stable.

If X is furthermore a (forward-)invariant set, then X is a domain of attraction under κ; otherwise, the domain of attraction can be estimated as the largest sublevel set contained in X, i.e., $S_V(v) = \{x | V(x) \leq v\}$ where v is the largest among all such choices. On the other hand, if one needs any trajectory starting on any subset X_0 of X to be attracted to the origin, then it suffices to let the above Lyapunov descent condition hold on the smallest sublevel set $S_V(v) \supseteq X_0$.

When the control law κ needs to be designed, a function $V : X \to [0, \infty)$, called *control-Lyapunov function*, can be artificially assigned, with a given rate of its descent $\sigma(\|x\|)$ imposed on the control law:

$$\nabla V(x)^\top (f(x) + g(x)\kappa(x)) \leq -\sigma(\|x\|). \qquad (4)$$

In this way, for any control policy κ, as long as (4) remains feasible for all $t > 0$ (i.e., is recursively feasible), the closed-loop stability is guaranteed with V being a Lyapunov function. Such a method is called *Lyapunov-based control*.

In the presence of exogenous inputs d, it is desirable that under the controller κ, the effect of d on the states x is small for the closed-loop system

$$\dot{x}(t) = f(x(t)) + g(x(t))\kappa(x(t)) + b(x(t))d(t). \qquad (5)$$

This property is known as *input-to-state stability* and characterized with a Lyapunov function (also called an input-to-state Lyapunov function) according to the following result of Sontag and Wang [26].

Fact 2. *Let $X \subseteq \mathbb{R}^n$ be a closed set containing the origin, and κ be a given continuous control policy on X. Suppose that there exists a continuously differentiable function $V : X \to \mathbb{R}$, satisfying*

$$w_1(\|x\|) \leq V(x) \leq w_2(\|x\|) \qquad (6)$$

for some $\mathcal{K}$-class functions w_1, w_2 and

$$\begin{aligned} \dot{V}(x) &= \nabla V(x)^\top (f(x) + g(x)\kappa(x) + b(x)d) \\ &\leq -\sigma(\|x\|) \text{ whenever } \|x\| \geq \zeta(\|d\|) \end{aligned} \qquad (7)$$

for some $\mathcal{K}$-class functions σ and ζ. Then the closed-loop system (5) is input-to-state stable, i.e., on any trajectory in X,

$$\|x(t)\| \leq \beta(\|x(0)\|, t) + \gamma\left(\max_{0 \leq \tau \leq t} \|d(\tau)\|\right), \qquad (8)$$

where β and γ belong to the $\mathcal{KL}$ and $\mathcal{K}$ classes, respectively.

As such, assuming that the exogenous disturbances d lie in a bounded set D, the Lyapunov function is decreasing outside of $\{x | \|x\| \leq M\}$, in which $M = \max_{d \in D} \zeta(\|d\|)$. If there is a sublevel set of the Lyapunov function V, $S_V(v)$, with $v \geq w_2(M)$, then $S_V(v)$ is an invariant set and hence any trajectory starting on $S_V(v)$ is attracted into $S_V(w_2(M))$.

The condition of Fact 2 is stronger than, and in the case of $\|d\| = 0$, implies the conditions in Fact 1. To achieve both asymptotic stability and (exogenous) input-to-state stability under control, one needs to design the controller which allows such a corresponding V that the two inequalities in Fact 2 are satisfied. Next we consider two classical examples for such Lyapunov-based controllers.

B. Lin-Sontag controller and MPC

We assume that the control inputs are constrained in $U = \{u|\, \|u\|_p \leq 1\}$, where $p \in [1, \infty]$. Given a control-Lyapunov function V, the "universal" control law of Lin and Sontag [3] is written as:

$$\kappa(x) = \begin{cases} -\kappa_0(x)(\mathbb{L}_g V(x))^\top, & \mathbb{L}_g V(x) \neq 0 \\ 0, & \mathbb{L}_g V(x) = 0 \end{cases} \quad (9)$$

in which

$$\kappa_0(x) = \frac{\mathbb{L}_f^* V(x) + \sqrt{(\mathbb{L}_f^* V(x))^2 + \|\mathbb{L}_g V(x)\|_q^4}}{\|\mathbb{L}_g V(x)\|_2^2 \left[1 + \sqrt{1 + \|\mathbb{L}_g V(x)\|_q^2}\right]}$$

with $\mathbb{L}_f^* V(x) = \nabla V(x)^\top f(x) + \rho V$ for some $\rho > 0$, $\mathbb{L}_g V(x) = \nabla V(x)^\top g(x)$, and $q \in [1, \infty]$ such that $1/p + 1/q = 1$. Consider $X = \{x|\mathbb{L}_f^* V(x) \leq \|\mathbb{L}_g V(x)\|_q\}$. Then, $\dot{V}(x) \leq -\rho V(x)$ holds for all $x \in X$, and therefore the largest sublevel set of V contained in X is guaranteed to be a domain of attraction under κ, on which the control-Lyapunov function is guaranteed to decay exponentially with rate ρ [5]. In the presence of exogenous disturbances, we should redefine $\mathbb{L}_f^* V(x)$ by appending an additional term of $\|\mathbb{L}_b V(x)\|\zeta^{-1}(\|x\|)$ and rewrite the input-to-state stability condition into

$$\begin{aligned} \forall x \in X, \mathbb{L}_f V(x) - \|\mathbb{L}_g V(x)\|_q \\ + \|\mathbb{L}_b V(x)\|\zeta^{-1}(\|x\|) + \rho V(x) \leq 0. \end{aligned} \quad (10)$$

Hence, $\dot{V} \leq -\rho V - \|\mathbb{L}_b V(x)\|\zeta^{-1}(\|x\|) + \mathbb{L}_b V(x) d$ does not exceed $-\rho V$ when $\|x\| \geq \zeta(\|d\|)$, thus satisfying the inequality in Fact 2.

