Centralized and distributed online learning for sparse time-varying optimization

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

Centralized and distributed online learning for sparse time-varying optimization

Sophie M. Fosson, *Member, IEEE*

***Abstract*—The development of online algorithms to track time-varying systems has drawn a lot of attention in the last years, in particular in the framework of online convex optimization. Meanwhile, sparse time-varying optimization has emerged as a powerful tool to deal with widespread applications, ranging from dynamic compressed sensing to parsimonious system identification. In most of the literature on sparse time-varying problems, some prior information on the system's evolution is assumed to be available. In contrast, in this paper, we propose an online learning approach, which does not employ a given model and is suitable for adversarial frameworks. Specifically, we develop centralized and distributed algorithms, and we theoretically analyze them in terms of dynamic regret, in an online learning perspective. Further, we propose numerical experiments that illustrate their practical effectiveness.**

***Index Terms*—Time-varying systems; sparse optimization; online learning; dynamic regret; Douglas-Rachford splitting; distributed iterative soft thresholding.**

I. Introduction

Time-varying optimization has attracted an increasing attention in the last years in machine learning, control, and signal processing, motivated by the observation that usually real-world systems vary with time. Examples are widespread, including big data streams [1], model predictive control [2], resource allocation [3], online learning [4], dynamic identification [5], and tracking moving agents [6]. Other applications are illustrated in [7], and in the recent survey [8].

Formally, by time-varying optimization, we mean a sequence of optimization problems of the kind $\min f_t$, where $t = 0, \dots, T$ is the time variable. If the problem can be solved off-line, i.e., after time T, then it can be considered as static. Usually, this is not the case: the goal is to track the optimal points as long as the optimization problem varies. This calls for online algorithms, that provide solutions in the system's time-scale, which might be very fast. Moreover, the minimization of f_t might involve the processing of large data; therefore, the development of prompt tracking strategies is challenging.

The literature on online algorithms for time-varying systems is mainly settled in convex optimization, which encompasses a number of applications and is mathematically tractable. To mention some examples, convex functionals are used to model problems of online system identification in [9], [10], tracking moving targets in [11], and dynamic magnetic resonance imaging in [12]. Most of the theoretical analyses on time-varying convex optimization are oriented towards evaluating the tracking error at time t. If $x_t^\star$ is the minimizer of f_t and x_t is the estimate provided by the online algorithm, the tracking error can be defined as a distance between $x_t^\star$ and x_t, see, e.g., [11], [13], [14], [15]. As the system is time-varying, the tracking error is not expected to converge to zero in time: an algorithm is considered successful if it guarantees a bounded tracking error, with a sufficiently small bound.

A different research line addressed, e.g., in [1], [4], [3], [16], [5], proposes the analysis of the *dynamic regret*, a popular performance metric in *online learning*, see [17]. Online learning is the sub-field of machine learning that aims at iteratively learning time-varying models, by assuming a game theoretic or adversarial framework. Specifically, a sequence of rounds is considered: at each iteration t, a learner plays an action x_t; then, an adversary chooses and reveals a loss function f_t. Thus, the learner suffers a loss $f_t(x_t)$. In turn, given the knowledge of f_t, the learner plays a new action x_{t+1}, and suffers a new loss $f_{t+1}(x_{t+1})$, and so on. The dynamic regret is the cumulative difference between the learner's loss $f_t(x_t)$ and $f_t(x_t^\star)$, the last one being the best possible loss in hindsight.

Differently from the tracking error metric, the dynamic regret metric evaluates the performance on f_t. This is particularly useful when one aims to control the cost functional. For example, in system identification (see Section VII.A), f_t describes the features (e.g., consistency with data and sparsity) of the model that one aims to identify, therefore it is more relevant to track the value of f_t than the error on $x_t^\star$. Moreover, the cumulative nature of the dynamic regret well captures the long-time behavior of systems in which violations of instantaneous error bounds are admitted. Recently, the use of dynamic regret has gained an increasing attention in a number of applications, e.g., online learning [18], [19], distributed optimization [20], multi-robot coordination [21], video reconstruction and traffic surveillance [1], network resource allocation [3], and maneuvering target tracking [16].

The online algorithms proposed in the literature for time-varying convex optimization are usually based on iterative procedures. Both centralized and distributed approaches are developed. Concerning the centralized methods, in [1], a dynamic mirror descent is proposed; in [4], a gradient descent strategy is analyzed; in [3], a modified online saddle-point scheme is studied; in [22], [23], prediction-correction methods are developed for constrained problems. At the same time, several distributed schemes are proposed: in [11], decentralization is based on the alternating direction method of multipliers (ADMM); in [24], prediction-correction methods are extended to networked systems; in [15], distributed gradient-based methods are proposed for quadratic problems; in [14], the focus is on continuous-time models; in [25], two decentralized variants of Nesterov primal-dual algorithm are developed; in [16], the mirror descent strategy is decentralized. In [20], a distributed approach based on auxiliary optimization is

S. M. Fosson is with the Dipartimento di Automatica e Informatica (DAUIN), Politecnico di Torino, Torino, Italy (e-mail: sophie.fosson@polito.it)

provided.

Within time-varying convex optimization, an important subset is represented by *sparsity* promoting problems, that is, problems whose solution is induced to be a vector with many zero entries. Sparsity is nowadays widely studied as it makes it possible to build parsimonious models from large data. In system identification, machine learning, the call for parsimonious models is rapidly increasing to deal with the increasing complexity of systems or with the need of running in small devices, like smartphones. In signal processing, sparse convex optimization has gained a lot of attention with the advent of compressed sensing (CS, [26]), which states that sparse signals can be recovered from few linear measurements. In system identification, the CS paradigm is exploited in the estimation of sparse ARX models from a limited number of observations, see [5], [27], [28].

The literature on sparse time-varying optimization (STVO) is quite recent and mainly focused on dynamic CS. Most of the works on the topic assume some prior information on the system's evolution. In [29], [30], [31], [32], the aim is to track time-varying sparse signals which evolve according to Markov models; a Kalman filtering approach is exploited. In [33], a finite bound for the tracking error is assessed, under boundedness assumptions on the the signal and its derivative. We refer the reader to [12] for a complete review.

The goal of this paper is to develop novel strategies and theoretical results for STVO, in terms of dynamic regret, without prior information on the dynamics. The lack of prior information can be interpreted as an adversarial framework, where the functional is modified arbitrarily. However, it is intuitive that a completely disordered evolution cannot be tracked: the online estimation performance is expected to improve in case of slowly varying systems.

In the literature on time-varying convex optimization, most of the theoretical results exploit the assumption that the f_t's are differentiable, see, e.g., [1], [4], [24], [23], [16], [20]. This can not be applied to sparse problems, which usually envisage an ℓ_1 regularizer. The non-differentiable case is considered in [5], [21], where the dynamic regret is analyzed for proximal gradient methods applied on composite cost functionals.

In particular, in [5] an online algorithm based on iterative soft thresholding (IST) is proposed for STVO from compressed measurements and analyzed in terms of dynamic regret. IST corresponds to the proximal gradient method or to forward-backward splitting applied to ℓ-regularized composite functionals [34], [35]. This paper extends [5] in two main directions. First, we propose and analyze a different centralized online algorithm, based on the Douglas-Rachford splitting [36], [37], [38]. Second, we develop and analyze an online distributed algorithm, which extends the distributed iterative soft thresholding algorithm (DISTA) proposed in [39] to the time-varying setting. For both centralized and distributed algorithms, we study the dynamic regret and we present numerical simulations to illustrate their practical effectiveness.

