

# Online Learning of Feasible Strategies in Unknown Environments

Santiago Paternain and Alejandro Ribeiro

**Abstract**—Define an environment as a set of convex constraint functions that vary arbitrarily over time and consider a cost function that is also convex and arbitrarily varying. Agents that operate in this environment intend to select actions that are feasible for all times while minimizing the cost's time average. Such action is said optimal and can be computed *offline* if the cost and the environment are known a priori. An *online* policy is one that depends causally on the cost and the environment. To compare online policies to the optimal offline action define the fit of a trajectory as a vector that integrates the constraint violations over time and its regret as the cost difference with the optimal action accumulated over time. Fit measures the extent to which an online policy succeeds in learning feasible actions while regret measures its success in learning optimal actions. This paper proposes the use of online policies computed from a saddle point controller which are shown to have fit and regret that are either bounded or grow at a sublinear rate. These properties provide an indication that the controller finds trajectories that are feasible and optimal in a relaxed sense. Concepts are illustrated throughout with the problem of a shepherd that wants to stay close to all sheep in a herd. Numerical experiments show that the saddle point controller allows the shepherd to do so.

## I. INTRODUCTION

The motivation for this paper is the navigation of a time varying convex environment defined as a set of convex constraints that an agent must satisfy at all times. The constraints are unknown a priori, vary arbitrarily in time in a possibly discontinuous manner, and are observed locally in space and causally in time. The goal of the agent is to find a feasible strategy that satisfies all of these constraints. This paper shows that an online version of the saddle point algorithm of Arrow and Hurwicz [1] executed by the agent succeeds in finding such strategy. If the agent wants to further minimize a convex cost, we show that the same algorithm succeeds in finding an strategy that is feasible at all times and optimal on average.

To understand the contribution of this paper it is important to observe that the navigation problem outlined above can be mathematically formulated as the solution of a convex program [2]–[6] whose solution is progressively more challenging when we progress from deterministic settings to stochastic and online settings. Indeed, in a determinist setting the cost and constraints are fixed. This yields a canonical convex optimization problem that can be solved with extremum seeking controllers based on gradient descent [7]–[10], primal-dual methods [1], [11]–[14], or interior point methods [15, Chapter 11]. In a stochastic setting cost and constraints are not constant

but vary randomly according to a stationary distribution. The agent's goal is then expressed as the selection of an action that minimizes the expected value of the objective function while satisfying constraints in an average sense [16]–[18]. This problem is more complicated than its deterministic counterpart but it can be solved using, e.g., stochastic gradient descent [19]–[21] or stochastic quasi-Newton's methods [22].

In this paper we consider online formulations in which cost and constraints can vary arbitrarily, perhaps strategically, and where the goal is to find an action that is good on average and that satisfies the constraints at all times – assuming such an action exists, which, when functions change strategically, restricts adversarial actions. In this case, *unconstrained* cost minimization can be formulated in the language of regret [23]–[25] whereby agents operate online by selecting plays that incur a cost selected by nature. The cost functions are revealed to the agent ex post and used to adapt subsequent plays. The goodness of these *online* policies are determined by comparing to the optimal action chosen *offline* by a clairvoyant agent that has prescient access to the cost. Regret is defined as the difference of the accumulated cost attained online and the optimal offline cost. It is a remarkable fact that an online version of gradient descent is able to find plays whose regret grows at a sublinear rate when the cost is a convex function [26], [27] – therefore suggesting vanishing per-play penalties of online plays with respect to the clairvoyant play.

The constrained optimization equivalent of gradient descent is the saddle point method applied to the determination of a saddle point of the Lagrangian function [1]. This method interprets each constraint as a separate potential and descends on a linear combination of their gradients. The coefficients of this linear combination are multipliers that adapt dynamically so as to push the agent to the optimal solution in the feasible region. Saddle point algorithms and variations have been widely studied [11]–[14] and used in various domains such as decentralized control [28], [29] and image processing, see e.g. [30]. Our observation is that since an online version of gradient descent succeeds in achieving small regret, it is not unreasonable to expect an online saddle point method to succeed in finding feasible actions with small regret.

The main contribution of this paper is to prove that this expectation turns out to be true. We show that an online saddle point algorithm that observes costs and constraints ex post succeeds in finding policies that are feasible and have small regret. Central to this development is the definition of a viable environment as one in which there exist an action that satisfies the time varying constraints at all times and the introduction of the notion of fit (Section II). The latter

Work in this paper is supported by NSF CNS-1302222 and ONR N00014-12-1-0997. The authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania, 200 South 33rd Street, Philadelphia, PA 19104. Email: {spater, aribeiro}@seas.upenn.edu.

is defined as a vector that contains the time integrals of the constraints evaluated across the trajectory and is the analogous of regret for the satisfaction of constraints. In the same way in which the accumulated payoff of the online trajectory is compared with the payoff of the offline trajectory, fit compares the accumulation of the constraints along the trajectory with the feasibility of an offline viable strategy. As such, a trajectory can achieve small fit by becoming feasible at all times or by alternating periods in which the constraints are violated with periods in which the constraints are satisfied with slack. This notion of fit is appropriate for constraints that have a cumulative nature. For cases where this is not appropriate we introduce the notion of saturated fit in which only violations of the constraint are accumulated. A trajectory with small saturated fit is one in which the constraints are violated by a significant amount only for a short period of time.

Technical developments begin with the derivation of a projected gradient controller to limit the growth of regret in an environment without constraints (Section III). The purpose of this section is to introduce tools and to clarify connections with existing literature in discrete time [26], [27] and continuous time regret [31]–[33]. An important conclusion here is that regret in continuous time can be bounded by a constant that is independent of the time horizon, as opposed to the sublinear growth that is observed in discrete time.

We then move onto the main part of the paper in which we propose to control fit and regret growth with the use of an online saddle point controller that moves along a linear combination of the negative gradients of the instantaneous constraints and the objective function. The coefficients of this linear combination are adapted dynamically as per the instantaneous constraint functions (Section IV). This online saddle point controller is a generalization of (offline) saddle point in the same sense that an online gradient controller generalizes (offline) gradient descent. We show that if there exists an action that satisfies the environmental constraints at all times, the online saddle point controller achieves bounded fit if optimality is not of interest (Theorem 2). When optimality is considered, the controller achieves bounded regret and a fit that grows sublinearly with the time horizon (Theorem 3). Analogous results are derived for saturated fit. I.e., it is bounded by a constant when optimality is not of interest and grows sublinearly otherwise (corollaries 2 and 3). Throughout the paper we illustrate concepts with the problem of a shepherd that has to stay close to his herd (Section II-B). A numerical analysis of this problem closes the paper (Section V) except for concluding remarks (Section VI).

**Notation.** A multivalued function  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  is defined by stacking component functions, i.e.,  $f := [f_1, \dots, f_m]^T$ . The notation  $\int f(x)dx := [\int f_1(x)dx, \dots, \int f_m(x)dx]^T$  represents a vector stacking individual integrals. An inequality  $x \leq y$  between vectors  $x, y \in \mathbb{R}^n$  is interpreted componentwise. An inequality  $x \leq c$  between a vector  $x = [x_1, \dots, x_n]^T \in \mathbb{R}^n$  and a scalar  $c \in \mathbb{R}$  means that  $x_i \leq c$  for all  $i$ .

## II. VIABILITY, FEASIBILITY AND OPTIMALITY

We consider a continuous time environment in which an agent selects actions that result in a time varying set of

penalties. Use  $t$  to denote time and let  $X \subseteq \mathbb{R}^n$  be a closed convex set from which the agent selects action  $x \in X$ . The penalties incurred at time  $t$  for selected action  $x$  are given by the value  $f(t, x)$  of the vector function  $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ . We interpret the vector penalty function  $f$  as a definition of the environment. Our interest is in situations where the agent is faced with an environment  $f$  and must choose an action  $x \in X$  – or perhaps a trajectory  $x(t)$  – that guarantees nonpositive penalties  $f(t, x(t)) \leq 0$  for all times  $t$  not exceeding a time horizon  $T$ . Since the existence of this trajectory depends on the specific environment we define a viable environment as one in which it is possible to select an action with nonpositive penalty for times  $0 \leq t \leq T$  as we formally specify next.

**Definition 1 (Viable environment).** We say that an environment  $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^m$  is viable over the time horizon  $T$  for an agent that selects actions  $x \in X$  if there exists a feasible action  $x^\dagger \in X$  such that

$$f(t, x^\dagger) \leq 0, \quad \text{for all } t \in [0, T]. \quad (1)$$

The set  $X^\dagger := \{x^\dagger \in X : f(t, x^\dagger) \leq 0, \text{ for all } t \in [0, T]\}$  is termed the feasible set of actions.

Since for a viable environment it is possible to have multiple feasible actions it is desirable to select one that is optimal with respect to some criterion of interest. Introduce then the objective function  $f_0 : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$ , where for a given time  $t \in [0, T]$  and action  $x \in X$  the agent suffers a loss  $f_0(t, x)$ . The optimal action is defined as the one that minimizes the accumulated loss  $\int_0^T f_0(t, x) dt$  among all viable actions, i.e.,

$$x^* := \operatorname{argmin}_{x \in X} \int_0^T f_0(t, x) dt \quad (2)$$

s.t.  $f(t, x) \leq 0$ , for all  $t \in [0, T]$ .

For the definition in (2) to be valid the function  $f_0(t, x)$  has to be integrable with respect to  $t$ . In subsequent definitions and analyses we also require integrability of the environment  $f$  as well as convexity with respect to  $x$  as we formally state next.

**Assumption 1.** The functions  $f(t, x)$  and  $f_0(t, x)$  are integrable with respect to  $t$  in the interval  $[0, T]$ .

**Assumption 2.** The functions  $f(t, x)$  and  $f_0(t, x)$  are convex with respect to  $x$  for all times  $t \in [0, T]$ .

If the environment  $f(t, x)$  and functions  $f_0(t, x)$  are known beforehand, finding the action in a viable environment that minimizes the total aggregate cost is equivalent to solving the convex optimization problem in (2) for which a number of algorithms are known. Here, we consider the problem of adapting a strategy  $x(t)$  when the functions  $f(t, x)$  and  $f_0(t, x)$  are *arbitrary* and *revealed causally*. I.e., we want to choose the action  $x(t)$  using observations of viability  $f(t, x)$  and cost  $f_0(t, x)$  in the open interval  $[0, t)$ . This implies that  $f(t, x(t))$  and  $f_0(t, x(t))$  are not observed before choosing  $x(t)$ . The action  $x(t)$  is chosen *ex ante* and the corresponding viability  $f(t, x(t))$  and cost  $f_0(t, x(t))$  are incurred *ex post*. Further observe that the constraints and objective functions may change abruptly if the number of discontinuities in these are finite for finite  $T$ . This makes the problem different

from time varying optimization in which the goal is to track the optimal argument of  $f_0(t, x)$  subject to the constraint  $f(t, x) \leq 0$  under the assumption that these functions change continuously and at a sufficiently small rate [34]–[36].

#### A. Regret and fit

We evaluate the performance of trajectories  $x(t)$  through the concepts of regret and fit. To define regret we compare the accumulated cost  $\int_0^T f_0(t, x(t)) dt$  incurred by  $x(t)$  with the cost incurred by the optimal action  $x^*$  defined in (2),

$$\mathcal{R}_T := \int_0^T f_0(t, x(t)) dt - \int_0^T f_0(t, x^*) dt. \quad (3)$$

Analogously, we define the fit of the trajectory  $x(t)$  as the accumulated penalties  $f(t, x(t))$  incurred for times  $t \in [0, T]$ ,

$$\mathcal{F}_T := \int_0^T f(t, x(t)) dt. \quad (4)$$

The regret  $\mathcal{R}_T$  and fit  $\mathcal{F}_T$  can be interpreted as performance losses associated with online causal operation as opposed to offline clairvoyant operation. If  $\mathcal{F}_T$  is positive in a viable environment we are in a situation in which, had the environment be known a priori, we could have selected an action  $x^\dagger$  with  $f(t, x^\dagger) \leq 0$ . The fit measures how far the trajectory  $x(t)$  comes from achieving that goal. As in the case of the fit, if the regret  $\mathcal{R}_T$  is large we are in a situation in which prior knowledge of environment and cost would have resulted in the selection of the action  $x^*$  – and in that sense  $\mathcal{R}_T$  indicates how much we regret not having had that information available.

Because of the cumulative nature of fit, it is possible to achieve small fit by alternating between actions for which the constraint functions take positive and negative values. This is valid when cumulative constraints are an appropriate model, which happens for quantities that can be stored or preserved in some sense – such as energy budgets enforced through average power constraints. For situations where this is not appropriate, we define the saturated fit in which constraint slacks are saturated to a small constant  $\delta$ . Formally, let  $\delta > 0$  be a positive constant and define the function  $\bar{f}_\delta(t, x) = \max\{f(t, x), -\delta\}$ . Then, the  $\delta$ -saturated fit is defined as

$$\bar{\mathcal{F}}_T = \int_0^T \bar{f}_\delta(t, x(t)) dt. \quad (5)$$

Since  $\bar{f}_\delta(t, x)$  is the pointwise maximum of two convex functions with respect to the actions, it is a convex function itself and  $\bar{\mathcal{F}}_T$  is not different than the fit for the environment defined by  $\bar{f}_\delta(t, x)$ . By taking small values of  $\delta$  we can reduce the negative portion of the fit to be as small as desired.

A good learning strategy is one in which  $x(t)$  approaches  $x^*$ . In that case, the regret and fit grow for small  $T$  but eventually stabilize or, at worst, grow at a sublinear rate. Considering regret  $\mathcal{R}_T$  and fit  $\mathcal{F}_T$  separately, this observation motivates the definitions of feasible trajectories strongly feasible trajectories, and strong optimal trajectories that we formally state next.