In contrast to the explicitness of the Lin-Sontag formula, in MPC the control law $u = \kappa(x)$ is implicitly determined by the quantity $\hat{u}(0)$ as a part of the solution to the following optimal control problem:

$$\begin{aligned} V(x) = \min_{\hat{u}(\cdot), \hat{x}(\cdot)} & \int_0^T \ell(\hat{x}(t), \hat{u}(t)) dt + \ell_{\mathrm{f}}(\hat{x}(T)) \\ \text{s.t.}\ & \dot{\hat{x}}(t) = f(\hat{x}(t)) + g(\hat{x}(t))\hat{u}(t),\ t \in [0, T] \\ & \hat{x}(t) \in X,\ \hat{u}(t) \in U,\ \hat{x}(0) = x,\ \hat{x}(T) \in X_{\mathrm{f}}. \end{aligned} \quad (11)$$

Here, $\hat{x}$ and $\hat{u}$ are the predicted state and control input trajectories, ℓ, ℓ_{f} are the stage cost and terminal cost functions (assumed to be continuous), T is the length of the prediction horizon, and X, U, X_{f} are state, control input and terminal state constraints, respectively. We assume for convenience that the optimal control problems can be solved in continuous time without any error induced from discretization and numerical errors. We assume that the following regularity conditions are satisfied: (i) U, X and $X_{\mathrm{f}} \subseteq X$ are compact sets containing zero in their interiors; (ii) $\ell(0, 0) = 0$, $\ell_{\mathrm{f}}(0) = 0$; (iii) $\forall x \in X, \forall u \in U$, $\ell(x, u) \geq w(x)$ and $\ell_{\mathrm{f}}(x) \leq w_{\mathrm{f}}(x)$ for some $\mathcal{K}$-class functions w, w_{f}.

The asymptotic and input-to-state stability conditions of MPC [4] are implicitly related to a control-Lyapunov function, namely the optimal objective value $V(x)$ as a function of x. Specifically, if there exists an auxiliary control law $\kappa_{\mathrm{f}} : X_{\mathrm{f}} \to U$ such that $\mathbb{L}_{f+g\kappa_{\mathrm{f}}} \ell_{\mathrm{f}}(x) \leq -\ell(x, \kappa_{\mathrm{f}}(x))$, then starting from any initial point x such that (11) is feasible, asymptotic stability is achieved with $\dot{V}(x) < 0$ whenever $x \neq 0$. The MPC achieves (exogenous) input-to-state stability if we further have

$$\mathbb{L}_{f+g\kappa_{\mathrm{f}}} \ell_{\mathrm{f}}(x) + \|\mathbb{L}_b \ell_{\mathrm{f}}(x)\|\zeta^{-1}(\|x\|) + \ell(x, \kappa_{\mathrm{f}}(x)) \leq 0. \quad (12)$$

As such, the requirement on the control-Lyapunov function $V(x)$ for MPC is implicitly contained in the conditions for the stage cost function ℓ and terminal cost function ℓ_{f}.

III. Optimal control-Lyapunov function design

A. General formulation

Now we provide a general formulation for optimally designing control-Lyapunov functions. We first parameterize the control-Lyapunov function to be designed, $V : X \to [0, \infty)$, linearly with a vector of parameters $\theta \in \mathbb{R}^{n_\theta}$, i.e.,

$$V(x) = \omega(x)^\top \theta \quad (13)$$

for some function $\omega : X \to \mathbb{R}^{n_\theta}$, so that the optimization of function V is performed on finite dimensions. In MPC where the control-Lyapunov function is reflected by the stage cost ℓ and terminal cost ℓ_{f}, we consider

$$\ell(x) = \omega(x)^\top \theta, \ \ \ell_{\mathrm{f}}(x) = \omega_{\mathrm{f}}(x)^\top \theta. \quad (14)$$

The parameters θ can be constrained with an a priori range $\Theta \subseteq \mathbb{R}^{n_\theta}$, which specifies the shape that the control-Lyapunov function can have. For simplicity, we assume that the conditions on the control-Lyapunov function specified by the bounding functions w_1, w_2, w, w_{f} as well as the conditions that $\ell(0, 0) = 0$, $\ell_{\mathrm{f}}(0) = 0$ can be translated into $\theta \in \Theta$. Such a range Θ is also assumed to be simple in the sense that the projection operator $\mathrm{proj}_\Theta : \mathbb{R}^{n_\theta} \to \Theta$, $\theta \mapsto \arg\min_{\theta' \in \Theta} \|\theta - \theta'\|_2^2$ can be evaluated exactly.

The constraint on the control-Lyapunov function has been discussed previously in §II-B as (10) and (12). Both of these conditions can be compactly written as

$$\psi(x, \theta) \leq 0, \ \ \forall x \in X \quad (15)$$

with a continuous function $\psi : X \times \mathbb{R}^{n_\theta} \to \mathbb{R}$, which form an infinite number of constraints on θ indexed by $x \in X$. We note that, however, since $\psi(x, \theta)$ is not necessarily concave in x, the typical bilevel programming approach of rewriting the constraint as $\max_{x \in X} \psi(x, \theta) \leq 0$ and using the Karush-Kuhn-Tucker (KKT) conditions [24] to convert it into a finite number of constraints is not applicable here. In fact, as one can observe from (10) with $V(x) = \omega(x)^\top \theta$, $\psi(x, \theta)$ is not guaranteed to be convex in θ either.

Under this parameterization, the Lyapunov-based controller is written as $u = \kappa(x|\theta)$. The performance of the controller can be evaluated with a finite or infinite number of simulation scenarios indexed by a random vector ξ, whose probability distribution is supported on Ξ, which is an either finite or bounded and closed set. Under scenario ξ, by solving the following differential equations, the closed-loop system is simulated with corresponding initial point x_ξ^0, time span $T_\xi > 0$, and bounded exogenous disturbance signals $d_\xi(\cdot)$:

$$\begin{aligned} \dot{x}_\xi(t) &= f(x_\xi(t)) + g(x_\xi(t))u_\xi(t) + b(x_\xi(t))d_\xi(t) \\ u_\xi(t) &= \kappa(x_\xi(t)|\theta),\ x_\xi(0) = x_\xi^0. \end{aligned} \tag{16}$$

Based on the simulated state and input trajectories, which depend on θ and hence can be denoted as $x_\xi(t|\theta)$ and $u_\xi(t|\theta)$ respectively, we can define a performance cost function $c_\xi : X \times U \to \mathbb{R}$, assumed to be continuously differentiable, for each scenario $\xi \in \Xi$ in the form of the following integral

$$\phi_\xi(\theta) = \int_0^{T_\xi} c_\xi(x_\xi(t|\theta), u_\xi(t|\theta))dt. \tag{17}$$

The objective function is then defined as the expectation of $\phi_\xi(\cdot)$ under the distribution of ξ on Ξ.