The paper is organized as follows. In Section II, we introduce our specific formulation of STVO. In Section III, we illustrate the online strategy based on splitting, and in Section IV, we theoretically analyze it terms of dynamic regret. In Section V, we propose the distributed online strategy, which is analyzed in Section VI. Section VII is devoted to numerical simulations. Finally, we draw some conclusions.

II. Problem statement

In this paper, we consider STVO problems that can be modeled via composite functionals, given by the sum of a time-varying cost functional and a sparsity-promoting regularizer. Specifically, we consider the following model:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f_t(x), \quad t=0,\dots,T \\ &f_t(x) := h_t(x) + \lambda\|x\|_1 \end{aligned} \tag{1}$$

where $\lambda > 0$ and h_t's are quadratic and strongly convex:

$$h_t(x) := \frac{1}{2}x^T Q_t x + \phi_t^T x, \tag{2}$$

$Q_t \in \mathbb{R}^{n,n}$ being symmetric positive definite, and $\phi_t \in \mathbb{R}^n$.

Strong convexity is often exploited for time-varying optimization, because it implies contractivity, hence stronger convergence properties, as studied in [4], [11], [1], [25], [24], [23], [16]. In line with these works, we assume strong convexity. We mention that in [40], a possible alternative to strong convexity is illustrated, based on bounded α-averaged operators, which might be investigated in future work.

Problem (1)-(2) is the basis for a large class of STVO problems; we illustrate some examples.

1) Elastic-net: Let us consider a time-varying CS problem: given $t = 0,\dots,T$, we aim at the online recovery of a sparse signal $\widetilde{x}_t \in \mathbb{R}^n$ (i.e., $\widetilde{x}_t$ has $k_t \ll n$ non-zero components) from compressed, linear measurements. More precisely, at each t, we observe

$$y_t = A_t\widetilde{x}_t + e_t, \quad A_t \in \mathbb{R}^{m,n},\ m < n, \tag{3}$$

where $e_t \in \mathbb{R}^m$ is a possible measurement noise. The goal is to recover $\widetilde{x}_t$ given y_t and A_t, knowing that $\widetilde{x}_t$ is sparse. As illustrated in [5], this model can be applied for compressed system identification of linear systems with time-varying parameters. We specify that the exact knowledge of k_t is not required. According to CS theory, an efficient way to tackle this problem is the convex relaxation called Lasso, which consists in the minimization of $\frac{1}{2}\|y_t - A_t x\|_2^2 + \lambda\|x\|_1$, where $\lambda > 0$; see [41], [35] for details. The presence of the ℓ_1-norm regularizer supports sparsity. If the number of measurements m is sufficiently large, $\widetilde{x}_t$ can be recovered via Lasso, with a bias proportional to the design parameter λ. We refer the reader to [42] for a complete overview on this topic.

As $m < n$, the least-squares term in Lasso is not strongly convex. A variation of Lasso, known as Elastic-net [43], enjoys this property by the addition of a Tikhonov ℓ_2-norm regularizer:

$$\begin{aligned} &f_t(x) := \frac{1}{2}\|y_t - A_t x\|_2^2 + \frac{\mu}{2}\|x\|_2^2 + \lambda\|x\|_1 \\ &\lambda > 0, \mu > 0 \end{aligned} \tag{4}$$

As a difference from Lasso, Elastic-net promotes a grouping effect of correlated variables instead of selecting just one of them and discarding the others. Moreover, the solution of Lasso necessarily has no more than m non-zero values, see

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

[44], while this limitation is not present in Elastic-net. The effectiveness of Elastic-net is exploited in many applications, ranging from micro-array classification [43] to indoor localization [45]. We remark that λ and μ might be time-varying as well. In particular, λ is generally designed by using prior knowledge on the sparsity level k_t: the higher the sparsity is, the higher the weight of the ℓ_1 should be, see, e.g., [42]. In this paper, we assume for simplicity that these parameters are constant.

2) MPC with sparse control: Quadratic, strongly convex models are usually exploited in MPC to predict a dynamic system behavior and optimize its control. Recently, the problem of reducing the number of active control inputs in MPC has been gaining an increasing interest in the literature, see, e.g., [46], [47]. This is known as sparse control, and can be tackled by introducing an ℓ_1 regularizer. The final aim of sparse control is to reduce consumption and transmission costs.

3) Sparse iterative learning control: The problem of reducing the number of control inputs is investigated in iterative learning control (ILC) as well. The purpose of ILC is the online optimization of repeated systems, with outstanding application in robotics and mechatronics. As in MPC, quadratic cost functionals are widely exploited in ILC [48]. In [49], the use of ℓ_1 regularizers is proven to be effective to obtain a reliable sparse control.

A. Performance metric: dynamic regret

Our ultimate goal is to solve Problem (1)-(2). More precisely, we are interested in computing the minimizer $x_t^\star = \underset{x\in\mathbb{R}^n}{\operatorname{argmin}} f_t(x)$, which represents to variable to track. In principle, the problem can be solved at each t through any convex optimization method, as the f_t's are convex. However, we aim to solve the problem online, that is, $x_t^\star$ should be estimated between instant t, when y_t and A_t are revealed, and instant $t+1$, when the next data acquisition is performed. Therefore, running a convex optimization algorithm might be not feasible if the time-scale is fast and the dimension n is large. We thus aim at developing fast, suboptimal strategies to track the minima with satisfactory accuracy. The first step to pursue this goal is to choose a suitable performance metric.

In game theory and online learning, a popular performance metric is the dynamic regret, denoted by $\mathbf{Reg}_T^d$, which is defined as follows (see, e.g., [50], [4]):

$$\mathbf{Reg}_T^d(x_1^\star,\dots,x_T^\star) := \sum_{t=1}^{T}\left(f_t(x_t) - f_t(x_t^\star)\right)$$

where

$$x_t^\star := \underset{x\in\mathcal{X}}{\operatorname{argmin}}\, f_t(x) \tag{5}$$

and x_t is the action played by the online algorithm in $[t-1,t)$, thus before that f_t is revealed. $\mathcal{X}\subseteq\mathbb{R}^n$ is the feasible space. Intuitively, an online algorithm is successful if its $\mathbf{Reg}_T^d$ is sublinear, because this implies that, on average, it performs as well as the clairvoyant opponent that plays the optimal action $x_t^\star$ [51], [52]. The possibility of achieving a sublinear $\mathbf{Reg}_T^d$ depends on the evolution and on the regularity of the f_t's. A quantity that well captures the system's evolution is the *path length*, defined as the cumulative distance between reference points [4], [52]; in our setting, we can consider $\sum_{t=1}^T \left\|x_t^\star - x_{t-1}^\star\right\|_2$ as path length.

As to online algorithms, the following requirements are fundamental: (a) if, for each t, the distance $\left\|x_t^\star - x_{t-1}^\star\right\|_2$ between successive minima is bounded, then the estimation error is bounded; (b) if $\left\|x_t^\star - x_{t-1}^\star\right\|_2$ tends to zero, i.e., the system tends to converge, also the estimation error should tend to zero. Requirements (a) and (b) are well captured by the dynamic regret, as illustrated, e.g., in [50], [4].

B. Summary of previous literature on regret analysis

In this section, we briefly overview the previous results on regret analysis in time-varying optimization and online learning.