**Definition 2.** Given an environment  $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ , a cost  $f_0 : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$ , and a trajectory  $x(t)$  we say that:

**Feasibility.** The trajectory  $x(t)$  is feasible in the environment if the fit  $\mathcal{F}_T$  grows sublinearly with  $T$ . I.e., if there exist a function  $h(T)$  with  $\limsup_{T \rightarrow \infty} h(T)/T = 0$  and a constant vector  $C$  such that for all times  $T$  it holds,

$$\mathcal{F}_T := \int_0^T f(t, x(t)) dt \leq Ch(T). \quad (6)$$

**Strong Feasibility.** The trajectory  $x(t)$  is strongly feasible in the environment if the fit  $\mathcal{F}_T$  is bounded for all  $T$ . I.e., if there exists a constant vector  $C$  such that for all times  $T$  it holds,

$$\mathcal{F}_T := \int_0^T f(t, x(t)) dt \leq C. \quad (7)$$

**Strong optimality.** The trajectory  $x(t)$  is strongly optimal in the environment if the regret  $\mathcal{R}_T$  is bounded for all  $T$ . I.e., if there exists a constant  $C$  such that for all times  $T$  it holds,

$$\mathcal{R}_T := \int_0^T f_0(t, x(t)) dt - \int_0^T f_0(t, x^*) dt \leq C. \quad (8)$$

Having the regret satisfy  $\mathcal{R}_T \leq C$  irrespectively of  $T$  is an indication that  $f_0(t, x(t))$  is close to  $f_0(t, x^*)$  so that the integral stops growing. This is not necessarily so because we can also achieve small regret by having  $f_0(t, x(t))$  oscillate above and below  $f_0(t, x^*)$  so that positive and negative values of  $f_0(t, x(t)) - f_0(t, x^*)$  cancel out. In general, the possibility of having small regret by a trajectory that does not approach  $x^*$  is a limitation of the concept of regret. Alternatively, we can think of the optimal offline policy  $x^*$  as fixing a budget for cost accumulated across time. An optimal online policy meets that budget up to a constant  $C$  – perhaps by overspending at some times and underspending at some other times.

Likewise, when the fit satisfies  $\mathcal{F}_T \leq C$  irrespectively of  $T$ , it suggests that  $x(t)$  approaches the feasible set. This need not be true as it is possible to achieve bounded fit by having  $f(t, x(t))$  oscillate around 0. Thus, as in the case of regret, we can interpret strongly feasible trajectories as meeting the *accumulated* budget  $\int_0^T f(t, x(t)) dt \leq 0$  up to a constant term  $C$ . This is in contrast with feasible actions  $x^\dagger$  that meet the budget  $f(t, x^\dagger) \leq 0$  for all times. Feasible trajectories differ from strongly feasible trajectories in that the fit is allowed to grow at a sublinear rate. This means that feasible trajectories do not meet the *accumulated* budget within a constant  $C$  but do meet the *time averaged* budget  $(1/T) \int_0^T f(t, x(t)) dt \leq 0$  within that constant. The notion of optimality – as opposed to strong optimality – could have been defined as a case in which regret is bounded by a sublinear function of  $T$ . This is not necessary here because our results state strong optimality.

In this work we solve three different problems: (i) Finding strongly optimal trajectories in unconstrained environments. (ii) Finding strongly feasible trajectories. (iii) Finding feasible, strongly optimal trajectories. We develop these solutions in sections III, IV-A, and IV-B, respectively. Before that, we present two pertinent remarks and we clarify concepts with the introduction of an example.

**Remark 1 (Not every trajectory is strongly feasible).** In definition (7) we consider the integral of a measurable function

in a finite interval, hence it is always bounded by a constant. Yet if the latter depends on the time horizon  $T$ , the trajectory is not strongly feasible, because it is not uniformly bounded for all time horizons  $T$ . The same remark is valid for the definitions of strongly optimal and feasible.

**Remark 2 (Connection with Stochastic Optimization).**

One can think about the online learning framework as a generalization of the stochastic optimization setting (see e.g. [19], [37]). In the latter, the objective and constraint functions depend on a random vector  $\theta \in \mathbb{R}^p$ . Formally, the cost is a function  $f_0 : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}$  and the constraints are given by a multivalued function  $f : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m$ . The constrained stochastic optimization problem can be then formulated as

$$\begin{aligned} x^* := \operatorname{argmin} \mathbb{E}[f_0(x, \theta)] \\ \text{s.t.} \quad \mathbb{E}[f(x, \theta)] \leq 0, \end{aligned} \quad (9)$$

where the above expectations are with respect to the random vector  $\theta$ . When the process that determines the temporal evolution of the random vector  $\theta_t$  is stationary, the expectations can be replaced by time averages. In that sense problem (9) is equivalent to the problem of generating trajectories that are feasible and optimal in the sense of Definition 2.

**B. The shepherd problem**

Consider a target tracking problem in which an agent – the shepherd – follows a group of  $m$  targets – the sheep. Specifically, let  $z(t) = [z_1(t), z_2(t)]^T \in \mathbb{R}^2$  denote the position of the shepherd at time  $t$ . To model smooth paths for the shepherd introduce a polynomial parameterization so that each of the position components  $z_k(t)$  can be written as

$$z_k(t) = \sum_{j=0}^{n-1} x_{kj} p_j(t), \quad (10)$$

where  $p_j(t)$  are polynomials that parameterize the space of possible trajectories. The action space of the shepherd is then given by the vector  $x = [x_{10}, \dots, x_{1,n-1}, x_{20}, \dots, x_{2,n-1}]^T \in \mathbb{R}^{2n}$  that stacks the coefficients of the parameterization in (10).

Further define  $y_i(t) = [y_{i1}(t), y_{i2}(t)]^T$  as the position of the  $i$ th sheep at time  $t$  for  $i = 1, \dots, m$  and introduce a maximum allowable distance  $r_i$  between the shepherd and each of the sheep. The goal of the shepherd is to find a path  $z(t)$  that is within distance  $r_i$  of sheep  $i$  for all sheep. This can be captured by defining an  $m$ -dimensional environment  $f$  with each component function  $f_i$  defined as

$$f_i(t, x) = \|z(t) - y_i(t)\|^2 - r_i^2 \quad \text{for all } i = 1..m. \quad (11)$$

That the environment defined by (11) is viable means that it is possible to select a vector of coefficients  $x$  so that the shepherd's trajectory given by (10) stays close to all sheep for all times. To the extent that (10) is a loose parameterization – we can approximate arbitrary functions with sufficiently large index  $n$ , if the time horizon is fixed and not allowed to tend to infinity –, this simply means that the sheep are sufficiently close to each other at all times. E.g., if  $r_i = r$  for all times, viability is equivalent to having a maximum separation between sheep smaller than  $2r$ .

As an example of a problem with an optimality criterion say that the first target – the black sheep – is preferred in that the shepherd wants to stay as close as possible to it. We can accomplish that by introducing the objective function

$$f_0(t, x) = \|z(t) - y_1(t)\|^2. \quad (12)$$

Alternatively, we can require the shepherd to minimize the work required to follow the sheep. This behavior can be induced by minimizing the integral of the acceleration which in turn can be accomplished by defining the optimality criterion [cf. (2)],

$$f_0(t, x) = \|\ddot{z}(t)\| = \left\| \left[ \sum_{j=0}^{n-1} x_{1j} \ddot{p}_j(t), \sum_{j=0}^{n-1} x_{2j} \ddot{p}_j(t) \right] \right\|. \quad (13)$$

Trajectories  $x(t)$  differ from actions in that they are allowed to change over time, i.e., the constant values  $x_{kj}$  in (10) are replaced by the time varying values  $x_{kj}(t)$ . A feasible or strongly feasible trajectory  $x(t)$  means that the shepherd is repositioning to stay close to all sheep. An optimal trajectory with respect to (12) is one in which he does so while staying as close as possible to the black sheep. An optimal trajectory with respect to (13) is one in which the work required to follow the sheep is minimized. In all three cases we apply the usual caveat that small fit and regret may be achieved with stretches of underachievement following stretches of overachievement.

### III. UNCONSTRAINED REGRET IN CONTINUOUS TIME.

Before considering the feasibility problem we consider the following unconstrained minimization problem. Given an unconstrained environment ( $f(t, x) \equiv 0$ ) our goal is to generate strong optimal trajectories  $x(t)$  in the sense of Definition 2, selecting actions from a closed convex set  $X$ , i.e.,  $x(t) \in X$  for all  $t \in [0, T]$ . Given the convexity of the objective function with respect to the action, as per Assumption 2, it is natural to consider a gradient descent controller. To avoid restricting attention to functions that are differentiable with respect to  $x$ , we work with subgradients. For a convex function  $g : X \rightarrow \mathbb{R}$  a subgradient  $g_x$  satisfies the

$$g(y) \geq g(x) + g_x(x)^T(y - x) \quad \text{for all } y \in X. \quad (14)$$

In general, subgradients are defined at all points for all convex functions. At the points where the function  $f$  is differentiable the subgradient and the gradient coincide. In the case of vector functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  we group the subgradients of each component into a matrix  $f_x(x) \in \mathbb{R}^{n \times m}$  defined as

$$f_x(x) = [f_{1,x}(x) \quad f_{2,x}(x) \quad \cdots \quad f_{m,x}(x)], \quad (15)$$

where  $f_{i,x}(x)$  is a subgradient of  $f_i(x)$ . In addition, since the action must always be selected from the set  $X$  we define the controller in a way that the actions are the solution of a projected dynamical system over the set  $X$ . The solution has been studied in [38] and we define the notion as follow.

**Definition 3** (Projected dynamical system). *Let  $X$  be a closed convex set.*

**Projection of a point.** For any  $z \in \mathbb{R}^n$ , there exists a unique element in  $X$ , denoted  $P_X(z)$  such that

$$P_X(z) = \operatorname{argmin}_{y \in X} \|y - z\|. \quad (16)$$

**Projection of a vector at a point.** Let  $x \in X$  and  $v$  a vector, the projection of  $v$  over the set  $X$  at the point  $x$  is

$$\Pi_X(x, v) = \lim_{\delta \rightarrow 0^+} (P_X(x + \delta v) - x) / \delta. \quad (17)$$

**Projected dynamical system.** Given a closed convex set  $X$  and a vector field  $F(t, x)$  which takes elements from  $\mathbb{R} \times X$  into  $\mathbb{R}^n$  the projected differential equation associated with  $X$  and  $F$  is defined to be

$$\dot{x}(t) = \Pi_X(x, F(t, x)). \quad (18)$$

In the above projection if the point  $x$  is in the interior of  $X$  then the projection is not different from the original vector field, i.e.,  $\Pi_X(x, F(t, x)) = F(t, x)$ . On the other hand if the point  $x$  is in the border of  $X$ , then the projection is just the component of the vector field that is tangential to the set  $X$  at the point  $x$ . Let's consider for instance the case where the set  $X$  is a box in  $\mathbb{R}^n$ . Let  $X = [a_1, b_1] \times \dots \times [a_n, b_n]$  where  $a_1 \dots a_n$  and  $b_1 \dots b_n$  are real numbers. Then for each component of the vector field we have that

$$\Pi_X(x, F(t, x))_i = \begin{cases} 0 & \text{if } x_i = a_i \text{ and } F(t, x)_i < 0, \\ 0 & \text{if } x_i = b_i \text{ and } F(t, x)_i > 0, \\ F(t, x)_i & \text{otherwise.} \end{cases} \quad (19)$$

Therefore, when the projection is included, the proposed controller takes the form of the following projected dynamical system:

$$\dot{x} = \Pi_X(x, -\varepsilon f_{0,x}(t, x)), \quad (20)$$

where  $\varepsilon > 0$  is the gain of the controller. Before stating the first theorem we need a Lemma concerning the relation between the original vector field and the projected vector field. This lemma is used in the proofs of theorems 1, 2 and 3.

**Lemma 1.** Let  $X$  be a convex set and  $x_0 \in X$  and  $x \in X$ . Then

$$(x_0 - x)^T \Pi_X(x_0, v) \leq (x_0 - x)^T v. \quad (21)$$

*Proof:* See Appendix A. ■

Let's define an Energy function  $V_{\bar{x}} : \mathbb{R}^n \rightarrow \mathbb{R}$  as

$$V_{\bar{x}}(x) = \frac{1}{2}(x - \bar{x})^T(x - \bar{x}). \quad (22)$$

Where  $\bar{x} \in X \subset \mathbb{R}^n$  is an arbitrary fixed action. We are now in conditions to present the first theorem, which states that the solution of the gradient controller defined in (20) is a strongly optimal trajectory, i.e., with bounded regret for all  $T$ .

**Theorem 1.** Let  $f_0 : \mathbb{R} \times X \rightarrow \mathbb{R}$  be cost function satisfying assumptions 1 and 2, with  $X \subseteq \mathbb{R}^n$  convex. The solution  $x(t)$  of the online projected gradient controller in (20) is strongly optimal in the sense of Definition 2. In particular, the regret  $\mathcal{R}_T$  can be bounded by

$$\mathcal{R}_T \leq V_{x^*}(x(0)) / \varepsilon, \quad \text{for all } T \quad (23)$$

where  $V_{\bar{x}}$  is the Energy function in (22).