Based on the above discussion, the optimal control-Lyapunov function design problem of interest is now formulated as the following semi-infinite stochastic programming problem:

$$\begin{aligned} &\min_{\theta\in\Theta} \phi(\theta) = \mathbb{E}_\xi\left[\phi_\xi(\theta)\right] \\ &\text{s.t. } \psi(x,\theta) \le 0, \forall x \in X. \end{aligned} \tag{18}$$

In the following subsection, we will show some different specializations of the above generic formulation under different parameterizations, choice of simulation scenarios, and control strategies (Lin-Sontag and MPC).

B. Specialized formulations

1) Linear parameterization: A typical choice of linear parameterization can be a quadratic form, i.e.,

$$V(x) = \theta_{11}x_1^2 + \theta_{12}x_1x_2 + \cdots + \theta_{n_xn_x}x_{n_x}^2. \tag{19}$$

If the corresponding bounding functions for V, w_1 and w_2, are chosen also as quadratic functions, $w_i(\|x\|) = a_i\|x\|^2$, $i = 1, 2$, then the condition $w_1(\|x\|) \le V(x) \le w_2(\|x\|)$ can be easily translated to

$$\theta \in \Theta = \{\theta | a_1 I \preceq \mathrm{mat}(\theta) \preceq a_2 I\}. \tag{20}$$

where $\mathrm{mat} : \mathbb{R}^{n_x^2} \to \mathbb{R}^{n_x\times n_x}$ maps the vector θ into a corresponding matrix. By specifying a_1 and a_2, the control-Lyapunov function is required to have an elliptic contour on which the ratio between the longest axis length and the shortest axis length does not exceed $\sqrt{a_2/a_1}$.

A more complicated option is to restrict V on the collection of sum-of-squares (SOS) polynomials of x, i.e., $V(x) = \sum_i p_i(x)^2$ with each p_i being a polynomial of degree not exceeding d. It is known [10] that such a function V can be expressed as

$$V(x) = m(x)^\top Hm(x) \tag{21}$$

where $m : X \to \mathbb{R}^{n_b}$ ($n_b = \binom{n_x+d_p}{n_x} - 1$), whose components comprise all non-constant n_x-variate monomials of degrees not exceeding d_p, and H is a matrix such that there is an $L \in \mathbb{R}^{n_b\times n_b}$ satisfying $m(x)Lm(x) \equiv 0$ and making $H + L \succeq 0$. Now, denoting $\omega(x) = m(x) \otimes m(x)$ where $\otimes$ stands for Kronecker product, we have $V(x) = \omega(x)^\top\theta$, $\theta \in \Theta$, with

$$\Theta = \{\theta | \mathrm{mat}(\theta + l) \succeq 0, \omega(x)^\top l \equiv 0\}. \tag{22}$$

Here $\omega(x)^\top l \equiv 0$ can be converted to explicit linear equality constraints on l. For example, when $n_x = 1$, $d_p = 2$, we have $m(x) = [1, x, x^2]^\top$, $\omega(x) = [1, x, x^2, x, x^2, x^3, x^2, x^3, x^4]^\top$, and hence l should satisfy $l_2 + l_4 = 0$, $l_3 + l_5 + l_7 = 0$ and $l_6 + l_8 = 0$. Hence we may denote these algebraic constraints on l as $l \in \Lambda$.

If w_1 and w_2 are also chosen as SOS polynomials, i.e., $w_i(\|x\|) = \omega(x)^\top h_i$, $i = 1, 2$, then the condition $w_1(\|x\|) \le V(x) \le w_2(\|x\|)$ can be met by requiring $w_2(\|x\|) - V(x)$ and $V(x) - w_1(\|x\|)$ both be SOS. That is,

$$\begin{aligned} \Theta = \{\theta | &\mathrm{mat}(\theta - h_1 - l_1) \succeq 0, \\ &\mathrm{mat}(h_2 + l_2 - \theta) \succeq 0,\ l_1, l_2 \in \Lambda\}. \end{aligned} \tag{23}$$

2) Scenarios and performance evaluation: The scenarios indexed by $\xi \in \Xi$ to evaluate the performance of the controller $\kappa(\cdot|\theta)$ can be chosen or designed by the user, based on the understanding of typical operating conditions of the system and the purpose of deploying the controller. Such considerations can include (i) initialization at any randomized state on X (i.e., the main goal of the controller is to attract the states), (ii) initialization at any point on X that is an equilibrium under a different input value (i.e., the controller is used for transitioning the system between setpoints), and (iii) induction of exogenous disturbance signals $d_\xi(\cdot)$ as random noises, steps or oscillations (i.e., the controller needs to reject common disturbances of some typical magnitude). As such, the objective function directly and practically evaluates the controller performance under the circumstances that the user anticipates to encounter. The number of simulation scenarios can be finite or infinite.

Based on the simulation scenarios, the performance cost function $c(x, u)$ can be flexibly defined. Such a cost can penalize various undesirable dynamic behaviors and can be expressed as a weighted sum of the corresponding terms. These penalization terms can include (i) the squared errors of states x_i^2, $i = 1, \dots, n_x$ or outputs y_i^2 ($y = h(x)$ for some defined output mapping h) and control inputs u_i, $i = 1, \dots, n_u$, (ii) large deviations from the origin, e.g., $(|x_i| - a_i)^2$ if $|x_i| \ge a_i$, (iii) large rate of changes, which aims to guarantee that the state trajectories are smooth, e.g., $\|\dot{x}\|^2 = \|f(x) + g(x)u\|^2$, and (iv) overshooting in the direction opposite to the initial condition, e.g., x_i^2 if $x_i x_i^0 < 0$.

