Online Gradient Descent for Linear Dynamical Systems

Marko Nonhoff* Matthias A. Müller*

* Institute of Automatic Control, Leibniz University Hannover, 30167 Hannover, Germany.

Abstract: In this paper, online convex optimization is applied to the problem of controlling linear dynamical systems. An algorithm similar to online gradient descent, which can handle time-varying and unknown cost functions, is proposed. Then, performance guarantees are derived in terms of regret analysis. We show that the proposed control scheme achieves sublinear regret if the variation of the cost functions is sublinear. In addition, as a special case, the system converges to the optimal equilibrium if the cost functions are invariant after some finite time. Finally, the performance of the resulting closed loop is illustrated by numerical simulations.

Keywords: Online convex optimization, linear systems, online learning, online gradient descent, predictive control, real-time optimal control

1. INTRODUCTION

Online convex optimization is an extension of classical numerical optimization to the case where an algorithm operates online in an unknown environment. Whereas in convex optimization the goal is to minimize a given cost function subject to known constraints (Boyd and Vandenberghe, 2004; Nesterov, 2018), in an online convex optimization (OCO) problem the cost function to be minimized is time-varying and the algorithm only has access to past information. Specifically, at every time $t$, the algorithm has to choose an action $y_t \in \mathcal{Y}$ from an action set $\mathcal{Y}$ based on the actions chosen at previous time instances and the corresponding observed cost functions. Then, the environment reveals a new cost function $L_t : \mathcal{Y} \to \mathbb{R}$, which leads to the cost $L_t(y_t)$. The goal is to minimize the total cost in $T$ stages. The classical OCO framework was introduced in (Zinkevich, 2003) and has received considerable interest as a tool for online optimization and learning (see Shalev-Shwartz (2012); Hazan (2016) for an overview).

The performance of OCO algorithms is commonly characterized by regret, which is defined as the gap between the algorithm’s performance and some offline optimum in hindsight. In (Hazan et al., 2007), several algorithms which achieve low static regret, i.e., low regret with respect to the best constant action, are presented. Dynamic benchmarks are proposed in (Besbes et al., 2015; Mokhtari et al., 2016). Sublinear regret is generally desirable because it implies that the algorithm’s performance is asymptotically on average no worse than the benchmark. One major advantage of the OCO framework is its ability to handle time-invariant as well as time-varying constraints (Paternain and Ribeiro, 2016; Cao and Liu, 2019).

In particular, an online version of gradient descent termed online gradient descent (OGD) has proven to be a simple algorithm that achieves low regret in the OCO setting (Hazan, 2016). In OGD, at every time instant $t$, the action $y_t$ is chosen as $y_t = \Pi_\gamma(y_{t-1} - \gamma \nabla f_t(y_{t-1}))$, where $\Pi_\gamma(y)$ denotes a projection of a point $y$ onto the convex constraint set $\mathcal{Y}$, $\gamma \in \mathbb{R}$ is a step size parameter, and $f_t(y)$ is the cost function to be minimized. Hence, instead of solving an optimization problem at every time instant $t$, only one gradient descent step on the previous cost function is employed to reduce computational complexity.

Whereas classical OCO does not consider coupling between time instances and, therefore, no underlying dynamical system, some combinations of OCO with dynamical models have been studied recently. On the one hand, in (Hall and Willett, 2015), algorithms for online prediction of incoming data are proposed, where the data is generated by a dynamical system. It is shown that the regret of the proposed algorithms is low if the environment follows the model of the underlying system. On the other hand, coupling between time instances in the OCO setting has been considered by introducing a switching or cost $d(y_t - y_{t-1})$ to study the effect of a time coupled cost function (Tanaka, 2006; Lin et al., 2013). In (Li et al., 2018), the switching cost is chosen as $\beta \|y_t - y_{t-1}\|^2$, where $\beta \in \mathbb{R}$ is a weighting parameter. This can be interpreted as an additional quadratic cost on the input $u_{t-1}$ of a single integrator system $y_t = y_{t-1} + u_{t-1}$. This approach is extended to general linear systems in (Li et al., 2019). However, this work strongly focuses on the case where predictions of future cost functions are available to the algorithm. It is shown that the algorithm’s regret can be reduced substantially by utilizing predictions. Moreover, in (Abbasi-Yadkori et al., 2014; Cohen et al., 2018; Akbari et al., 2019), linear dynamical systems and quadratic cost functions are considered. Therein, the best linear controller is chosen as the benchmark in the definition of regret and the proposed algorithms apply OCO to optimize over the set of stable, linear policies. This method is generalized to general convex cost functions in (Agarwal et al., 2019). A different approach is taken in (Colombino et al., 2020), where online optimization is used to steer a linear dynamical system to the solutions of time-varying convex optimization problems.
In contrast, many control algorithms able to handle dynamical systems equipped with a cost function as well as state and input constraints exist. In particular, Model Predictive Control (MPC) is able to minimize a given cost function while taking constraints explicitly into account (Rawlings and Mayne, 2009). However, classical MPC techniques require solving a potentially large optimization problem at every time instant. Therefore, inexact or suboptimal MPC algorithms have been proposed, which only complete a finite number of optimization iterations at every time instant (Scokaert et al., 1999; Diehl et al., 2005). Research in the field of suboptimal MPC typically focuses on stability, feasibility, and computational complexity. However, there are limited results on the closed-loop performance of suboptimal MPC schemes.

In this work, OCO for controlling a general linear controllable dynamical system equipped with a general convex cost function \( L_t(x, u) \) is considered. We focus on the cases where the cost functions are strongly convex and smooth, and we propose an algorithm which is similar to online gradient descent in classical OCO. We show that the proposed algorithm attains sublinear regret if the variation of the cost functions is sublinear in time. In contrast to existing results in OCO, which do not consider an underlying dynamical system, this requires novel algorithm design and analysis techniques since the dynamical system cannot move in an arbitrary direction in a single time instant. Moreover, we do not restrict the proposed algorithm to the set of stable, linear controllers. Compared to suboptimal MPC, we explicitly consider a time-varying stage cost which is usually not possible in the MPC literature. In particular, in MPC the cost function needs to be known a priori in order to obtain performance guarantees.