*Proof:* Consider an action trajectory  $x(t)$ , an arbitrary given action  $\bar{x} \in X$ , and the corresponding energy function  $V_{\bar{x}}(x(t))$  as per (22). The derivative  $\dot{V}_{\bar{x}}(x(t))$  of the energy function with respect to time is then given by

$$\dot{V}_{\bar{x}}(x(t)) = (x(t) - \bar{x})^T \dot{x}(t). \quad (24)$$

If the trajectory  $x(t)$  follows from the online projected gradient dynamical system in (20) we can substitute the trajectory derivative  $\dot{x}$  by the vector field value and reduce (24) to

$$\dot{V}_{\bar{x}}(x(t)) = (x(t) - \bar{x})^T \Pi_X(x(t), -\varepsilon f_{0,x}(t, x(t))). \quad (25)$$

Use now the result in Lemma 1 with  $v = -\varepsilon f_{0,x}(t, x(t))$  to remove the projection operator from (25) and write

$$\dot{V}_{\bar{x}}(x(t)) \leq -\varepsilon(x(t) - \bar{x})^T f_{0,x}(t, x(t)). \quad (26)$$

Using the defining equation of a subgradient (14), we can upper bound the inner product  $-(x(t) - \bar{x})^T f_{0,x}(t, x(t))$  by the difference  $f_0(t, \bar{x}) - f_0(t, x(t))$  and transform (26) into

$$\dot{V}_{\bar{x}}(x(t)) \leq \varepsilon(f_0(t, \bar{x}) - f_0(t, x(t))). \quad (27)$$

Rearranging terms in the preceding inequality and integrating over time yields

$$\int_0^T f_0(t, x(t)) dt - \int_0^T f_0(t, \bar{x}) dt \leq -\frac{1}{\varepsilon} \int_0^T \dot{V}_{\bar{x}}(x(t)) dt. \quad (28)$$

Since the primitive of  $\dot{V}_{\bar{x}}(x(t))$  is  $V_{\bar{x}}(x(t))$  we can evaluate the integral on the right hand side of (28) and further use the fact that  $V_{\bar{x}}(x) \geq 0$  for all  $x \in \mathbb{R}^n$  to conclude that

$$-\int_0^T \dot{V}_{\bar{x}}(x(t)) dt = V_{\bar{x}}(x(0)) - V_{\bar{x}}(x(T)) \leq V_{\bar{x}}(x(0)). \quad (29)$$

Combining the bounds in (28) and (29) we have that

$$\int_0^T f_0(t, x(t)) dt - \int_0^T f_0(t, \bar{x}) dt \leq V_{\bar{x}}(x(0)) / \varepsilon. \quad (30)$$

Since the above inequality holds for an arbitrary point  $\bar{x} \in \mathbb{R}^n$  it holds for  $\bar{x} = x^*$  in particular. When making  $\bar{x} = x^*$  in (30) the left hand side reduces to the regret  $\mathcal{R}_T$  associated with the trajectory  $x(t)$  [cf. (3)] and in the right hand side we have  $V_{x^*}(x(0)) / \varepsilon = V_{x^*}(x(0)) / \varepsilon$ . Eq. (23) follows because (30) is true for all times  $T$ . This implies that the trajectory is strongly optimal according to (8) in Definition 2. ■

The strong optimality of the online projected gradient controller in (20) that we claim in Theorem 1 is not a straightforward generalization of the optimality of gradient controllers in constant convex potentials. The functions  $f_0$  are allowed to change arbitrarily over time and are not observed until after the cost  $f_0(t, x(t))$  has been incurred.

Since the initial value of the Energy function  $V_{x^*}(x(0))$  is the square of the distance between  $x(0)$  and  $x^*$ , the bound on the regret in (23) shows that the closer we start to the optimal point the smaller the accumulated cost is. Likewise, the larger the controller gain  $\varepsilon$ , the smaller the bound on the regret is. Theoretically, we can make this bound arbitrarily small. This is not possible in practice because larger  $\varepsilon$  entails

trajectories with larger derivatives which cannot be implemented in systems with physical constraints. In the example in Section II-B the derivatives of the state  $x(t)$  control the speed and acceleration of the shepherd. The physical limits of these quantities along with an upper bound on the cost gradient  $f_{0,x}(t, x)$  can be used to estimate the largest allowable gain  $\varepsilon$ .

Another observation regarding the bound on the regret is that it does not depend on the function that we are minimizing –except for the location of the point  $x^*$ . For instance by scaling a function the bound on the regret is kept constant if the same gain  $\varepsilon$  can be selected. This is not surprising since a scaling in the function implies a bigger cost but it also entails a larger action derivative, which allows to track better changes on the function. However, if a bound on the maximum allowed gain exists then the regret bound cannot be invariant to scalings.

**Remark 3.** In discrete time systems where  $t$  is a natural variable and the integrals in (3) are replaced by sums, online gradient descent algorithms are used to reduce regret; see e.g. [26], [27]. The online gradient controller in (20) is a direct generalization of online gradient descent to continuous time. This similarity notwithstanding, the result in Theorem 1 is stronger than the corresponding bound on the regret in discrete time which states a sublinear growth at a rate not faster than  $\sqrt{T}$  if the cost function is convex [26], and  $\log T$  if the cost function is strictly convex [27]. The key where this difference lies is in the fact that discrete time algorithms have to “pay” to switch from the action at time  $t$  to the action at time  $t + 1$ . In the proofs of [26], [27] a term related to the norm square of the gradient is present in the upper bound on the regret while in continuous time this term is absent. The bound on the norm of the gradient is related to the selecting a different action. As in the case of fictitious plays that lead to no regret in the continuous time but not in discrete time (see e.g. [31], [39], [40]) the bounds on the regret in continuous time are tighter than in discrete time for online gradient descent.

#### IV. SADDLE POINT ALGORITHM

Given an environment  $f(t, x)$  and an objective function  $f_0(t, x)$  verifying assumptions 1 and 2 we set our attention towards two different problems: design a controller whose solution is a strongly feasible trajectory and a controller whose solution is a feasible and strongly optimal trajectory. As already noted, when the environment is known beforehand the problem of finding such trajectories is a constrained convex optimization problem, which we can solve using the saddle point algorithm of Arrow and Hurwicz [1]. Following this idea, let  $\lambda \in \Lambda = \mathbb{R}_+^m$ , be a multiplier and define the time-varying Lagrangian associated with the online problem as

$$\mathcal{L}(t, x, \lambda) = f_0(t, x) + \lambda^T f(t, x). \quad (31)$$

Saddle point methods rely on the fact that for a constrained convex optimization problem, a pair is a primal-dual optimal solution if and only if the pair is a saddle point of the Lagrangian associated with the problem; see e.g. [15]. The main idea of the algorithm is then to generate trajectories that descend in the opposite direction of the gradient of the

Lagrangian with respect to  $x$  and that ascend in the direction of the gradient with respect to  $\lambda$ .

Since the Lagrangian is differentiable with respect to  $\lambda$ , we denote by  $\mathcal{L}_\lambda(t, x, \lambda) = f(t, x)$  the derivative of the Lagrangian with respect to  $\lambda$ . On the other hand, since the functions  $f_0(\cdot, x)$  and  $f(\cdot, x)$  are convex, the Lagrangian is also convex with respect to  $x$ . Thus, its subgradient with respect to  $x$  always exist, let us denote it by  $\mathcal{L}_x(t, x, \lambda)$ . Let  $\varepsilon$  be the gain of the controller, then following the ideas in [1] we define a controller that descends in the direction of the subgradient with respect to the action  $x$

$$\begin{aligned} \dot{x} &= \Pi_X(x, -\varepsilon \mathcal{L}_x(t, x, \lambda)) \\ &= \Pi_X(x, -\varepsilon(f_{0,x}(t, x) + f_x(t, x)\lambda)), \end{aligned} \quad (32)$$

and that ascends in the direction of the subgradient with respect to the multiplier  $\lambda$

$$\dot{\lambda} = \Pi_\Lambda(\lambda, \varepsilon \mathcal{L}_\lambda(t, x, \lambda)) = \Pi_\Lambda(\lambda, \varepsilon f(t, x)). \quad (33)$$

The projection over the set  $X$  in (32) is done to assure that the trajectory is always in the set of possible actions. The operator  $\Pi_\Lambda(\lambda, f)$  is a projected dynamical system in the sense of Definition 3 over the set  $\Lambda$ . This projection is done to assure that  $\lambda(t) \in \mathbb{R}_+^m$  for all times  $t \in [0, T]$ . An important observation regarding (32) and (33) is that the environment is observed locally in space and causally in time. The values of the environment constraints and its subgradients are observed at the current trajectory position  $x(t)$  and the values of  $f(t, x(t))$  and  $f_x(t, x(t))$  affect the derivatives of  $x(t)$  and  $\lambda(t)$  only. Notice that if the environment function satisfies  $f(t, x) \equiv 0$  we recover the algorithm defined in (20) as a particular case of the saddle point controller.

A block diagram for the controller in (32) - (33) is shown in Figure 1. The controller operates in an environment to which it inputs at time  $t$  an action  $x(t)$  that results in a penalty  $f(t, x(t))$  and cost  $f_0(t, x(t))$ . The value of these functions and their subgradients  $f_x(t, x(t))$  and  $f_{0,x}(t, x(t))$  are observed and fed to the multiplier and action feedback loops. The action feedback loop behaves like a weighted gradient descent controller. We move in the direction given by a linear combination of the the gradient of the objective function  $f_{0,x}(t, x(t))$  and the constraint subgradients  $f_{i,x}(t, x(t))$  weighted by their corresponding multipliers  $\lambda_i(t)$ . Intuitively, this pushes  $x(t)$  towards satisfying the constraints and to the minimum of the objective function in the set where constraints are satisfied. However, the question remains of how much weight to give to each constraint. This is the task of the multiplier feedback loop. When constraint  $i$  is violated we have  $f_i(t, x(t)) > 0$ . This pushes the multiplier  $\lambda_i(t)$  up, thereby increasing the force  $\lambda_i(t)f_{i,x}(t, x(t))$  pushing  $x(t)$  towards satisfying the constraint. If the constraint is satisfied, we have  $f_i(t, x(t)) < 0$ , the multiplier  $\lambda_i(t)$  being decreased, and the corresponding force decreasing. The more that constraint  $i$  is violated, the faster we increase the multiplier, and the more we increase the force that pushes  $x(t)$  towards satisfying  $f_i(t, x(t)) < 0$ . If the constraint is satisfied, the force is decreased and may eventually vanish altogether if we reach the point of making  $\lambda_i(t) = 0$ .

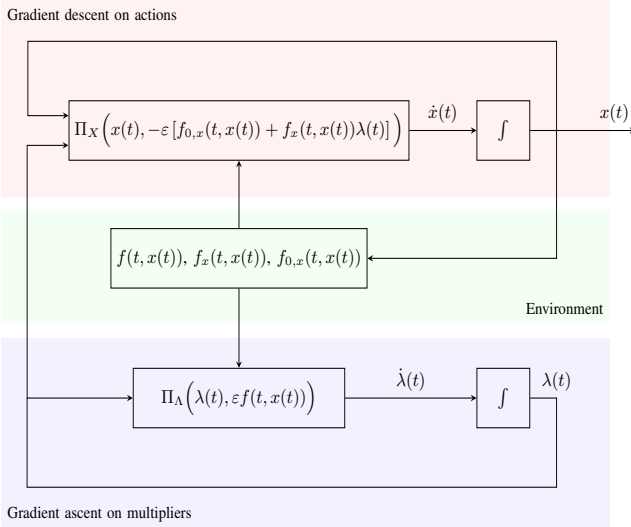


Fig. 1: Block diagram of the saddle point controller. Once that action  $x(t)$  is selected at time  $t$ , we measure the corresponding values of  $f(t, x(t))$ ,  $f_x(t, x(t))$  and  $f_{0,x}(t, x(t))$ . This information is fed to the two feedback loops. The action loop defines the descent direction by computing weighted averages of the subgradients  $f_x(t, x(t))$  and  $f_{0,x}(t, x(t))$ . The multiplier loop uses  $f(t, x(t))$  to update the corresponding weights.

#### A. Strongly feasible trajectories

We begin by studying the saddle point controller defined by (32) and (33) in a problem in which optimality is *not* taken into account, i.e.,  $f_0(t, x) \equiv 0$ . In this case the action descent equation of the controller (32) takes the form:

$$\dot{x} = \Pi_X(x, -\varepsilon \mathcal{L}_x(t, x, \lambda)) = \Pi_X(x, -\varepsilon f_x(t, x)\lambda), \quad (34)$$

while the multiplier ascent equation (33) remains unchanged. The bounds to be derived for the fit ensure that the trajectories  $x(t)$  are strongly feasible in the sense of Definition 2. To state the result consider an arbitrary fixed action  $\bar{x} \in X$  and an arbitrary multiplier  $\bar{\lambda} \in \Lambda$  and define the energy function

$$V_{\bar{x}, \bar{\lambda}}(x, \lambda) = \frac{1}{2} (\|x - \bar{x}\|^2 + \|\lambda - \bar{\lambda}\|^2). \quad (35)$$

We can then bound fit in terms of the initial value  $V_{\bar{x}, \bar{\lambda}}(x(0), \lambda(0))$  of the energy function for properly chosen  $\bar{x}$  and  $\bar{\lambda}$  as we formally state next.

**Theorem 2.** *Let  $f : \mathbb{R} \times X \rightarrow \mathbb{R}^m$ , satisfying assumptions 1 and 2, where  $X \subseteq \mathbb{R}^n$  is a convex set. If the environment is viable, then the solution  $x(t)$  of the dynamical system defined by (34) and (33) is strongly feasible for all  $T > 0$ . Specifically, the fit is bounded by*

$$\mathcal{F}_{T,i} \leq \min_{x^\dagger \in X^\dagger} \frac{1}{\varepsilon} V_{x^\dagger, e_i}(x(0), \lambda(0)), \quad (36)$$

where  $e_i$  with  $i = 1..m$  form the canonical base of  $\mathbb{R}^m$ .