Table I: Main results on static and dynamic regret in the literature. Each row of the table represents a paper. We distinguish convex (C) and strongly convex (SC) models, and we indicate if algorithms are gradient-based (G) or else. Finally, we specify the main assumptions: $\beta > 0$ is a suitable fixed value; $\eta_t \to 0$ denotes the need for vanishing learning parameters, which is undesired for large or infinite time horizons; $\Delta_t := \left\|x_t^\star - x_{t-1}^\star\right\|_2$.

	f_t	Alg.	$\mathbf{Reg}_T^s$	$\mathbf{Reg}_T^d$	Assumptions
[50]	C	G	$O(\sqrt{T})$	$O\left(\sqrt{T}(1+\sum\Delta_t)\right)$	$\|\nabla f_t\|\leq\beta$, $\eta_t\to 0$
[51]	SC	Newton	$O(\log T)$		$\|\nabla f_t\|\leq\beta$, $\eta_t\to 0$
[53]	C; SC	COMID	$O(\sqrt{T}); O(\log T)$		$\|\nabla h_t\|\leq\beta$
[54]	C; SC	ADMM	$O(\sqrt{T}); O(\log T)$		$\|\nabla h_t\|\leq\beta$
[55]	C; SC	ADMM	$O(\sqrt{T}); O(\log T)$		$\|\nabla h_t\|\leq\beta$, $\eta_t\to 0$
[56]	C	ADMM	$O(\sqrt{T})$		$\|\nabla h_t\|\leq\beta$, $\eta_t\to 0$
[4]	SC	G		$O(1+\sum\Delta_t)$	$\|\nabla f_t\|\leq\beta$
[5]	El.-net (SC)	O-IST		$O\left(1+\sum\Delta_t+\sum\Delta_t^2\right)$	
[57]	C	ADMM	$O(\sqrt{T})$		

Problems of the kind $\min_{x\in\mathbb{R}^n}\sum_t f_t(x)$ with composite functionals $f_t(x) = h_t(x) + g(x)$ are widely considered in the literature. These problems are intrinsically static: new indirect data are acquired at each time t to estimate a static optimization variable. This is usually analyzed in terms of *static* regret, which is defined as $\mathbf{Reg}_T^s := \min_x \sum_{t=1}^T \left(f_t(x_t) - f_t(x)\right)$.

In Table I, we summarize the main results (in chronological order) on regret analysis in the literature. Even though our interest is in the dynamic regret, we also report results on static regret for completeness.

The main algorithms proposed for the static problem are COMID [53], based on mirror descent, and different variants of online ADMM in [54], [55], [56]; in particular, in [56] a distributed setting is considered. As shown in Table I, all these methods achieve a static regret of order $O(\sqrt{T})$ for convex cost functionals, and in some cases an improvement to $O(\log T)$ is obtained in case of strong convexity. As illustrated in the table, most of these methods are driven by decreasing sequences of parameters, which are generally exploited to improve the convergence properties. However, this tool can not be used for tracking problems, where T is possibly infinite.

Less work is devoted to the dynamic regret analysis in tracking problems. The main result is provided by [4], which analyzes the dynamic regret of a gradient descent method for strongly convex functionals. In [5], an online iterative

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

soft thresholding (O-IST) method obtains similar performance limited to the Elastic-net model, while not requiring a bounded subgradient.

C. O-IST for quadratic problems

Before presenting the main algorithms, we retrieve O-IST proposed in [5] for Problem (4), and we generalize it to Problem (1)-(2). This paragraph provides the background to understand the distributed algorithm presented in Section V.

O-IST consists in performing a soft thresholding iteration at each t. This is a successful strategy in the sense that $\mathbf{Reg}_T^d$ (see Table I) is sublinear whenever the path length is sublinear. In particular, this implies that (a) x_t converges to $x_t^\star$ when $\left\|x_t^\star - x_{t-1}^\star\right\|_2$ is null or decreasing as, e.g., $\frac{1}{t^\beta}$, $\beta \in (0,1]$, and (b) we have a steady state tracking error for $\left\|x_t^\star - x_{t-1}^\star\right\|_2 > c$, for some $c > 0$.

In Algorithm 1, we adapt O-IST [5] to Problem (1)-(2). Differently from [5], we run $r \geq 1$ soft thresholding iterations at each t, where r depends on the time available between t and $t+1$. In Algorithm 1, the operator $\mathbb{S}_\beta : \mathbb{R}^n \to \mathbb{R}^n$, $\beta > 0$, is the component-wise soft thresholding operator, defined as follows: for $z \in \mathbb{R}$, $\mathbb{S}_\beta[z] = z - \beta$ if $z > \beta$; $\mathbb{S}_\beta[z] = z + \beta$ if $z < -\beta$; $\mathbb{S}_\beta[z] = 0$ otherwise; see, e.g., [35] for details. Moreover, the dynamic regret analysis for O-IST for Problem (1) can be straightforward derived from the results in [5].

Step 4 of Algorithm 1 can be derived as follows. Given a generic problem $\frac{1}{2}x^T Q x + \phi^T x + \lambda\|x\|_1$, a direct minimization over x is not possible, due to the presence of both the ℓ_1 term and the term $x^T Q x$ which couples the variables. In order to decouple the variables, a surrogate term $\frac{1}{2}(x-b)^T\left[\frac{1}{\tau}I - Q\right](x-b)$ can be added, where $b \in \mathbb{R}^n$ is an auxiliary variable and τ is designed such that $\frac{1}{\tau}I - Q$ is positive definite. In this way, the surrogate term is always non negative, and the global minimum is the same by adding it. An alternated minimization is then performed: with respect to b, the minimum is obtained for $b = x$; with respect to x, the problem is decoupled and can be solved by soft thresholding [35], [5]. O-IST is successful in terms of dynamic regret; however, in practice IST methods are observed to be not very fast, which, in the online version, reduce the promptness to sudden changes [5]. For this motivation, in this paper we develop a faster online strategy based on Douglas-Rachford splitting, whose convergence properties in the static framework can be leveraged to obtain good tracking properties in the dynamic framework.

Algorithm 1 O-IST for Problem (1)-(2)

input: $\lambda > 0$, $\tau > 0$, $x_0 = 0$; at time $t = 0, \dots, T$, Q_t and ϕ_t;
output: in $[t, t+1)$, an estimate x_{t+1} of $x_t^\star$;
1: **for** $t = 0, \dots, T$ **do**
2: $\quad \mathring{x}_0 = x_t$; Q_t and ϕ_t are revealed;
3: $\quad$ **for** $h = 1, \dots, r$ **do**
4: $\quad\quad \mathring{x}_h = \mathbb{S}_\lambda\left[\mathring{x}_{h-1} - \tau Q_t \mathring{x}_{h-1} - \tau\phi_t\right]$
5: $\quad$ **end for**
6: $\quad x_{t+1} = \mathring{x}_r$
7: **end for**

We remark that accelerated versions of IST might be investigated as well to speed up O-IST, based, e.g., on Nesterov accelerations [58] or FISTA [59]. However, these methods are driven by time-varying, convergent parameters, which makes their application more difficult in a tracking context, where the time horizon is possibly infinite. For this motivation, we focus on splitting methods.

Concerning the distributed setting, in static sparse recovery, a decentralization of IST is proposed in [39]. By leveraging [39], in the second part of this work, we develop a distributed online version of IST and we analyze its dynamic regret.

We specify that an online distributed splitting methods could be conceived as well, as distributed/parallel splitting algorithms are widely applied in sparse optimization, see, e.g., [60], [61]. However, a rigorous dynamic regret analysis of an online distributed splitting is rather technical, thus left for future work.

