Robust Optimization via Continuous-Time Dynamics

Keivan Ebrahimi, Nicola Elia, and Umesh Vaidya

Abstract—**We propose a dynamical system-based approach for solving robust optimization problems in a general convex-concave framework. Our proposed approach offers a novel solution to robust optimization problems by introducing a continuous-time dynamical system, which we call $\mathcal{RO}$ dynamics. Unlike the well-known primal-dual gradient dynamics for solving regular optimization problems, our new approach does not rely on the gradient of the Lagrangian function to derive the vector field. In $\mathcal{RO}$ dynamics, the uncertain variable is treated as a dynamical state, and we demonstrate that the globally asymptotically stable equilibrium point of $\mathcal{RO}$ dynamics can recover robust optimal solutions for a general class of convex-concave robust optimization problems. This is achieved without any prior knowledge of the formulation of the specific problem. Furthermore, our approach is amenable to decentralized implementation, enabling the dynamics to autonomously derive an unknown system to solve the robust optimization problem by using output feedback only. This is while the existing solutions such as the robust counterpart and scenario-based random sampling assume that the problem formulation is completely known a priori. In contrast, our proposed method is not constrained by these limitations and is suitable for real-time applications in dynamic environments. To demonstrate the effectiveness of our approach, we provide a simulation example of a robust location and placement problem with time-varying anchor locations, among others. Our proposed dynamical system-based approach offers better properties compared to traditional approaches and demonstrates the potential to solve a broad range of robust optimization problems in a decentralized manner.**

I. INTRODUCTION

With emerging applications that require solving real-time optimization problems in a reactive manner, this paper describes how interacting dynamical systems can find a robust solution for a broad class of robust optimization problems. This is particularly relevant in situations where physical systems must be steered towards optimal operating conditions, without prior knowledge of the functions to be optimized or the model of the uncertainty affecting the optimization. Specifically, we propose a real-time approach to solving robust optimization problems, which has become increasingly important in recent years [1]. Optimization problems often involve various forms of uncertainty in the problem data.

Two different approaches can be used: stochastic optimization, where uncertainty is treated as a random variable, or robust optimization ($\mathcal{RO}$), where uncertainty is assumed to be deterministic and bounded. Unlike stochastic optimization, $\mathcal{RO}$ does not require any known probability distributions in the problem data. Instead, $\mathcal{RO}$ assumes that the uncertain data reside within a predefined uncertainty set, for which constraint violation cannot be tolerated. For more information on robust and stochastic optimization-based approaches, see [2], [3], and the references therein. Early works on $\mathcal{RO}$ include [4], which considered robust linear optimization with ellipsoidal uncertainty sets, and [5], which presented exact solutions of inexact linear programs as a simple case of $\mathcal{RO}$.

The problem of robust linear programming was studied in [6], while robust conic-quadratic optimization and robust semi-definite optimization were discussed in [7] and [8], respectively. Refer to [3] and [9] for a survey of different $\mathcal{RO}$ problem solutions. [10] and [11] showed that some of the machine learning algorithms such as the norm-regularized support vector machine and the Lasso problem could be interpreted as $\mathcal{RO}$ problems. In addition to that, several papers focused on making machine learning problems robust against outliers, parameter uncertainties, and data perturbations by techniques such as adversarial training [12]–[18].

One of the main standard approaches for solving $\mathcal{RO}$ problems involves constructing a robust counterpart (RC) equivalent to the $\mathcal{RO}$ problem [2]. This widely used method essentially tries to find a deterministic equivalent to $\mathcal{RO}$ problem through a reformulation. In this sense, the practicality of robust programming depends on whether or not its RC is computationally tractable. An overview of different RO problems with tractable conjugates can be found in [19] Table 1 for some of which RC cannot be found. The reformulation approach to solving the $\mathcal{RO}$ problem, which is often a challenging, albeit usually convex, optimization problem, has the deficiency of suffering from case-by-case scenarios depending on the specific form of the uncertainty constraint and the specific form of the uncertainty set. In other words, depending on how simple the uncertainty set looks and based on the optimization problem type (whether it is linear programming, quadratic programming, second-order cone programming, semi-definite programming, etc.), an RC is being calculated and provided at hand. These reformulations may be computationally more expensive than other approaches [20]. Hence, the absence of a unified framework for solving a general class of $\mathcal{RO}$ problems without prior knowledge of the problem formulation is strongly felt.

By calculating a concave conjugate of nonlinear constraint functions and supporting the function of uncertainty sets, [21] and [19] used the Fenchel duality to obtain a tractable

RC for new classes of robust nonlinear optimization (RNO) problems. It may be the case that no closed form is available for the convex conjugate function of an uncertainty set or convex/concave conjugates of a non-linear constraint. In this case, these approaches do not obtain the RC for many sets of nonlinear uncertainties and constraints.

An alternative randomized approach based on constraint sampling is an approximate probabilistic relaxation solution to $\mathcal{RO}$ problems as seen in [22] and [23] where a finite set of high-dimensional deterministic optimization problems obtained from sampling are solved. This approach does not require concavity of the constrain in the uncertain variable but due to the large number of required scenarios to approximate the stochasticity of these problems, the stochastic optimization involves formulating a large-scale scenario program, which is in general computationally demanding. To address this issue, [24] presented two methods. The first is to decompose the large-scale scenario program into distributed scenario programs using the alternating-direction multiplier method (ADMM). The second method is to use a soft communication scheme to reduce the required communication between the subproblems.

In another effort, [25] incorporated the min-max behavior of $\mathcal{RO}$ problems to solve them by an oracle-based approximate robust optimization algorithm based on oracle-based subgradient descent and interior point methods. The proposed algorithms find an approximate robust solution using a number of calls to an oracle that solves the original (non-robust) problem. However, the solution would be approximated with some predetermined accuracy, η, with the number of iterations growing to $\mathcal{O}(\frac{1}{\eta})$, as the algorithm approximates the RC by invoking the Oracle a polynomial number of times. In [26], a cutting-plane algorithm treats $\mathcal{RO}$ problem as semi-infinite programming and iterates between optimization problem with fixed uncertainties values and posterior worst-case analysis to possibly find the robust solution. The worst-case analysis depends on finding pessimization oracles that may be exact or approximate. The robust problem cannot be solved by the cutting plane method if the pessimization is approximate, that is, when the worst-case functions are being computed approximately. In another effort, [27] tried to alleviate the computational cost of Oracle-based methods by finding the approximate solution for $\mathcal{RO}$ problem iteratively with first-order oracles.

Robust optimization over time (ROOT) is another line of research in which robust optimization and dynamic optimization (a certain type of optimization to model uncertainties resulting from environmental changes) are combined. In these problems, robust solutions usually do not remain desirable forever and need to be changed after some major environmental change. [28] provides a review of ROOT papers that includes a classification of these problems with their respective definitions.

Compared with existing $\mathcal{RO}$ methods, we point out that most existing algorithms are in a one-shot manner, as they require a priori knowledge of the uncertainty set model in order to even write the optimization problem to be solved. In contrast, our approach is independent of knowledge of the uncertainty set model as long as it is convex and the adversary

variables enter concavely in the constraints and cost function. This opens up the possibility of having distributed real-time implementations of such systems in many settings where the uncertainty model and exact knowledge of the cost function and constraints are not available.

In this paper, we provide the architecture of such a dynamical system that converges to the optimal robust solution for a large class of $\mathcal{RO}$ problems.

Although our approach is inspired by the work on primal-dual systems [29], [30], it presents significant architectural and convergence challenges described in the paper. A key technical contribution is finding the right architecture of the dynamical system for which a non-standard Lyapunov function can help to prove global convergence to an optimal $\mathcal{RO}$ solution. The proposed continuous-time optimization system can solve a general class of $\mathcal{RO}$ problems where the cost function and the constraints are convex (concave) with respect to the decision variables (uncertain variables) and the uncertainty sets are convex[1]. This class includes all the cases in [19, Table 1].

The remainder of this paper is organized as follows. In Section III, we present the problem statement of $\mathcal{RO}$ in a slightly generalized form. In Section IV, the characterization of saddle property and KKT optimality conditions along with the Lagrangian function for $\mathcal{RO}$ problem are provided. The main results on how to construct the $\mathcal{RO}$ dynamics is presented in Section V. This section also includes the Lyapunov-based global convergence result. We then present simulation results in Section VII followed by conclusions in Section VIII. Finally, detailed proofs are mentioned in the Appendix.

II. NOTATIONS

We define $i = 0, \cdots, N$ as $i_{[N]}$, and $i = 1, \cdots, N$ as $i_{[N]}^+$. In addition, $[\cdot]_\eta^+$ shows the positive projection defined as follows for the scalar-valued function P

$$[P]_\eta^+ := \begin{cases} P & \text{if} & P > 0 \text{ or } \eta > 0 \\ 0 & \text{otherwise} \end{cases} . \quad (1)$$

For vector-valued P, the projection is defined element-wise.

III. ROBUST OPTIMIZATION PROBLEMS

Given an objective $f_0(x)$ to optimize, subject to constraints $f_i(x, u_i) \leq 0$ with uncertain parameters, $\{u_i\}$, a classical robust optimization problem has the following form

$$\mu := \min_x f_0(x)$$
$$\text{s.t.} \quad f_i(x, u_i) \leq 0, \quad \forall u_i \in \mathcal{U}_i, \quad i_{[N]}^+.$$

where $x \in \mathbb{R}^n$ is a vector of decision variables, f_0 and f_i are $\mathbb{R}^n \to \mathbb{R}$ functions, and the uncertainty parameters $u_i \in \mathbb{R}^{m_i}$ are assumed to take arbitrary values in certain convex compact uncertainty sets $\mathcal{U}_i \subseteq \mathbb{R}^{m_i}$. The above problem is typically a semi-infinite optimization due to the cardinality of

---

[1]Preliminary convergence results for a special formulation appeared without proofs in [31]. In this paper, we generalize the problem formulation and provide complete proofs, which are the core of our contributions.

the $\mathcal{U}_i$. However, each robust constraint can be rewritten as a maximization problem as[2]

$$\mu := \quad \min_x f_0(x) \\ \text{s.t.} \quad \max_{u_i \in \mathcal{U}_i} f_i(x, u_i) \leq 0, \quad i_{[N]}^+ \quad . \tag{2}$$

We consider a slight variation of (2), which takes the following under the assumptions stated below.

$$\mathcal{RO}_0 \begin{cases} \mu := \min_x \max_{u_0 \in \mathcal{U}_0} f_0(x, u_0) \\ \text{s.t.} \quad \max_{u_i \in \mathcal{U}_i} f_i(x, u_i) \leq 0 , \ i_{[N]}^+, \\ \\ \mathcal{U}_i := \{u_i \in \mathbb{R}^{m_i} \ : \ h_{ij}(u_i) \leq 0, \ j_{[K_i]}\} , \ i_{[N]} \end{cases} \tag{3}$$

The problem in (3) reduces to the one in (2) when $\mathcal{U}_0$ is a singleton. Following [32] and without loss of generality, we consider the $\mathcal{RO}$ problem dealing with constraint-wise uncertainties where each constraint f_i is only a function of u_i. The functions f_i and h_{ij} have scalar values with the following assumptions.