This paper is organized as follows. Section 2 defines the problem setting and states our proposed algorithm. A theoretical analysis of the closed loop and the main theorem are given in Section 3. In Section 4, we illustrate the algorithm’s performance by numerical simulations. Section 5 concludes the paper.

**Notation:** For a vector \( x \in \mathbb{R}^n \), \( \|x\| \) denotes the Euclidean norm, whereas for a matrix \( A \in \mathbb{R}^{n \times m} \), \( \|A\| \) denotes the corresponding induced matrix norm and \( A^T \) the transposed of the matrix. We define by \( \mathbb{N}_{[a,b]} \) the set of natural numbers in the interval \([a,b]\). The gradient of a function \( f(x) \) is denoted by \( \nabla f(x) \). Additionally, \( I_n \) is the identity matrix of size \( n \times n \), and \( 0_{n \times m} \) is the matrix of all zeros of size \( n \times m \).

## 2. Setting and Algorithm

### 2.1 Problem setup

We consider discrete-time linear systems of the form

\[
x_t = Ax_{t-1} + Bu_t \tag{1}
\]

with a given initial condition \( x_0 \in \mathbb{R}^n \), where \( x_t \in \mathbb{R}^n \) are the states of the system, \( u_t \in \mathbb{R}^m \) are the control inputs, and \( A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m} \).

**Remark 1.** Note that we slightly deviate from the usual notation for linear systems (i.e., \( x_{t+1} = Ax_t + Bu_t \)) to facilitate notation in the proposed OCO approach.

At every time instant \( t \in \mathbb{N}_{[1,T]} \), we choose a control action \( u_t \) which is applied to system (1). Then, afterwards, a cost function \( L_t: \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \) is revealed which results in the cost \( L_t(x_t, u_t) \) before we move on to the next time step. As is standard in OCO, we measure our algorithm’s performance by regret. In our case, regret is defined as

\[
\mathcal{R} := \sum_{t=1}^{T} L_t(x_t, u_t) - L_t(x_t^*, u_t^*).
\]

Here, the state and input sequences \( x^* = (x_1^*, \ldots, x_T^*) \) and \( u^* = (u_1^*, \ldots, u_T^*) \) are defined as the solution to the optimization problem

\[
\min_{u_1, u_2, \ldots, u_T} \sum_{t=1}^{T} L_t(x_t, u_t) \quad \text{s.t.} \quad x_t = Ax_{t-1} + Bu_t.
\]

Hence, \( (x_t^*, u_t^*) \) denote the optimal states and inputs at time \( t \) in hindsight, with full knowledge of the cost functions \( L_t \). The regret \( \mathcal{R} \) can therefore be viewed as a measure of how much we regret receiving information about the cost functions \( L_t \) only after we choose a control input \( u_t \). We do not consider any cost on the initial condition \( x_0 \) at time \( t = 0 \) since it cannot be influenced by the algorithm’s decisions, i.e., control inputs \( u_t \).

Similar to various works in OCO (see, e.g., Mokhtari et al. (2016); Li et al. (2018)), we assume the cost functions \( L_t \) to be separable, strongly convex, and smooth as stated in the following assumption.

**Assumption 2.** For every \( t \in \mathbb{N}_{[0,T]} \), the cost function \( L_t \) satisfies the following conditions:

1. \( L_t(x, u) = f^t(x) + f^t(u) \).
2. \( f^t(x) \) is \( \alpha_x \)-strongly convex and \( l_x \)-smooth,
3. \( f^t(u) \) is \( \alpha_u \)-strongly convex and \( l_u \)-smooth,

where an \( \alpha \)-strongly convex function \( f : \mathbb{R}^n \to \mathbb{R} \) satisfies

\[
f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle + \frac{\alpha}{2} \|y-x\|^2, \quad \forall x, y \in \mathbb{R}^n,
\]

and \( l \)-smoothness means that \( f \) satisfies

\[
f(y) \leq f(x) + \langle \nabla f(x), y-x \rangle + \frac{l}{2} \|y-x\|^2, \quad \forall x, y \in \mathbb{R}^n.
\]

We note the fact that \( l \)-smoothness of a convex function \( f \) implies the following Lipschitz condition on the gradient of \( f \) (Nesterov, 2018) \( \|\nabla f(x) - \nabla f(y)\| \leq l \|x-y\| \).

Moreover, we define \( \theta_t = \arg\min_{x} \ f^t(x) \) and \( \eta_t = \arg\min_{u} \ f^t(u) \). In the following, we assume that the minima are attained and therefore finite and, due to convexity, unique. Hence, at each time \( t \) when the cost \( L_t(x, u) \) is measured, that is \( t \in \mathbb{N}_{[1,T]} \), \( (\theta_t, \eta_t) \) is the minimizer of \( L_t(x, u) \). In contrast to the trajectories \( x^* \) and \( u^* \), the sequences \( \theta = (\theta_1, \ldots, \theta_T) \) and \( \eta = (\eta_1, \ldots, \eta_T) \) do in general not satisfy the system dynamics (1).

So far, the optimizer \( (\theta_t, \eta_t) \) and the cost functions \( L_t \) are only defined for \( t \in \mathbb{N}_{[1,T]} \). For the remainder of this work, we fix without loss of generality \( L_0(x, u) = f_0^x(x) + f_0^u(u) \) such that Assumption 2 is satisfied, \( \arg\min_{x} f_0^x(x) = \theta_0 \), and \( \arg\min_{u} f_0^u(u) = \eta_0 \). The values of \( \theta_0 \) and \( \eta_0 \) can be defined arbitrarily, and a convenient choice for our subsequent analysis is given below.