III. O-DR: Online Douglas-Rachford splitting

In this section, we present an online splitting algorithm to tackle Problem (1), based on the Douglas-Rachford (DR) method [36], [38]. First, we briefly review the classical batch DR algorithm in a static framework.

DR is an iterative algorithm that tackles the minimization of cost functionals of the kind $f(x) = h(x) + g(x)$, $x \in \mathbb{R}^n$. The procedure can be formulated as follows. Given the proximal operator, defined by

$$\operatorname{prox}_{\gamma h}(z) := \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \left[\gamma h(x) + \frac{1}{2}\|x - z\|_2^2\right], \qquad (6)$$

where $\gamma > 0$, for each $t = 0, \dots, T_{stop}$,

$$\begin{aligned} u_t &= \operatorname{prox}_{\gamma g}(2x_t - z_t) \\ z_{t+1} &= z_t + 2\alpha(u_t - x_t) \\ x_{t+1} &= \operatorname{prox}_{\gamma h}(z_{t+1}) \end{aligned} \qquad (7)$$

where $\alpha > 0$, and T_{stop} is the instant where some stop criterion is met. The procedure can be equivalently written as

$$\begin{aligned} z_{t+1} &= \mathcal{R}(z_t) \\ x_{t+1} &= \operatorname{prox}_{\gamma h}(z_{t+1}) \end{aligned} \qquad (8)$$

where $\mathcal{R} := (1-\alpha)I + \alpha(2\operatorname{prox}_{\gamma h} - I)(2\operatorname{prox}_{\gamma g} - I)$, I being the identity operator. The sequence x_t is proven to converge to $x^\star = \underset{h}{\operatorname{argmin}}\,(x) + g(x)$ (while z_t converges to a fixed point of $\mathcal{R}$: $z^\star = \mathcal{R}z^\star$) when h and g are proper closed and convex, and $\alpha \in (0,1)$, see [38] and references therein for more details. The case $\alpha = 1$, also known as Peaceman-Rachford splitting method, converges faster than the case $\alpha \in (0,1)$, under strong convexity assumptions on h, see [38, Section III].

We remark that DR is equivalent to ADMM [37] for convex problems $h(x) + g(x)$. More precisely, ADMM tackles more general problems of kind $h(x) + g(z)$ subject to $Ax + Bz = c$, and actually is the dual version of DR. The equivalence between DR and ADMM is widely studied in the literature, see, e.g., [62] and references therein. In this paper, we leverage the DR formulation; however, the ADMM formulation is possible as well, and yields the same theoretical and numerical results.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

To apply DR to Problem (1) (for the moment, in the static case $Q_t = Q$, $\phi_t = \phi$), we set $h(x) = x^T Q x + \phi^T x$, and $g(x) = \lambda \|x\|_1$. The steps in (7) are made explicit in Algorithm 2. To unburden the notation, we set $\gamma = 1$, and we consider the Peaceman-Rachford version $\alpha = 1$ [38]. These values are observed to be suitable for the proposed setting; an optimal tuning is beyond the scope of the paper and left for future analysis.

Algorithm 2 Batch DR for Problem (1)-(2)

input: $\lambda > 0$, $\mu > 0$, $Q \in \mathbb{R}^{n,n}$, $\phi \in \mathbb{R}^n$, $z_0 = 0$, $x_0 = 0$
output: at time T_{stop}, an estimate $x_{T_{stop}}$ of $x^\star$
for $t = 0, \dots, T_{stop}$ **do**
  $u_t = \mathbb{S}_\lambda \left[2x_t - z_t\right]$
  $z_{t+1} = z_t + 2(u_t - x_t)$
  $x_{t+1} = [Q + I]^{-1} \left[z_{t+1} - \phi\right]$
end for

Afterwards, following the rationale of O-IST, we propose O-DR, that performs r DR steps at each t. This is summarized in Algorithm 3.

Algorithm 3 O-DR for Problem (1)-(2)

input: $\lambda > 0$, $\mu > 0$, $z_0 = 0$, $x_0 = 0$; at $t = 0, \dots, T$: $Q_t \in \mathbb{R}^{n,n}$, $\phi_t \in \mathbb{R}^n$
output: in $[t, t+1)$, an estimate x_{t+1} of $x_t^\star$
for $t = 0, \dots, T$ **do**
  $\mathring{z}_0 = z_t$, $\mathring{x}_0 = x_t$, Q_t and ϕ_t are revealed
  for $h = 1, \dots, r$ **do**
    $\mathring{u}_h = \mathbb{S}_\lambda \left[2\mathring{x}_h - \mathring{z}_h\right]$
    $\mathring{z}_{h+1} = \mathring{z}_h + 2(\mathring{u}_h - \mathring{x}_h)$
    $\mathring{x}_{h+1} = [Q_t + I]^{-1} \left[\mathring{z}_{h+1} - \phi_t\right]$
  end for
  $z_{t+1} = \mathring{z}_{r+1}$
  $x_{t+1} = \mathring{x}_{r+1}$
end for

A. Related literature

The research on online splitting methods is very active in these years. In particular, the idea of performing one ADMM/DR iteration at each time step to tackle dynamic problems is known in the literature. However, previous work is mainly focused on the online estimation of a static quantity, which yields to online ADMM/DR procedures different from O-DR (Algorithm 3). Specifically, in [54], an online ADMM, called OADM, is proposed to tackle static problems of kind $\min_x \sum_{t=1}^{T} f_t(x) + g(x)$. The idea is to update the estimation of the global minimum at each t, when new measurements are acquired, i.e., a new f_t is revealed. Then, OADM is conceived to tackle the online estimation of a static quantity, which is intrinsically different from the tracking problem proposed in this paper. Similarly to O-DR, OADM perform one ADMM iteration at each time step. Differently from O-DR, in OADM a Bregman divergence term is added to obtain good static regret properties. The analysis in [54] is specific for static regret; in particular, some step size parameters of the algorithms are required to increase or decrease as T, which can not be applied in a tracking context, where T may be infinite. In [55], the same static problem is tackled with slightly different online ADMM procedures, based the addition of proximal operators or ℓ_2 regularization terms. Similarly to [54], the so-obtained online ADMM procedures work for decreasing/increasing time step parameters, which requires a finite T, and prevents their application to tracking problems. The tracking capabilities of ADMM/DR have been investigated more recently, with particular attention to specific practical problems. In [63], ADMM is used for real-time optimization of power systems; in [64], the tracking capabilities of ADMM are tested in a dynamic beamforming problem. The algorithms proposed in these works are based on the idea of performing one ADMM iteration at each time step, and bounds for their limit errors are studied. In [65], ADMM is used to track the solution of a stochastic sequence of problems, parametrized by a discrete time Markov process. Finally, in [66], a dynamic ADMM procedure that performs one ADMM iteration at each time step is analyzed for the dynamic sharing problem: under technical assumptions (in particular, the time-varying cost functional is sum of strongly convex functions), the convergence to a neighborhood of the optimal time-varying point is proven. Moreover, a numerical experiment on dynamic Lasso is illustrated, even though Lasso does not enjoy the above mentioned technical conditions.

IV. Dynamic regret analysis for O-DR

In this section, we show how $\mathbf{Reg}_T^d$ for O-DR depends on the system's evolution, in terms of path length. This is achieved through some intermediate results. For this purpose, we assume that the evolution of the system is bounded.