Assumption 1. *The functions $h_{ij}(u_i)$ are convex in u_i for $i_{[N]}$ and $j_{[K_i]}^+$. The function $f_0(x, u_0)$ is strictly convex in x for any $u_0 \in \mathcal{U}_0$, and concave in u_0 for any x. Also, for $i_{[N]}^+$, $f_i(x, u_i)$ is convex in x for fixed u_i and concave in $u_i \in \mathcal{U}_i$, for fixed x. Finally, the functions f_i and h_i $i_{[N]}$ are C^1 with local Lipschitz gradients.*

Assumption 2. *Existence of optimal solutions and strong duality.*

A1 *$\mathcal{RO}$ problem (4) is feasible. An optimal min-max solution $(x^\star, u^\star)$ exists and μ in the $\mathcal{RO}$ problem (4) is finite.*

A2 *$\mathcal{RO}$ problem satisfies the Slater constraint qualification, for both the upper level optimization (5) and the lower level (6) [32], that is, for all $i_{[N]}$ and all $j_{[K_i]}^+$, there exist $u_i \in \mathcal{U}_i$ such that $h_{ij}(u_i) < 0$, and for $i_{[N]}^+$, there exists $x \in \mathbb{R}^n$ such that, $\mathcal{F}_i(x) < 0$.*

Remark 1. *It should be noted that the uncertainty is often parametrized affinely in $\mathcal{RO}$ problem; hence, the concavity property of the f_i functions is automatically satisfied.*

Remark 2. *Assumption [A2] guarantees that $\mathcal{RO}$ problem enjoys strong duality for upper and lower level optimization problems (5) and (6) [33, Section 5.2.3, 5.9.1]. Moreover, it enforces that saddle point and optimal dual solutions exist [32]. For conditions less restrictive than those of Slater, the interested reader may refer to [34].*

Finally, we introduce the problem we consider in the paper,

a slight generalization of $\mathcal{RO}_0$ in (3).

$$\mathcal{RO} \begin{cases} \mu := \min_x \max_{u_0 \in \mathcal{U}_0} f_0(x, u_0) + \sum_{i=1}^N c_i \max_{u_i \in \mathcal{U}_i} f_i(x, u_i) \\ \\ \text{s.t.} \quad \max_{u_i \in \mathcal{U}_i} f_i(x, u_i) \leq 0 , \ i_{[N]}^+, \\ \\ \mathcal{U}_i := \{u_i \in \mathbb{R}^{m_i} \ : \ h_{ij}(u_i) \leq 0, \ j_{[K_i]}\} , \ i_{[N]} \end{cases} \tag{4}$$

where $c_i \geq 0$ for $i_{[N]}^+$. This setting can be seen as an elementary regularized version of more common formulations $\mathcal{RO}_0$ obtained for $c_i = 0$, $i_{[N]}^+$. The role of c_i is further clarified in Remark 3.

It is convenient to rewrite $\mathcal{RO}$ in the following form

$$\mu := \min_x \mathcal{F}_0(x) + \sum_{i=1}^N c_i \mathcal{F}_i(x) \tag{5} \\ \text{s.t.} \quad \mathcal{F}_i(x) \leq 0 , \ i_{[N]}^+ ,$$

where

$$\mathcal{F}_i(x) := \max_{u_i \in \mathcal{U}_i} f_i(x, u_i) , \ i_{[N]} . \tag{6}$$

In this paper, we often call (6) as the "lower optimization problems", and the minimization in (5) as the "upper optimization".

As already mentioned, our formulation includes the typical robust optimization formulation $\mathcal{RO}_0$

$$\mu = \quad \min_x f_0(x) \\ s.t. \quad \mathcal{F}_i(x) \leq 0, \ i_{[N]}^+ \quad , \tag{7}$$

or the form below that is popular in the machine learning context [35], [36]

$$\mu = \min_x \max_{u \in \mathcal{U}} f_0(x, u) . \tag{8}$$

We finally introduce the following assumption valid for most of the paper.

Assumption 3. A3 $c_i > 0$ for $i_{[N]}^+$.

Remark 3. *The assumption [A3] is made for technical reasons to simplify and streamline the derivations. It is used to impose arbitrarily small regularization terms in order to provide convergence proof in the event that some of the robust constraints are inactive. On the other hand, if a constraint is active, the corresponding c_is can be zero. So, all c_is can be equal to zero under the classical but stronger assumption that all the robust constraints are active.*

A. Robust feasible solution and robust counterpart

A meaningful solution to $\mathcal{RO}$ problem (4) has to be immune against the uncertainties in the sense that the solution vector x should satisfy the constraints for all u_i's within the uncertainty sets[3]. Such vector x is called a robust feasible solution (RFS). One approach to solving the problem (4) is to try to compute (6) in closed form.

---

[2]Since the uncertainty set is compact, the supremum is attained within the set; therefore, we can replace "sup" with "max" in our formulation.

[3]Similar to what is meant by feasibility in Robust Control [37].

For various important combinations of constraint functions f_i and uncertainty sets $\mathcal{U}_i$, $(i > 0)$ it is possible to obtain an explicit convex function $\mathcal{F}_i$ [2]. A classical example is when f_i is linear in x for fixed u_i and linear in u_i for fixed x, while $\mathcal{U}_i$ is an ellipsoidal set. Then, $\mathcal{F}_i$ can be easily derived as an explicit second-order conic function. In this case, $\mathcal{RO}$ problem (4) becomes a nominal optimization problem (not affected by uncertainty) known as the explicit Robust Counterpart (RC):

$$\min_{\mathcal{F}_i(x) \leq 0} f_0(x) . \tag{9}$$

As shown in [32], the RC is always a convex optimization problem under Assumption 1. While the RC is known for important classes of problems as described in [3], this approach requires problem-specific derivations; moreover, RC is generally difficult to find (See Section VII for an example). Instead, our proposed approach has a dynamical system that simultaneously finds the best RFS and the worst parameters u_i' s, independently of the specifics of the constraint functions and the uncertainty sets.

IV. DUALITY, KKT CONDITIONS AND SADDLE PROPERTY

The basic idea of this paper is to combine the usually separated and nested optimization (5) and (6) into one Lagrangian optimization.

To derive the Lagrangian function for $\mathcal{RO}$ problem (4), we introduce Lagrange multipliers $\lambda_i \geq 0$ for $i_{[N]}^+$, and let $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_N)^\top \in \mathbb{R}_+^N$ where the non-negative orthant of $\mathbb{R}^N$ is denoted by $\mathbb{R}_+^N$. Note that μ in $\mathcal{RO}$ problem (4) can be written in short-hand notation as follows

$$\mu = \min_x \max_{\lambda \geq 0} \mathcal{F}_0(x) + \sum_{i=1}^N (c_i + \lambda_i)\, \mathcal{F}_i(x) , \tag{10}$$

where $\mathcal{F}_i(x)$ is given by (6). By introducing Lagrange multipliers $v_{ij} \geq 0$ for the maximization problem in (6) and defining

$$v_i := (v_{i1}, \cdots, v_{iK_i})^\top \in \mathbb{R}_+^{K_i}, \ h_i := (h_{i1}, \cdots, h_{iK_i})^\top ,$$

the lower level strong duality (according to Assumption 2) yields

$$\mathcal{F}_i(x) = \max_{u_i \in \mathbb{R}^{m_i}} \min_{v_i \geq 0} f_i(x, u_i) - v_i^\top h_i(u_i) . \tag{11}$$

Defining the Lagrangian $\tilde{\mathcal{L}}_i : \mathbb{R}^n \times \mathbb{R}^{m_i} \times \mathbb{R}^{K_i} \to \mathbb{R}$ for the lower level maximization problem (11) as

$$\tilde{\mathcal{L}}_i(x, u_i, v_i) := f_i(x, u_i) - v_i^\top h_i(u_i), \ i_{[N]}^+, \tag{12}$$

and

$$\tilde{\mathcal{L}}_0(x, u_0, v_0) := f_0(x, u_0) - v_0^\top h_0(u_0) . \tag{13}$$

In the optimization problem (10), μ can be written as

$$\mu = \min_x \max_{\lambda \geq 0} \max_{u_0} \min_{v_0 \geq 0} \tilde{\mathcal{L}}_0(x, u_0, v_0) \tag{14}$$
$$+ \sum_{i=1}^N (c_i + \lambda_i) \max_{u_i} \min_{v_i \geq 0} \tilde{\mathcal{L}}_i(x, u_i, v_i) .$$

or combining the independent maximizations and minimizations, and collecting the variables, we have

$$\mu = \min_x \max_{\lambda \geq 0} \max_u \min_{v \geq 0} \tag{15}$$
$$\left(\tilde{\mathcal{L}}_0(x, u_0, v_0) + \sum_{i=1}^N (c_i + \lambda_i)\, \tilde{\mathcal{L}}_i(x, u_i, v_i) \right) .$$

Hence, we showed the equivalence of (4) and (14). The complete Lagrangian $\mathcal{L}(x, \lambda, u, v)$ for $\mathcal{RO}$ problem is derived as

$$\mathcal{L}(x, \lambda, u, v) := f_0(x, u_0) - v_0^\top h_0(x, u_0)$$
$$+ \sum_{i=1}^N (c_i + \lambda_i)\big(f_i(x, u_i) - v_i^\top h_i(x, u_i)\big) , \tag{16}$$

which leads to

$$\mu = \min_x \max_{\lambda \geq 0} \max_u \min_{v \geq 0} \mathcal{L}(x, \lambda, u, v) .$$

From Remark 2, strong duality holds for both upper and lower optimizations. Thus, we can switch the order of max and min as follows

$$\mu = \max_{\lambda \geq 0} \ \min_x \min_{v \geq 0} \max_u \mathcal{L}(x, \lambda, u, v) . \tag{17}$$

Definition 1. *We denote an optimal solution of (17), $z^\star := (x^\star, \lambda^\star, u^\star, v^\star)$, as an "optimal $\mathcal{RO}$ solution", where $(x^\star, u^\star)$ is an optimal solution for (4), and $(\lambda^\star, v^\star)$ are optimal values for the corresponding dual variables according to the principles in [33, Section 5.9.1].*

A. KKT optimality conditions

As strong duality holds for $\mathcal{RO}$ problem, any optimal solution, $z^\star = (x^\star, \lambda^\star, u^\star, v^\star)$, satisfies the following Karush-Kuhn-Tucker (KKT) conditions, and vice-versa [33]

$$\nabla_x f_0(x^\star, u_0^\star) + \sum_{i=1}^N (c_i + \lambda_i^\star) \nabla_x f_i(x^\star, u_i^\star) = 0, \tag{18}$$

$$\nabla_{u_i} f_i(x^\star, u_i^\star) - v_i^{\star\top} \nabla_{u_i} h_i(u_i^\star) = 0, \ i_{[N]}, \tag{19}$$

$$v_{ij}^\star \geq 0, \ h_{ij}(u_i^\star) \leq 0, \ v_{ij}^\star h_{ij}(u_i^\star) = 0, \ j_{[K_i]}^+, i_{[N]} \tag{20}$$

$$\lambda_i^\star \geq 0, \ f_i(x^\star, u_i^\star) \leq 0, \ \lambda_i^\star f_i(x^\star, u_i^\star) = 0, \ i_{[N]}^+ \tag{21}$$

where $\nabla_x f$ is the notation for the gradient of a function f w.r.t. x.

B. Saddle property of the optimal $\mathcal{RO}$ solution

As the $\mathcal{RO}$ problem has two levels, we have different saddle property related to the bi-level optimization problem. For the $\mathcal{RO}$ problem as a whole, there is a saddle property based on the Lagrangian function (16), which we call the $\mathcal{RO}$ saddle property. In the context of Lagrangian duality, a saddle point of the Lagrangian function is a point where the function is minimized with respect to the convex variables and maximized with respect to the concave variables. The following lemma addresses the connection between the optimal $\mathcal{RO}$ solution and $\mathcal{RO}$ saddle property (proof in the appendix).

Lemma 1. *[Saddle property] Let $z^\star = (x^\star, \lambda^\star, u^\star, v^\star)$ be an optimal $\mathcal{RO}$ solution. Then, for all $x, \lambda \geq 0, u, v \geq 0$, $z^\star$ satisfies the $\mathcal{RO}$ saddle property, namely,*

$$\mathcal{L}(x^\star, \lambda, u, v^\star) \leq \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) \leq \mathcal{L}(x, \lambda^\star, u^\star, v) . \quad (22)$$

Proving Lemma 1 is one of the main contributions of our paper. Lemma 1 proves that the saddle point property holds true for the $\mathcal{RO}$ problem despite the fact that the Lagrangian in (22) is not jointly convex-concave. The proof is given in the appendix and this lemma is being used to show the monotonicity property of the Lyapunov function in Lemma 3.