If the cost function \( L_t \) is allowed to change arbitrarily at every time step, we cannot expect to achieve a low regret.
Therefore, we consider the path length as a measure for the variation of the cost functions $L_t$. Path length is defined as the accumulative absolute difference of the optimizer of the cost function at two consecutive time steps:

$$\text{Path length} := \sum_{i=1}^{T} \left\| \theta_t - \theta_{t-1} \right\| + \sum_{i=1}^{T} \left\| \eta_t - \eta_{t-1} \right\|.$$ 

Path length plays an important role in dynamic regret analysis (Mokhtari et al., 2016). In (Li et al., 2018) it is shown for single integrator systems that sublinear regret can be achieved if the path length is sublinear in $T$.

In this paper we consider tracking cost functions as described in the following assumption.

**Assumption 3.** For all $t \in \mathbb{N}_{[1,T]}$, $\theta_t$ and $\eta_t$ satisfy

$$\theta_t = A\theta_t + B\eta_t.$$

Assumption 3 states that the minimum $(\theta_t, \eta_t)$ of the cost function $L_t(x,u)$ is a steady-state with respect to the system dynamics (1), meaning that the control objective is to track a priori unknown and online changing setpoints. Relaxing this assumption to general convex cost functions (Faulwasser et al., 2018) is part of our ongoing work.

Last, we assume that system (1) is controllable and require the norm of $A \in \mathbb{R}^{n \times n}$ to be bounded as follows:

**Assumption 4.** $(A,B)$ is controllable with controllability index $\mu$, i.e.,

$$\text{rank } S_n = \text{rank } (B A B \ldots A^{\mu-1} B) = n.$$

Moreover, $A$ satisfies $\|A\| < \frac{L_{x}+\alpha_x}{2(\varepsilon-\alpha_x)}$.

The norm of $A$ can be interpreted as a measure for the stability/instability of system (1). Achieving low regret for a sequence of cost functions $f^\mu_t(x)$ by applying an algorithm similar to OGD means that the gradient descent step needs to counteract the instability of the system. Hence, it is natural to require an upper bound for the norm of $A$ in terms of the smoothness and convexity of the stage cost $f^\mu_t(x)$. In particular, if the cost function is given by $L_t(x,u) = \frac{\alpha_x}{2} \|x - \theta_t\|^2 + f^\mu_t(u)$ for some $\beta > 0$, we obtain $\alpha_x = \beta \varepsilon$. Hence, in this case, any controllable system satisfies Assumption 4.

### 2.2 Online gradient descent for linear systems

Before we state our algorithm, we first define two useful matrices. The matrix $W = \begin{pmatrix} 0_{m \times (\mu-1)m} & 0_{m \times m} \\ I_{(\mu-1)m} & 0_{(\mu-1)m \times m} \end{pmatrix}$ shifts a vector by $m$ components, whereas the matrix $e = \begin{pmatrix} 0_{m \times (\mu-1)m} \\ I_m \end{pmatrix}$ extracts the last $m$ components.

The proposed OCO scheme is given in Algorithm 1, where we set $g_t = 0$ if $t < 1$ in (4) and (6). Note that the inverse $(S_cS_c^T)^{-1}$ exists due to controllability in Assumption 4. In our setting, at every time step $t$, given the previous state $x_{t-1}$ and cost function $L_{t-1}(x,u)$, Algorithm 1 computes a control input $u_t$ which is then applied to system (1). Afterwards, a new cost function $L_t$ is revealed resulting in the cost $L_t(x_t,u_t)$.

Since no cost function is known at the first time instant, a standard method in OCO is to apply an arbitrary initialization input $v_0$. At time $t = 1$, Algorithm 1 computes

$$v_1 = v_0 - \gamma_0 \nabla f^\mu_0(v_0)$$

and

$$g_1 = -\gamma_0 S_c^T (S_cS_c^T)^{-1} \nabla f^\mu_0(\hat{x}_{1+\mu-1}).$$

Therefore, we fix $\theta_0 = \hat{x}_\mu$ and $\eta_0 = v_0$, which yields $v_1 = v_0$, $g_1 = 0$, and, hence, $u_1 = v_1 = v_0$.

Roughly speaking, the proposed algorithm employs OGD twice to seek the optimal input $\eta_t$ and the optimal state $\theta_t$. First, we apply OGD in (2) to track the optimal input. Next, we would like to apply OGD again on the states of the system which would yield $B\theta_t = B\theta_t - \gamma_x \nabla f_{t-1}(Ax_{t-1} + Bu_t)$. Unfortunately, this is not possible if the system is not 1-step controllable, i.e., $\text{rank}(B) < n$. Instead, as illustrated in Figure 1, the algorithm predicts an input sequence for the next $\mu$ time steps and the corresponding system state $\mu$ time steps ahead in (4). Then, an additional input sequence $u^{\text{OGD}}_t = (u^{\text{OGD}}_1 \ldots u^{\text{OGD}}_{\mu-1}) \in \mathbb{R}^{m}$ is the vector created by stacking the components of the additional input sequence. Moreover, because we want $v_t$ to converge to the optimal input $\eta_t$, we need $g_t$ such that it does not contribute much to the input $f^\mu_0(u_t)$. Therefore, we choose $g_t$ to be the solution of

$$\min_{g_t \in \mathbb{R}^{m}} \|g_t\|^2 \text{ s.t. } S_c g_t = -\gamma_x \nabla f^\mu_{t-1}(\hat{x}_{t+\mu-1}).$$

Solving this optimization problem analytically yields (5). By employing the matrices $W$ and $e$ as in (6), we extract the required input $u^{\text{OGD}}_t$ from $g_t$ at time $t+i$, $i \in \mathbb{N}_{\{0,\mu-1\}}$, to complete one gradient descent step at time $t + \mu - 1$.

### 3. THEORETICAL ANALYSIS

In this section, we state our main result which gives a bound on the regret of Algorithm 1. To shorten notation, let $\Theta_{x} = \sum_{t=1}^{T} \left\| \theta_t - \theta_{t-1} \right\|^2$ and $H_x = \sum_{t=1}^{T} \left\| \eta_t - \eta_{t-1} \right\|^2$. 