Assumption 1. *The sequence of minimizers $x_t^\star$ of f_t in Problem* (1)-(2) *is bounded, i.e., there exist $M_Q > 0$ and $M_\phi > 0$ such that $\max_t \|Q_t\|_2 \leq M_Q$ and $\max_t \|\phi_t\|_2 \leq M_\phi$, which implies $\max_t \|x_t^\star\|_2 \leq M^\star$ for some $M^\star > 0$.*

We remark that this is an assumption on the system's evolution, while we do not force any boundedness on the algorithm's evolution x_t or $f_t(x_t)$. This is marks a difference with respect to [4], where the boundedness of ∇f_t is required [4, Assumption 3], which excludes, for example, quadratic cost functionals over non-compact state spaces.

In [38], a novel convergence rate analysis is proposed for DR when the cost functional $h(x) + g(x)$ enjoys the additional properties that $h(x)$ is σ-strongly convex, i.e., $h - \frac{\sigma}{2}\|\cdot\|_2^2$ is convex, and β-smooth, i.e., $\frac{\beta}{2}\|\cdot\|_2^2 - h$ is convex, see [38, Section II] for details. From that result, we derive the following corollary on the contractivity of DR for Problem (1)-(2).

Corollary 1. *Let us consider the static Problem* (1)-(2)*, that is, $f_t = f$ for each t. Let $(x_t, z_t)_{t=0,\dots T_{stop}}$ be the sequence generated by batch DR (Algorithm 2) and let $(x^\star, z^\star)$ be its limit point. Then, z_t converges Q-linearly to $z^\star$; more precisely, for each $t = 0, \dots, T_{stop}$*

$$\|z_{t+1} - z^\star\|_2 \leq \delta \, \|z_t - z^\star\|_2 \tag{9}$$

where

$$\delta = \max\left(\frac{1-\sigma}{1+\sigma}, \frac{\beta - 1}{\beta + 1}\right) < 1 \tag{10}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

where σ and β respectively are the minimum non-null and the maximum eigenvalues of Q. Moreover,

$$\left\|x_{t+1} - x^\star\right\|_2 \le q \left\|z_t - z^\star\right\|_2 \tag{11}$$

where $q = \frac{\delta}{1+\sigma}$.

Proof. The proof is based on [38, Theorem 2, Corollary 1]. First, we notice that [38, Assumption 2] is satisfied by Problem (1)-(2): $h(x)$ is β-smooth and σ-strongly convex with parameters as defined after (10). Therefore, [38, Theorem 2] holds, which states that z_t in Algorithm 2 converges Q-linearly to $z^\star$, with contraction parameter δ as defined in (10).

Since $\mathrm{prox}_h(z) = (Q+I)^{-1}(z-\phi)$, then prox_h is $\frac{1}{1+\sigma}$-Lispchitz continuous. Hence, we prove (11) as follows:

$$\begin{aligned}\left\|x_{t+1} - x^\star\right\|_2 &= \left\|\mathrm{prox}_h(z_{t+1}) - \mathrm{prox}_h(z^\star)\right\|_2 \\ &\le \frac{1}{1+\sigma}\left\|z_{t+1} - z^\star\right\|_2 \le \frac{\delta}{1+\sigma}\left\|z_t - z^\star\right\|_2.\end{aligned}$$

□

Now, we exploit Corollary 1 in the dynamic case. For each f_t, let $(x_t^\star, z_t^\star)$ be the limit point of DR, where $x_t^\star$ is the minimum of f_t, and $z_t^\star = \phi_t + (Q_t+I)x_t^\star$, given that $(x_t^\star, z_t^\star)$ is a fixed point for the map defined by one iteration in Algorithm 2. If r iterations are played at time t, we derive the following result from Corollary 1.

Corollary 2. *For O-DR (Algorithm 3), the following properties hold:*

$$\left\|z_{t+1} - z_t^\star\right\|_2 \le \delta^r \left\|z_t - z_t^\star\right\|_2 \tag{12}$$

and

$$\left\|x_{t+1} - x_t^\star\right\|_2 \le q^r \left\|z_t - z_t^\star\right\|_2. \tag{13}$$

By using Corollary 2, we prove that, at t, the distance between the played action and the current minimum is controlled by the distance between successive minima. In the following, we name:

$$\begin{aligned}&\Delta_{z_t} := \|z_t - z_t^\star\|_2, \\ &\Delta_{x_t}^\star := \|x_t^\star - x_{t-1}^\star\|_2, \quad \Delta_{z_t}^\star := \|z_t^\star - z_{t-1}^\star\|_2.\end{aligned} \tag{14}$$

From Assumption 1, $z_t^\star = \phi_t + (Q_t + I)x_t^\star$ is bounded; then, there exists $M_{z\star} > 0$ such that $\Delta_{z_t}^\star \le M_{z\star}$. This is used to prove the following result.

Lemma 1. *For O-DR,*

$$\begin{aligned}(a)\quad &\sum_{t=1}^T \Delta_{z_t} \le c_1 + c_2 \sum_{t=1}^T \Delta_{z_t}^\star \\ (b)\quad &\sum_{t=1}^T \Delta_{z_t}^2 \le c_3 + c_4 \sum_{t=1}^T \Delta_{z_t}^\star + c_5 \sum_{t=1}^T {\Delta_{z_t}^\star}^2\end{aligned}$$

where

$$\begin{aligned}&c_1 = \frac{\delta^r}{1-\delta^r}\left(\Delta_{z_0} - \Delta_{z_T}\right), \quad c_2 = \frac{1}{1-\delta^r}, \\ &c_3 = \frac{\delta^{2r}\left(\Delta_{z_0}^2 - \Delta_{z_T}^2\right) + 4M_{z\star}\delta^{2r}\left(\Delta_{z_0} - \Delta_{z_T}\right) + 4M_{z\star}\delta^r c_1}{1-\delta^{2r}}, \\ &c_4 = \frac{4M_{z\star}\delta^r c_2}{1-\delta^{2r}}, \quad c_5 = \frac{1}{1-\delta^{2r}}.\end{aligned} \tag{15}$$

Proof. By the triangle inequality and Corollary 2, for each $t = 1, \ldots, T$,

$$\begin{aligned}\Delta_{z_t} &= \|z_t - z_t^\star \pm z_{t-1}^\star\|_2 \le \|z_t - z_{t-1}^\star\|_2 + \Delta_{z_t}^\star \\ &\le \delta^r \Delta_{z_{t-1}} + \Delta_{z_t}^\star.\end{aligned} \tag{16}$$

By summing over $t = 1, \ldots, T$, we prove (a):

$$(1-\delta^r)\sum_{t=1}^T \Delta_{z_t} \le \delta^r\left(\Delta_{z_0} - \Delta_{z_T}\right) + \sum_{t=1}^T \Delta_{z_t}^\star.$$

To prove (b), first we use (16) and the fact that $\Delta_{z_t}^\star \le M_{z\star}$:

$$\begin{aligned}\Delta_{z_t}^2 &\le \delta^{2r}\Delta_{z_{t-1}}^2 + \Delta_{z_t}^{\star 2} + 2\delta^r \Delta_{z_{t-1}}\Delta_{z_t}^\star \\ &\le \delta^{2r}\Delta_{z_{t-1}}^2 + \Delta_{z_t}^{\star 2} + 4M_{z\star}\delta^r\Delta_{z_{t-1}}.\end{aligned}$$

Then, we sum over over $t = 1, \ldots, T$:

$$\sum_{t=1}^T \Delta_{z_t}^2 \le \frac{\delta^{2r}\left(\Delta_{z_0}^2 - \Delta_{z_T}^2\right) + \sum_{t=1}^T\left(\Delta_{z_t}^{\star 2} + 4M_{z\star}\delta^r\Delta_{z_{t-1}}\right)}{1-\delta^{2r}}. \tag{17}$$

Since $\sum_{t=1}^T \Delta_{z_{t-1}} = \Delta_{z_0} - \Delta_{z_T} + \sum_{t=1}^T \Delta_{z_t}$, by applying (a), we have

$$\sum_{t=1}^T \Delta_{z_{t-1}} \le \Delta_{z_0} - \Delta_{z_T} + c_1 + c_2\sum_{t=1}^T \Delta_{z_t}^\star.$$

By substituting this bound in (17), the thesis is obtained, with constants as defined in (15). □

The following two lemmas highlight properties of the quadratic functional in (1).