3) Stochastic gradient oracles: To solve a stochastic programming problem, it is needed to evaluate or estimate the gradient of $\phi(\theta)$. For the Lin-Sontag controller, the sensitivity analysis of the differential equation in (16) gives the following differential equation for $\partial x/\partial\theta$, i.e., the dependence of state trajectory on the parameters:

$$\frac{d}{dt}\left(\frac{\partial x}{\partial \theta}\right) = \frac{\partial F}{\partial x}\frac{\partial x}{\partial \theta} + \frac{\partial F}{\partial \theta} \tag{24}$$

which has the initial condition $\partial x/\partial\theta = 0$. Here F refers to the right-hand side of the state equation, i.e., $f(x) + g(x)\kappa(x|\theta) + b(x)w$. To derive (24), we assume that $\|\mathbb{L}_b(\omega(x)^\top\theta)\|\zeta^{-1}(\|x\|)$ is continuously differentiable for any $x \in X$ and $\theta \in \Theta$. The explicit dependence of $\kappa(\cdot|\theta)$

on θ allows the computation of $\partial\kappa/\partial x$ and $\partial\kappa/\partial\theta$ using the Leibniz rule through the partial derivatives with respect to $\mathbb{L}_f V(x)$, $\mathbb{L}_g V(x)$, $V(x)$ and $\|\mathbb{L}_b(\omega(x)^\top\theta)\|\zeta^{-1}(\|x\|)$. Hence, the partial derivatives of the right-hand side of the first equation in (16) are given by

$$\frac{\partial F}{\partial x} = \frac{df}{dx} + \frac{dg}{dx}\kappa + g\frac{\partial\kappa}{\partial x} + \frac{db}{dx}w, \quad \frac{\partial F}{\partial\theta} = g\frac{\partial\kappa}{\partial\theta}. \tag{25}$$

Solving (24) for $\partial x/\partial\theta$ under any given scenario ξ, then we can subsequently compute

$$\nabla\phi_\xi(\theta) = \int_0^{T_\xi}\left[\left(\frac{\partial c_\xi}{\partial x} + \frac{\partial c_\xi}{\partial u}\frac{\partial\kappa}{\partial x}\right)\frac{\partial x}{\partial\theta} + \frac{\partial c_\xi}{\partial u}\frac{\partial\kappa}{\partial\theta}\right]dt. \tag{26}$$

Therefore it is possible to directly compute $\nabla\phi_\xi$ by doing a sensitivity simulation along with the state simulation. Assuming that $\nabla\phi_\xi$ as a random vector is integrable under the probability measure of ξ for any $\theta\in\Theta$, then we have $\mathbb{E}_\xi[\nabla\phi_\xi] = \nabla\phi$. In other words, we have a stochastic first-order oracle that is computationally available.

For MPC, we can not analytically obtain the dependence of the simulated trajectories on the parameters θ. Nevertheless, we assume that for any $\xi\in\Xi$, $\theta\in\Theta$, (11) is recursively feasible and the partial derivatives exist, although they not readily computable, so that $\nabla\phi_\xi$ and hence $\nabla\phi$ still exist. Further assuming that the gradient of ϕ is Lipschitz continuous with constant $L_\phi>0$, we can adopt the Nesterov smoothing technique [27], where a random direction $\eta\in\mathbb{R}^{n_\theta}$ under the standard normal distribution is used:

$$\hat\nabla\phi_\xi^\mu(\theta,\eta) = \mu^{-1}\left(\phi_\xi(\theta+\mu\eta) - \phi_\xi(\theta)\right)\eta. \tag{27}$$

Here $\mu>0$ is called the smoothing constant. The construction in (27) is an unbiased gradient estimate for the smoothed function $\phi^\mu(\theta) = \mathbb{E}_\eta[\phi(\theta+\mu\eta)]$ (i.e., $\mathbb{E}_{\xi,\eta}[\hat\nabla\phi_\xi^\mu(\theta,\eta)] = \nabla\phi^\mu(\theta)$). The gradient of the smoothed function has a difference from that of ϕ linearly bounded by μ: $\|\nabla\phi^\mu - \nabla\phi\| \le \mu L_\phi(n_\theta+3)^{3/2}/2$. Hence as $\mu\to 0$, the error induced by smoothing vanishes. The estimate (27) is said to be a stochastic zeroth-order oracle for $\nabla\phi$ [28].

IV. A STOCHASTIC PROXIMAL PRIMAL-DUAL ALGORITHM

The algorithms for stochastic programming problems with semi-infinite constraints in generic nonconvex settings have not been well studied so far. The algorithm recently developed in Boob et al. [25] provides a theoretically established approach applicable to (18). Although the algorithm seems to have a high complexity and its practical numerical performance has not been reported to a wide range of benchmark problems, we adopt it nevertheless to demonstrate the application of our proposed method, without excluding the possibility that other more efficient algorithms may appear. Next, we review the basic properties of this algorithm. For this, we let

$$\psi_+(x,\theta) = \max\left(\psi(x,\theta), 0\right) \tag{28}$$

and assign a distribution of x supported on X (e.g., a uniform distribution). Then the infinite constraints become an expectation constraint, i.e., (18) is reformulated as

$$\begin{aligned} &\min_{\theta\in\Theta}\phi(\theta) = \mathbb{E}_\xi\left[\phi_\xi(\theta)\right] \\ &\text{s.t. } \psi(\theta) := \mathbb{E}_x\left[\psi_+(x,\theta)\right] \le 0. \end{aligned} \tag{29}$$

The following assumptions are needed for the algorithm:

- $\nabla\phi$ is Lipschitz continuous with constant $L_\phi>0$, and $\forall\theta,\theta'\in\Theta$, $\forall s\in\partial\psi(\theta)$ (∂ is the subgradient set), $\exists L_\psi>0$, such that
$$\psi(\theta') - \psi(\theta) - s^\top(\theta'-\theta) \le (L_\psi/2)\|\theta'-\theta\|^2. \tag{30}$$
- $\exists M>0$, such that $\forall\theta,\theta'\in\Theta$,
$$|\psi(\theta') - \psi(\theta)| \le M\|\theta'-\theta\|. \tag{31}$$
- $\forall\theta\in\Theta$, $\exists\sigma_\phi,\sigma_\psi,\sigma_{\nabla\phi},\sigma_{\partial\psi}>0$, such that
$$\begin{aligned} &\mathbb{E}_\xi\left[\phi_\xi(\theta)\right] = \phi(\theta),\ \ \mathrm{Var}_\xi\left[\phi_\xi(\theta)\right] \le \sigma_\phi^2, \\ &\mathbb{E}_\xi\left[\nabla\phi_\xi(\theta)\right] = \nabla\phi(\theta),\ \ \mathrm{Var}_\xi\left[\nabla\phi_\xi(\theta)\right] \le \sigma_{\nabla\phi}^2, \\ &\mathbb{E}_x\left[\psi_+(x,\theta)\right] = \psi(\theta),\ \ \mathrm{Var}_x\left[\psi_+(x,\theta)\right] \le \sigma_\psi^2, \\ &\forall s\in\partial\psi_+(x,\theta),\ \mathbb{E}_x\left[s\right]\in\partial\psi(\theta), \mathrm{Var}_x\left[s\right] \le \sigma_{\partial\psi}^2. \end{aligned} \tag{32}$$