**Algorithm 1** (OGD for linear dynamical systems)

| Given step sizes $\gamma_0$ and $\gamma_x$, initialization $v_0$, $x_0$, and state vector $x_{t-1}$. At time $t \in [1,T]$: |
|---|
| **Input OGD** |
| $v_t = v_{t-1} - \gamma_t \nabla f^\mu_{t-1}(v_{t-1})$ |
| **Prediction** |
| $\hat{V}_t = \begin{pmatrix} v_t \\ \vdots \\ v_t \end{pmatrix} \in \mathbb{R}^{\mu m}$ |
| **State OGD** |
| $g_t = -\gamma_x S_c^T (S_cS_c^T)^{-1} \nabla f^\mu_{t-1}(\hat{x}_{t+\mu-1})$ |
| **Output** |
| $u_t = v_t + \sum_{i=0}^{\mu-1} e W^i g_{t-i}$ |
Theorem 5. Let Assumptions 2, 3, and 4 be satisfied and let \( \alpha_u > \frac{2\sqrt{2}}{2\sqrt{2}+1} \). Given step sizes \( \frac{2\sqrt{2}}{2\sqrt{2}+1} < \gamma_c \leq \frac{2}{1+\alpha_u} \) and \( \frac{2}{2\sqrt{2}+1} < \gamma_2 \leq \frac{2}{1+\alpha_u} \), there exist constants \( \beta \geq 0 \) and \( A \gamma_2 > 0 \), such that for each \( T \in \mathbb{N} \geq 1 \), the regret of Algorithm 1 can be upper bounded by

\[
\mathcal{R} \leq C_\mu + \Lambda_\theta \Theta_T + \Lambda_\eta H_T,
\]

where \( C_\mu = z_2 \sum_{t=1}^{T-1} \| x_t - \theta_t \|^2 \).

The proof of Theorem 5 is given in the appendix.

Theorem 5 states that the regret of Algorithm 1 can be upper bounded, where the bound depends linearly on \( \Theta_T \) and \( H_T \), up to a constant \( C_\mu \). First, note that the step sizes \( \gamma_c \) and \( \gamma_2 \) are well defined. Due to the lower bound on \( \alpha_u \) in Theorem 5 we have

\[
\frac{2\sqrt{2}}{2\sqrt{2}+1} = \frac{2}{2\sqrt{2}+1} < \frac{2}{1+\alpha_u} = \frac{2}{1+\alpha_u}.
\]

In addition, the upper bound on \( \| A \| \) in Assumption 4 yields

\[
\frac{2\sqrt{2}}{2\sqrt{2}+1} = \frac{1}{2\sqrt{2}+1} < \frac{1}{1+\alpha_u} = \frac{1}{1+\alpha_u}.
\]

Second, \( C_\mu = z_2 \sum_{t=1}^{T-1} \| x_t - \theta_t \|^2 \) can be bounded in terms of \( \Theta_T \) and \( H_T \) as well. While we omit a detailed derivation due to space limitations, we note that, since \( \Theta_T \) and \( H_T \) are compact sets, if the path length

\[
T \sum_{t=1}^{T} \| x_t - x_{t-1} \|^2 + \sum_{t=1}^{T} \| u_t - u_{t-1} \|^2
\]

is sublinear in \( T \). The corollary below gives a sufficient condition for sublinear regret.

**Corollary 7.** Let \( \theta_t \in \mathcal{D}_\theta \) and \( \eta_t \in \mathcal{D}_\eta \) for all \( t \in \mathbb{N} \), where \( \mathcal{D}_\theta \) and \( \mathcal{D}_\eta \) are compact sets. If the path length

\[
T \sum_{t=1}^{T} \| \theta_t - \theta_{t-1} \|^2 + \sum_{t=1}^{T} \| \eta_t - \eta_{t-1} \|^2
\]

is sublinear in \( T \), then the regret of Algorithm 1 is sublinear in \( T \).

**Proof of Corollary 7.** Compactness of \( \mathcal{D}_\theta \) and \( \mathcal{D}_\eta \) implies \( \| \theta_t - \theta_{t-1} \| \leq d_\theta \) for some \( d_\theta \in \mathbb{R} \) and \( \| \eta_t - \eta_{t-1} \| \leq d_\eta \) for some \( d_\eta \in \mathbb{R} \) for all \( t \in \mathbb{N} \). Hence, we have

\[
\Theta_T = \sum_{t=1}^{T} \| \theta_t - \theta_{t-1} \|^2 \leq d_\theta T \sum_{t=1}^{T} \| \theta_t - \theta_{t-1} \|, \quad H_T = \sum_{t=1}^{T} \| \eta_t - \eta_{t-1} \|^2 \leq d_\eta T \sum_{t=1}^{T} \| \eta_t - \eta_{t-1} \|.
\]

Thus, \( \Theta_T \) and \( H_T \) are sublinear in \( T \). Theorem 5 states that the regret of Algorithm 1 is at most linear in \( \Theta_T \) and \( H_T \). Therefore, the regret of the proposed algorithm is sublinear in \( T \).

**Remark 8.** Consider, as a special case, that for some finite time \( t' \) the minimizer of the cost functions satisfy \( \theta_t = \theta_{t'} \) and \( \eta_t = \eta_{t'} \) for all \( t \geq t' \). Optimality of \( \theta_t, \eta_t \) and the fact that \( \Theta_T \) and \( H_T \) are both finite implies

\[
\mathcal{R} \leq \sum_{t=1}^{T} f_t^{\alpha}(x_t) - f_t^{\alpha}(\theta_t) + f_t^{\alpha}(u_t) - f_t^{\alpha}(\eta_t)
\]

\[
\leq C_\mu + \Lambda_\theta \Theta_T + \Lambda_\eta H_T = C_\mu + \Lambda_\theta \Theta_{t'} + \Lambda_\eta H_{t'}, \quad \Lambda \in \mathbb{R}
\]

for some \( \Lambda \) independent of \( T \), where the second inequality is by Theorem 5 (compare Remark 6). We also have \( f_t^{\alpha}(x_t) - f_t^{\alpha}(\theta_t) \geq 0 \) as well as \( f_t^{\alpha}(u_t) - f_t^{\alpha}(\eta_t) \geq 0 \). Now, if we let \( T \to \infty \), we obtain \( \lim_{T \to \infty} x_t = \theta_t \) and \( \lim_{T \to \infty} u_t = \eta_t \), i.e., the closed loop converges to the optimal equilibrium.