*Proof:* Consider action trajectories  $x(t)$  and multiplier trajectories  $\lambda(t)$  and the corresponding energy function  $V_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t))$  in (35) for arbitrary given action  $\bar{x} \in X$  and multiplier  $\bar{\lambda} \in \Lambda$ . The derivative  $\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t))$  of the energy with respect to time is then given by

$$\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) = (x(t) - \bar{x})^T \dot{x}(t) + (\lambda(t) - \bar{\lambda})^T \dot{\lambda}(t). \quad (37)$$

Substitute the action and multiplier derivatives by their corresponding values given in (34) and (33) to reduce (37) to

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) = & (x(t) - \bar{x})^T \Pi_X(x, -\varepsilon f_x(t, x(t))\lambda(t)) \\ & + (\lambda(t) - \bar{\lambda})^T \Pi_\Lambda(\lambda, \varepsilon f(t, x(t))). \end{aligned} \quad (38)$$

Then, using the result of Lemma 1 for both  $X$  and  $\Lambda$ , the following inequality holds:

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) \leq & \varepsilon (\bar{x} - x(t))^T f_x(t, x(t))\lambda(t) \\ & + \varepsilon (\lambda(t) - \bar{\lambda})^T f(t, x(t)). \end{aligned} \quad (39)$$

Notice that  $f(t, x)\lambda(t)$  is a convex function with respect to the action, therefore we can upper bound the inner product  $(\bar{x} - x(t))^T f_x(t, x(t))\lambda(t)$  by the quantity  $f(t, \bar{x})^T \lambda(t) - f(t, x(t))^T \lambda(t)$  and transform (39) into

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) \leq & \varepsilon (f(t, \bar{x}) - f(t, x(t)))^T \lambda(t) \\ & + \varepsilon (\lambda(t) - \bar{\lambda})^T f(t, x(t)). \end{aligned} \quad (40)$$

Further note that in the above equation the second and the third term are opposite. Thus, it reduces to

$$\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) \leq \varepsilon [\lambda^T(t) f(t, \bar{x}) - \bar{\lambda}^T f(t, x(t))]. \quad (41)$$

Rewriting the above expression and then integrating both sides with respect to time from  $t = 0$  to  $t = T$  we obtain

$$\begin{aligned} \varepsilon \int_0^T (\bar{\lambda}^T f(t, x(t)) - \lambda^T(t) f(t, \bar{x})) dt \\ \leq - \int_0^T \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) dt. \end{aligned} \quad (42)$$

Integrating the right side of the above equation we obtain

$$\begin{aligned} - \int_0^T \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) dt \\ = V_{\bar{x}, \bar{\lambda}}(x(0), \lambda(0)) - V_{\bar{x}, \bar{\lambda}}(x(T), \lambda(T)). \end{aligned} \quad (43)$$

Then using the fact that  $V_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) \geq 0$  for all  $t$ , yields

$$- \int_0^T \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) dt \leq V_{\bar{x}, \bar{\lambda}}(x(0), \lambda(0)). \quad (44)$$

Then, combining (42) and (44), we have that

$$\int_0^T \bar{\lambda}^T f(t, x(t)) - \lambda^T(t) f(t, \bar{x}) dt \leq (V_{x^\dagger, \bar{\lambda}}(x(0), \lambda(0))) / \varepsilon. \quad (45)$$

Since the environment is viable, there exist a fixed action  $x^\dagger$  such that  $f(t, x^\dagger) \leq 0$  for all  $t \geq 0$ . Then choosing  $\bar{x} = x^\dagger$ , since  $\lambda(t) \geq 0$  for all  $t$ , we have that

$$\lambda^T(t) f(t, x^\dagger) \leq 0 \quad \forall t \in [0, T]. \quad (46)$$

Therefore the left hand side of (45) can be lower bounded by

$$\bar{\lambda}^T \int_0^T f(t, x(t)) dt \leq (V_{x^\dagger, \bar{\lambda}}(x(0), \lambda(0))) / \varepsilon. \quad (47)$$

Choosing  $\bar{\lambda} = e_i$  where  $e_i$  is the  $i$ th element of the canonical base of  $\mathbb{R}^m$ , we have that for all  $i = 1..m$ :

$$\int_0^T f_i(t, x(t)) dt \leq (V_{x^\dagger, e_i}(x(0), \lambda(0))) / \varepsilon. \quad (48)$$

Notice that since the above inequality holds for any  $x^\dagger \in X^\dagger$  it is also true for the particular  $x^\dagger$  that minimizes the right hand side. The left hand side of the above inequality is the  $i$ th component of the fit. Thus, since the  $m$  components of the fit of the trajectory generated by the saddle point algorithm are bounded for all  $T$ , the trajectory is strongly feasible with the specific upper bound stated in (36). ■

Theorem 2 assures that if an environment is viable for an agent that selects actions over a set  $X$ , the solution of the dynamical system given by (34) and (33) is a trajectory  $x(t)$  that is strongly feasible in the sense of Definition 2. This result is not trivial, since the function  $f$  that defines the environment is observed causally and can change arbitrarily over time. In particular, the agent could be faced with an adversarial environment that changes the function  $f$  in a way that makes the value of  $f(t, x(t))$  larger. The caveat is that the choice of the function  $f$  must respect the viability condition that there exists a feasible action  $x^\dagger$  such that  $f(t, x^\dagger) \leq 0$  for all  $t \in [0, T]$ . This restriction still leaves significant leeway for strategic behavior. E.g., in the shepherd problem of Section II-B we can allow for strategic sheep that observe the shepherd's movement and respond by separating as much as possible. The strategic action of the sheep are restricted by the condition that the environment remains viable, which in this case reduces to the not so stringent condition that the sheep stay in a ball of radius  $2r$  if all  $r_i = r$ .

Since the initial value of the energy function  $V_{x^\dagger, e_i}(x(0), \lambda(0))$  is the square of the distance between  $x(0)$  and  $x^\dagger$  added to a term that depends on the distance between the initial multiplier and  $e_i$ , the bound on the fit in (36) shows that the closer we start to the feasible set the smaller the accumulated constraint violation becomes. Likewise, the larger the gain  $\varepsilon$ , the smaller the bound on the fit is. As in section III we observe that increasing  $\varepsilon$  can make the bound on the fit arbitrarily small, yet for the same reasons discussed in that section this can't be done.

Further notice that for the saddle point controller defined by (34) and (33) the action derivatives are proportional not only to the gain  $\varepsilon$  but to the value of the multiplier  $\lambda$ . Thus, to select gains that are compatible with the system's physical constraints we need to determine upper bounds in the multiplier values  $\lambda(t)$ . An upper bound follows as a consequence of Theorem 2 as we state in the following corollary.

**Corollary 1.** *Given the controller defined by (34) and (33) and assuming the same hypothesis of Theorem 2, if the set of actions  $X$  is bounded in norm by  $R$ , then the multipliers  $\lambda$  are bounded for all times by*

$$0 \leq \lambda_i(t) \leq (4R^2 + 1), \text{ for all } i = 1, \dots, m. \quad (49)$$

*Proof:* First of all notice that according to (33) a projection over the positive orthant is performed for the multiplier update. Therefore, for each component of the multiplier we have that  $\lambda_i(t) \geq 0$  for all  $t \in [0, T]$ . On the other hand, since the trajectory of the multipliers is defined by  $\dot{\lambda}(t) = \Pi_\Lambda(\lambda(t), \varepsilon f(t, x(t)))$ , while  $\lambda(t) > 0$  we have that  $\dot{\lambda}(t) = \varepsilon f(t, x(t))$ . Let  $t_0$  be the first time instant for which

$\lambda_i(t) > 0$  for a given  $i \in \{1, 2, \dots, m\}$ , i.e.,

$$t_0 = \inf \{t \in [0, T], \lambda_i(t) > 0\}. \quad (50)$$

In addition, let  $T_0^*$  be the first time instant greater than  $t_0$  where  $\lambda_i(t) = 0$ , if this time is larger than  $T$  we set  $T_0^* = T$ , formally this is

$$T_0^* = \max \{\inf \{t \in (t_0, T], \lambda_i(t) > 0\}, T\}. \quad (51)$$

Further define  $t_{s+1} = \inf \{t \in [T_s^*, T], \lambda_i(t) > 0\}$ , and

$$T_s^* = \max \{\inf \{t \in (t_s, T], \lambda_i(t) > 0\}, T\}. \quad (52)$$

From the above definition it holds that in any time in the interval  $(T_s^*, t_{s+1}]$ , we have  $\lambda_i(t) = 0$ . And therefore in those intervals the multipliers are bounded. Consider now  $\tau \in (t_s, T_s^*]$ . In this case it holds that

$$\int_{t_s}^{\tau} \dot{\lambda}_i(t) dt = \int_{t_s}^{\tau} \varepsilon f_i(t, x(t)) dt. \quad (53)$$

Notice that the right hand side of the above equation is, proportional to the  $i$ th component of the fit restricted to the time interval  $[t_0, \tau]$ . In Theorem 2 it was proved that the  $i$ th component of the fit is bounded for all time horizons by  $V_{x^\dagger, e_i}(x(t_s), 0)/\varepsilon$ . In this particular case we have that

$$V_{x^\dagger, e_i}(x(t_s), 0) = \frac{1}{2} ((x(t_s) - x^\dagger)^2 + (0 - e_i)^2), \quad (54)$$

and since for any  $x \in X$  we have that  $\|x\| \leq R$ , we conclude

$$V_{x^\dagger, e_i}(x(t_s), 0) \leq \frac{1}{2} ((2R)^2 + 1^2). \quad (55)$$

Therefore, for all  $\tau \in (t_s, T_s^*]$   $\lambda_i(\tau) \leq \frac{1}{2} (4R^2 + 1^2)$ . This completes the proof that the multipliers are bounded. ■

The bound in Corollary 1 ensures that action derivatives  $\dot{x}(t)$  remain bounded if the subgradients are. This means that action derivatives increase, at most, linearly with  $\varepsilon$  and is not compounded by an arbitrary increase of the multipliers.

The cumulative nature of the fit does not guarantee that the constraint violation is controlled. This is because time intervals of constraint violations can be compensated by time intervals where the constraints are negative. Thus, it is of interest to show that the saddle point controller archives bounded saturated fit for all time horizon. We formalize this result next.

**Corollary 2.** *Let the hypothesis of Theorem 2 hold. Let  $\delta > 0$  and let  $\bar{F}_T$  be the saturated fit defined in (5). Then, the solution of the dynamical system (34) and (33) when  $f(t, x)$  is replaced by  $\bar{f}_\delta(t, x) = \max \{f(t, x), -\delta\}$  archives a bounded saturated fit. Furthermore the bound is given by*

$$\bar{F}_{T,i} \leq \min_{x^\dagger \in X^\dagger} \frac{1}{\varepsilon} V_{x^\dagger, e_i}(x(0), \lambda(0)), \quad (56)$$

where  $e_i$  with  $i = 1..m$  form the canonical base of  $\mathbb{R}^m$ .

*Proof:* Since  $\bar{f}_\delta(t, x)$  is the pointwise maximum of two convex functions, it is a convex function itself. As a consequence of Theorem 2 the fit for the environment  $\bar{f}_\delta(t, x)$  satisfies

$$\int_0^T \bar{f}_\delta(t, x(t)) dt \leq \min_{x^\dagger \in X^\dagger} \frac{1}{\varepsilon} V_{x^\dagger, e_i}(x(0), \lambda(0)). \quad (57)$$



The fact that the left hand side of the above equation corresponds to the saturated fit [c.f. (5)] completes the proof. ■

The above result establishes that a trajectory that follows the saddle point dynamics for the environment defined by  $\bar{f}_\delta(t, x)$  achieves bounded saturated fit. This means that it is possible to adapt the controller (34) and (33), so that the fit is bounded while not alternating between periods of large under and over satisfaction of the constraints

### B. Strongly optimal feasible trajectories

This section presents bounds on the growth of the fit and the regret of the trajectories  $x(t)$  that are solutions of the saddle point controller defined by (32) and (33). These bounds ensure that the trajectory is feasible and strongly optimal in the sense of Definition 2. To derive these bounds we need the following assumption regarding the objective function.

**Assumption 3.** There is a finite constant  $K$  independent of the time horizon  $T$  such that for all  $t$  in the interval  $[0, T]$ .

$$K \geq f_0(t, x^*) - \min_{x \in X} f_0(t, x), \quad (58)$$

where  $x^*$  is the solution of the offline problem (2).

The existence of the bound in (58) is a mild requirement. Since the function  $f_0(t, x)$  is convex, for any time  $t$  it is lower bounded if the action space is bounded, as is the case in most applications of practical interest. The only restriction imposed is that  $\min_{x \in X} f_0(t, x)$  does not become progressively smaller with time so that a uniform bound  $K$  holds for all times  $t$ . The bound can still hold if  $X$  is not compact as long as the span of the functions  $f_0(t, x)$  is not unbounded below. A consequence of Assumption 3 is that the regret cannot *decrease* faster than a linear rate as we formally state in the following lemma.

**Lemma 2.** Let  $X \subset \mathbb{R}^n$  be a convex set. If Assumption 3 holds, then the regret defined in (3) is lower bounded by  $-KT$  where  $K$  is the constant defined in (58), i.e.,

$$\mathcal{R}_T \geq -KT. \quad (59)$$

*Proof:* See Appendix B. ■

Observe that regret is a quantity that we want to make small and, therefore, having negative regret is a desirable outcome. The result in Lemma 2 puts a floor on how much we can succeed in making regret negative. Using the bound in (59) and the definition of the energy function in (35) we can formalize bounds on the regret and the fit, for an action trajectory  $x(t)$  that follows the saddle point dynamics in (32) and (33).