Lemma 2. *For each t, and for any $x \in \mathbb{R}^n$,*

$$f_t(x) - f_t(x_t^\star) \le \alpha_1 \left\|x - x_t^\star\right\|_2 + \alpha_2 \left\|x - x_t^\star\right\|_2^2 \tag{18}$$

where $\alpha_1 = M_Q M^\star + M_\phi + \lambda$ and $\alpha_2 = \frac{M_Q}{2}$.

Proof. The joint descent theorem, see [67, Corollary 3], states that given a differentiable $h : \mathbb{R}^n \to \mathbb{R}$ with μ-Lipschitz gradient, and $g : \mathbb{R}^n \to \mathbb{R}$, for any $x, y, z \in \mathbb{R}^n$, the following inequality holds: $h(x) + g(x) \le h(y) + g(y) + (x-y)^T(\nabla h(z) + \widetilde{\nabla} g(x)) + \frac{\mu}{2}\|z - x\|_2^2$, where $\widetilde{\nabla}$ denotes the subgradient. Then, we obtain the thesis by considering $h(x) = x^T Q_t x + \phi_t^T x$, $g(x) = \lambda\|x\|_1$, and $y = z = x_t^\star$, and by noticing that $\nabla h(x_t^\star) \le M_Q M^\star + M_\phi$, ∇h is M_Q-Lipschitz, and $\widetilde{\nabla} g \le 1$. □

Remark 1. *We notice that, even though specialized for the quadratic case, the previous results actually hold for any functional $h(x) + \lambda\|x\|_1$ with strongly convex g with Lipschitz gradient. In this paper, we focus on the quadratic case as it is the most common example of functionals with Lipschitz gradient and it is prone to decentralization, see Section V.*

Based on these results, we prove the main result.

Theorem 1. *O-DR for Problem* (1)-(2) *(Algorithm 3) has the following dynamic regret bound:*

$$\mathbf{Reg}_T^d \le \eta_0 + \sum_{t=1}^T\left(\eta_1\Delta_{z_t}^\star + \eta_2{\Delta_{z_t}^\star}^2 + \eta_3\Delta_{x_t}^\star + \eta_4{\Delta_{x_t}^\star}^2\right)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

where $\eta_i > 0$, $i = 0, \dots, 4$ are assessed in the proof.

In particular, this theorem implies that if the path lengths $\sum_{t=1}^{T} \Delta^\star_{z_t}$ and $\sum_{t=1}^{T} \Delta^\star_{x_t}$ are sublinear, then also $\mathbf{Reg}_T^d$ is sublinear, i.e., the algorithm is successful, in line with previous results shown in Table I.

Proof. From Lemma 2, by considering $x = x_t$,

$$f_t(x_t) - f_t(x_t^\star) \leq \alpha_1 \|x_t - x_t^\star\|_2 + \alpha_2 \|x_t - x_t^\star\|_2^2 . \quad (19)$$

From Corollary 2, we know that $\left\|x_t - x_{t-1}^\star\right\|_2 \leq q^r \|z_{t-1} - z_{t-1}^\star\|_2 = q^r \Delta_{z_{t-1}}$. Then, by applying the triangle inequality,

$$\|x_t - x_t^\star\|_2 \leq \left\|x_t - x_{t-1}^\star\right\|_2 + \Delta^\star_{x_t} \leq q^r \Delta_{z_{t-1}} + \Delta^\star_{x_t}. \quad (20)$$

Since $(a-b)^2 \leq 2a^2 + 2b^2$ for any $a, b \in \mathbb{R}$, we have

$$\|x_t - x_t^\star\|_2^2 \leq 2q^{2r} \Delta^2_{z_{t-1}} + 2\Delta^{\star^2}_{x_t}. \quad (21)$$

By substituting (20) and (21) in (19), we conclude:

$$\begin{aligned} f_t(x_t) - f_t(x_t^\star) &\leq \alpha_1 \left(q^r \Delta_{z_{t-1}} + \Delta^\star_{x_t}\right) + \\ &+ 2\alpha_2 \left(q^{2r} \Delta^2_{z_{t-1}} + \Delta^{\star^2}_{x_t}\right) \\ &= \zeta_1 \Delta_{z_{t-1}} + \zeta_2 \Delta^2_{z_{t-1}} + \zeta_3 \Delta^\star_{x_t} + \zeta_4 \Delta^{\star^2}_{x_t} \end{aligned}$$

where $\zeta_1 = \alpha_1 q^r$, $\zeta_2 = 2\alpha_2 q^{2r}$, $\zeta_3 = \alpha_1$, $\zeta_6 = 2\alpha_2$. Now, let us sum over $t = 1, \dots, T$:

$$\begin{aligned} \mathbf{Reg}_T^d &= \sum_{t=1}^{T} \left(f_t(x_t) - f_t(x_t^\star)\right) \\ &\leq \kappa + \sum_{t=1}^{T} \left(\zeta_1 \Delta_{z_t} + \zeta_2 \Delta^2_{z_t} + \zeta_3 \Delta^\star_{x_t} + \zeta_4 \Delta^{\star^2}_{x_t}\right) \end{aligned}$$

where $\kappa = \zeta_1(\Delta_{z_0} - \Delta_{z_T}) + \zeta_2(\Delta^2_{z_0} - \Delta^2_{z_T})$. Then, we apply Lemma 1:

$$\begin{aligned} \mathbf{Reg}_T^d &\leq \kappa + \zeta_1 \left(c_1 + c_2 \sum_{t=1}^{T} \Delta^\star_{z_t}\right) + \zeta_2 c_3 + \\ &+ \zeta_2 \left(c_4 \sum_{t=1}^{T} \Delta^\star_{z_t} + c_5 \sum_{t=1}^{T} \Delta^{\star^2}_{z_t}\right) + \zeta_3 \sum_{t=1}^{T} \Delta^\star_{x_t} + \zeta_4 \sum_{t=1}^{T} \Delta^{\star^2}_{x_t}. \end{aligned}$$

The thesis is obtained with $\eta_0 = \zeta_1 c_1 + \zeta_2 c_3 + \kappa$, $\eta_1 = \zeta_1 c_2 + \zeta_2 c_4$, $\eta_2 = \zeta_2 c_5$, $\eta_3 = \zeta_3$, $\eta_4 = \zeta_4$. □

V. O-DISTA: Distributed online IST

As mentioned in the introduction, several works in the literature are concerned with distributed methods for online convex optimization, see, e.g., [11], [14], [15], [24], [25]. For this motivation, we propose a distributed algorithm for STVO, that consists in a decentralization of the IST algorithm proposed in [5], based on the DISTA algorithm developed in[39]. Specifically, we reformulate DISTA for Problem (1)-(2), and we prove its contraction properties. Then, we analyze its dynamic regret in Section VI.