**4. SIMULATIONS**

We illustrate the effectiveness of the proposed algorithm through numerical simulations. We randomly choose the matrix \( A \) as

\[
\begin{pmatrix}
1.05 & 0.7 & 1.75 \\
0.35 & 0.7 & 1.05 \\
1.40 & 1.05 & 1.855
\end{pmatrix}
\]

which yields a controllable system with \( \mu = 3 \). We set \( T = 30 \), the cost function to \( L_t(x,u) = f_t^{\alpha}(x) + f_t^{\alpha}(u) = \frac{1}{2} \| x - \theta_t \|^2 + \frac{1}{2} \| u - \eta_t \|^2 \), and choose \( \gamma_c = 0.98 \) and \( \gamma_x = 0.995 \). It is easy to see that this choice satisfies all conditions in Theorem 5 since \( f_t^{\alpha} \) and \( f_t^{\alpha} \) are \( \alpha \)-strongly convex and \( l \)-smooth, where \( \alpha = l = 1 \). Algorithm 1 is initialized with \( x_0 = (0 \ 0 \ 0)^T \) and \( v_0 = 0 \).

In the first experiment, the sequences \( \theta \) and \( \eta \) are chosen randomly, i.e., \( \theta_t \) is sampled randomly from a uniform distribution over the interval \([-5, 5]\) and \( \theta_1 \) is calculated such that Assumption 3 is satisfied. Then, at every time instant \( t \in \mathbb{N}[2,30] \), the minimizer \( (\theta_t, \eta_t) \) are modified with a probability of 0.1. If they change, \( \eta_t \) is again sampled randomly from the interval \([-5, 5]\) and \( \theta_t \) is calculated accordingly. Hence, the control objective is to track a priori unknown setpoints. Figure 2 presents the resulting closed-loop trajectories for all three states and the input trajectory \( u_t \). The closed loop closely tracks the desired setpoints, and it converges to the optimal states and control input within a few time steps whenever the cost function is changed.

In a second experiment, the system and the cost function are chosen as before and 1000 simulations are conducted,
Fig. 2. Blue, solid: The three states $x_i, i \in \{1, 2, 3\}$, and the input $u_i$; red, dashed: The optimal states $\theta_i, i \in \{1, 2, 3\}$, and input $\eta_t$.

where we set $T = 500$. At every time step $t$, the optimal state and input $(\theta_t, \eta_t)$ is changed with a probability of $0.25 \frac{1}{|N_{1,1000}|}$, where $j \in N_{1,1000}$ is the number of the simulation run. As before, if the cost function is changed, the new value for $\eta_t$ is sampled randomly from the interval $[-5, 5]$ and $\theta_t$ is calculated such that Assumption 3 is satisfied. Thereby, we cover a wide range of path lengths. Figure 3 shows the resulting total cost and path length for each simulation run. Apparently, the total cost grows linearly with the path length as stated in Theorem 5.

5. CONCLUSION

In this paper, we apply online convex optimization to linear dynamical systems equipped with a cost function and propose a first online algorithm for this problem. We derive a bound on the regret of the algorithm and show that it achieves sublinear regret if the variation of the cost functions, measured in terms of path length, is sublinear in time. The performance of the proposed algorithm is illustrated by numerical examples.

Since we do not consider any constraints in this work, an interesting direction for future research is to investigate how input and state constraints can be satisfied by an OCO algorithm. Moreover, Assumption 3 could be relaxed, and more efficient algorithms than OGD could be applied. Finally, predictions on the future cost functions could be incorporated to improve the algorithm’s performance.

REFERENCES

Abbasi-Yadkori, Y., Bartlett, P., and Kanade, V. (2014). Tracking adversarial targets. In Proceedings of the 31st International Conference on Machine Learning, volume 32, 369–377.

Agarwal, N., Bullins, B., Hazan, E., Kakade, S., and Singh, K. (2019). Online control with adversarial disturbances. In Proceedings of the 36th International Conference on Machine Learning, volume 97, 111–119.

Akbari, M., Gharesifard, B., and Linder, T. (2019). An Iterative Riccati Algorithm for Online Linear Quadratic Control. arXiv e-prints. ArXiv:1912.09451.

Besbes, O., Gur, Y., and Zeevi, A. (2015). Non-stationary stochastic optimization. Operations Research, 63(5), 1227–1244.

Boyd, S. and Vandenberghe, L. (2004). Convex Optimization. Cambridge University Press, New York, NY, USA.

Cao, X. and Liu, K.J.R. (2019). Online convex optimization with time-varying constraints and bandit feedback. IEEE Transactions on automatic control, 64(7), 2665–2680.

Cohen, A., Hazan, E., Agarwal, A., and Kale, S. (2007). Logarithmic regret algorithms for online convex optimization. Machine Learning, 69(2), 169–192.

Cohen, A., Hasidim, A., Koren, T., Lutz, N., Mansour, Y., and Talwar, K. (2018). Online linear quadratic control. In Proceedings of the 35th International Conference on Machine Learning, volume 80, 1029–1038.

Colombo, M., DallAnese, E., and Bernstein, A. (2020). Online optimization as a feedback controller: Stability and tracking. IEEE Transactions on Control of Network Systems, 7(1), 422–432.

Diehl, M., Findeisen, R., Allgöwer, F., Bock, H.G., and Schlöder, J.P. (2005). Nominal stability of real-time iteration scheme for nonlinear model predictive control. IEEE Proceedings - Control Theory and Applications, 152(3), 296 – 308.

Faulwasser, T., Grüne, L., and Müller, M.A. (2018). Economic nonlinear model predictive control. Foundations and Trends® in Systems and Control, 5(1), 1–98.