**Theorem 3.** Let  $X \subset \mathbb{R}^n$  be a compact convex set and let  $f : \mathbb{R} \times X \rightarrow \mathbb{R}^m$  and  $f_0 : \mathbb{R} \times X \rightarrow \mathbb{R}$ , be functions satisfying assumptions 1, 2 and 3. If the environment is viable, then the solution of the system defined by (32) and (33) is a trajectory  $x(t)$  that is feasible and strongly optimal for all time horizons  $T > 0$  if the gain  $\varepsilon > 1$ . In particular, the fit is bounded by

$$\mathcal{F}_{T,i} \leq \mathcal{O}(\sqrt{KT}, \varepsilon^0), \quad (60)$$

and the regret is bounded by

$$\mathcal{R}_T \leq \frac{1}{\varepsilon} V_{x^*,0}(x(0), \lambda(0)), \quad (61)$$

where  $V_{\bar{x}, \bar{\lambda}}(x, \lambda)$  is the energy function defined in (35),  $x^*$  is the solution to the problem (2) and  $K$  is the constant defined in (58). The notation  $\mathcal{O}(\varepsilon^0)$  refers to a function that is constant with respect to the gain  $\varepsilon$ .

*Proof:* See Appendix C. ■

Theorem 3 assures that if the environment is viable for an agent selecting actions from a bounded set  $X$ , the solution of the saddle point dynamics defined in (32)-(33) is a trajectory that is feasible and strongly optimal. The bounds on the fit in theorems 2 and 3 prove a trade off between optimality and feasibility. If optimality of the trajectory is not of interest it is possible to get strongly feasible trajectories with fit that is bounded by a constant independent of the time horizon  $T$  (cf. Theorem 2). When an optimality criterion is added to the problem, its satisfaction may come at the cost of a fit that may increase as  $\sqrt{T}$ . An important consequence of this difference is that even if we could set the gain  $\varepsilon$  to be arbitrarily large, the bound on the fit cannot be made arbitrarily small. This bound would still grow as  $\sqrt{KT}$ . The result in Theorem 3 also necessitates Assumption 3 as opposed to Theorem 2.

As in the cases of theorems 1 and 2 it is possible to have the environment and objective function selected strategically. Further note that, again, the initial value of the energy function used to bound regret is related with the square of the distance between the initial action and the optimal offline solution of problem (2). It also follows from the proof that this distance is related to the bound on the fit. Thus, the closer we start from this action the tighter the bounds will be. We next show that similar results holds for the saddle point dynamics if we consider the notion of saturated fit in lieu of fit.

**Corollary 3.** Let the hypothesis of Theorem 3 hold. Let  $\delta > 0$  and let  $\bar{\mathcal{F}}_T$  be the saturated fit defined in (5). Then, the solution of the dynamical system (32) and (33), when  $f(t, x)$  is replaced by  $\bar{f}_\delta(t, x) = \max\{f(t, x), -\delta\}$  achieves a regret satisfying (61) and saturated fit that is bounded by

$$\bar{\mathcal{F}}_{T,i} \leq \mathcal{O}(\sqrt{KT}, \varepsilon^0). \quad (62)$$

*Proof:* Same as Corollary 2. ■

The above result establishes that a trajectory that follows the saddle point dynamics for the environment defined by  $\bar{f}_\delta(t, x)$  achieves bounded saturated fit. This means that it is possible to adapt the controller (32) and (33), so that the growth of the fit is controlled while not alternating between periods of large under and over satisfaction of the constraints. In the next section we evaluate the performance of the saddle point controller, after a pertinent remark on the selection of the gain.

**Remark 4 (Gain depending on the Time Horizon).** If it were possible to select the gain as a function of the time horizon  $T$ , fit could be bounded by a constant that does not grow with  $T$ . Take (88) and choose  $\bar{\lambda} = e_i T$ , where  $e_i$  is the  $i$ -th component of the canonical base of  $\mathbb{R}^m$  we have that

$$T \int_0^T f_i(t, x(t)) dt \leq (V_{x^*, T e_i}(x(0), \lambda(0))) / \varepsilon + KT. \quad (63)$$

With this selection of  $\bar{\lambda}$  the function  $V_{x^*, T e_i}(x(0), \lambda(0))$  grows like  $T^2$ . Dividing both sides of the above equation by

$T$  we have that the  $i$ -th component of the fit is bounded by

$$\mathcal{F}_{T,i} \leq \mathcal{O}(T)/\varepsilon + K. \quad (64)$$

If the gain is set to have order  $\mathcal{O}(T)$ , the right hand side of (64) becomes of order  $\mathcal{O}(T^0)$ . This means that fit can be bounded by a constant that does not depend on  $T$ .

## V. NUMERICAL EXPERIMENTS

We evaluate performance of the saddle point algorithm defined by (32)-(33) in the solution of the shepherd problem introduced in Section II-B. We determine sheep paths using a perturbed polynomial characterization akin to the one in (10). Specifically, letting  $p_j(t)$  be elements of a polynomial basis, the path  $y_i(t) = [y_{i,1}(t), y_{i,2}(t)]^T$  of the  $i$ th sheep is given by

$$y_{i,k}(t) = \sum_{j=0}^{n_i-1} y_{i,k,j} p_j(t) + w_{i,k}(t), \quad (65)$$

where  $k = 1, 2$  denotes different path components,  $n_i$  the dimension of the base that parameterizes the path followed by sheep  $i$ , and  $y_{i,k,j}$  represent the corresponding  $n_i$  coefficients. The noise terms  $w_{i,k}(t)$  are Gaussian white with zero mean, standard deviation  $\sigma$  and independent across components and sheep. Their purpose is to obtain more erratic paths.

To determine  $y_{i,k,j}$  we make  $w_{i,k}(t) = 0$  in (65) and require all sheep to start at  $y_i(0) = [0, 0]^T$  and finish at  $y_i(T) = [1, 1]^T$ . A total of  $L$  random points  $\{\tilde{y}_l\}_{l=1}^L$  are then drawn independently and uniformly at random in the unit box  $[0, 1]^2$ . Sheep  $i = 1$  is required to pass through points  $\tilde{y}_l$  at times  $lT/(L+1)$ , i.e.,  $y_1(lT/(L+1)) = \tilde{y}_l$ . For each of the other sheep  $i \neq 1$  we draw  $L$  random offsets  $\{\Delta\tilde{y}_{i,l}\}_{l=1}^L$  uniformly at random from the box  $[-\Delta, \Delta]^2$  and require the  $i$ th sheep path to satisfy  $y_i(lT/(L+1)) = \tilde{y}_l + \Delta\tilde{y}_{i,l}$ . Paths  $y_i(t)$  are then chosen as those that minimize the path integral of the acceleration squared subject to the constraints of each path

$$\begin{aligned} y_i^* &= \arg\min \int_0^T \|\ddot{y}_i(t)\|^2 dt, \\ \text{s.t. } y_i(0) &= [0, 0]^T, \quad y_i(T) = [1, 1]^T, \\ y_i(lT/(L+1)) &= \tilde{y}_l + \Delta\tilde{y}_{i,l}, \end{aligned} \quad (66)$$

where, by construction  $\Delta\tilde{y}_{1,l} = 0$ . The paths in (66) can be computed as solutions of a quadratic program [41]. Let  $y_i^*(t)$  be the trajectory given by (65) when we set  $y_{i,k,j} = y_{i,k,j}^*$ . We obtain the paths  $y_{i,k}(t)$  by adding  $w_{i,k}(t)$  to  $y_i^*(t)$ .

In subsequent numerical experiments we consider  $m = 5$  sheep, a time horizon  $T = 1$ , and set the proximity constraint in (11) to  $r_i = 0.3$ . We use the polynomial basis  $p_j(t) = t^j$  in both, (10) and (65). The number of basis elements in both cases is set to  $n = n_i = 30$ . To generate sheep paths we consider a total of  $L = 3$  randomly chosen intermediate points, set the variation parameter to  $\Delta = 0.1$ , and the perturbation standard deviation to  $\sigma = 0.1$ . These problem parameters are such that the environment is most likely viable in the sense of Definition 1. We check that this is true by solving the offline feasibility problem. If the environment is not viable a new one is drawn before proceeding to the implementation of (32)-(33).

We emphasize that even if the path of the sheep is known to us, the information is not used by the controller. The latter is only fed information of the position of the sheep at the current time, which it uses to evaluate the environment functions  $f_i(t, x)$  in (11), their gradients  $f_{ix}(t, x)$  and the gradient of  $f_0(t, x)$ . In the first problem considered  $f_0(t, x)$  is identically zero, in the second takes the form of (12) and in the last problem the form of (13). Since the agent is dynamicless, there are not physical constraints on the derivatives of the system, therefore the gain  $\varepsilon$  in (32)-(33) can be set to have any value.

### A. Strongly feasible trajectories

We consider a problem without optimality criterion in which case (32)-(33) simplifies to (34)-(33) and the strong feasibility result in Theorem 2 applies. The system's behavior is illustrated in Figure 2 when the gain is set to  $\varepsilon = 50$ . In this problem the average and maximal speed of the sheep is  $5.1\text{km/h}$  and  $14.8\text{km/h}$  respectively while for the shepherd these are  $6.1\text{km/h}$  and  $18.3\text{km/h}$  for the selected gain. This speeds are in the range of reasonable velocities for this particular problem. A qualitative examination of the sheep and shepherd paths shows that the shepherd succeeds in following the herd. A more quantitative evaluation is presented in Figure 3 where we plot the instantaneous constraint violation  $f_i(t, x(t))$  with respect to each sheep for the trajectories  $x(t)$ . Observe the oscillatory behavior that has the constraint violations  $f_i(t, x(t))$  hovering at around  $f_i(t, x(t)) = 0$ . When the constraints are violated, i.e., when  $f_i(t, x(t)) > 0$ , the saddle point controller drives the shepherd towards a position that makes him stay within  $r_i$  of all sheep. When a constraint is satisfied we have  $f_i(t, x(t)) < 0$ . This drives the multiplier  $\lambda_i(t)$  towards 0 and removes the force that pushes the shepherd towards the sheep (c.f. Figure 3). The absence of this force makes the constraint violation grow and eventually surpass the maximum tolerance  $f_i(t, x(t)) = 0$ . At this point the multipliers start to grow and, as a consequence, to push the shepherd back towards proximity with the sheep.

The behavior observed in Figure 3 does not contradict the result in Theorem 2 which gives us a guarantee on fit, not on instantaneous constraint violations. The components of the fit are shown in Figure 4a where we see that they are indeed bounded. Thus, the trajectory is feasible in the sense of Definition 2, even if the instantaneous problem's constraints are being violated at specific time instances. Further note that the fit is not only bounded but actually becomes negative. This is a consequence of the relatively large gain  $\varepsilon = 50$  which helps the shepherd to respond quickly to the sheep movements. The fit for a second experiment in which the gain is reduced to  $\varepsilon = 5$  is shown in Figure 4b. In this case the fit stabilizes at a positive value. This behavior is expected because reducing  $\varepsilon$  decreases the speed with which the shepherd can adapt to changes in the sheep paths. More to the point, the bound on the fit in Theorem 2 is inversely proportional to the gain  $\varepsilon$ . The paths and instantaneous constraints violations for  $\varepsilon = 5$  are not shown but they are qualitatively similar to the ones shown for  $\varepsilon = 50$  in figures 2 and 3.

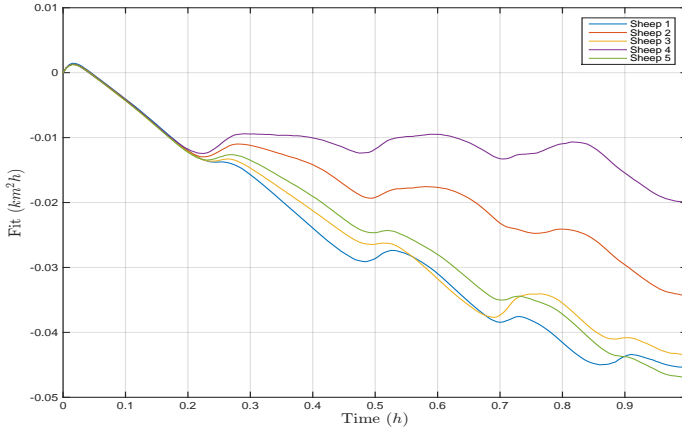
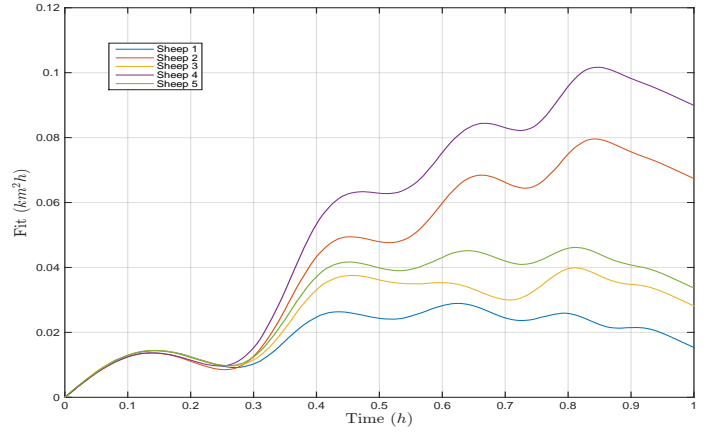
(a) Experiment with gain  $\varepsilon = 50$ .(b) Experiment with gain  $\varepsilon = 5$ .

Fig. 4: Fit  $\mathcal{F}_T$  for two different controller gains in the feasibility-only problem (Section V-A). Fit is bounded in both cases as predicted by Theorem 2. As is also predicted by Theorem 2, the larger the value of the gain  $\varepsilon$  the smaller the bound on the fit of the shepherd's trajectory.