Under the above assumptions, for any fixed $\bar\theta\in\Theta$, define $\bar\phi(\theta|\bar\theta) := \phi(\theta) + L_\phi\|\theta-\bar\theta\|^2$, $\bar\psi_+(x,\theta|\bar\theta) := \psi_+(x,\theta) + L_\psi\|\theta-\bar\theta\|^2$, which are strongly convex functions of θ, and consider the following *proximal point problem*:

$$\begin{aligned} &\min_{\theta\in\Theta}\ \bar\phi(\theta|\bar\theta) \\ &\text{s.t. } \ \bar\psi(\theta|\bar\theta) := \mathbb{E}_x\left[\bar\psi_+(x,\theta|\bar\theta)\right] \le 0. \end{aligned} \tag{33}$$

(For practical implementation, we need to estimate the Lipschitz constants L_ϕ and L_ψ, e.g., by sampling ξ and x on a grid of θ and using numerical differentiation.) Letting the dual variable (Lagrangian multiplier) associated with the constraint be $\upsilon\ge 0$, the Lagrangian of the problem (29) is $\bar\phi(\theta|\bar\theta) + \upsilon\bar\psi(x|\bar\theta)$, which, in a stochastic setting and linearized at a point $\tilde\theta$, can be approximated by

$$\begin{aligned} \tilde\Lambda_{\xi,x}^{\bar\theta,\tilde\theta}(\theta,\upsilon) := &\left[\bar\phi_\xi(\tilde\theta|\bar\theta) + \nabla\bar\phi_\xi(\tilde\theta|\bar\theta)^\top(\theta-\tilde\theta)\right] \\ &+\upsilon\left[\bar\psi_+(x,\tilde\theta|\bar\theta) + \nabla\bar\psi_+(x,\tilde\theta|\bar\theta)^\top(\theta-\tilde\theta)\right]. \end{aligned} \tag{34}$$

Algorithm 1, called the constrained extrapolation method, is a stochastic and linearized Lagrangian-based algorithm for solving (33). In Algorithm 1, Line 2 is an extrapolation of constraint violation using the recent two iterations (β_t is the step size of extrapolation, usually fixed at 1; ς_t is the extrapolated value of violation). Line 3 is a subgradient ascent step of the dual variable with (inverse) step size τ_t. Line 4 is the primal update by minimizing the stochastic linearized Lagrangian, regularized by the distance from the previous iteration. Line 6 returns the solution based on intermediate iterations, where γ_t is the relative weight for the t^{th} iteration. The convergence and complexity of Algorithm 1 is ensured by the following algorithm [25, Theorem 2.3].

Theorem 1. *Suppose that the assumptions in* (30)*,* (31)*, and* (32) *hold. Setting* $\gamma_t = \beta_t = 1$*, and* η_t *and*

for $t = 0, 1, \ldots, T-1$ **do**
  $\varsigma_t \in -\beta_t \psi_+(x_{t-1}, \theta_{t-1}|\bar{\theta}) + (1+\beta_t) \cdot [\bar{\psi}_+(x_{t-1}, \theta_{t-1}|\bar{\theta}) + \partial\bar{\psi}_+(x_{t-1}, \theta_{t-1}|\bar{\theta})(\theta_t - \theta_{t-1})]$;
  $\upsilon_{t+1} := [\upsilon_t + \varsigma_t/\tau_t]_+$;
  $\theta_{t+1} \in \arg\min_{\theta\in\Theta} \tilde{\Lambda}^{\bar{\theta},\theta_t}_{\xi_t,x_t}(\theta, \upsilon_{t+1}) + \frac{\eta_t}{2}\|\theta - \theta_t\|^2$;
end
Return: $\theta = \left(\sum_{t=0}^{T-1}\gamma_t\right)^{-1}\left(\sum_{t=0}^{T-1}\gamma_t\theta_{t+1}\right)$;

Algorithm 1: Constraint extrapolation method.

for $k = 1, 2, \ldots, K$ **do**
  $\theta_k := \theta^*(\theta_{k-1}, \epsilon_k, \epsilon^*_k)$ using Algorithm 1;
end
Randomly choose $\kappa \in \{1, 2, \ldots, K\}$;
return θ_κ;

Algorithm 2: Semi-infinite stochastic programming based on an inexact proximal point method.

τ_t as sufficiently large constant values, the number of iterations T_ϵ to reach a stochastic (ϵ, ϵ')-optimal solution when $\epsilon' \sim O(\epsilon)$ is of the order $O(\epsilon^{-2})$. Here a stochastic (ϵ, ϵ')-optimal solution refers to a $\theta^* \in \Theta$ such that $\mathbb{E}_\xi\left[\bar{\phi}_\xi(\theta^*|\bar{\theta}) - \min_{\theta\in\Theta}\bar{\phi}(\theta|\bar{\theta})\right] \leq \epsilon$ and $\mathbb{E}_x\left[\bar{\psi}_+(x, \theta^*|\bar{\theta})\right] \leq \epsilon'$.

Since given any $\bar{\theta} \in \Theta$ and stochastic errors $\epsilon, \epsilon' > 0$, Algorithm 1 returns an inexact optimal solution to (33) in a stochastic sense, which we may denote as $\theta^*(\bar{\theta}, \epsilon, \epsilon')$, an outer loop of iterations (indexed by k) can be applied, as summarized in Algorithm 2. The convergence and complexity properties are given below [25, Theorem 3.17].