V. DYNAMICAL SYSTEM SOLVING $\mathcal{RO}$

So far, we have characterized the "optimization" properties of the Robust Optimization problem under study. In this paper, we are interested in understanding if there is a continuous-time dynamical system that can solve $\mathcal{RO}$ and how it would operate. Our main motivations are two: (1) Understanding from a dynamical system perspective how $\mathcal{RO}$ can be solved. (2) Studying how physically interacting systems with very simple capabilities and or "intelligence" can cooperate to solve complex optimization problems (learning, estimation, and decision) well outside the single element capabilities. While there are now answers to these questions for large classes of convex optimization problems [30], [38], this is the first work, to the best of our knowledge, that addresses $\mathcal{RO}$ problems. Our task turned out to be quite non-trivial, as explained below.

The basic method to obtain a continuous-time dynamics that solves a constrained convex optimization problem goes back to [29]. The main idea is quite intuitive. The primal dynamics evolves with the negative gradient of the problem's Lagrangian function, w.r.t, the primal variable, that is x, while the dual dynamics evolves with the positive gradient of the Lagrangian w.r.t to the dual variables, that is, λ_is. The primal descent and dual ascent dynamics is globally converging to the optimal solution under minor assumptions. The proof is based on a simple quadratic Lyapunov function. However, from a system point of view, it is the passivity of the gradient of a convex function that provides the convergence mechanism [39], [40].

For the $\mathcal{RO}$ problem, the standard approach does not work due to the nested structure of optimization. Thus, the natural match between intuitive primal-descent dual-ascent dynamics via a traditional quadratic Lyapunov function is broken. It turns out that it is not easy to find the right combination of dynamics and Lyapunov function that show global convergence.

This section presents the main contribution of the paper: a continuous-time dynamical system named "$\mathcal{RO}$ dynamics" (together with an appropriate "Lyapunov function") whose solutions globally converge to a solution of $\mathcal{RO}$ problem. Let

$$M = \sum_{i=0}^{N} m_i , \; K = \sum_{i=0}^{N} K_i .$$

Consider the following $\mathcal{RO}$ dynamics defined on $\mathbb{S} := \mathbb{R}^n \times \mathbb{R}_+^N \times \mathbb{R}^M \times \mathbb{R}_+^K$

$$\begin{cases} \dot{x} = -\nabla_x f_0(x, u_0) - \sum_{i=1}^{N} (c_i + \lambda_i) \nabla_x f_i(x, u_i) \\ \dot{\lambda}_i = [f_i(x, u_i) - v_i^\top h_i(u_i)]_{\lambda_i}^+, \; i_{[N]}^+ \\ \dot{u}_i = \nabla_{u_i} f_i(x, u_i) - \sum_{j=1}^{K_i} v_{ij} \nabla_{u_i} h_{ij}(u_i), \; i_{[N]} \\ \dot{v}_0 = [h_0(u_0)]_{v_0}^+ \\ \dot{v}_i = [(c_i + \lambda_i) \, h_i(u_i)]_{v_i}^+, \; i_{[N]}^+ \end{cases} . \quad (23)$$

A. Examples and comments

It is interesting to describe the structure using special examples.

1) min-max: Consider the following $\mathcal{RO}$ problem with no constraints

$$\mu = \min_x \max_{u_0 : h_0(u_0) \leq 0} f_0(x, u_0) .$$

Such problems are popular in machine learning. In this case, the continuous-time dynamical system is given by

$$\begin{cases} \dot{x} = -\nabla_x f_0(x, u_0) \\ \dot{u}_0 = \nabla_{u_0} f_0(x, u_0) - v_0^\top \nabla_{u_0} h_0(u_0) \\ \dot{v}_0 = [h_0(u_0)]_v^+ \end{cases} .$$

Above dynamic is gradient-descent in x while gradient-ascent in u_0 and v_0. We must note that often in machine learning applications, the constraints on u_0 are simple boxes, and the dual variables v_0 are omitted in place of a simple set projection.

2) One uncertain constraint: Another simple example is the following $\mathcal{RO}$ problem which follows the standard setting where the constraint is active and c_1 is zero.

$$\begin{aligned} \mu \quad &= \quad \min_x f_0(x) \\ s.t. \quad &\max_{h_1(u_1) \leq 0} f_1(x, u_1) \leq 0 \end{aligned}$$

The dynamical system equations are

$$\begin{cases} \dot{x} = -\nabla_x f_0(x) - \lambda_1 \nabla_x f_1(x, u_1) \\ \dot{\lambda}_1 = [f_1(x, u_1) - v_1^\top h_1(u_1)]_{\lambda_1}^+ \\ \dot{u}_1 = \nabla_{u_1} f_1(x, u_1) - v_1^\top \nabla_{u_1} h_1(u_1) \\ \dot{v}_1 = [\lambda_1 h_1(u_1)]_{v_1}^+ \end{cases} .$$

This example shows the following significant differences with the primal-dual dynamical system for solving standard optimization problems appeared in [30], [41].

1) First, the $\mathcal{RO}$ dynamics has additional states associated with the worst-case constraint u_i and the associated multiplier v_1.
2) Another difference is that the $\mathcal{RO}$ dynamics vector field is not completely derived as negative/positive gradients of the Lagrangian function $\mathcal{L}$. In particular, the vector

field for the state u_1 is not obtained as the positive gradient of the Lagrangian function $\mathcal{L}$.

3) Finally, we note the presence of λ_1, the dual variable, in the upper optimization in the dynamics of v_1, the dual variable of the lower optimization. At first glance, this seems strange, since one could expect the differential equations for u_1 and v_1 to be simply the primal-ascent and dual-descent, respectively, of the lower optimization problem. This point will be discussed further after we present the stability results in Remark 4.

B. Equivalence of the KKT and equilibrium points

The following lemma relates the optimal (KKT) points of (4) and the equilibrium points of the dynamics of $\mathcal{RO}$.

Lemma 2. *[Optimal solution and equilibrium point] Under Assumptions 1 and 2, any optimal $\mathcal{RO}$ solution based on Definition 1 is an equilibrium point of $\mathcal{RO}$ dynamics (23) and vice versa.*

Proof. Any equilibrium point, $\bar{z} = (\bar{x}, \bar{\lambda}, \bar{u}, \bar{v})$ of $\mathcal{RO}$ dynamics (23) satisfies

$$\nabla_x f_0(\bar{x}, \bar{u}_0) + \sum_{i=1}^{N}(c_i + \bar{\lambda}_i)\nabla_x f_i(\bar{x}, \bar{u}_i) = 0\,,$$
$$f_i(\bar{x}, \bar{u}_i) - \bar{v}_i^\top h_i(\bar{u}_i) \leq 0,\ \bar{\lambda}_i \geq 0\,,$$
$$\bar{\lambda}_i(f_i(\bar{x}, \bar{u}_i) - \bar{v}_i^\top h_i(\bar{u}_i)) = 0\,,$$
$$\nabla_{u_i} f_i(\bar{x}, \bar{u}_i) - \bar{v}_i^\top \nabla_{u_i} h_i(\bar{u}_i) = 0\,,$$
$$h_{0j}(\bar{u}_0) \leq 0,\ \bar{v}_{0j} \geq 0,\ \bar{v}_{0j}h_{0j}(\bar{u}_0) = 0\,,$$
$$(c_i + \bar{\lambda}_i)h_{ij}(\bar{u}_i) \leq 0,\ \bar{v}_{ij} \geq 0,\ \bar{v}_{ij}(c_i + \bar{\lambda}_i)h_{ij}(\bar{u}_i) = 0\,,$$

for $i^+_{[N]}, j^+_{[K_i]}$, while any optimal point satisfies below KKT conditions

$$\nabla_x f_0(x^\star, u_0^\star) + \sum_{i=1}^{N}(c_i + \lambda_i^\star)\nabla_x f_i(x^\star, u_i^\star) = 0, \tag{24}$$

$$\nabla_{u_i} f_i(x^\star, u_i^\star) - v_i^{\star\top}\nabla_{u_i} h_i(u_i^\star) = 0, \quad i \in [N], \tag{25}$$

$$v_{ij}^\star \geq 0,\ h_{ij}(u_i^\star) \leq 0,\ v_{ij}^\star h_{ij}(u_i^\star) = 0,\ j \in [K_i],\ i \in [N] \tag{26}$$

$$\lambda_i^\star \geq 0,\ f_i(x^\star, u_i^\star) \leq 0,\ \lambda_i^\star f_i(x^\star, u_i^\star) = 0,\ i \in [N] \tag{27}$$

Substituting $z^\star$ for $\bar{z}$, and using the fact that $(c_i + \lambda_i^\star) > 0$ for all $i^+_{[N]}$, it is immediate to verify that the optimal point $z^\star$, satisfying (24)-(27), also satisfies the above equilibrium conditions and therefore is an equilibrium point of $\mathcal{RO}$ dynamics (23).

On the other hand, since $c_i > 0$ and $c_i + \bar{\lambda}_i > 0$, for $i^+_{[N]}$; then, $h_{ij}(\bar{u}_i) \leq 0$, and $\bar{v}_{ij}h(\bar{u}_{ij}) = 0$ for $i_{[N]}, j^+_{[K_i]}$. Substituting these properties in the rest of the equilibrium conditions, we see that $\bar{z}$ satisfies the KKT conditions therefore is optimal for $\mathcal{RO}$.

□   □

We denote the $\mathcal{RO}$ dynamics (23) compactly with the shorthand notation $\dot{z} = \mathcal{Z}^{\mathcal{RO}}(z)$.

C. Lyapunov Function

In this subsection, we present the Lyapunov function[4] as the second important contribution of the paper that is tightly connected with the proposed dynamical system.

Lemma 3. *[Monotonicity property] Let $z^\star = (x^\star, \lambda^\star, u^\star, v^\star)$ be an optimal $\mathcal{RO}$ solution (Definition 1).*

Let $V : \mathbb{S} \to \mathbb{R}_+$ defined as

$$V = \frac{1}{2}\big(\|x - x^\star\|^2 + \|\lambda - \lambda^\star\|^2 + \|u_0 - u_0^\star\|^2 +$$
$$\sum_{i=1}^{N}(c_i + \lambda_i^\star)\|u_i - u_i^\star\|^2 + \sum_{i=0}^{N}\|v_i - v_i^\star\|^2\big)\,, \tag{28}$$

then, the Lie-derivative of V along $\mathcal{Z}^{\mathcal{RO}}$ at $z = (x, \lambda, u, v)$ is $\nabla V(z)^\top \dot{z} \leq 0$.