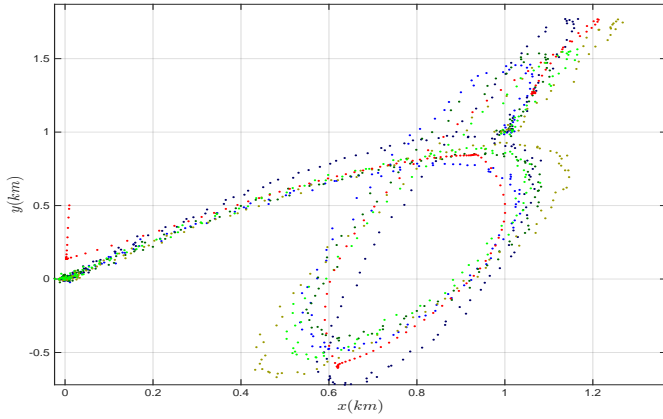
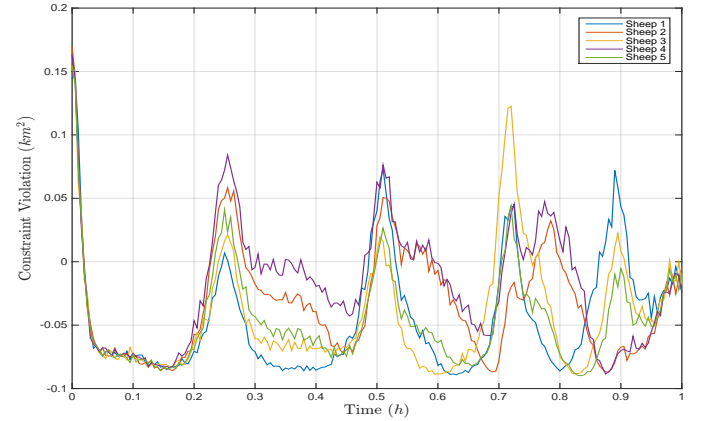


Fig. 2: Path of the sheep and the shepherd for the feasibility-only problem (Section V-A) when the gain of the saddle point controller is set to be  $\varepsilon = 50$ . The shepherd succeed in following the herd since its path – in red – is close to the path of all sheep.

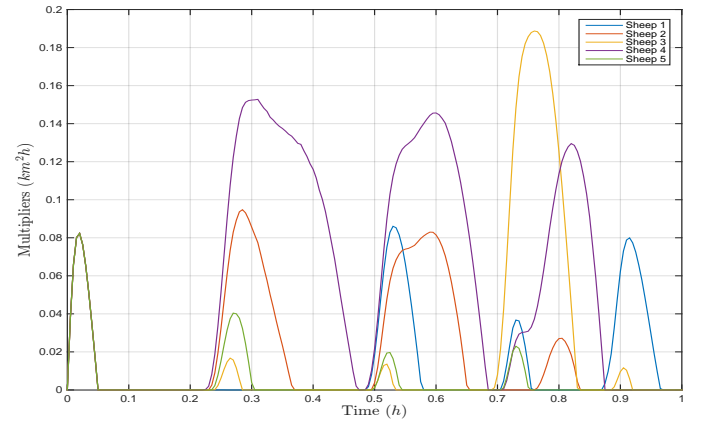
### B. Preferred sheep problem

Besides satisfying the constraints in (11), the shepherd wishes to follow the first (black) sheep as close as possible. This translates into the optimality criterion (12). Since the sheep trajectories are viable the hypotheses of Theorem 3 hold. Thus, for a shepherd following the dynamics (32) and (33), the resulting trajectory is feasible and strongly optimal.

Given that the trajectory is guaranteed to be feasible, we expect to have the fit bounded by a sublinear function of  $T$ . This does happen, as can be seen in the fit trajectories illustrated in Figure 5 where a gain  $\varepsilon = 50$  is used. In fact, the fit does not grow and is bounded by a constant for all time horizons  $T$ . The trajectory is therefore not only feasible but strongly feasible. This does not contradict Theorem 3 because strong feasibility implies feasibility. The reason why it's reasonable to see bounded fit here is that the objective function pushing the shepherd closer to the sheep is, in a sense, redundant with the constraints that push the shepherd to stay closer to all sheep. This redundancy can be also observed in the



(a) Instantaneous constraint value.



(b) Temporal evolution of the multipliers.

Fig. 3: Relationship between the instantaneous value of the constraints and their corresponding multipliers for the feasibility-only problem (Section V-A). At the times in which the value of a constraint is positive, its corresponding multiplier increases. When the value of the multipliers is large enough a decrease of the value of the constraint function is observed. Once the constraint function is negative the corresponding multiplier decreases until it reaches zero.

fact that the fit in this problem (c.f. Figure 5) is smaller than the fit in the problem of Section V-A (c.f. Figure 4a). To explain why this may happen, focus on the value of the multipliers in Figure 3b between, e.g., times  $0.07h < t < 0.21h$ . During this time the multipliers are equal to zero because all constraints are satisfied. As a consequence, the Lagrangian subgradient with respect to the action is identically zero in the time interval. In turn, this implies that the action is constant and no effort is made to reduce the value of the constraints. If the optimality criterion was present, the shepherd would be pushed towards the black sheep and fit would be further reduced.

The regret corresponding to the trajectory for this experiment with  $\varepsilon = 50$  is shown in Figure 6. Since the trajectory is strongly optimal as per Theorem 3, we expect regret to be bounded. This is the case in Figure 6. The path of the shepherd is not shown for this experiment as it is qualitatively analogous to the one in Figure 2 for the feasibility-only problem considered in Section V-A.

### C. Minimum acceleration problem

We consider, an environment defined by the distances between the shepherd and the sheep given by (11), with the minimum acceleration objective defined in (13). Since the construction of the target trajectories gives a viable environment we satisfy, again, the hypotheses of Theorem 3. Hence, for a shepherd following the dynamics given by (32) and (33), the action trajectory is feasible and strongly optimal. In this section the gain of the controller is set to  $\varepsilon = 50$ .

A feasible trajectory implies that the fit must be bounded by a function that grows sublinearly with the time horizon  $T$ . Notice that this is the case in Figure 8. Periods of growth of the fit are observed, yet the presence of inflection points is an evidence of the growth being controlled. The fit in this problem is larger than the one in problem V-B (c.f. figures 5 and 8). This result is predictable since the constraints and the objective function push the action in different directions. For instance, suppose that all constraints are satisfied and that the Lagrange multipliers are zero. Then, the subgradient of the Lagrangian is equal to the subgradient of the objective function. Hence the action will be modified trying to minimize the acceleration without taking the constraints (distance with the sheep) into account. Hence, pushing the action to the boundary of the feasible set. In this problem, this translates into the fact that the shepherd does not follow the sheep as closely as in the problems in sections V-A and V-B (c.f. Figure 7).

Since the trajectory is strongly optimal, we should observe a regret bounded by a constant. This is the case in Figure 9, where in fact we observe negative regret for some time intervals. Negative regret implies that the trajectory of the shepherd is incurring a total cost that is smaller than the one associated with the optimal solution. Notice that while the optimal fixed action minimizes the total cost as defined in (2) it does not minimize the objective at all times. Thus, by selecting different actions the shepherd can suffer smaller instantaneous losses than the ones associated with the optimal fixed action. If this is the case, regret – which is the integral of the difference between these two losses – can be negative.

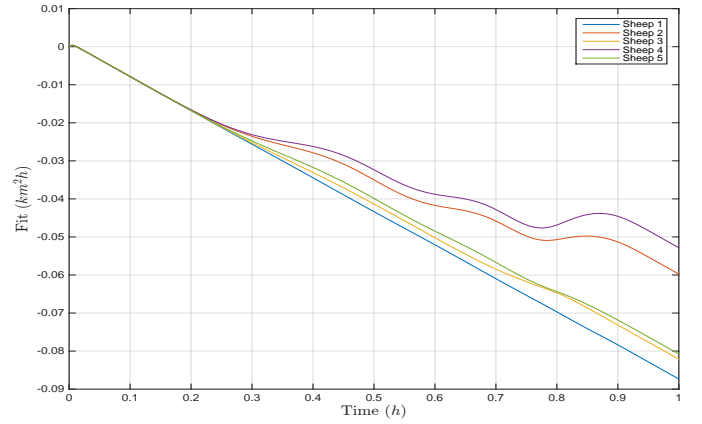


Fig. 5: Fit  $\mathcal{F}_T$  for the preferred sheep problem (Section V-B) when the gain of the saddle point controller is set to be  $\varepsilon = 50$ . As predicted by Theorem 3 the trajectory is feasible since the fit is bounded, and, in fact, appears to be strongly feasible. Since the subgradient of the objective function is the same as the subgradient of the first constrain the fit is smaller than in the pure feasibility problem (c.f. Figure 4).

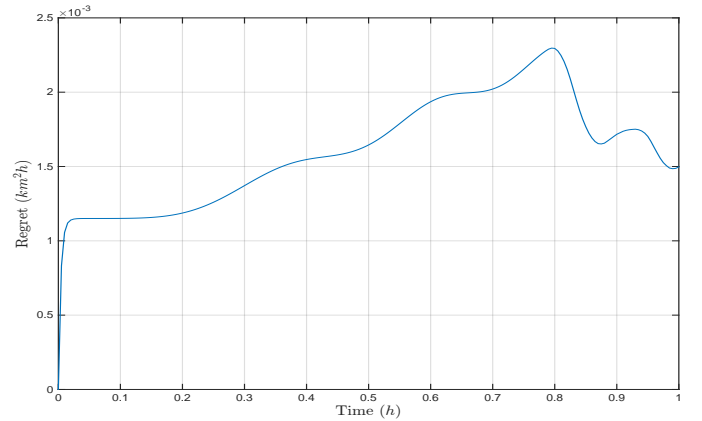


Fig. 6: Regret  $\mathcal{R}_T$  for the preferred sheep problem (Section V-B) when the gain of the saddle point controller is set to be  $\varepsilon = 50$ . The trajectory is strongly optimal, as predicted by Theorem 3, since the regret is bounded by a constant. The initial increment in the regret is due to the fact that the shepherd starts away from the first sheep while in the optimal offline trajectory would start close to it.

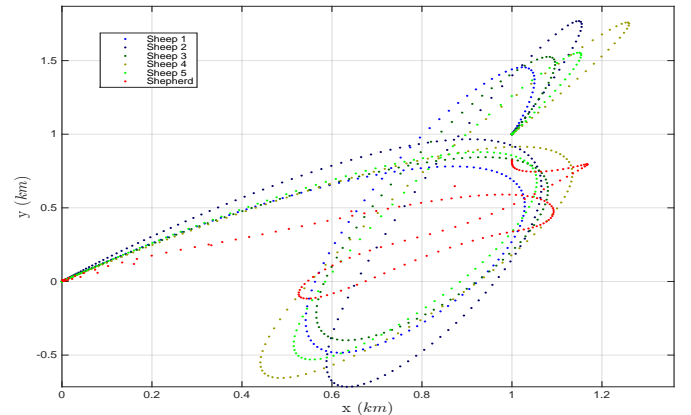


Fig. 7: Path of the sheep and the shepherd for the minimum acceleration problem (Section V-C) when the gain of the saddle point controller is set to be  $\varepsilon = 50$ . Observe that the shepherd path – in red – is not as close to the path of the sheep as in Figure 2. This is reasonable because the objective function and the constraints push the shepherd in different directions.

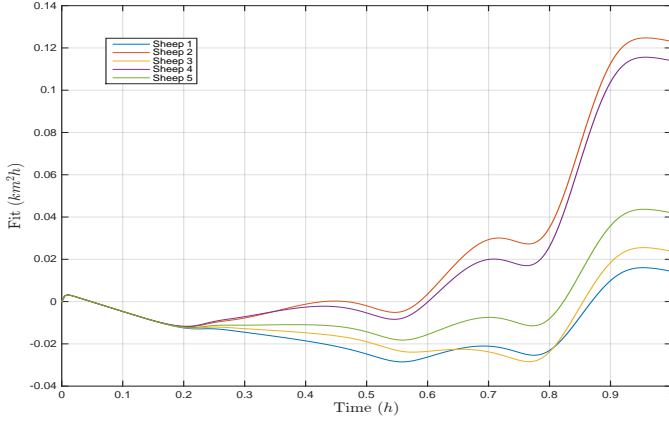


Fig. 8: Fit  $\mathcal{F}_T$  for the minimum acceleration problem (Section V-C) when the gain of the saddle point controller is set to  $\varepsilon = 50$ . Since the fit is bounded, the trajectory is feasible in accordance with Theorem 3. Since the gradient of the objective function and the gradient of the feasibility constraints tend to point in different directions, the fit is larger than in the preferred sheep problem (c.f Figure 5).

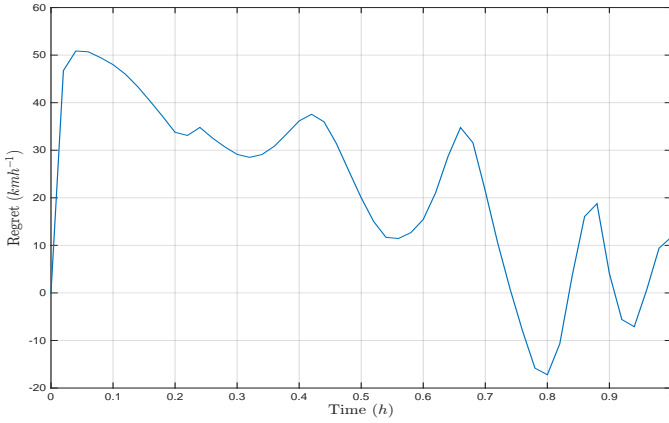


Fig. 9: Regret  $\mathcal{R}_T$  for the minimum acceleration problem (Section V-C) when the gain of the saddle point controller is set to be  $\varepsilon = 50$ . The trajectory is strongly optimal as predicted by Theorem 3. Observe that regret is negative due to the fact that the agent is allowed to select different actions at different times as opposed to the clairvoyant player that is allowed to select a fixed action.