Theorem 2. *Suppose that the assumptions for Theorem 1 hold and further assume that the Mangasarian-Fromovitz constraint qualification (MFCQ) is satisfied at all limit points of the sequence $\{\theta_t\}_{t=1}^T$ in Algorithm 1 under $\bar{\theta} = \theta_k$ for $k = 1, \ldots, K$. Then under Algorithm 2,*

- *all the dual variables during the iterations are bounded, i.e., $\exists B > 0$ such that $|\upsilon_t| \leq B$, and*
- *the returned θ_κ is a stochastic $(\varepsilon, \varepsilon')$-optimal solution, in which*

$$\varepsilon = \frac{2\Gamma}{K}\max(1, 4(L_\phi + BL_\psi)), \quad \varepsilon' = \frac{2\Omega}{L_\phi K},$$
$$\Omega = \sum_{k=1}^{K}\left(\epsilon_k + B\epsilon'_k\right), \quad \Gamma = c + \Omega \text{ for some } c > 0. \tag{35}$$

The latter point implies that for any desirable final error $\varepsilon > 0$, to obtain a stochastic $(\varepsilon, \varepsilon)$-optimal solution, if setting all $\epsilon_k, \epsilon'_k \sim O(\varepsilon)$, then $O(\varepsilon^{-1})$ outer loops of iterations will be needed, each of which uses $O(\varepsilon^{-2})$ inner iterations. Therefore the overall complexity becomes $O(\varepsilon^{-3})$.

Remark 1. *For the proximal point problem (33), MFCQ at a point θ^* means that if the constraint is active here ($\bar{\psi}(\theta^*|\bar{\theta}) = 0$), there must exist $\nu \in N_\Theta(\theta^*)$, where $N_\Theta(\theta^*) = \{\nu|\nu^\top(\theta - \theta^*) \leq 0, \forall\theta \in \Theta\}$ is the normal cone of Θ at θ^*, such that $\min_{s\in\partial\bar{\psi}(\theta^*|\bar{\theta})} s^\top\nu > 0$.*

Remark 2. *In our implementation of the above-mentioned algorithm we make the following three modifications.*

- *First, in each inner iteration, instead of using only one scenario for $\xi \in \Xi$ and one sample for $x \in X$, we adopt* minibatches *to reduce the variances of the stochastic quantities in (34), thus reducing the number of iterations needed. Specifically, using multiple scenarios $\xi_1, \ldots, \xi_{S_\xi}$, we replace the stochastic gradient $\nabla\phi_\xi(\theta)$ by $\frac{1}{S_\xi}\sum_{s=1}^{S_\xi}\nabla\phi_{\xi_s}(\theta)$, whose variance is reduced by a factor of S_ξ. Analogously, a minibatch sampling with size S_x can be used to evaluate $\bar{\psi}_+(x_t, \theta_t|\bar{\theta})$.*
- *We also note that the stochastic optimality solution is unverifiable by definition, since the true optimum can not be known a priori. Instead, we check* primal *and* dual residuals, *i.e., $r = \partial\tilde{\Lambda}/\partial\theta$ and $r' = \partial\tilde{\Lambda}/\partial\upsilon$. In deterministic optimization problems, one can calculate the residuals directly for all the intermediate solutions and terminate the iterations when the residuals are below their tolerances. However, for stochastic problems such a direct calculation is not possible. Hence, for every period of Δ iterations, we approximate the residuals by*
$$\begin{cases} r_t &= \frac{1}{\Delta}\left\|\sum_{t'=t-\Delta}^{t-1}\eta_{t'}(\theta_{t'+1} - \theta_{t'})\right\| \\ r'_t &= \frac{1}{\Delta}\sum_{t'=t-\Delta+1}^{t}\bar{\psi}_+(x_{t'}, \theta_{t'}|\bar{\theta}) \end{cases}. \tag{36}$$
When these residuals are below their corresponding thresholds, the inner iterations are terminated.
- *Finally, upon termination of inner iterations, instead of assigning an equal weight $\gamma = 1$ to all t, we only average the last Δ solutions since the incipient iterations should be far away from convergence. Also for the outer iterations (Algorithm 2), instead of randomly choosing a $\kappa \in \{1, 2, \ldots, K\}$, we simply set final tolerances on the residuals and choose the solution from the last outer iteration.*

V. Case Study

For illustration, we now apply the proposed method on a continuously stirred tank reactor system modeled by [29]:

$$\begin{aligned}\dot{C} =& 1 - C - 0.0637C\exp\left[37.7\left(1 - T^{-1}\right)\right] + 0.1d \\ \dot{T} =& 1 - T + 0.0106C\exp\left[37.7\left(1 - T^{-1}\right)\right] \\ &+ 0.1356(Q - T).\end{aligned} \tag{37}$$

d is an exogenous disturbance in the feed concentration. The steady state $Q^{\text{ss}} = 1.048$, $C^{\text{ss}} = 0.5196$, $T^{\text{ss}} = 1.0764$ is open-loop unstable. The system equations (37) are translated and scaled by $x_1 := (C - C^{\text{ss}})/0.20$, $x_2 := (T - T^{\text{ss}})/0.03$ and $u := (Q - Q^{\text{ss}})/0.40$ to obtain a form of $\dot{x} = f(x) + g(x)u$ with $f(0) = 0$.

The optimal design of a control-Lyapunov function is formulated specifically as follows. First, we parameterize V in a quadratic form with x_1 and x_2 equally weighted, i.e.,

$$V(x) = \frac{1}{2}\left(x_1^2 + 2\theta x_1 x_2 + x_2^2\right) \tag{38}$$

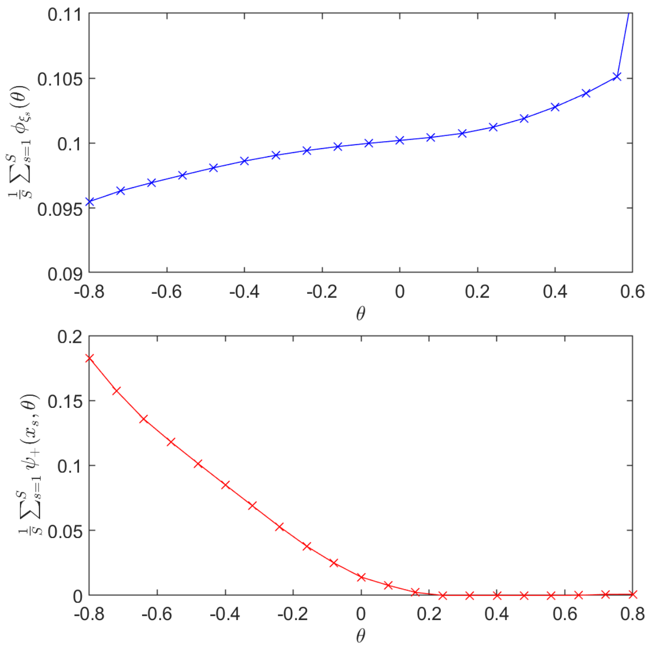

Fig. 1. Estimated ϕ and ψ functions.