Let us consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes, whose cardinality is denoted by $|\mathcal{V}|$, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. $\mathcal{E}$ enjoys the property: $(v, w) \in \mathcal{E}$ implies $(w, v) \in \mathcal{E}$; moreover, $(v, v) \in \mathcal{E}$ for each $v \in \mathcal{V}$. We denote by d_v the degree of $v \in \mathcal{V}$, i.e., the number of edges incident to v, while d_{m} and d_{M} are the minimum and maximum degree of $\mathcal{G}$, respectively.

Let $X := (x_1, \dots, x_{|\mathcal{V}|}) \in \mathbb{R}^{n,|\mathcal{V}|}$. At each t, in the philosophy of [39, Equation (9)], we formulate the following problem:

$$\begin{aligned} &\min_{X \in \mathbb{R}^{n,|\mathcal{V}|}} F_t(X) \\ &F_t(X) := \sum_{v \in \mathcal{V}} \left[\frac{1}{2} x_v^T Q_{v,t} x_v + \phi_{v,t}^T x_v + \lambda \|x_v\|_1 + \right. \\ &\left. + \frac{1}{2\tau d_{\mathrm{M}}} \sum_{w \in \mathcal{N}_v} \|\overline{x}_w - x_v\|_2^2 \right] \end{aligned} \quad (22)$$

where $\overline{x}_w := \frac{1}{d_w} \sum_{u \in \mathcal{N}_w} x_u$, and $\tau > 0$ is a weight that will be assessed later. The term $\sum_{w \in \mathcal{N}_v} \|\overline{x}_w - x_v\|_2^2$ promotes consensus among the local estimates. Problem (22) is a relaxation of [39, Problem 8], and does not guarantee to reach a consensus; more precisely, one can guarantee consensus by suitably weighting the term $\sum_{w \in \mathcal{N}_v} \|\overline{x}_w - x_v\|_2^2$, see [39, Theorem 2] for details. In our experiments, we observe that the formulation (22) is sufficient to reach a substantial agreement; therefore, for simplicity, we do not discuss the weight tuning.

The motivation to use $\overline{x}_w$ instead of x_w (which would equivalently support consensus) is rather technical; in a nutshell, it makes easier to split each iteration in two steps: one of local communication and one of individual descent, as illustrated in algorithms 4 and 5. Finally, we remark that, differently from [39], we do not require that $\mathcal{G}$ is a regular graph.

At time t, each $v \in \mathcal{V}$ is assumed to know local data $Q_{v,t} \in \mathbb{R}^{n,n}$ and $\phi_{v,t} \in \mathbb{R}^n$. Nodes aim to track the minimizer of (22), denoted as $x_t^\star \in \mathbb{R}^n$, by leveraging local information and local communication. As for the centralized case of O-IST, see [5], the minimum of F_t cannot be obtained in closed form, while it can be achieved via alternated minimization of a surrogate functional F_t^s. $F_t^s(x)$ is obtained from $F_t(x)$ by adding a quadratic term to deal and by substituting the local mean $\overline{x}_w$ with a local auxiliary variable, which makes the problem separable in each component. More precisely, given $C = (c_1, \dots, c_{|\mathcal{V}|}) \in \mathbb{R}^{n,|\mathcal{V}|}$, $B = (b_1, \dots, b_{|\mathcal{V}|}) \in \mathbb{R}^{n,|\mathcal{V}|}$, and $R_{v,t}^\tau := \frac{1}{\tau} I - Q_{v,t}$ we define

$$\begin{aligned} F_t^s(X, C, B) := \sum_{v \in \mathcal{V}} &\left[\frac{1}{2} x_v^T Q_{v,t} x_v + \phi_{v,t}^T x_v + \right. \\ &+ \lambda \|x_v\|_1 + \frac{1}{2\tau d_{\mathrm{M}}} \sum_{w \in \mathcal{N}_v} \|c_w - x_v\|_2^2 + \\ &\left. + \frac{1}{2}(x_v - b_v)^T R_{v,t}^\tau (x_v - b_v)\right]. \end{aligned} \quad (23)$$

As discussed for O-IST, we assume that τ is designed such that $R_{\tau,v,t}$ is positive definite. Different τ's might be considered at each node; here we consider a unique value to simplify the notation.

It is straightforward to verify that $F_t(X) = F_t^s(X, \overline{X}, X)$, where $\overline{X} = (\overline{x}_1, \dots, \overline{x}_{|\mathcal{V}|})$; moreover, if $X_t^\star = \operatorname{argmin} F_t$, then $(X^\star, \overline{X^\star}, X^\star) = \operatorname{argmin} F_t^s(X, C, B)$, which is achieved by alternated minimization over $X \in \mathbb{R}^{n,|\mathcal{V}|}$, $C \in \mathbb{R}^{n,|\mathcal{V}|}$, $B \in \mathbb{R}^{n,|\mathcal{V}|}$. In particular, the minimization with respect to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

x_v can be done separately in each component, leveraging the fact that, given $a > 0$, $\operatorname{argmin}_{x\in\mathbb{R}} \frac{1}{2}ax^2 + bx + \lambda|x| = \mathbb{S}_{\lambda/a}[-b/a]$. Moreover, since $\sum_{v\in\mathcal{V}}\sum_{w\in\mathcal{N}_v}\|c_w - x_v\|_2^2 = \sum_{v\in\mathcal{V}}\sum_{w\in\mathcal{N}_v}\|c_v - x_w\|_2^2$ for undirected graphs, the minimum with respect to c_v is achieved at $c_v = \overline{x}_v$. Finally, the minimum with respect to b_v is $b_v = x_v$ provided that $R_{\tau,v,t}$ is positive definite. The so-obtained iterative procedure is reported in Algorithm 4. As in the centralized case, first, we illustrate the static case $F_t = F$; then, we provide the online formulation, denoted as O-DISTA, in Algorithm 5.

Algorithm 4 Batch DISTA for Problem (22)

input: $\lambda > 0$, $\tau > 0$; for each $v \in \mathcal{V}$, $x_{v,0} = 0$, Q_v, ϕ_v, $R_v^\tau = \frac{1}{\tau}I - Q_v$
output: at time T_{stop}, for each $v \in \mathcal{V}$, an estimate $x_{v,T_{stop}}$ of $x^\star$

for $t = 0, \dots, T_{stop}$ **do**
  If t is even, for any $v \in \mathcal{V}$,

$$c_{v,t+1} = \overline{x}_{v,t}$$
$$x_{v,t+1} = x_{v,t}$$

  If t is odd, for any $v \in \mathcal{V}$,

$$c_{v,t+1} = c_{v,t}$$
$$x_{v,t+1} = \mathbb{S}_{\frac{\lambda\tau}{1+d_v/d_{\mathrm{M}}}}\left[\frac{\tau R_v^\tau x_{v,t} + \frac{d_v}{d_{\mathrm{M}}}\overline{c}_{v,t} - \tau\phi_v}{1 + d_v/d_{\mathrm{M}}}\right]$$

end for

Algorithm 5 O-DISTA for Problem (22)

input: $\lambda > 0$, $\tau > 0$; for each $v \in \mathcal{V}$, $x_{v,0} = 0$; at time t, $Q_{v,t}$, $\phi_{v,t}$, $R_{v,t}^\tau = \frac{1}{\tau}I - Q_{v,t}$
output: in $[t, t+1)$, for each $v \in \mathcal{V}$, an estimate $x_{v,t+1}$ of $\widetilde{x}_t$

for $t = 0, \dots, T$ **do**
  for $h = 0, \dots, r$ **do**
    For each $v \in \mathcal{V}$, $\mathring{x}_{v,0} = x_{v,t}$
    If h is even, for any $v \in \mathcal{V}$,

$$c_{v,h+1} = \overline{\mathring{x}}_{v,h}$$
$$\mathring{x}_{v,h+1} = \mathring{x}_{v,h}$$