Proof. Note that $\nabla V(z)^\top$ equals to

$$(x - x^\star)^\top\big(-\nabla_x f_0(x, u_0) - \sum_{i=1}^{N}(c_i + \lambda_i)\nabla_x f_i(x, u_i)\big)$$
$$+ \sum_{i=1}^{N}(\lambda_i - \lambda_i^\star)[f_i(x, u_i) - v_i^\top h_i(u_i)]_{\lambda_i}^+$$
$$+ (u_0 - u_0^\star)^\top(\nabla_{u_0} f_0(x, u_0) - v_0^\top \nabla_{u_0} h_0(u_0))$$
$$+ \sum_{i=1}^{N}(c_i + \lambda_i^\star)(u_i - u_i^\star)^\top(\nabla_{u_i} f_i(x, u_i) - v_i^\top \nabla_{u_i} h_i(u_i))$$
$$+ (v_0 - v_0^\star)^\top[h_0(u_0)]_{v_0}^+ + \sum_{i=1}^{N}(v_i - v_i^\star)^\top[(c_i + \lambda_i)\,h_i(u_i)]_{v_i}^+\,. \tag{29}$$

By convex and concave under-estimator properties according to Assumption 1, one can write [33, Section 3.1.3]

$$(x^\star - x)^\top\nabla_x f_i(x, u_i) \leq f_i(x^\star, u_i) - f_i(x, u_i)\,, \tag{30}$$
$$(u_i - u_i^\star)^\top\nabla_{u_i} f_i(x, u_i) \leq f_i(x, u_i) - f_i(x, u_i^\star)\,, \tag{31}$$
$$(u_i^\star - u_i)^\top\nabla_{u_i} h_{ij}(u_i) \leq h_{ij}(u_i^\star) - h_{ij}(u_i)\,. \tag{32}$$

Moreover, using the fact that the projection operator is non-expansive, we have that

$$(\lambda_i - \lambda_i^\star)[f_i(x, u_i) - v_i^\top h_i(u_i)]_{\lambda_i}^+$$
$$\leq (\lambda_i - \lambda_i^\star)(f_i(x, u_i) - v_i^\top h_i(u_i))\,, \tag{33}$$
$$(v_0 - v_0^\star)^\top[h_0(u_0)]_{v_0}^+ \leq (v_0 - v_0^\star)^\top h_0(u_0), \tag{34}$$
$$(v_i - v_i^\star)^\top[(c_i + \lambda_i)\,h_i(u_i)]_{v_i}^+$$
$$\leq (v_i - v_i^\star)^\top(c_i + \lambda_i)\,h_i(u_i)\,. \tag{35}$$

---

[4]With slight abuse of notation.

By substitution, we get

$$\nabla V(z)^\top \dot{z} \leq$$

$$f_0(x^\star, u_0) - f_0(x, u_0) + \sum_{i=1}^{N}(c_i + \lambda_i)(f_i(x^\star, u_i) - f_i(x, u_i))$$

$$+ \sum_{i=1}^{N}(c_i + \lambda_i) - (c + \lambda_i^\star))(f_i(x, u_i) - v_i^\top h_i(u_i))$$

$$+ f_0(x, u_0) - f_0(x, u_0^\star) + v_0^\top(h_0(u_0^\star) - h_0(u_0))$$

$$+ \sum_{i=1}^{N}(c + \lambda_i^\star)(f_i(x, u_i) - f_i(x, u_i^\star) + v_i^\top(h_i(u_i^\star) - h_i(u_i)))$$

$$+ (v_0 - v_0^\star)^\top h_0(u_0) + \sum_{i=1}^{N}(v_i - v_i^\star)^\top(c_i + \lambda_i)h_i(u_i) .$$

$$(36)$$

After simplification and rearrangement we obtain

$$\begin{aligned}\nabla V(z)^\top \dot{z} \leq &\ f_0(x^\star, u_0) - (v_0^\star)^\top h_0(u_0) \\ &- f_0(x, u_0^\star) + v_0^\top h_0(u_0^\star) \\ &+ \sum_{i=1}^{N}(c + \lambda_i)\Big(f_i(x^\star, u_i) - (v_i^\star)^\top h_i(u_i))\Big) \\ &- \sum_{i=1}^{N}(c + \lambda_i^\star)\Big(f_i(x, u_i^\star) - v_i^\top h_i(u_i^\star))\Big) .\end{aligned}$$

Adding and subtracting $\mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star)$ yields

$$\begin{aligned}\nabla V(z)^\top \dot{z} \leq &\ \mathcal{L}(x^\star, \lambda, u, v^\star) - \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) \quad (37) \\ &+ \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) - \mathcal{L}(x, \lambda^\star, u^\star, v) .\end{aligned}$$

Note that from Lemma 1 (saddle property),

$$\begin{aligned}\mathcal{L}(x^\star, \lambda, u, v^\star) - \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) &\leq 0 , \\ \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) - \mathcal{L}(x, \lambda^\star, u^\star, v) &\leq 0 .\end{aligned}$$

Hence,

$$\begin{aligned}\nabla V(z)^\top \dot{z} \leq &\ \mathcal{L}(x^\star, \lambda, u, v^\star) - \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) \quad (38) \\ &+ \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) - \mathcal{L}(x, \lambda^\star, u^\star, v) \leq 0 .\end{aligned}$$

$$\square$$
$$\square$$

Remark 4. *To elaborate on the issue with the gradient descent-ascent primal-dual dynamics for robust optimization and the reason we need a modified $\mathcal{RO}$ dynamics, note that although $\mathcal{RO}$ problem is jointly-convex in x and v, it is not jointly concave in λ and u. Hence, the primal-dual dynamics (with λ_i in $\dot{u}_i$ dynamics) is not converging to the $\mathcal{RO}$ solution if λ_i reaches zero before u_i reaches its optimal value. In other words, the u_i dynamics freezes when λ_i is near zero. That is why we modified the primal-dual dynamics with the new $\mathcal{RO}$ dynamics which changes the stability results and proofs significantly. The reader should notice how the unintuitive dynamics (see discussion in Section V-A) fits the computation of the Lie derivative based on the unconventional function V which is quadratic in the error w.r.t. an optimal solution $z^\star = (x^\star, \lambda^\star, u^\star, v^\star)$ but weights the error in the u variables*

w.r.t. the optimal dual variables, $\lambda^\star$, of the upper optimization. Discovering both the dynamical system and the Lyapunov function is our main achievement in this paper, as they do not follow from simple extension of existing approaches.

Remark 5. *The convergence speed analysis of $\mathcal{RO}$ dynamics in the discrete-time version of our algorithm appears in [42].*

D. Solutions: Existence, Uniqueness, and Continuity

We need to guarantee that the switching dynamical system (23) has well defined solutions. To do so, we show that our system satisfies the conditions for applicability of existing results [41]. The development of this section is in the Appendix Section IX-B.

E. Convergence Analysis

Considering $\gamma(t)$ as a solution of $\mathcal{RO}$ dynamics (23) defined on the time interval $[0, \infty)$, omega-limit set is defined as

$$\begin{aligned}\Omega(\gamma) := \{y \in \mathbb{S} \mid \exists \{t_k\}_{k=1}^{\infty} \subset [0, \infty) \\ \text{with } \lim_{k \to \infty} t_k = \infty \text{ and } \lim_{k \to \infty} \gamma(t_k) = y\} . \quad (39)\end{aligned}$$

The next lemma presents an invariant property for the omega-limit set of any solution of the $\mathcal{RO}$ dynamics, which contributes in showing the convergence result.

Lemma 4. *[41, Lemma 4.4][Invariance of omega-limit set] The omega-limit set of any solution of the $\mathcal{RO}$ dynamics (23) starting from any point $\mathbb{S}$ is invariant.*

Finally, we have the main convergence result.

Theorem 4. *[Convergence] Under Assumptions 1 and 2, the trajectories of $\mathcal{RO}$ dynamics (23) converge to an optimal $\mathcal{RO}$ solution for any initial condition in $\mathbb{S} = \mathbb{R}^n \times \mathbb{R}_+^N \times \mathbb{R}^M \times \mathbb{R}_+^K$. In other words, the set of optimal $\mathcal{RO}$ solutions is globally asymptotically stable on $\mathbb{S}$ under $\mathcal{RO}$ dynamics (23).*

Proof. Let $z^\star = (x^\star, \lambda^\star, u^\star, v^\star)$ be an optimal $\mathcal{RO}$ solution under Assumptions 1 and 2. Lemma 6 implies that a unique solution of $\mathcal{RO}$ dynamics $\mathcal{Z}^{\mathcal{RO}}$ exists starting from any point in the compact set $\mathbb{P} = V^{-1}(\leq \delta) \cap \mathbb{S}$ for any $\delta > 0$ [5]. We now call this solution $\bar{\gamma}(t)$.

From Lemma 4, the omega-limit set of $\bar{\gamma}(t)$ is invariant and invariance principle for discontinuous Caratheodory systems [41, Proposition 2.1] (simplified version of [43, Proposition 3]) implies that $\bar{\gamma}(t)$ converges to the largest invariant set in $\text{cl}(\mathcal{M})$ where $\mathcal{M} := \{z \in \mathbb{P} \mid \mathfrak{L}_{\mathcal{Z}^{\mathcal{RO}}} V(z) = 0\}$.

Next, we characterize the set $\mathcal{M}$ where $\mathfrak{L}_{\mathcal{Z}^{\mathcal{RO}}} V(z) = 0$ by defining

$$\begin{aligned}\bar{\mathcal{M}} := \{z \in \mathbb{P} \mid \lambda \geq 0, v_i \geq 0, \forall i, \\ \mathcal{L}(x^\star, \lambda, u, v^\star) - \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) = 0, \\ \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) - \mathcal{L}(x, \lambda^\star, u^\star, v) = 0\} . \quad (40)\end{aligned}$$

From the inequality in (38), it follows that $\mathcal{M} \subseteq \bar{\mathcal{M}}$. We then prove that every point in $\bar{\mathcal{M}}$ is an optimal $\mathcal{RO}$ solution.

[5]As V is radially unbounded, the set $\mathbb{P}$ is always compact.

From the strict convexity of f, it follows that $x = x^\star$ on $\bar{\mathcal{M}}$. From (40), any point in $\bar{\mathcal{M}}$ achieves the optimal cost of $\mathcal{RO}$. Let $\bar{z} = (x^\star, \bar\lambda, \bar{u}, \bar{v}) \in \bar{\mathcal{M}}$. Then, in general,

$$\mathcal{L}(x^\star, \bar\lambda, \bar{u}, v^\star) \leq \mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) ,$$
$$\mathcal{L}(x^\star, \lambda^\star, u^\star, v^\star) \leq \mathcal{L}(x^\star, \lambda^\star, u^\star, \bar{v}) .$$

But since $\bar{z} \in \bar{\mathcal{M}}$, the equality must hold for the above equations. This means that

$$\bar{v} = \arg \max_{v \geq 0} \ \mathcal{L}(x^\star, \lambda^\star, u^\star, v) ,$$
$$(\bar\lambda, \bar{u}) = \arg \max_{u, \lambda \geq 0} \ \mathcal{L}(x^\star, \lambda, u, v^\star) .$$

Therefore, $\bar{z}$ is an optimal $\mathcal{RO}$ solution. Therefore, any point in $\bar{\mathcal{M}}$ is an optimal $\mathcal{RO}$ solution. On the other hand, any optimal $\mathcal{RO}$ solution is an equilibrium of $\mathcal{RO}$ dynamics (23) according to Lemma 2 and therefore is in $\mathcal{M}$. Thus, $\mathcal{M} = \bar{\mathcal{M}}$. As $\delta > 0$ is arbitrary, we conclude that the set of optimal $\mathcal{RO}$ solutions is globally asymptotically stable on $\mathbb{S}$.

Note that $\mathcal{M}$ can contain an uncountable infinite set of points. If the optimal $\mathcal{RO}$ solution is not unique, these correspond to the set of optimal $\mathcal{RO}$ solutions and to the set of non-isolated equilibria of $\mathcal{RO}$ dynamics (23) from Lemma 2. Moreover, the convergence of each solution in Theorem 4 is to a point in the set of optimal $\mathcal{RO}$ solutions. This means that the omega-limit set of any solution $\gamma(t)$ is a singleton which follows from (39) and Lemma 3. $\qquad\square\qquad\square$

Corollary 1. *Under Assumptions 1 and 2, let $z = (x^\star, \lambda^\star, u^\star, v^\star)$ be an optimal solution. Assume all robust constraints are strictly active, that is, $\lambda_i^\star > 0$, $i_{[N]}^+$. Then, the $\mathcal{RO}$ dynamics (23) converges to an optimal solution.*

Proof. The setup of the corollary satisfies the assumptions of Theorem 4, as $c_i + \lambda_i^\star > 0$, $i_{[N]}^+$. $\qquad\square$

VI. Convergence with Inactive Constraints

The convergence proof is based on the mild assumption that $c_i + \lambda_i^\star > 0$. In this section we consider the optimization with $c_i = 0$, $i_{[N]}$, as

$$\mu := \min_x \ \mathcal{F}_0(x) \tag{41}$$
$$\text{s.t.} \quad \mathcal{F}_i(x) \leq 0 , \ i_{[N]}^+ ,$$

where there are inactive constraints.