Hall, E.C. and Willett, R.M. (2015). Online convex optimization in dynamic environments. IEEE Journal of Selected Topics in Signal Processing, 9(4), 647–662.

Hazar, E. (2016). Introduction to online convex optimization. Foundations and Trends® in Optimization, 2(3-4), 157–325.

Hazar, E., Agarwal, A., and Kale, S. (2007). Logarithmic regret algorithms for online convex optimization. Machine Learning, 69(2), 169–192.

Li, Y., Qu, G., and Li, N. (2018). Using predictions in online optimization with switching costs: A fast algorithm and a fundamental limit. 2018 Annual American Control Conference (ACC), 3008–3013.

Lin, M., Wierman, A., Andrew, L.L.H., and Theresa, E. (2013). Dynamic right-sizing for power-proportional data centers. IEEE/ACM Transactions on Networking, 21(5), 1378–1391.

Mokhtari, A., Shahrampour, S., Jadbabaie, A., and Ribeiro, A. (2016). Online optimization in dynamic environments: Improved regret rates for strongly convex problems. 2016 IEEE 55th Conference on Decision and Control (CDC), 7195–7201.

Nesterov, Y. (2018). Lectures on Convex Optimization, volume 137 of Springer Optimization and Its Applications. Springer International Publishing, 2 edition.

Fig. 3. Total cost of 1000 simulation runs over the path length of each run.
Appendix A. PROOF OF THEOREM 5

Before the formal proof of Theorem 5, we derive three auxiliary results. First, we study the closed-loop dynamics resulting from application of Algorithm 1 to system (1). Let \( t \in \mathbb{N}_{[1,T+\mu+1]} \), then we have by repeatedly applying the system dynamics (1)

\[
x_{t+\mu+1} = A^\mu x_{t+1} + \sum_{i=0}^{\mu-1} A^i B u_{t+i+\mu-1-i},
\]

where we use \( W^\mu = 0 \) in the second and the relation \( S_i W + A^\mu B e = A S_i e \) in the third line. Inserting (4) yields

\[
\hat{x}_{t+1} = A \hat{x}_t + B v_{t-\mu+2} + A S_i e_{t-\mu+1} + S_i E_{t-\mu+1} E_i \left( \hat{V}_{t-\mu+2} - \hat{V}_{t-\mu+1} \right).
\]

(4.2)

Third, we have the following result on the rate of convergence of gradient descent (Nesterov, 2018, Chapter 2.3.3, Theorem 2.3.4). For an \( \alpha \)-convex and \( l \)-smooth function \( f : \mathbb{R}^n \to \mathbb{R} \) to be minimized, one gradient descent step \( x_1 = x_0 - \gamma \nabla f(x_0) \) yields

\[
\| x_1 - \theta \| \leq \kappa \| x_0 - \theta \|, 
\]

(4.3)

where \( \theta = \arg \min_x f(x) \) and \( \kappa = 1 - \alpha \gamma_c \). Accordingly, we define \( \kappa_y = 1 - \alpha_y \gamma_c \) and \( \kappa_v = 1 - \alpha_v \gamma_c \). Before we prove Theorem 5, we introduce the following supporting lemma.

Lemma 9. Let Assumptions 2, 3, and 4 be satisfied. Let \( \alpha_u > 2\sqrt{2|A|} \gamma_u \) and \( \kappa = \gamma - \frac{2}{\kappa + \alpha} \gamma_c \). Given step sizes \( \eta_{t+1} = \eta_t - \gamma t \), the predicted states \( \hat{x}_t \) satisfy

\[
\sum_{t=1}^{\tau} \| \hat{x}_{t+\mu-1} - \theta_{t-1} \| ^2 \leq C_\theta / l \sum_{t=1}^{\tau-1} \| \theta_t - \theta_{t-1} \| ^2
\]

\[
+ C_\theta / l \sum_{t=1}^{\tau-1} \| \eta_{t-1} \| ^2, 
\]

\[
\text{where } \tau \in \mathbb{N}_{[1,T]}, C_\theta = \frac{8\epsilon_c (\| A \|^2 l^2 + S_i E_{ta} \| x_0 \|^2 l^2 \mu^2)}{(1-2\epsilon_c)(1-4\| A \| l^2 \kappa_y ^2)} 
\]

\[
\text{and } C_\theta = \frac{4|A| \epsilon_c / l}{1-4|A| l^2 \kappa_y ^2}. 
\]

Proof. Note that the bounds on the step sizes imply \( 1 - 2\kappa_y > 0 \) and \( 1 - 4 \| A \| ^2 \kappa_x ^2 > 0 \). By Jensen’s inequality we have

\[
\left\| \sum_{i=1}^p a_i \right\|^2 \leq \sum_{i=1}^p \lambda_i \| a_i \|^2, 
\]

(4.4)

where \( \sum_{i=1}^p a_i = 1 \) and \( \lambda_i > 0 \) for all \( i \in \mathbb{N}_{[1,p]} \). In particular, we can choose \( \lambda_i = p \) for all \( i \in \mathbb{N}_{[1,p]} \).