#### D. Saturated Fit

We apply the modified saddle point algorithm in the setting of Section V-B so to consider the saturated fit [c.f. (5)] in lieu of the fit. Since the construction of the target trajectories gives a viable environment the hypotheses of Corollary 3 are satisfied. Hence for a shepherd following the dynamics given by (32) and (33), the trajectories are such that have saturated fit bounded by a function that grows sub linearly and bounded regret. For the simulation in this section the gain of the controller is set to  $\varepsilon = 50$ . Observe that the shepherd succeeds in following the herd, since his path remains close to the sheep (c.f. Figure 10). As predicted by the Corollary 3 the fit of the trajectory is bounded by a function that grows sub linearly and the regret is bounded by a constant as it can be observed in figures 11 and 12 respectively. Further notice that the regret in this scenario is similar to the regret of the trajectory in the preferred sheep problem (c.f. Section V-B).

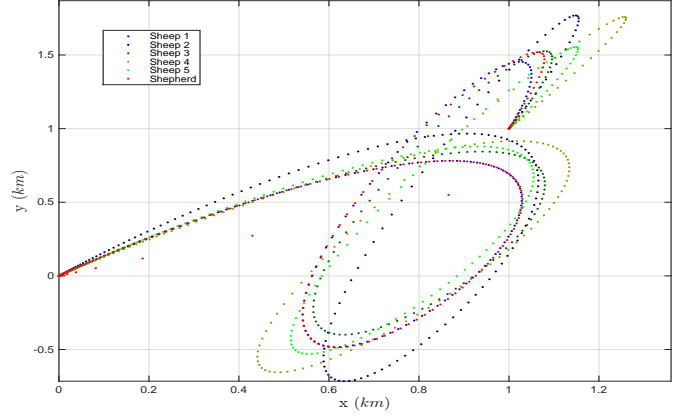


Fig. 10: Path of the sheep and the shepherd for preferred sheep problem when saturated fit is considered (Section V-D) and the gain of the saddle point controller is set to be  $\varepsilon = 50$ . The shepherd succeed in following the herd since its path – in red – is close to the path of all sheep.

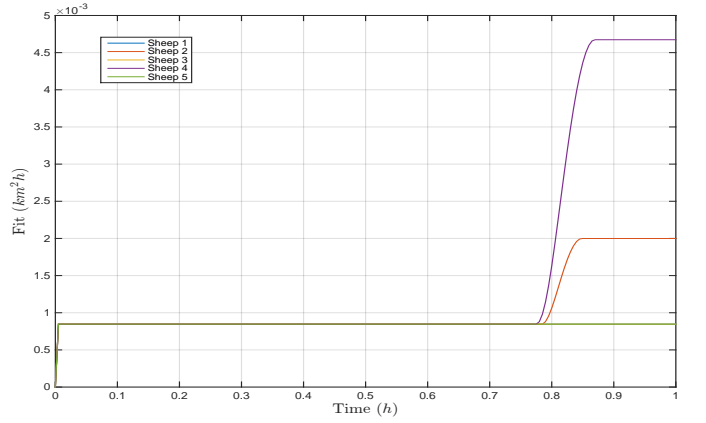


Fig. 11: Saturated fit  $\mathcal{F}_T^{sat}$  for the preferred sheep problem (Section V-D) when the gain of the saddle point controller is set to  $\varepsilon = 50$ . Since the saturated fit grows sublinearly in accordance with Corollary 3, the trajectory is feasible.

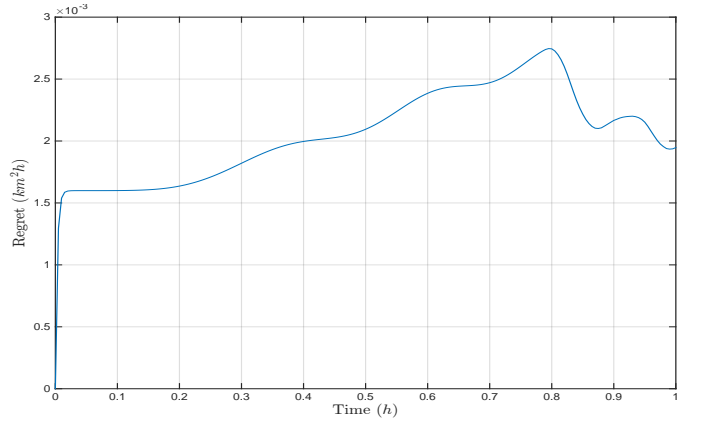


Fig. 12: Regret  $\mathcal{R}_T$  for the preferred sheep problem when saturated fit is considered (Section V-D) and the gain of the saddle point controller is set to be  $\varepsilon = 50$ . The regret is bounded as predicted by Corollary 3 and therefore the trajectory is strongly optimal. Notice that regret in this case is identical to regret in the preferred sheep problem when regular fit is considered (c.f. Figure 6).

## VI. CONCLUSION

We considered a continuous time environment in which an agent must select actions to satisfy a set of constraints that are time varying and unknown a priori. We defined a viable environment as one in which there is a fixed action that satisfies the constraints at all times. We defined the fit as the cumulated constraint violation and the notions of feasible and strongly feasible trajectories. Feasible trajectories are such that the fit is bounded by a constant independent of the time horizon, and strongly feasible trajectories are such that the fit is bounded by a sublinear function of the time horizon. An objective function was considered to select a strategy that meets an optimality criterion and we defined regret in continuous time as the difference between the cumulative costs of the agent and the best clairvoyant agent. We then defined strongly optimal trajectories as those for which the regret is bounded by a constant that is independent of the time horizon.

We proposed an online version of the saddle point controller of Arrow-Hurwicz to generate trajectories with small fit and regret. We showed that for any viable environment the trajectories that follow the dynamics of this controller are: (i) Strongly feasible if no optimality criterion is considered. (ii) Feasible and strongly optimal when an optimality criterion is considered. Numerical experiments on a shepherd that tries to follow a herd of sheep support these theoretical results.

Future research includes studying asymptotic convergence of the saddle point dynamics to the optimal trajectory and studying systems with second order dynamics. In this setting, it is possible to add a term in the objective function that penalizes the action derivative, therefore allowing to control it and maintaining in a desired range.

## APPENDIX

### A. Proof of Lemma 1

In order to develop this proof we need to define the tangent cone and to state Lemma 3 relating the projection of a vector over it and the projection over a convex set

**Definition 4** (Tangent cone). *Let  $X \subset \mathbb{R}^n$  be a closed convex set. We define the tangent cone to  $X$  at  $x_0$  as*

$$T_X(x_0) = \overline{\bigcup_{\theta > 0, x \in X} \theta(x - x_0)}. \quad (67)$$

The above union is over all the points of the set  $X$  and over all the positive reals  $\theta$ . Notice that the  $\bigcup_{\theta > 0} \theta(x - x_0)$  is the ray from  $x_0$  and intersecting the point  $x$ . Thus, the tangent cone is the closure of the cone formed by all rays emanating from  $x_0$  and intersecting at least one point  $x \in X$  with  $x \neq x_0$ .

**Lemma 3.** *Let  $X \in \mathbb{R}^n$  be a closed convex set, let  $x_0 \in X$  and let  $v \in \mathbb{R}^n$ . Then the projection of  $v$  over the set  $X$  at  $x_0$  defined in (16) is*

$$\Pi_X(x_0, v) = P_{T_X(x_0)}(v). \quad (68)$$

*Proof:* The proof follows from Lemma 4.6 in [38]. ■

*Proof of Lemma 1:* Consider the case in which  $x_0 \in \text{int}(X)$ . Then, for any  $v$  there exists a small enough  $\delta > 0$

such that  $x_0 + \delta v \in X$ . Hence  $P_X(x_0 + \delta v) = x_0 + \delta v$  and it holds that

$$P_X(x_0 + \delta v) - x_0 = v\delta. \quad (69)$$

Thus  $\Pi_X(x, v) = v$  and (21) is verified. When  $x_0 \in \partial X$  two cases are possible; either  $x_0 + \delta v \in T_X(x_0)$  for small enough  $\delta > 0$  or  $x_0 + \delta v \notin T_X(x_0)$  for all  $\delta > 0$ . In the first case because of Lemma 3 it is verified that

$$\Pi_X(x_0, v) = P_{T_X(x_0)}(v) = v. \quad (70)$$

And therefore (21) holds. Let us now consider the last case in which  $x_0 \in \partial X$  and  $x_0 + \delta v \notin T_X(x_0)$ . Because  $X$  is a convex set there exists a vector  $a \in \mathbb{R}^n$  with  $\|a\| = 1$  defining a supporting hyperplane at  $x_0$   $\mathcal{H} = \{x \in \mathbb{R}^n : a^T(x - x_0) = 0\}$ , and for all  $x \in X$  we have that

$$a^T(x - x_0) \leq 0. \quad (71)$$

If the set  $X$  is smooth at  $x_0$  then the border of the tangent cone at the point  $x_0$  is contained in the hyperplane  $\mathcal{H}$ , therefore  $\Pi_X(x_0, v) \subset \mathcal{H}$ . Thus,  $a^T \Pi_X(x_0, v) = 0$  and we have as well that  $a^T v \geq 0$ , otherwise there must exist a  $\delta > 0$  such that  $x_0 + \delta v \in T_X(x_0)$ . On the other hand if there is a corner at  $x_0$  there are infinite supporting hyperplanes. One of them verifies that  $a^T v \geq 0$  and contains the boundary of the tangent cone, thus  $a^T \Pi_X(x_0, v) = 0$ . Since  $\Pi_X(x_0, v)$  is the projection of  $v$  over the tangent cone, we have that:  $\Pi_X(x_0, v) = P_{T_X(x_0)}(v) = (a_\perp^T v) a_\perp$ , where  $a_\perp \in \mathbb{R}^n$  and verifies that  $a^T a_\perp = 0$  and  $\|a_\perp\| = 1$ . Projecting the vectors  $x_0 - x$  and  $v$  over  $a$  and  $a_\perp$ , we have

$$(x_0 - x)^T v = (x_0 - x)^T a v^T a + (x_0 - x)^T a_\perp v^T a_\perp. \quad (72)$$

From the previous discussion the above equation reduces to

$$(x_0 - x)^T v = (x_0 - x)^T a v^T a + (x_0 - x)^T \Pi_X(x_0, v). \quad (73)$$

By combining the fact that  $v^T a \geq 0$  and (71) the left hand side of the above equality can be lower bounded by

$$(x_0 - x)^T v \geq (x_0 - x)^T \Pi_X(x_0, v). \quad (74)$$

Hence we have proved the lemma for all possible cases. ■

### B. Proof of Lemma 2

Let  $x(t)$  be the action at time  $t$  when the agent follows the dynamics defined by (32) and (33), because of Assumption 3, we have that

$$f_0(t, x(t)) - f_0(t, x^*) \geq -K, \quad (75)$$

Integrating both sides of the above equation yields

$$\int_0^T f_0(t, x(t)) dt - \int_{t=0}^T f_0(t, x^*) dt \geq -KT. \quad (76)$$

Since the left hand side of the above equation is the regret up to time  $T$  defined in (3), the proof is completed.

### C. Proof of Theorem 3

Consider action trajectories  $x(t)$  and multiplier trajectories  $\lambda(t)$  and the corresponding energy function  $V_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t))$  in (35), for arbitrary given action  $\bar{x} \in \mathbb{R}^n$  and multiplier  $\bar{\lambda} \in \Lambda$ . The derivative  $\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t))$  is given by

$$\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) = (x(t) - \bar{x})^T \dot{x}(t) + (\lambda(t) - \bar{\lambda})^T \dot{\lambda}(t). \quad (77)$$

If the trajectories  $x(t)$  and  $\lambda(t)$  follow from the saddle point dynamical system defined by (32) and (33) respectively we can substitute the action and multiplier derivatives by their corresponding values and reduce (77) to

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) &= (x(t) - \bar{x})^T \Pi_X(x, -\varepsilon \mathcal{L}_x(t, x(t), \lambda(t))) \\ &\quad + (\lambda(t) - \bar{\lambda})^T \Pi_\Lambda(\lambda, \varepsilon \mathcal{L}_\lambda(t, x(t), \lambda(t))). \end{aligned} \quad (78)$$

Then, use Lemma 1 for both  $X$  and  $\Lambda$  to write

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) &\leq -\varepsilon (x(t) - \bar{x})^T \mathcal{L}_x(t, x(t), \lambda(t)) \\ &\quad + \varepsilon (\lambda(t) - \bar{\lambda})^T \mathcal{L}_\lambda(t, x(t), \lambda(t)). \end{aligned} \quad (79)$$

Since  $\mathcal{L}(t, x(t), \lambda(t))$  is a convex function, (14) takes the form  $-(x(t) - \bar{x})^T \mathcal{L}_x(t, x(t), \lambda(t)) \leq \mathcal{L}(t, \bar{x}, \lambda(t)) - \mathcal{L}(t, x(t), \lambda(t))$ . (80)

From the linearity of the Lagrangian with respect to  $\lambda$  we have  $(\lambda(t) - \bar{\lambda})^T \mathcal{L}_\lambda(t, x(t), \lambda(t)) = \mathcal{L}(t, x(t), \lambda(t)) - \mathcal{L}(t, x(t), \bar{\lambda})$ . (81)