In case of inactive constraints, (23) with $c = 0$ empirically converges with the $\lambda' s$ corresponding to inactive constraints converging to zero. However, the proof does not directly apply in this case as $(c_i + \lambda_i^\star) = 0$, for some i, and the Lyapunov function is no longer valid for some $\lambda_i^\star = 0$. Moreover, the u_i and v_i dynamics as well as the interaction with the x dynamics become irrelevant to the convergence but are still part of the dynamics. To address this case, we consider the perturbed problem with $c_i = \varepsilon > 0$ sufficiently small. Then dynamics (23), that is,

$$\begin{cases} \dot{x} = -\nabla_x f_0(x, u_0) - \sum_{i=1}^{N} (\varepsilon + \lambda_i) \nabla_x f_i(x, u_i) \\ \dot\lambda_i = [f_i(x, u_i) - v_i^\top h_i(u_i)]_{\lambda_i}^+, \ i_{[N]}^+ \\ \dot{u}_i = \nabla_{u_i} f_i(x, u_i) - \sum_{j=1}^{K_i} v_{ij} \nabla_{u_i} h_{ij}(u_i), \ i_{[N]} \\ \dot{v}_0 = [h_0(u_0)]_{v_0}^+ \\ \dot{v}_i = [(\varepsilon + \lambda_i) \, h_i(u_i)]_{v_i}^+, \ i_{[N]}^+ \end{cases} \tag{42}$$

converges to some μ_ε and $x_\varepsilon^\star$ from Theorem 4. It turns out that for ε sufficiently small, we can approximate the optimal cost and the optimal solution $x^\star$ arbitrarily well under compactness conditions.

Theorem 5. *For the problem $\mathcal{RO}$ (4) under Assumptions 1 and 2, let $x_\varepsilon^\star$ be the optimal solution where $c = \varepsilon$. Then $\lim_{\varepsilon \to 0} \mu_\varepsilon = \mu_0$, where μ_0 is the optimal cost of problem $\mathcal{RO}$ (4) with $\varepsilon = 0$, that is, (41). Furthermore, assuming the feasible set $\mathcal{C} = \{x \in \mathbb{R}^n \, | \, \mathcal{F}_i(x) \leq 0, \ i_{[N]}^+\}$ is compact, we have*

$$\lim_{\varepsilon \to 0} \| x_\varepsilon^\star - x^\star \| = 0 ,$$

with $x^\star$ be the optimal $\mathcal{RO}$ solution when $\varepsilon = 0$, that is, problem (41) (proof in the appendix).

Finally, we note that the dynamics in (42), is equivalent to the following dynamics by letting $\hat\lambda = \lambda + \varepsilon \mathbf{1}$

$$\begin{cases} \dot{x} = -\nabla_x f_0(x, u_0) - \sum_{i=1}^{N} \hat\lambda_i \nabla_x f_i(x, u_i) \\ \dot{\hat\lambda}_i = [f_i(x, u_i) - v_i^\top h_i(u_i)]_{\hat\lambda_i}^{\varepsilon+}, \ i_{[N]}^+ \\ \dot{u}_i = \nabla_{u_i} f_i(x, u_i) - \sum_{j=1}^{K_i} v_{ij} \nabla_{u_i} h_{ij}(u_i), \ i_{[N]} \\ \dot{v}_0 = [h_0(u_0)]_{v_0}^+ \\ \dot{v}_i = [\hat\lambda_i \, h_i(u_i)]_{v_i}^+, \ i_{[N]}^+ \end{cases} \tag{43}$$

Noting that above dynamics evolves on $\mathbb{S} = \mathbb{R}^n \times \mathbb{R}_{\varepsilon+}^N \times \mathbb{R}^M \times \mathbb{R}_+^K$, it shows that the perturbed dynamics can be obtained by simply perturbing λ_i projections with respect to ε instead of 0, by initializing $\lambda_i \geq \varepsilon > 0$.

VII. Simulations

In MATLAB, several solvers can be used to simulate ordinary differential equations (ODE). In the following simulation examples, "ode15s" is used, which is a solver for stiff problems.

A. Robust Quadratic Programming

Consider a robust QP problem as below

$$\min_{x \in \mathbf{R}^2} \ f(x) := -8x_1 - 16x_2 + x_1^2 + 4x_2^2$$
$$\text{s.t.} \max_{u \in \mathcal{U}} \ (a + Pu)^\top x \leq b , \tag{44}$$

where $x \in \mathbf{R}^2$ is the decision variable, and $a = [1 \ \ 1]^\top$, $P = I_2 \in \mathbf{R}^{2 \times 2}$ and $b = 5$ are given parameters. Variable u

is uncertain, for which, the uncertainty set is described by the intersection of five ellipsoids as below

$$\mathcal{U} := \{u \in \mathbb{R}^2 : h_j(u) \leq 0 \, , \, j_{[5]}^+\} \, ,$$

where $h_j(u) := u^\top Q_j u - 1, \, j_{[5]}^+$. Each Q_j is a symmetric positive semi-definite matrix and $\sum_{j=1}^5 Q_j \succ 0$. In this example, we assume that the following matrices specify ellipsoids

$$Q_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \, Q_2 = \begin{bmatrix} 5 & -2 \\ -2 & 4 \end{bmatrix}, \, Q_3 = \begin{bmatrix} 4 & 4 \\ 4 & 6 \end{bmatrix},$$

$$Q_4 = \begin{bmatrix} 3 & 0 \\ 0 & 8 \end{bmatrix}, \, Q_5 = \begin{bmatrix} 5 & 2 \\ 2 & 4 \end{bmatrix}.$$

The Lagrangian function can be written as

$$\mathcal{L} = f(x) + \lambda\big((a + Pu)^\top x - b - v^\top h(u)\big) \, .$$

where

$$h = [h_1, h_2, h_3, , h_4, h_5]^\top, \; v = [v_1, v_2, v_3, v_4, v_5]^\top.$$

We obtain the following dynamics according to $\mathcal{RO}$ dynamics (43)

$$\begin{cases} \dot{x} = -\begin{bmatrix} 2x_1 - 8 \\ 8x_2 - 16 \end{bmatrix} - \hat{\lambda}\,(a + Pu) \\[2mm] \dot{\hat{\lambda}} = \left[(a + Pu)^\top x - b - v^\top h(u)\right]_{\hat{\lambda}}^{\varepsilon+} \\[2mm] \dot{u} = P^\top x - 2\sum_{j=1}^5 Q_j u v_j \\[2mm] \dot{v} = \left[\hat{\lambda}\, h(u)\right]_v^+ \end{cases}$$

where $\hat{\lambda} = \lambda + \varepsilon$. The trajectories for this system starting from zero initial conditions are shown in Fig. 1. Note that the constraint is active, so we can set ε to zero according to Remark 3. The optimal value of x is $[2.2674, 1.6636]$ and the optimal cost is -28.5452. Five ellipsoids in the uncertainty set are plotted in Fig. 2. The blue star shows the optimal value of u at $[0.4046, 0.0909]$ which lies on the boundary of intersection of two of the ellipsoids corresponding to Q_3 and Q_5. Also note that for positive values of ε, the λ trajectory and convergence value may change but the solution x remains the same as the constraint is active.

The solution for the $\mathcal{RO}$ problem (44) can be verified by other methods. We can apply the technique in [22], in which random instances of uncertainties are sampled from the uncertainty set. Each of the uncertainty instances corresponds to a constraint. This results in a deterministic optimization problem with finitely many constraints. By picking 1115 samples from the intersection of ellipsoids uncertainty set and solving the derived deterministic optimization problem with 1115 constraints by CVX, the approximate robust optimal solution and the approximate optimal cost value are found $[2.2693, 1.6770]$ and -28.5873 respectively. The optimal cost function of our method has a larger value compared to the method in [22], as the latter approximates the RFS by taking finite samples. Hence, it cannot find the best value of u precisely and the optimal u; therefore, the optimal cost is approximated. The second method verifying our solution is robust counterpart [2]. Deriving the robust counterpart and solving the deterministic problem by CVX returns the same
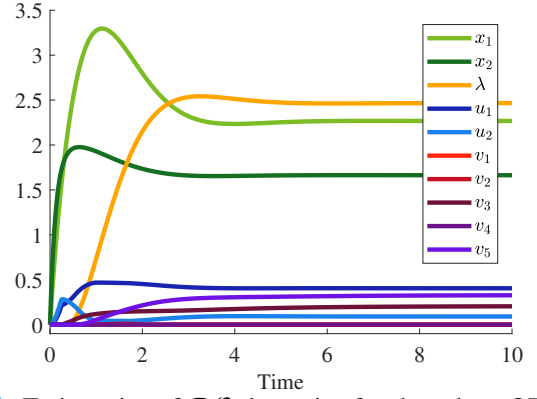

Fig. 1: Trajectories of $\mathcal{RO}$ dynamics for the robust QP problem with intersection of ellipsoids uncertainty set in Example A.
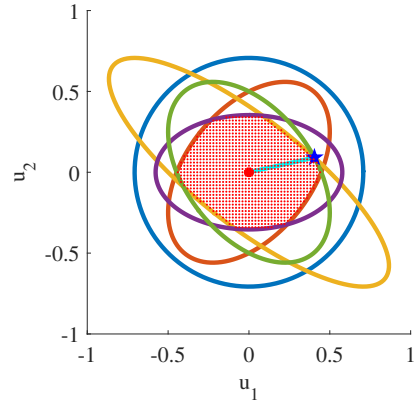

Fig. 2: Uncertainty set for the Example A plotted in u_2-u_1 space. The big red point at the origin represents the initial point of u in $\mathcal{RO}$ dynamics and the blue star depicts the optimal value of u on the intersection of two ellipsoids derived by our method. The small red grid points are the 1115 sampled points from the uncertainty set in randomized scenario method [22] to compare with our method.

solution compared to the $\mathcal{RO}$ dynamics. Our method works on the main $\mathcal{RO}$ problem to find the optimal solution without transforming it to a deterministic equivalent which can be a hassle and even impossible task, as will be shown in the following example.