Next, we derive three auxiliary results on the relation of \( v_t \) and \( \eta_t \). Remember that \( \eta_0 = v_0 = v_1 \). Hence, we have

\[
\sum_{t=1}^\tau \| v_{t+1} - \eta_t \|^2 \leq \sum_{t=1}^\tau \| v_t - \gamma t \nabla f_t (v_t) - \eta_t \|^2 
\]

(4.3)

\[
\leq \sum_{t=1}^\tau \| v_t - \eta_t \|^2, 
\]

(4.4)

\[
\leq 2\kappa_v \sum_{t=1}^\tau \| v_t - \eta_{t-1} \|^2 + 2\kappa_v \sum_{t=1}^\tau \| \eta_t - \eta_{t-1} \|^2 
\]

\[
\leq 2\kappa_v \sum_{t=1}^\tau \| v_{t+1} - \eta_t \|^2 + 2\kappa_v \sum_{t=1}^\tau \| \eta_t - \eta_{t-1} \|^2. 
\]

Since \( 1 - 2\kappa_y > 0 \), rearranging yields

\[
\sum_{t=1}^\tau \| v_{t+1} - \eta_t \|^2 \leq 2\kappa_v \frac{\tau - 1}{1 - 2\kappa_y} \sum_{t=1}^\tau \| \eta_t - \eta_{t-1} \|^2. 
\]

(4.5)

Moreover, we have
\[
\sum_{t=1}^{\tau} \|v_t - \eta_t\|^2 \leq \frac{2}{1 - 2\kappa_e^2 v} \sum_{t=1}^{\tau} \|v_t - \eta_{t-1}\|^2
\]

(\text{A.6})

Finally, due to optimality of \( \eta_t \) and, hence, \( \nabla f_t^i(\eta_t) = 0 \),
\[
\sum_{t=1}^{k-1} \|v_{t+1} - v_t\|^2 \leq 2 \sum_{t=1}^{k-1} \|\nabla f_t^i(v_t) - \nabla f_t^i(\eta_t)\|^2
\]
\[
\leq 2\kappa_t^2 \sum_{t=1}^{k-1} \|v_t - \eta_t\|^2 \leq \frac{2\kappa_t^2}{1 - 2\kappa_e^2 v} \sum_{t=1}^{k-1} \|\eta_t - \eta_{t-1}\|^2 ,
\]

(\text{A.7})

where we use the fact that \( \kappa_t \)-smoothness of \( f_t^i(u) \) implies a Lipschitz condition on its gradient in the second line.

Last, we show the bound on the predicted states. Since \( \theta_0 = \hat{x}_\mu \), we have
\[
\sum_{t=1}^{\tau} \|\hat{x}_{t+1} - \theta_{t-1}\|^2 \leq \sum_{t=1}^{\tau} \|\hat{x}_{t+1} - \theta_t\|^2
\]

(\text{A.2),(A.4)}
\[
\leq 4 \|A\|^2 \sum_{t=1}^{\tau-1} \left( 2 \|A(\hat{x}_{t+1} - x_c g_t - \theta_t)\|^2 + 4 \|S_{c} E_{01} E_{10}^T (\hat{V}_{t+1} - \hat{V}_t)\|^2 \right)
\]

\[
= 4 \|A\|^2 \sum_{t=1}^{\tau-1} \|\hat{x}_{t+1} - \theta_{t-1}\|^2 + \gamma_\delta \sum_{t=1}^{\tau-1} \left( \|\nabla f_t^i(\hat{x}_{t+1}) - \nabla f_t^i(\eta_{t-1})\|^2 + \|B(v_{t+1} - \eta_{t-1})\|^2 \right)
\]

(\text{A.4),(A.5)}
\[
\leq 4 \|A\|^2 \sum_{t=1}^{\tau-1} \|\theta_t - \theta_{t-1}\|^2 + \frac{4 \|B\|^2 \kappa_x^2}{1 - 2\kappa_e^2 v} \sum_{t=1}^{\tau-1} \|\eta_t - \eta_{t-1}\|^2
\]

\[
+ \frac{4 \|S_{c} E_{01}\|^2 (\mu - 1)}{1 - 2\kappa_e^2 v} \sum_{t=1}^{\tau-1} \|v_{t+1} - v_t\|^2
\]

(\text{A.3),(A.7)}
\[
\leq 4 \|A\|^2 \kappa_x^2 \sum_{t=1}^{\tau-1} \|\hat{x}_{t+1} - \theta_{t-1}\|^2 + 4 \|A\|^2 \sum_{t=1}^{\tau-1} \|\theta_t - \theta_{t-1}\|^2
\]

\[
+ \frac{4 \|S_{c} E_{01}\|^2 (1 - 4 \|A\|^2 \kappa_x^2)}{1 - 2\kappa_e^2 v} \sum_{t=1}^{\tau-1} \|\eta_t - \eta_{t-1}\|^2,
\]

Since \( 1 - 4 \|A\|^2 \kappa_x^2 > 0 \), rearranging yields the desired bound. \qed

\textbf{Proof of Theorem 5.} The proof consists of three parts. First, we derive a bound on the cost of the control inputs. In the second part, we derive a bound on \( \sum_{t=1}^{k} \|\hat{x}_{t+1} - x_c g_{t+1}\|^2 \), which will be useful to bound the cost on the states. The last part is to combine these results to find a bound on the regret of Algorithm 1.

First, we have for \( k \in \mathbb{N}_{[1,T]} \)
\[
\sum_{t=1}^{k} \|eW^i g_{t-1}\|^2 \leq \mu \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2
\]

(\text{A.4})
\[
\leq \mu \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2
\]

(\text{A.5})
\[
\leq \mu \sum_{t=1}^{k} \|eW^i g_{t-1}\|^2
\]

(\text{A.6})
\[
\leq 2 \sum_{t=1}^{k} \mu - 1 \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2
\]

where we use the fact that \( g_t = 0 \) for \( t \leq 1 \) in the second line. \( C_1 = \sum_{t=1}^{\mu-1} \|eW^i S_{c} S_{c}^T (S_{c} S_{c}^T)^{-1}\|^2 \) and \( \nabla f_t^i(\theta_{t-1}) = 0 \) in the third line, and Lipschitz continuity of the gradients in the last line. By Lemma 9, we obtain
\[
\sum_{t=1}^{T} \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2 \leq \mu \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2
\]

(\text{A.9})