and require $\theta \in \Theta = [-4/5, 4/5]$ so that on the ellipsoidal contour of $V(x)$, the length of the long axis does not exceed 3 times of that of short axis. Consider the Lin-Sontag controller subject to $u \in U = [-1, 1]$. Set $\rho = 0.5$ so that $V(x)$ decays exponentially with a time constant of 2 without exogenous disturbance, and $\zeta(|d|) = 3|d|$. We let $X = \{x | \|x\| \leq 1\}$, and a uniform distribution is used for $x \in X$ in (29). The control performance under each scenario ξ is simply evaluated by $c_\xi(x, u) = (x_1^2 + x_2^2)/2 + u^2$. For all the scenarios, we fix $x_\xi^0 = 0$ and $T_\xi = 10$, while the exogenous disturbance d is a constant signal that can be nonzero on $t \in [0, 5]$, with its magnitude D sampled from a probability density function $p(D) = 3(1 - D^2)/4$ supported on $[-1, 1]$.

We sample 21 points for $\theta \in [-0.8, 0.8]$ with equal distances and obtain 20 scenarios for $\xi \in \Xi$ and 200 samples for $x \in X$ to approximate $\phi(\theta)$ and $\psi_+(\theta)$. The graphs for the approximated ϕ and ψ_+ are plotted in Fig. 1, from which we estimate the Lipschitz constants as $L_\phi = 0.0292$ and $L_\psi = 0.0871$, implying that ψ and ϕ are slightly nonconvex. We also observe from the plot of ϕ that the control performance becomes better with smaller values of θ. However, as seen in the plot of ψ, the semi-infinite constraints are satisfied only on an interval in $(0, 4/5]$.

We implement Algorithm 2 to solve for the optimal θ value. In each inner iteration, the minibatch sizes are $S_\xi = 1$ and $S_x = 200$. The step sizes are set as $\eta_t = 0.1 \cdot 2^{k-1}$ and $\tau_t = 0.5$ for $k \leq 5$ and $\tau_t = 5 \cdot 2^{k-5}$ for $k \geq 6$. The residuals are set as $r_k = \max(10^{-4}, 10^{-3}/2^{k-1})$ and $r'_k = \max(10^{-4}, 10^{-2}/2^{k-2})$. The size of the memory for evaluating residuals and averaging solutions is set as $\Delta = 20$. With $K = 9$ outer iterations, both primal and dual residuals converge to below 10^{-4}, which we consider as satisfactory.

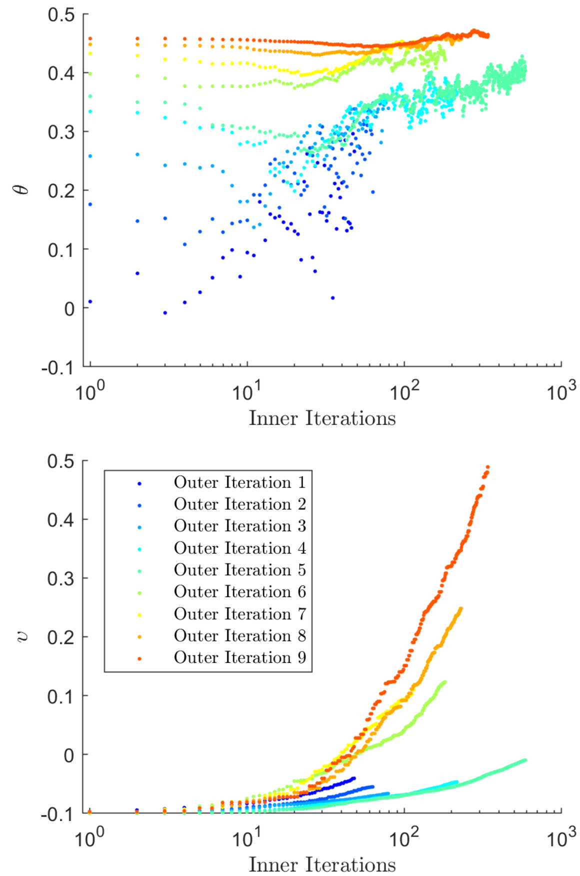

Fig. 2. Trajectory of primal and dual variable throughout iterations.

The trajectories of the primal variable θ and dual variable v throughout the iterations are shown in Fig. 2. The obtained optimal value is found to be $\theta^* = 0.4622$.

To verify the result, we plot the image of $\psi(x, \theta^*)$ over $x \in X$ in Fig. 3. It can be observed that the plotted surface is below 0, i.e., the semi-infinite input-to-state stability constraints are satisfied, except on a small corner of X due to the nonzero tolerance of the dual residual in the optimization algorithm.

VI. Conclusions

In this paper, we have proposed a generic semi-infinite stochastic programming formulation for the problem of optimally designing control-Lyapunov functions for nonlinear systems under an explicit Lin-Sontag Lyapunov-based controller or MPC. Specifically, the formulation uses (i) a linear parameterization of the control-Lyapunov function, (ii) constraints indexed by all states on a given region that specifies an exogenous input-to-state stability condition, and (iii) a cost function defined based on closed-loop trajectories under simulation scenarios. The problem formulation is scalable with increasing state dimensions, flexible for user specifications and oriented towards a practically optimal control performance.