B. Robust Nonlinear Optimization with no RC

Consider the following robust nonlinear optimization problem

$$\min_{x \in \mathbb{R}^2} \; f(x) := \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - 2)^2 \quad (45)$$

$$\text{s.t.} \; \max_{u \in \mathcal{U}} u^\top \begin{bmatrix} e^{x_1^2} \\ e^{x_2^2} \end{bmatrix} \leq b \, ,$$

where $x \in \mathbb{R}^2$, and $u = [u_1, u_2]^\top \in \mathbb{R}^2$, for which, the strictly convex uncertainty set is described by

$$\mathcal{U} := \{u \in \mathbb{R}^2 : e^{u_j^2} + u_j e^{\frac{1}{u_j}} \leq \rho_j \, , \, j = 1, 2\} \, . \quad (46)$$

As stated in [21] and [19], there is no closed-form convex conjugate for the constraint (convex in x) in (45) and no

closed-form conjugate for the convex uncertainty set in (46). This is the third case in [19, Table 1], for which there is no known method for obtaining RC. However, by writing the Lagrangian function as

$$\mathcal{L} = \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - 2)^2$$
$$+ \lambda \left(u^\top \left[\begin{array}{c} e^{x_1^2} \\ e^{x_2^2} \end{array} \right] - b - v^\top \left[\begin{array}{c} e^{u_1^2} + u_1 e^{\frac{1}{u_1}} - \rho_1 \\ e^{u_2^2} + u_2 e^{\frac{1}{u_2}} - \rho_2 \end{array} \right] \right),$$

where $h_j(u_j) = e^{u_j^2} + u_j e^{\frac{1}{u_j}} - \rho_j$ for $j = 1, 2$ as defined in the previous example, we can form the $\mathcal{RO}$ dynamics for the RNO problem (45) as

$$\dot{x} = - \left[\begin{array}{c} x_1 - 1 \\ x_2 - 2 \end{array} \right] - \hat{\lambda} \left[\begin{array}{cc} 2x_1 e^{x_1^2} & 0 \\ 0 & 2x_2 e^{x_2^2} \end{array} \right] u,$$

$$\dot{\hat{\lambda}} = \left[u^\top \left[\begin{array}{c} e^{x_1^2} \\ e^{x_2^2} \end{array} \right] - b - \sum_{j=1}^2 \left(v_j(e^{u_j^2} + u_j e^{\frac{1}{u_j}} - \rho_j) \right) \right]_{\hat{\lambda}}^{\varepsilon+},$$

$$\dot{u} = \left[\begin{array}{c} e^{x_1^2} \\ e^{x_2^2} \end{array} \right] - \sum_{j=1}^2 (2u_j e^{u_j^2} + e^{\frac{1}{u_j}} - \frac{1}{u_j} e^{\frac{1}{u_j}}) v_j,$$

$$\dot{v} = \left[\hat{\lambda} \left[\begin{array}{c} e^{u_1^2} + u_1 e^{\frac{1}{u_1}} - \rho_1 \\ e^{u_2^2} + u_2 e^{\frac{1}{u_2}} - \rho_2 \end{array} \right] \right]_v^+,$$

according to $\mathcal{RO}$ dynamics (43) where $\hat{\lambda} = \lambda + \varepsilon$. Similarly to the previous example, we can set ε to zero according to Remark 3 as the constraint is active. Fig. 3 shows the trajectory plot for $\rho_1 = 10$, $\rho_2 = 20$, and all the states initialized at 1. The robust optimal solution is $[0.5271, 0.7916]$ and the optimal cost value is 0.8419. We observe that the optimal value for u, which is $[1.4020, 1.6824]$ lies on the boundary of the uncertainty set. For positive values of ε, the trajectory and convergence value of λ may change, but the solution x remains the same as the constraint is active.

Applying the sampling method in [22] with 168 sampled points in the grid $0.2 \leq u_1 \leq 2$ and $0.2 \leq u_2 \leq 2$ within the uncertainty sets, and solving the many-constrained deterministic problem, the approximate robust optimal solution will be $[0.5376, 0.8193]$, and the approximate optimal cost value is 0.8039. We again observe that our solution is more exact with a larger cost function value. It is worth mentioning that the solution will be closer to ours by taking more samples in the randomized scenario approach with the cost of adding more constraints to the optimization problem[6]. The reader may imagine how inefficient this approach is when the main $\mathcal{RO}$ problem has many constraints. The robust counterpart method in [2] nor the one in [19, Table 1] cannot find the solution as there is no closed-form convex conjugate available for this complicated example. However, the $\mathcal{RO}$ dynamics easily finds the robust optimal solution.

C. Robust Dynamic Location Problem

We now consider the optimal cooperative and robust self-placement of autonomous vehicles, modeled as first-order

[6]For this example, the running time of the sampling scenario method with CVX is more than 200 times the running time for the $\mathcal{RO}$ dynamics method. Furthermore, the sampling method gives less accurate results.
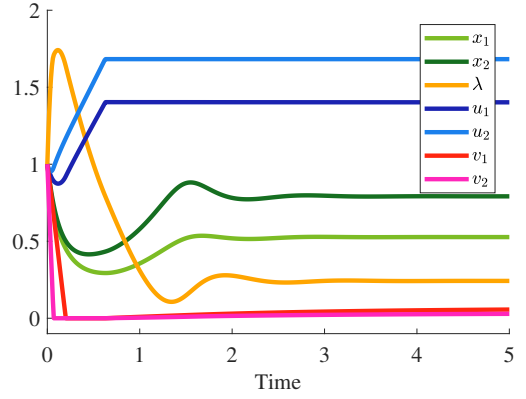

Fig. 3: Trajectory of the $\mathcal{RO}$ dynamics for the robust nonlinear optimization problem (45) in Example B.

kinematic points, to monitor multiple targets (initially assumed static). This is a dynamic generalization of a classical facility location problem [44] and here we provide a solution to the robust formulation of this problem using $\mathcal{RO}$ dynamics. The optimization problem is described by an undirected graph with N nodes and a set of edges $\mathbb{E}$. Among these nodes, the first N_1 nodes represent the fixed positions of the targets (also referred to as anchors), while the remaining $N_2 = N - N_1$ nodes represent the mobile agents. Each node $x_i \in \mathbb{R}^2$ denotes the location of an anchor or an agent. The anchors $x_1, \cdots, x_{N_1}$ have fixed positions and the agents $x_{N_1+1}, \cdots, x_N$ are mobile and can adjust their positions. The problem is to find the locations of the sensor nodes $x_{N_1+1}, \cdots, x_N$ to minimize the following cost

$$g(x_1, .., x_{N_1}) = \min_{x_{N_1+1}, \cdots, x_N} \sum_{(i,j) \in \mathbb{E}} f_{ij}(x_i, x_j),$$

which is the sum of some measure of "length" for each link. This problem and its generalizations have many applications [33], and can be solved efficiently and in a distributed fashion in continuous time [38]. Leaving the sensor nodes (agents) mobile and capable of computation in distributed mode, we obtain a distributed dynamical version *real-time* of the optimal placement. The sensor nodes, through local interactions, cooperate to find and move toward their globally optimal locations autonomously. The problem can also be under a set of constraints for the position of agents $x_{N_1+1}, \cdots, x_{N_1}$ to be in a specified convex set. Specifically, we consider the following robust location and placement problem

$$\min_{x_{N_1+1}, \cdots, x_N} \sum_{(i,j) \in \mathbb{E}} \frac{1}{2} w_{ij} \parallel x_i - x_j \parallel_2^2$$
$$\text{s.t. } \max_{u_i \in \mathcal{U}_i} (a_i + P_i u_i)^\top x_i \leq b_i, \ i = N_1 + 1, \cdots, N,$$

where the uncertainty u_i lies in $\mathcal{U}_i = \parallel u_i \parallel_2^2 \leq \rho_i^2$ and $x_{N_1} + 1, \cdots, x_N$ are moving agents. The $\mathcal{RO}$ dynamics for

this problem can be derived as follows for $i = N_1 + 1, \cdots, N$

$$
\begin{cases}
\dot{x}_i = \sum_j w_{ij}(x_i - x_j) - \hat{\lambda}_i\,(a_i + P_i u_i) \\[2mm]
\dot{\hat{\lambda}}_i = [(a_i + u_i P_i)^\top x_i - P_i u_i^\top c_i - b_i - v_i\,\mathcal{U}_i]_{\hat{\lambda}_i}^{\varepsilon+} \\[2mm]
\dot{u}_i = P_i^\top (x_i - c_i) - 2 v_i u_i \\[2mm]
\dot{v}_i = [\hat{\lambda}_i\,\mathcal{U}_i]
\end{cases}
.
$$

Note that this problem is naturally distributed and our dynamics reflects the distributed structure. We consider the setup shown in Figure 4 with five anchors and four agents. The figure also shows the (fixed) interconnection graph among all the elements of the problem. In this academic example, we consider the presence of a linear constraint defining the half-space where the agents can be. The constraint is simply

$$
\mathbf{1}' x_i \leq 2.5, \quad i = N_1 + 1, \cdots, N \ ,
$$

for all the agents. In addition, we would like the agents to find robust locations based on the uncertainty on the nominal slope 45° of the nominal constraint. Namely

$$
\mathbf{1}' x_i + u_i P' x_i \leq 2.5, \quad i = N_1 + 1, \cdots, N \ ,
$$

where $P' = \begin{bmatrix} 1 & -1 \end{bmatrix}$ and $\|u_i\|_2^2 \leq \rho^2$. For example, when $\rho = 1$, the constraint can be any line passing through $x' = (1.25, 1.25)$, including the horizontal and vertical ones. Figure 4 also shows the nominal linear constraint. The location of the shown agents is the optimal robust one w.r.t. the constraint being perturbed by the uncertainty u_i with $\|u_i\|_2^2 \leq 0.1$.

First, agents start at their initial locations at the origin (white circle), and their paths converge to the optimal robust location for $\rho^2 = 0.1$. Such locations are indicated by full-colored circles. We see that agents 1 and 4 (blue and green) are on the boundary of the robust feasible set identifiable with the larger sector in the figure. The uncertain constraint is not active for the other agents. However, their optimal location is indirectly influenced by the robust constraint active in agents 1 and 4 as all agents and anchors are interconnected.

In the next phase of the simulation, we rotate the location of the anchors clockwise at constant speed. Although agents only react to their local neighbors (agents/anchors), the interconnected dynamical system shows the ability to globally track the coordinated motion of the anchors within the robust feasible set. It is interesting to notice that the boundary of the feasible set is not defined a priori or hard-coded in the simulation, but it emerges from the interactions built in the dynamical system. Finally, after some time, the uncertainty changes and increases with size $\rho^2 = 1$ at time $t = 300$. This implies that no location should be feasible above the horizontal line passing through $(1.25, 1.25)$ and on the right of the vertical line passing through the same point. The location of the agents when the uncertainty changes is indicated by black empty circles. We note that, when the uncertainty changes, the nodes 1, 2, and 3 are located outside the new robust feasible set and, with a transient correction, move inside the robust SW quadrant for the rest of the time.

This example showcases the ability of the dynamical system to track changes online and adapt to uncertainty changes over
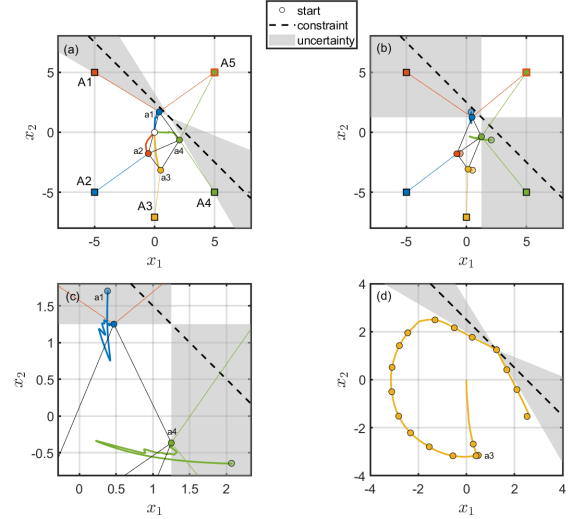

Fig. 4: Locations of agents and anchors, interactions between agents and anchors, and the nominal linear constraint in Example C.

time. We have obtained the behavior shown by appropriate scaling of the various differential equations involved. Note that the convergence is not affected by positive scaling of the differential equations. A relatively larger scaling on the dual variables in charge of the constraints makes the system response more reactive toward feasibility and more sensitive toward uncertainty changes. We leave to future research the systematic design of the optimization system for the desired real-time behavior.

VIII. Conclusions

We proposed a continuous-time dynamical system, $\mathcal{RO}$ dynamics, to solve robust optimization problems. This approach can solve convex-concave $\mathcal{RO}$ problems in a decentralized manner without requiring specific model knowledge. This approach offers an advantage over existing methods, which are based on robust counterpart and random sampling particularly for problems with large scales, complex constraint and uncertainty sets, real-time adaptation requirements or unknown dynamics. The implementation capability of $\mathcal{RO}$ dynamics, elaborated in this paper, opens new avenues for robustifying large-scale optimization problems in Machine Learning and Deep Learning against uncertainties, perturbations, and adversarial attacks, while further research is needed to analyze the computational cost and rigorously characterize the convergence rate of $\mathcal{RO}$ dynamics.