Having established a bound on the regret of the control inputs chosen by Algorithm 1, we derive a bound on \( \sum_{t=1}^{k} \|\hat{x}_{t+1} - x_c g_{t+1}\|^2 \) in the second part of the proof. First, let \( k \in \mathbb{N}_{[1,T-\mu+1]} \). The last component of \( \hat{V}_t \) and \( V_t^\mu \) is the same, therefore, we insert the matrix
\[
E_{10} = \left( \begin{array}{c} I_{(\mu-1)m} \\ 0_{m \times (\mu-1)m} \end{array} \right) \in \mathbb{R}^{mp \times (\mu-1)m} \quad \text{which yields}
\]
\[
\sum_{t=1}^{k} \|S_{c}(\hat{V}_t - V_t^\mu)\|^2 = \sum_{t=1}^{k} \|S_{c} E_{10} E_{10}^T (\hat{V}_t - V_t^\mu)\|^2
\]

\[
\leq \|S_{c} E_{10}\|^2 \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \|eW^i g_{t-1}\|^2
\]

(\text{A.5})
\[
= \|S_{c} E_{10}\|^2 \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \|v_{t+1} - v_t\|^2
\]

where we make use of a telescoping series in the last line. Next, by inserting (A.4), upper bounding \( i \), and finally positivity of the norm we have
\[
\sum_{t=1}^{k} \left\| S_c \left( \dot{V}_t - V^\mu \right) \right\|^2 \\
\leq \left\| S_c E_{10} \right\|^2 \sum_{t=1}^{k} \sum_{i=1}^{\mu-1} \sum_{j=1}^{i} \left\| v_{t+j} - v_{t+j-1} \right\|^2 \\
\leq \left\| S_c E_{10} \right\|^2 (\mu - 1)^2 \sum_{t=1}^{k} \sum_{i=1}^{\mu-1} \left\| v_{t+j} - v_{t+j-1} \right\|^2 \\
\leq \left\| S_c E_{10} \right\|^2 \left( \mu - 1 \right)^3 \sum_{t=1}^{k} \left\| v_{t+1} - v_t \right\|^2 \\
\leq C_2 \sum_{t=1}^{k+\mu-2} \left\| \eta_t - \eta_{t-1} \right\|^2 , \tag{A.10}
\]
where \( C_2 = \frac{2\left\| S_c E_{10} \right\|^2 \gamma_2^2 \left( \mu - 1 \right)^3}{1 - \gamma_2^2} \). Next, we apply (A.4) to obtain
\[
\sum_{t=1}^{k} \sum_{i=1}^{\mu-1} \left\| S_c (W^T)^i g_{t+i} \right\|^2 \leq \sum_{t=1}^{k} \sum_{i=0}^{\mu-1} \left\| S_c (W^T)^i g_{t+i} \right\|^2 \\
\leq \mu \sum_{t=1}^{k} \sum_{i=1}^{\mu-1} \left\| S_c (W^T)^i g_t \right\|^2 \\
\leq \mu \gamma_2^2 C_3 \sum_{t=1}^{k} \left\| \nabla f^x_t \left( x_{t+\mu-1} \right) - \nabla f^x_t \left( \theta_{t-1} \right) \right\|^2 , \tag{A.7}
\]
where \( C_3 = \frac{\sum_{i=0}^{\mu-1} \left\| S_c (W^T)^i S^T_c \right\|^2}{1 - \gamma_2^2} \) and we use the fact that \( \nabla f^x_t \left( \theta_t \right) = 0 \) due to optimality of \( \theta_t \) in the last line. Lipschitz continuity of the gradient yields
\[
\sum_{t=1}^{k} \left\| S_c (W^T)^i g_{t+i} \right\|^2 \leq \mu \gamma_2^2 C_3 \sum_{t=1}^{k} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 .
\]
By Lemma 9, we obtain
\[
\sum_{t=1}^{k} \left\| S_c (W^T)^i g_{t+i} \right\|^2 \leq \mu \gamma_2^2 C_3 \sum_{t=1}^{k} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 .
\]
Combining the last two results yields the desired bound
\[
\sum_{t=1}^{k} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 \\
\leq \frac{1}{2} \sum_{t=1}^{k+\mu-2} \left\| \dot{V}_t - V^\mu \right\|^2 , \tag{A.11}
\]
For \( p \) \( (\theta_t, \eta_t) \), we have
\[
\mathcal{R} \leq \sum_{t=1}^{T} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 + \frac{l_x T}{2} \sum_{t=1}^{T-\mu+1} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 + \frac{2l_u}{1 - 2\kappa_v^2} T H_T \\
+ \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \theta_t - \theta_{t-1} \right\|^2 + \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \eta_t - \eta_{t-1} \right\|^2 , \tag{A.12}
\]
We finally compute the regret of Algorithm 1. By optimality of \( \left( \theta_t, \eta_t \right) \), we have
\[
\mathcal{R} \leq \sum_{t=1}^{T} \left\| x_t - x^* \right\|^2 + \frac{l_x T}{2} \sum_{t=1}^{T-\mu+1} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 + \frac{2l_u}{1 - 2\kappa_v^2} T H_T \\
+ \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \theta_t - \theta_{t-1} \right\|^2 + \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \eta_t - \eta_{t-1} \right\|^2 . \tag{A.13}
\]
Finally, we apply (A.4) to the first sum and (A.9) to bound the cost on the control inputs. Afterwards, inserting (A.4) and (A.12) yields
\[
\sum_{t=1}^{k+\mu-2} \left\| \theta_t - \theta_{t-1} \right\|^2 \\
\leq \sum_{t=1}^{k} \left\| \theta_{t+\mu-1} - \theta_{t+\mu-1} \right\|^2 + \frac{2l_u}{1 - 2\kappa_v^2} T H_T \\
+ \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \theta_t - \theta_{t-1} \right\|^2 + \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \eta_t - \eta_{t-1} \right\|^2 + \mu \gamma_2^2 l_x \sum_{t=1}^{T-\mu+1} \left\| \eta_t - \eta_{t-1} \right\|^2 . \tag{A.14}
\]
Combining the last two results yields the desired bound
\[
\sum_{t=1}^{k} \left\| x_{t+\mu-1} - x_{t+\mu-1} \right\|^2 \\
\leq \frac{1}{2} \sum_{t=1}^{k+\mu-2} \left\| \dot{V}_t - V^\mu \right\|^2 , \tag{A.11}
\]
Before we combine all results to obtain a bound on the regret, we apply (A.4) to obtain