Combine expressions (80) and (81) to reduce (79) to

$$\dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) \leq \varepsilon (\mathcal{L}(t, \bar{x}, \lambda(t)) - \mathcal{L}(t, x(t), \bar{\lambda})). \quad (82)$$

Substituting the Lagrangians by the expression (31)

$$\begin{aligned} \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) &\leq \varepsilon [f_0(t, \bar{x}) + \bar{\lambda}^T(t) f(t, \bar{x}) \\ &\quad - f_0(t, x(t)) - \bar{\lambda}^T f(t, x(t))]. \end{aligned} \quad (83)$$

Rewriting the above inequality and integrating both sides with respect to the time from time  $t = 0$  to  $t = T$ , we obtain

$$\begin{aligned} \int_0^T f_0(t, x(t)) - f_0(t, \bar{x}) + \bar{\lambda}^T f(t, x(t)) - \bar{\lambda}^T f(t, \bar{x}) dt \\ \leq -\frac{1}{\varepsilon} \int_0^T \dot{V}_{\bar{x}, \bar{\lambda}}(x(t), \lambda(t)) dt. \end{aligned} \quad (84)$$

Using the result (44) the above equation reduces to yields

$$\begin{aligned} \int_0^T f_0(t, x(t)) - f_0(t, \bar{x}) + \bar{\lambda}^T f(t, x(t)) - \bar{\lambda}^T f(t, \bar{x}) dt \\ \leq \frac{1}{\varepsilon} V_{\bar{x}, \bar{\lambda}}(x(0), \lambda(0)). \end{aligned} \quad (85)$$

Since (85) holds for any  $\bar{x} \in X$  and any  $\bar{\lambda} \in \Lambda$ , it holds for  $\bar{x} = x^*$ ,  $\bar{\lambda} = 0$ . Since  $\lambda^T(t) f(t, x^*) dt \leq 0 \quad \forall t \in [0, T]$  we can lower bound the left hand side of (85) to obtain:

$$\int_0^T f_0(t, x(t)) - f_0(t, x^*) dt \leq \frac{1}{\varepsilon} V_{x^*, 0}(x(0), \lambda(0)). \quad (86)$$

Notice that the left hand side of the above equation is the definition of regret given in (3). Thus, we have showed that (61) holds and since the right hand side of the above equation

is a constant for all  $T$  we proved that the trajectory generated by the saddle point controller is strongly optimal. It remains to prove that the trajectory generated is feasible. Choosing  $\bar{x} = x^*$  in (85) and using the result of Lemma 2 yields

$$\begin{aligned} \int_0^T \bar{\lambda}^T f(t, x(t)) - \lambda^T(t) f(t, x^*) dt \\ \leq \frac{1}{\varepsilon} V_{x^*, \bar{\lambda}}(x(0), \lambda(0)) + KT. \end{aligned} \quad (87)$$

Since  $\lambda^T(t) f(t, x^*) dt \leq 0 \quad \forall t \in [0, T]$  the left hand side of the above equation is lower bounded by  $\bar{\lambda}^T \int_0^T f(t, x(t)) dt$ , yielding

$$\bar{\lambda}^T \int_0^T f(t, x(t)) dt \leq (V_{x^*, \bar{\lambda}}(x(0), \lambda(0))) / \varepsilon + KT. \quad (88)$$

Now let's choose  $\bar{\lambda} = [\mathcal{F}_T]^+ = \left[ \int_0^T f(t, x(t)) dt \right]^+$ . Let  $I = \{i = 1..m \mid \int_0^T f_i(t, x(t)) dt \geq 0\}$ . Notice that if  $i \notin I$ , then  $\bar{\lambda}_i \int_0^T f_i(t, x(t)) dt = 0$ . On the other hand, if  $i \in I$ ,  $\bar{\lambda}_i \int_0^T f_i(t, x(t)) dt = \left( \int_0^T f_i(t, x(t)) dt \right)^2 \geq 0$ . Thus,

$$\bar{\lambda}^T \int_0^T f(t, x(t)) dt = \left\| [\mathcal{F}_T]^+ \right\|^2. \quad (89)$$

Write then inequality (88) for the particular choice of  $\bar{\lambda}$  as

$$\left\| [\mathcal{F}_T]^+ \right\|^2 \leq \frac{1}{\varepsilon} V_{x^*, [\mathcal{F}_T]^+}(x(0), \lambda(0)) + KT. \quad (90)$$

Use the definition of the energy function  $V_{\bar{x}, \bar{\lambda}}(x, \lambda)$  given in (35) to write the above inequality as

$$\left\| [\mathcal{F}_T]^+ \right\|^2 \leq \frac{1}{\varepsilon} \left( \|x(0) - x^*\|^2 + \left\| [\mathcal{F}_T]^+ - \lambda(0) \right\|^2 \right) + KT. \quad (91)$$

Expand the second square in the right hand side of the above expression and re arrange terms to write

$$\begin{aligned} \left\| [\mathcal{F}_T]^+ \right\|^2 + \lambda^T(0) [\mathcal{F}_T]^+ \frac{2}{\varepsilon - 1} \\ \leq \frac{1}{\varepsilon - 1} \left( \|x(0) - x^*\|^2 + \|\lambda(0)\|^2 \right) + KT \frac{\varepsilon}{\varepsilon - 1}. \end{aligned} \quad (92)$$

Adding in both sides of the above inequality  $\|\lambda(0)\|^2 \left( \frac{1}{\varepsilon - 1} \right)^2$ , then factorizing the left hand side the above inequality yields

$$\begin{aligned} \left\| [\mathcal{F}_T]^+ + \lambda(0) \frac{1}{\varepsilon - 1} \right\|^2 &\leq \frac{1}{\varepsilon - 1} \|x(0) - x^*\|^2 + KT \frac{\varepsilon}{\varepsilon - 1} \\ &\quad + \frac{\|\lambda(0)\|^2}{\varepsilon - 1} \left( 1 + \frac{1}{\varepsilon - 1} \right). \end{aligned} \quad (93)$$

Since the term  $\lambda(0)/(\varepsilon - 1)$  is constant with respect to  $T$  it is the case that the norm of  $[\mathcal{F}_T]^+$  is bounded by a function that grows like  $\sqrt{T}$ . On the other hand it also holds that  $\|[\mathcal{F}_T]^+\|$  is bounded by a constant function of the gain  $\varepsilon$ . These observations lead to the conclusion that

$$\|[\mathcal{F}_T]^+\| \leq \mathcal{O}(\sqrt{KT}, \varepsilon^0). \quad (94)$$

The above inequality implies that for any  $i \in I$  it is the case that  $\mathcal{F}_{T,i} \leq \mathcal{O}(\sqrt{KT}, \varepsilon^0)$ . If  $i \notin I$  it means that  $\mathcal{F}_{T,i} < 0$  and it trivially satisfies (60). Which proves that the trajectories



that are solution of the saddle point controller defined by (32) and (33) are feasible since they are bounded by a sublinear function of the time horizon for all  $T$ .

## REFERENCES

- [1] K. J. Arrow and L. Hurwicz, *Studies in linear and nonlinear programming*. CA: Stanford University Press, 1958.
- [2] E. Rimón and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *Robotics and Automation, IEEE Transactions on*, vol. 8, no. 5, pp. 501–518, 1992.
- [3] C. W. Warren, "Global path planning using artificial potential fields," in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*, pp. 316–321, IEEE, 1989.
- [4] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Rob. Res.*, vol. 5, pp. 90–98, Apr. 1986.
- [5] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on robotics and automation*, vol. 16, no. 5, pp. 615–620, 2000.
- [6] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," in *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, vol. 1, pp. 256–263, IEEE, 2000.
- [7] M. W. Hirsch, S. Smale, and R. L. Devaney, *Differential equations, dynamical systems, and an introduction to chaos*, vol. 60. Academic press, 2004.
- [8] M. Krstić and H.-H. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, no. 4, pp. 595–601, 2000.
- [9] K. B. Ariyur and M. Krstic, *Real-time optimization by extremum-seeking control*. John Wiley & Sons, 2003.
- [10] Y. Tan, D. Nešić, and I. Mareels, "On non-local stability properties of extremum seeking control," *Automatica*, vol. 42, no. 6, pp. 889–903, 2006.
- [11] A. Nedić and A. Ozdaglar, "Subgradient methods for saddle-point problems," *Journal of optimization theory and applications*, vol. 142, no. 1, pp. 205–228, 2009.
- [12] H. Uzawa, "Iterative methods for concave programming," *Studies in linear and nonlinear programming*, vol. 6, 1958.
- [13] D. Maistroskii, "Gradient methods for finding saddle points," *Matekon*, vol. 14, no. 1, pp. 3–22, 1977.
- [14] 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.
- [15] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [16] N. Atanasov, J. Le Ny, N. Michael, and G. J. Pappas, "Stochastic source seeking in complex environments," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 3013–3018, IEEE, 2012.
- [17] S.-i. Azuma, M. S. Sakar, and G. J. Pappas, "Stochastic source seeking by mobile robots," *Automatic Control, IEEE Transactions on*, vol. 57, no. 9, pp. 2308–2321, 2012.
- [18] S.-J. Liu and M. Krstic, "Stochastic source seeking for nonholonomic unicycle," *Automatica*, vol. 46, no. 9, pp. 1443 – 1453, 2010.
- [19] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.
- [20] M. Schmidt, N. L. Roux, and F. Bach, "Minimizing finite sums with the stochastic average gradient," *arXiv preprint arXiv:1309.2388*, 2013.
- [21] J. Konečný and P. Richtárik, "Semi-stochastic gradient descent methods," *arXiv preprint arXiv:1312.1666*, 2013.
- [22] A. Mokhtari and A. Ribeiro, "Res: Regularized stochastic bfgs algorithm," *Signal Processing, IEEE Transactions on*, vol. 62, no. 23, pp. 6089–6104, 2014.
- [23] D. Blackwell *et al.*, "An analog of the minimax theorem for vector payoffs," *Pacific Journal of Mathematics*, vol. 6, no. 1, pp. 1–8, 1956.
- [24] V. Vapnik, *The nature of statistical learning theory*. Springer, 2000.
- [25] S. Shalev-Shwartz, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2011.
- [26] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *ICML*, pp. 928–936, 2003.
- [27] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Machine Learning*, vol. 69, no. 2–3, pp. 169–192, 2007.
- [28] S. H. Low and D. E. Lapsley, "Optimization flow control- i: basic algorithm and convergence," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 6, pp. 861–874, 1999.
- [29] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.
- [30] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.
- [31] Y. Viossat and A. Zapechelnuyk, "No-regret dynamics and fictitious play," *Journal of Economic Theory*, vol. 148, no. 2, pp. 825–842, 2013.
- [32] S. Sorin, "Exponential weight algorithm in continuous time," *Mathematical Programming*, vol. 116, no. 1–2, pp. 513–528, 2009.
- [33] J. Kwon and P. Mertikopoulos, "A continuous-time approach to online optimization," *arXiv preprint arXiv:1401.6956*, 2014.
- [34] A. Y. Popkov, "Gradient methods for nonstationary unconstrained optimization problems," *Automation and Remote Control*, vol. 66, no. 6, pp. 883–891, 2005.
- [35] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Interior point method for dynamic constrained optimization in continuous time," *arXiv preprint arXiv:1510.01396*, 2015.
- [36] V. M. Zavala and M. Anitescu, "Real-time nonlinear optimization as a generalized equation," *SIAM Journal on Control and Optimization*, vol. 48, no. 8, pp. 5444–5467, 2010.
- [37] V. S. Borkar *et al.*, "Stochastic approximation," *Cambridge Books*, 2008.
- [38] D. Zhang and A. Nagurney, "On the stability of projected dynamical systems," *J. Optim. Theory Appl.*, vol. 85, pp. 97–124, Apr. 1995.
- [39] S. Hart and A. Mas-Colell, "A general class of adaptive strategies," *Journal of Economic Theory*, vol. 98, no. 1, pp. 26–54, 2001.
- [40] H. P. Young, "The evolution of conventions," *Econometrica: Journal of the Econometric Society*, pp. 57–84, 1993.
- [41] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2011.

PLACE  
PHOTO  
HERE

**Santiago Paternain** received the B.Sc. degree in electrical engineering from Universidad de la República Oriental del Uruguay, Montevideo, Uruguay in 2012. Since August 2013, he has been working toward the Ph.D. degree in the Department of Electrical and Systems Engineering, University of Pennsylvania. His research interests include optimization and control of dynamical systems.

PLACE  
PHOTO  
HERE

**Alejandro Ribeiro** received the B.Sc. degree in electrical engineering from the Universidad de la República Oriental del Uruguay, Montevideo, in 1998 and the M.Sc. and Ph.D. degree in electrical engineering from the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis in 2005 and 2007. From 1998 to 2003, he was a member of the technical staff at Bell-south Montevideo. After his M.Sc. and Ph.D studies, in 2008 he joined the University of Pennsylvania (Penn), Philadelphia, where he is currently the

Rosenbluth Associate Professor at the Department of Electrical and Systems Engineering. His research interests are in the applications of statistical signal processing to the study of networks and networked phenomena. His current research focuses on wireless networks, network optimization, learning in networks, networked control, robot teams, and structured representations of networked data structures. Dr. Ribeiro received the 2012 S. Reid Warren, Jr. Award presented by Penn's undergraduate student body for outstanding teaching, the NSF CAREER Award in 2010, and student paper awards at the 2013 American Control Conference (as adviser), as well as the 2005 and 2006 International Conferences on Acoustics, Speech and Signal Processing. Dr. Ribeiro is a Fulbright scholar and a Penn Fellow