IX. Appendix

A. Lemma 1 [Saddle property]

Essentially, we want to show that

$$
\mathcal{L}(x^\star, \lambda, u, v^\star) \leq \mu \leq \mathcal{L}(x, \lambda^\star, u^\star, v) \ .
$$

Under Assumption 2, an optimal solution $x^\star$ exists and based on Assumption 1, this optimal solution is unique as $f_0(x, u_0)$

is strictly convex in x for any $u_0 \in \mathcal{U}_0$. Consider any optimal solution $(x^\star, \lambda^\star, u^\star, v^\star)$ for $\mathcal{RO}$ problem (17).
For each of the lower level optimizations, let

$$\eta_i = \max_{u_i \in \mathcal{U}_i} f_i(x^\star, u_i) = f_i(x^\star, u_i^\star) = \tilde{\mathcal{L}}_i(x^\star, u_i^\star, v_i^\star), \; i_{[N]} \; .$$

From the corresponding saddle property, it follows that for all u_i, $i_{[N]}$,

$$f_i(x^\star, u_i) - (v_i^\star)^\top h_i(u_i) \le \eta_i \; . \tag{47}$$

From the upper level saddle property, for all $\lambda_i \ge 0$, $i_{[N]}^+$,

$$\eta_0 + \sum_i (c_i + \lambda_i)\eta_i \le f_0(x^\star, u_0^\star) + \sum_{i=1}^N (c_i + \lambda_i^\star)\eta_i = \mu \; .$$

Substituting the lower bound on η_i, (47), in the left hand side, it follows that for all $u_i, i_{[N]}$ and $\lambda_i \ge 0, i_{[N]}^+$,

$$\begin{aligned}
&\mathcal{L}(x^\star, \lambda, u, v^\star) \\
=& f_0(x^\star, u_0) - (v_0^\star)^\top h_0(u_0) \\
&+ \sum_{i=1}^N (c_i + \lambda_i)(f_i(x^\star, u_i) - (v_i^\star)^\top h_i(u_i)) \\
\le& \mu \; ,
\end{aligned}$$

where we have used $c_i + \lambda_i \ge 0$. We next use lower saddle property in the other direction, namely, for all $v_i \ge 0$,

$$\begin{aligned}
\eta_i = f_i(x^\star, u_i^\star) &= f_i(x^\star, u_i^\star) - (v_i^\star)^\top h_i(u_i^\star) \\
&\le f_i(x^\star, u_i^\star) - (v_i)^\top h_i(u_i^\star) \; ,
\end{aligned}$$

which implies that for all $v_i \ge 0$, $-v_i^\top h(u^\star) \ge 0$, $i_{[N]}$. Using this property and the fact that $(c_i + \lambda_i^\star) \ge 0$, it follows that for all $v_i \ge 0$, $i_{[N]}^+$,

$$-(v_0)^\top h_0(u_0^\star) - \sum_{i=1}^N (c_i + \lambda_i^\star)(v_i)^\top h_i(u_i^\star) \ge 0 \; . \tag{48}$$

The upper level saddle property implies

$$\mu \le f_0(x, u_0^\star) + \sum_{i=1}^N (c_i + \lambda_i^\star) f_i(x, u_i^\star)$$

for all x. Combining this with (48), we obtain

$$\begin{aligned}
\mu &\le f_0(x, u_0^\star) - (v_0)^\top h_0(u_0^\star) \\
&+ \sum_{i=1}^N (c_i + \lambda_i^\star)(f_i(x, u_i^\star) - (v)^\top h_i(u_i^\star)) \\
&= \mathcal{L}(x, \lambda^\star, u^\star, v)
\end{aligned}$$

for all x and $v_i \ge 0, i_{[N]}$. $\qquad\square$

B. $\mathcal{RO}$ dynamics solutions properties

$\mathcal{RO}$ dynamics (23) can be viewed as switched dynamical system with a discontinuous right-hand side. The conditions guaranteeing the existence and uniqueness of the solution and continuity w.r.t. initial conditions, for a general discontinuous dynamical system are provided in [45, Theorem 2.5]. In this section, we show that our $\mathcal{RO}$ dynamics (23) satisfies the refined conditions presented in [41].

To prove the existence and uniqueness of solutions for (23), and also the continuity of solutions w.r.t. the initial conditions, there are two main steps. The first step is showing that $\mathcal{RO}$ dynamics is a particular case of projected dynamical systems. The second step requires $\mathcal{RO}$ dynamics (23) satisfying the monotonicity property, which is our main result.

Definition 2. *(Projection operator) If $\mathcal{K}$ is a closed convex set, for any point $\bar{y} \in \mathbb{R}^q$, the point projection of $\bar{y}$ on the set $\mathcal{K}$ can be written as*

$$proj_{\mathcal{K}}(\bar{y}) = argmin_{y \in \mathcal{K}} \parallel y - \bar{y} \parallel \; .$$

For $\bar{y} \in \mathbb{R}^n$ and $y \in \mathcal{K}$, vector projection of $\bar{y}$ at y w.r.t. $\mathcal{K}$ is

$$\Pi_{\mathcal{K}}(y, \bar{y}) = lim_{\delta \to 0^+} \frac{proj_{\mathcal{K}}(y + \delta\bar{y}) - y}{\delta} \; . \tag{49}$$

Note that the map $proj_{\mathcal{K}}$ is Lipschitz on $\mathbb{R}^q$ with constant $L = 1$ [46, Proposition 2.4.1].

Definition 3. *[Projected dynamical system [47]] Considering a differential equation $\dot{y} = F(y)$ with $F : \mathbb{R}^q \to \mathbb{R}^q$, the associated projected dynamical system is defined as*

$$\dot{y} = \Pi_{\mathcal{K}}(y, F(y)), \; y(0) \in \mathcal{K} \; . \tag{50}$$

Lemma 5. *($\mathcal{RO}$ dynamics as a projected dynamics) $\mathcal{RO}$ dynamics (23) can be written as a projected dynamical system according to Definition 3.*

Proof. The proof of Lemma 5 follows along the lines of the construction outlined in [41]. Details omitted. $\qquad\square$
$\qquad\square$

Proposition 6. *If F in the projected dynamical system (50) is Lipschitz on $\mathcal{K}$, we have the following existence, uniqueness, and continuity w.r.t. the initial condition results for the solutions of the projected dynamics (50):*

1) *For any $y_0 \in \mathcal{K}$, there exists a unique solution $t \to y(t)$ of the projected system (50) with $y(0) = y_0$ in $[0, \infty)$.*
2) *Consider a sequence of points $\{y_k\}_{k=1}^\infty \subset \mathcal{K}$ with $lim_{k \to \infty} = y$. Then, the sequence of solutions $\{t \to \gamma_k(t)\}_{k=1}^\infty$ of the projected dynamics (50) with $\gamma_k(0) = y_k$ for all k, converges to the solution $t \to \gamma(t)$ of (50) with $\gamma(0) = y$ uniformly on every compact set of $[0, \infty)$.*

The ability to write $\mathcal{RO}$ dynamics (23) as a projected dynamical system along with the monotonicity property is used in the proof of the existence, uniqueness and continuity of the solutions of the set $\mathbb{S}$.

Lemma 6. *(Existence, uniqueness and continuity of solutions) $\gamma : [0, T) \to \mathbb{S}$ is defined as a Caratheodory solution of $\mathcal{Z}^{\mathcal{RO}}$ in the interval $[0, T)$ if γ is absolutely continuous on $[0, T)$ and satisfies $\dot{\gamma}(t) = \mathcal{Z}^{\mathcal{RO}}(\gamma(t))$ almost everywhere in $[0, T)$. Under Assumptions 1 and 2, and starting from any point $z \in \mathbb{S}$, a unique solution to $\mathcal{RO}$ dynamics (23) exists and remains in $\mathbb{S} \cap V^{-1}(\le V(z))$. Also, if a sequence of points $\{z_k\}_{k=1}^\infty \subset \mathbb{S}$ converges to z as $k \to \infty$, the sequence of solutions $\{t \to \gamma_k(t)\}_{k=1}^\infty$ of $\mathcal{Z}^{\mathcal{RO}}$ starting at these points (that is, $\gamma_k(0) = z_k$ for all k) converge uniformly to the solution $t \to \gamma(t)$ on every compact set of $[0, \infty)$.*

The proof of this lemma follows closely along the lines of proof for the existence and uniqueness of solution for the primal-dual dynamical system from [41, Lemma 4.3].

C. Theorem 5

Based on the optimal solution $x^\star$ for $\mathcal{RO}$ problem,

$$\mu = \min_{\mathcal{F}_i(x) \leq 0} \mathcal{F}_0(x) \, , \, \mu = \mathcal{F}_0(x^\star) \, .$$

As the cost function of μ_ε is smaller than or equal to that of $\mathcal{RO}$ and the feasible sets of the two problems are equal,

$$\mu_\varepsilon - \mu \leq 0 \, . \tag{51}$$

Since $x^\star$ minimizes $\mathcal{F}_0(x)$ over the constraint set, $\mu = \mathcal{F}_0(x^\star) \leq \mathcal{F}_0(x_\varepsilon^\star)$. Adding and subtracting $\varepsilon \sum_{i=1}^{n} \mathcal{F}_i(x_\varepsilon^\star)$ in the RHS and using (51) yields $\varepsilon \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star) \leq \mu_\varepsilon - \mu \leq 0$.

Following a similar argument as before by comparing $\mu(\varepsilon_0)$ and μ_ε for $\varepsilon_0 \geq \varepsilon$, we now let

$$\mu_\varepsilon = \tilde{\mathcal{F}}_0(x_\varepsilon^\star) = \mathcal{F}_0(x_\varepsilon^\star) + \varepsilon \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star) \, ,$$

$$\mu(\varepsilon_0) = \tilde{\mathcal{F}}_0(x_{\varepsilon_0}^\star) + \delta \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) \, ,$$

where $\delta = \varepsilon_0 - \varepsilon$. Then, $\mu_\varepsilon \geq \mu(\varepsilon_0)$, but because $x_\varepsilon^\star$ is optimal for μ_ε, we have $\tilde{\mathcal{F}}_0(x_\varepsilon^\star) \leq \tilde{\mathcal{F}}_0(x_{\varepsilon_0}^\star)$. As $x_\varepsilon^\star$ is feasible for $\mu(\varepsilon_0)$,

$$\tilde{\mathcal{F}}_0(x_{\varepsilon_0}^\star) + \delta \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) = \mu(\varepsilon_0) \leq \tilde{\mathcal{F}}_0(x_\varepsilon^\star) + \delta \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star) \, .$$

Combining the two inequalities,

$$\delta \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) - \delta \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star) \leq \tilde{\mathcal{F}}_0(x_\varepsilon^\star) - \tilde{\mathcal{F}}_0(x_{\varepsilon_0}^\star) \leq 0 \, ,$$

which implies that $\sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) \leq \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star)$. Thus,

$$\varepsilon \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) \leq \varepsilon \sum_{i=1}^{N} \mathcal{F}_i(x_\varepsilon^\star) \leq \mu_\varepsilon - \mu \leq 0 \, . \tag{52}$$

Since $\sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) = \frac{\mu(\varepsilon_0) - \mathcal{F}_0(x_{\varepsilon_0}^\star)}{\varepsilon_0}$ is bounded, we have

$$\varepsilon \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_0}^\star) \leq \mu_\varepsilon - \mu \leq 0 \, , \tag{53}$$

which means that μ_ε converges to μ as $\varepsilon \to 0$.

To prove the second part of the theorem, that is, $x_\varepsilon^\star \to x^\star$ as $\varepsilon \to 0$, consider any sequence $\{\varepsilon_n\}$ converging to 0. Let $\{x_n^\star\}$ be the corresponding sequence of optimal solutions for $\mu(\varepsilon_n)$.