We have also discussed a stochastic proximal primal-dual algorithm for semi-infinite stochastic programming, where

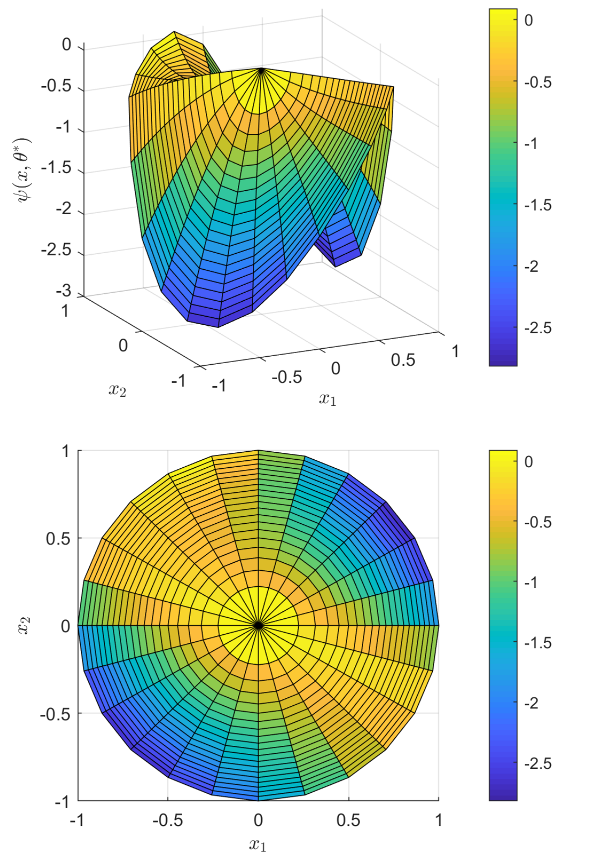

Fig. 3. Image of $\psi(x,\theta^*)$ in surface plot (upper) and heat map (below).

the primal and dual variables are updated with minibatch stochastic gradient descent and ascent, respectively, in an outer iteration with proximal convexification. The algorithm is essentially driven by the collection of simulation trajectory data and the iterative evaluation of such data. A case study on a chemical reactor system is used for demonstration. To this end, the complexity of the algorithm in its practical implementation for problems of higher dimensions, especially in the steps of estimating Lipschitz parameters and sampling random scenarios, will be of critical importance and needs further investigation.

References

[1] A. M. Lyapunov, *General problem of the stability of motion (in Russian)*. Moscow-Leningrad: Gostekhizdat, 1950.

[2] H. K. Khalil, *Nonlinear systems*, 3rd ed. Upper Saddle River: Prentice Hall, 2002.

[3] Y. Lin and E. D. Sontag, "A universal formula for stabilization with bounded controls," *Syst. Control Lett.*, vol. 16, no. 6, pp. 393–397, 1991.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[5] P. Mhaskar, N. H. El-Farra, and P. D. Christofides, "Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control," *Syst. Control Lett.*, vol. 55, no. 8, pp. 650–659, 2006.

[6] M. Heidarinejad, J. Liu, and P. D. Christofides, "Economic model predictive control of nonlinear process systems using Lyapunov techniques," *AIChE J.*, vol. 58, no. 3, pp. 855–870, 2012.

[7] Z. Wu, F. Albalawi, Z. Zhang, J. Zhang, H. Durand, and P. D. Christofides, "Control Lyapunov-barrier function-based model predictive control of nonlinear systems," *Automatica*, vol. 109, p. 108508, 2019.

[8] H. Oyama and H. Durand, "Integrated cyberattack detection and resilient control strategies using Lyapunov-based economic model predictive control," *AIChE J.*, vol. 66, no. 12, p. e17084, 2020.

[9] P. Kokotović and M. Arcak, "Constructive nonlinear control: a historical perspective," *Automatica*, vol. 37, no. 5, pp. 637–662, 2001.

[10] G. Chesi, *Domain of attraction: analysis and control via SOS programming*. London: Springer, 2011.

[11] M. Tacchi, C. Cardozo, D. Henrion, and J. B. Lasserre, "Approximating regions of attraction of a sparse polynomial differential system," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3266–3271, 2020.

[12] M. Souaiby, A. Tanwani, and D. Henrion, "Computation of Lyapunov functions under state constraints using semidefinite programming hierarchies," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6281–6286, 2020.

[13] M. Bardi and I. Capuzzo-Dolcetta, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Boston: Birkhäuser, 1997.

[14] A. Rantzer, "A dual to Lyapunov's stability theorem," *Syst. Control Lett.*, vol. 42, no. 3, pp. 161–168, 2001.

[15] U. Vaidya and P. G. Mehta, "Lyapunov measure for almost everywhere stability," *IEEE Trans. Autom. Control*, vol. 53, no. 1, pp. 307–323, 2008.

[16] A. Raghunathan and U. Vaidya, "Optimal stabilization using Lyapunov measures," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1316–1321, 2013.

[17] A. Abate, D. Ahmed, M. Giacobbe, and A. Peruffo, "Formal synthesis of Lyapunov neural networks," *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 773–778, 2020.

[18] N. Gaby, F. Zhang, and X. Ye, "Lyapunov-net: A deep neural network architecture for lyapunov function approximation," in *Proc. 61st Conf. Decis. Control (CDC)*. IEEE, 2022, pp. 2091–2096.

[19] S. Chen, M. Fazlyab, M. Morari, G. J. Pappas, and V. M. Preciado, "Learning Lyapunov functions for hybrid systems," in *Proc. 24th Int. Conf. Hybrid Syst. Comput. Control*, 2021, pp. 1–11.

[20] I. E. Grossmann, B. A. Calfa, and P. Garcia-Herreros, "Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes," *Comput. Chem. Eng.*, vol. 70, pp. 22–34, 2014.

[21] W. Tang and P. Daoutidis, "A bilevel programming approach to the convergence analysis of control-Lyapunov functions," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4174–4179, 2019.

[22] ——, "Lyapunov dynamic flexibility of nonlinear processes," in *Computer Aided Chemical Engineering*. Elsevier, 2019, vol. 47, pp. 35–40.

[23] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: Modeling and theory*. Philadelphia: SIAM, 2014.

[24] M. López and G. Still, "Semi-infinite programming," *Eur. J. Oper. Res.*, vol. 180, no. 2, pp. 491–518, 2007.

[25] D. Boob, Q. Deng, and G. Lan, "Stochastic first-order methods for convex and nonconvex functional constrained optimization," *Math. Program.*, vol. 197, 2023.

[26] E. D. Sontag and Y. Wang, "On characterizations of the input-to-state stability property," *Syst. Control Lett.*, vol. 24, no. 5, pp. 351–359, 1995.

[27] Yu. Nesterov, "Smooth minimization of non-smooth functions," *Math. Program.*, vol. 103, no. 1, pp. 127–152, 2005.

[28] S. Ghadimi and G. Lan, "Stochastic first- and zeroth-order methods for nonconvex stochastic programming," *SIAM J. Optim.*, vol. 23, no. 4, pp. 2341–2368, 2013.

[29] R. Aris, *Elementary chemical reactor analysis*, 2nd ed. Boston: Butterworths, 1989.