As $\mathcal{C}$ is compact and the same for both perturbed and original problem, $x_n^\star \in \mathcal{C}$ is bounded. Therefore, there exists a convergent sub-sequence $x_{n_t}^\star$ that converges to, say, $\hat{x} \in \mathcal{C}$, as $\varepsilon_{n_t} \to 0$, since $\mathcal{C}$ is closed by assumption.

This implies that $\mathcal{F}_0(x_{n_t}^\star) \to \mathcal{F}_0(\hat{x})$, since $\mathcal{F}_0(x)$ is continuous. Because $\hat{x}$ is feasible, $\mathcal{F}_0(\hat{x}) \geq \mu$. However, $\mathcal{F}_0(\hat{x}) > \mu$ is impossible since $\mu(\varepsilon_{n_t}) \to \mu$, from the first part of the proof, and $\mu(\varepsilon_{n_t}) = \mathcal{F}_0(x_{n_t}^\star) + \varepsilon_{n_t} \sum_{i=1}^{N} \mathcal{F}_i(x_{n_t}^\star) \to \mathcal{F}_0(\hat{x})$, since from (52) and (53), $\lim_{\varepsilon_{n_t} \to 0} \varepsilon_{n_t} \sum_{i=1}^{N} \mathcal{F}_i(x_{\varepsilon_{n_t}}) = 0$.

Therefore, $\mathcal{F}_0(\hat{x}) = \mu = \mathcal{F}_0(x^\star)$. Since $f(x)$ is strictly convex, $\hat{x} = x^\star$. Since every convergent sub-sequence converges to $x^\star$, the whole sequence converges to it. Since the sequence was arbitrary, we have that $x_\varepsilon^\star \to x^\star$ as $\varepsilon \to 0$.

REFERENCES

[1] Z. He, S. Bolognani, J. He, F. Dörfler, and X. Guan, "Model-free nonlinear feedback optimization," *arXiv preprint arXiv:2201.02395*, 2022.

[2] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*. Princeton University Press, 2009.

[3] D. Bertsimas, D. B. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Review*, vol. 53, no. 3, pp. 464–501, 2011.

[4] A. L. Soyster, "Convex programming with set-inclusive constraints and applications to inexact linear programming," *Operations Research*, vol. 21, pp. 1154–1157, 1973.

[5] J. E. Falk, "Exact solutions of inexact linear programs," *Operations Research*, vol. 24, pp. 783–787, 1976.

[6] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operational Research Letters*, vol. 25, no. 1, pp. 1–13, 1999.

[7] A. Ben-Tal, A. Nemirovski, and C. Roos, "Robust solutions of uncertain quadratic and conic-quadratic problems," *Mathematical Operations Research*, vol. 13, no. 2, pp. 535–560, 2002.

[8] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operational Research*, vol. 23, no. 4, pp. 769–805, 1998.

[9] H. Beyer and B. Sendhoff, "Robust optimization: A comprehensive survey," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 33–34, pp. 3190–3218, 2007.

[10] H. Xu, C. Caramanis, and S. Mannor, "Robustness and regularization of support vector machines," *Journal of Machine Learning Research*, vol. 10, pp. 1485–1510, 2009.

[11] H. Xu, C. Caramanis, and S. Mannor, "Robust regression and lasso," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3561–3574, 2010.

[12] G. Lanckriet, C. Bhattacharyya, and M. Jordan, "A robust minimax approach to classification," *Journal of Machine Learning Research*, vol. 3, pp. 555–582, 2003.

[13] C. Bhattacharyya, L. Grate, M. Jordan, L. El Ghaoui, and I. Mian, "Robust sparse hyperplane classifiers: Application to uncertain molecular profiling data," *Journal of Computational Biology*, vol. 11, no. 6, pp. 1073–1089, 2004.

[14] C. Bhattacharyya, P. Shivaswamy, and A. Smola, "A second order cone programming formulation for classifying missing data," *Advances in Neural Information Processing Systems*, pp. 153–160, 2004.

[15] T. Trafalis and R. Gilbert, "Robust support vector machines for classification and computational issues," *Optimization Methods and Software*, vol. 22, no. 1, pp. 187–198, 2007.

[16] H. Namkoong and J. C. Duchi, "Stochastic gradient methods for distributionally robust optimization with f-divergences," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[17] A. Sinha, H. Namkoong, and J. Duchi, "Certifiable distributional robustness with principled adversarial training," in *International Conference on Learning Representations*, 2018.

[18] Y. Esfandiari, K. Ebrahimi, A. Balu, N. Elia, U. Vaidya, and S. Sarkar, "A saddle-point dynamical system approach for robust deep learning," *arXiv preprint arXiv:1910.08623*, 2019.

[19] B. L. Gorissen and D. Hertog, "Robust nonlinear optimization via the dual," *Optimization Online*, 2015.

[20] D. Bertsimas, I. Dunning, and M. Lubin, "Reformulation versus cutting-planes for robust optimization," *Computational Management Science*, vol. 13, no. 2, pp. 195–217, 2016.

[21] A. Ben-Tal, D. Hertog, and J. P. Vial, "Deriving robust counterparts of nonlinear uncertain inequalities," *Mathematical Programming*, vol. 149, no. 1, pp. 265–299, 2015.

[22] G. Calafiore and M. C. Campi, "Uncertain convex programs: Randomized solutions and confidence levels," *Mathematical Programming*, vol. 102, no. 1, pp. 25–46, 2004.

[23] G. C. Calafiore, "Random convex programs," *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3427–3464, 2010.

[24] V. Rostampour and T. Keviczky, "Distributed computational framework for large-scale stochastic convex optimization," *Energies*, vol. 14, no. 1, 2021.

[25] A. Ben-Tal, E. Hazan, T. Koren, and S. Mannor, "Oracle-based robust optimization via online learning," *Operations Research*, vol. 63, no. 3, pp. 628–638, 2015.

[26] A. Mutapcic and S. Boyd, "Cutting-set methods for robust convex optimization with pessimizing oracles," *Optimization Methods and Software*, vol. 24, no. 3, pp. 381–406, 2009.

[27] N. Ho-Nguyen and F. K. Karzan, "Online first-order framework for robust convex optimization," *Operations Research*, vol. 66, no. 6, 2018.

[28] D. Yazdani, M. Omidvar, D. Yazdani, J. Branke, T. Nguyen, A. Gandomi, Y. Jin, and X. Yao, "Robust optimization over time: A critical review," *IEEE Transactions on Evolutionary Computation*, 2023.

[29] K. Arrow, L. Hurwitz, and H. Uzawa, *Studies in Linear and Non-linear Programming*. Stanford University Press, 1958.

[30] D. Feijer and F. Paganini, "Stability of primal-dual gradient dynamics and applications to network optimization," *Automatica*, vol. 46, no. 12, pp. 1974–1981, 2010.

[31] K. Ebrahimi, N. Elia, and U. Vaidya, "A continuous time dynamical system approach for solving robust optimization," in *2019 18th European Control Conference (ECC)*, pp. 1479–1485, IEEE, 2019.

[32] A. Ben-Tal and A. Beck, "Duality in robust optimization: Primal worst equals dual best," *Operations Research Letters*, 2009.

[33] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[34] V. Jeyakumar and G. Y. Li, "Strong duality in robust convex programming: Complete characterizations," *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3384–3407, 2010.

[35] H. Rafique, M. Liu, Q. Lin, and T. Yang, "Weakly-convex concave min-max optimization: Provable algorithms and applications in machine learning," *Optimization Methods and Software*, vol. 37, 2022.

[36] G. Zhang, "Understanding minimax optimization in modern machine learning," Master's thesis, University of Waterloo, 2021.

[37] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*. Prentice Hall, New Jersey, 1995.

[38] J. Wang and N. Elia, "A control perspective for centralized and distributed convex optimization," *50th IEEE Conference on Decision and Control and European Control Conference, Orlando, FL*, pp. 3800–3805, 2011.

[39] J. W. Simpson-Porco, "Input/output analysis of primal-dual gradient algorithms," *54th Annual Allerton Conference, Allerton House, UIUC, Illinois, USA*, 2016.

[40] K. C. Kosaraju, V. Chinde, R. Pasumarthy, A. Kelkar, and N. M. Singh, "Stability analysis of constrained optimization dynamics via passivity techniques," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 91–96, 2018.

[41] A. Cherukuri, E. Mallada, and J. Cortes, "Asymptotic convergence of constrained primal-dual dynamics," *Systems and Control Letters*, vol. 87, pp. 10–15, 2016.

[42] K. Ebrahimi, N. Elia, and U. Vaidya, "Robust optimization via discrete-time saddle point algorithm," in *58th Conference on Decision and Control (CDC), Nice, France*, 2019.

[43] A. Bacciotti and F. Ceragioli, "Nonpathological lyapunov functions and discontinuous carathéodory systems," *Automatica*, vol. 42, no. 3, pp. 453–458, 2006.

[44] R. Farahani, M. Abedian, and S. Sharahi, *Dynamic Facility Location Problem*. Contributions to Management Science. Physica, 2009.

[45] A. Nagurney and D. Zhang, *Projected Dynamical Systems and Variational Inequalities with Applications*, vol. 2. Springer Science & Business Media, 2012.

[46] F. H. Clarke, *Optimization and Nonsmooth Analysis*. Wiley, 1983.

[47] A. Nagurney and D. Zhang, "Projected dynamical systems and variational inequalities with applications," *International Series in Operations Research and Management Science*, vol. 2, 1996.

Keivan Ebrahimi received bachelor's and master's degree in electrical engineering from Sharif University of Technology, Tehran, Iran in 2012 and 2014. He is a Ph.D. candidate in electrical and computer engineering at Iowa State University (ISU), Ames, IA, USA. He is currently a principal data scientist at Tarana Wireless in Milpitas, CA, USA. He received the College of Engineering fellowship at Iowa State University in 2015 and has been selected for the Rock Star Spot Award for stellar contribution and performance in View Inc. COVID-SENSE product, May 2020. He filed three patents to the United States Patents and Trademarks Office for "Environmental Adjustment using Artificial Intelligence", "Identifying and Reducing Health Risks and Tracking Occupancy in a Facility", and "Immersive Collaboration of Remote Participants via Media Displays". His research interests include adversarial machine learning and deep learning, computer vision, robust optimization, control theory and dynamical systems.

Prof. Nicola Elia received the Laurea degree in Electrical Engineering from the Politecnico di Torino, Turin, Italy in 1987 and the Ph.D. degree in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA in 1996. He is currently the Vincentine-Hermes-Luh Chair Professor of Electrical and Computer Engineering at the University of Minnesota (UMN), Twin Cities, MN, USA. Before joining UMN in 2018, and since 1999, he was a faculty with the department of Electrical and Computer Engineering at Iowa State University, Ames, IA, USA. He was a Postdoctoral Associate at the Laboratory for Information and Decision Systems at MIT from 1996 to 1999. He was a Control Engineer with the Fiat research Center, Turin, Italy, from 1987 to 1990. Dr. Elia received the NSF Career Award and he is a Fellow of the IEEE. His research interests include computational methods for controller design, communication systems with access to feedback, control with communication constraints, and network distributed systems.

Prof. Umesh Vaidya received the Ph.D. degree in mechanical engineering from the University of California at Santa Barbara, Santa Barbara, CA, in 2004. He was a Research Engineer at the United Technologies Research Center (UTRC), East Hartford, CT, USA. He is currently a professor in the Department of Mechanical Engineering, Clemson University, S.C., USA. Before joining Clemson University in 2019, and since 2006, he was a faculty with the department of Electrical and Computer Engineering at Iowa State University. He is the recipient of 2012 National Science Foundation CAREER award. His current research interests include dynamical systems and control theory.