    If h is odd, for any $v \in \mathcal{V}$,

$$c_{v,h+1} = c_{v,h}$$
$$\mathring{x}_{v,h+1} = \mathbb{S}_{\frac{\lambda\tau}{1+d_v/d_{\mathrm{M}}}}\left[\frac{\tau R_{v,t}^\tau \mathring{x}_{v,h} + \frac{d_v}{d_{\mathrm{M}}}\overline{c}_{v,h} - \tau\phi_{v,t}}{1 + d_v/d_{\mathrm{M}}}\right]$$

  end for
  For each $v \in \mathcal{V}$, $x_{v,t+1} = \mathring{x}_{v,r}$
end for

VI. Dynamic Regret Analysis for O-DISTA

In this section, we analyze the dynamic regret for O-DISTA, by extending the results in [5, Section IV] to the distributed case. We start by proving the contractivity, under the following assumption.

Assumption 2. *Let $\tau < \min_{v,t}\|Q_{v,t}\|_2^{-2}$, and $\theta_\tau =: \max_{v,t}\|\tau R_{v,t}^\tau\|_2^2 < 1$, where $R_{v,t}^\tau = \frac{1}{\tau}I - Q_{v,t}$. We assume that $\rho := 2\left(1 + \frac{d_{\mathrm{m}}}{d_{\mathrm{M}}}\right)^{-2}\left(\theta_\tau + \frac{d_{\mathrm{M}}}{d_{\mathrm{m}}}\right) < 1$.*

Roughly speaking, Assumption 2 holds if the difference between minimum and maximum degrees is not too large, which is a reasonable requirement in practice. In particular, Assumption 2 holds for any θ_τ if $d_{\mathrm{M}} = d_{\mathrm{m}}$, i.e., if $\mathcal{G}$ is regular.

Lemma 3. *If $X(t) \in \mathbb{R}^{n,|\mathcal{V}|}$ is the sequence produced by O-DISTA, and $X_t^\star \in \mathbb{R}^{n,|\mathcal{V}|}$ is the minimizer of (22), then*

$$\|X_{t+1} - X_t^\star\|_F \le \rho^{r/2}\|X_t - X_t^\star\|_F \tag{24}$$

where $\|\cdot\|_F$ denotes the Frobenious norm. Then, under Assumption 2, contractivity holds.

Proof. Let $r = 1$, and let $\Gamma_t : \mathbb{R}^{n,|\mathcal{V}|} \mapsto \mathbb{R}^{n,|\mathcal{V}|}$ be the map defined by $(\Gamma_t X)_v = \mathbb{S}_{\frac{\lambda\tau}{1+d_v/d_{\mathrm{M}}}}\left[\frac{\tau R_{v,t}^\tau x_v + \frac{d_v}{d_{\mathrm{M}}}\overline{\overline{x}}_v - \tau\phi_{v,t}}{1+d_v/d_{\mathrm{M}}}\right]$. As $X_t^\star = (x_{1,t}^\star, \dots, x_{|\mathcal{V}|,t}^\star)$ is the minimum of $F_t(X)$, then $(X_t^\star, \overline{X_t^\star}, X_t^\star)$ is the minimum of $F^s(X, C, B)$, and $X_t^\star$ is a fixed point for Γ_t. Thus, since $\mathbb{S}_\beta$ is non-expansive [35],

$$\left\|x_{v,t+1} - x_{v,t}^\star\right\|_2^2 \le$$
$$\le \frac{1}{\mathrm{g}}\left\|\tau R_{v,t}^\tau(x_{v,t} - x_{v,t}^\star) + \frac{d_v}{d_{\mathrm{M}}}(\overline{\overline{x}}_{v,t} - \overline{\overline{x^\star}}_{v,t})\right\|_2^2$$

where $\mathrm{g} = \left(1 + \frac{d_v}{d_{\mathrm{M}}}\right)^2$ and $\overline{\overline{x}}_v = \frac{1}{d_v}\sum_{w\in\mathcal{N}_v}\frac{1}{d_w}\sum_{u\in\mathcal{N}_w}x_u$. By applying the Cauchy-Schwarz inequality $\left(\sum_{i=1}^n a_i\right)^2 \le n\sum_{i=1}^n a_i^2$, we compute the following bound:

$$\|x_{v,t+1} - x_{v,t}^\star\|_2^2 \le$$
$$\le \frac{2}{\mathrm{g}}\left\|\tau R_{v,t}^\tau\right\|_2^2\left\|x_{v,t} - x_{v,t}^\star\right\|_2^2 + \frac{2d_v^2}{\mathrm{g}d_{\mathrm{M}}^2}\|\overline{\overline{x}}_{v,t} - \overline{\overline{x^\star}}_{v,t}\|_2^2$$
$$\le \frac{2\theta_\tau}{\mathrm{g}}\left\|x_{v,t} - x_{v,t}^\star\right\|_2^2 + \frac{2d_v^2}{\mathrm{g}d_{\mathrm{M}}^2}\left\|\frac{1}{d_v}\sum_{w\in\mathcal{N}_v}(\overline{x}_{w,t} - \overline{x^\star}_{w,t})\right\|_2^2$$
$$\le \frac{2\theta_\tau}{\mathrm{g}}\left\|x_{v,t} - x_{v,t}^\star\right\|_2^2 + \frac{2d_v}{\mathrm{g}d_{\mathrm{M}}^2}\sum_{w\in\mathcal{N}_v}\|\overline{x}_{w,t} - \overline{x^\star}_{w,t}\|_2^2$$
$$\le \frac{2\theta_\tau}{\mathrm{g}}\left\|x_{v,t} - x_{v,t}^\star\right\|_2^2 + \frac{2}{\mathrm{g}d_{\mathrm{M}}}\sum_{w\in\mathcal{N}_v}\frac{1}{d_w}\sum_{u\in\mathcal{N}_w}\|x_{u,t} - x_{u,t}^\star\|_2^2$$
$$\le \frac{2\theta_\tau}{\mathrm{g}}\left\|x_{v,t} - x_{v,t}^\star\right\|_2^2 + \frac{2}{\mathrm{g}d_{\mathrm{M}}d_{\mathrm{m}}}\sum_{w\in\mathcal{N}_v}\sum_{u\in\mathcal{N}_w}\|x_{u,t} - x_{u,t}^\star\|_2^2.$$

By summing over $v \in \mathcal{V}$, since $\sum_{v\in\mathcal{V}}\sum_{w\in\mathcal{N}_v}\sum_{u\in\mathcal{N}_w}\|x_{u,t} - x_{u,t}^\star\|_2^2 \le d_{\mathrm{M}}^2\sum_{v\in\mathcal{V}}\|x_{v,t} - x_{v,t}^\star\|_2^2$ and $\mathrm{g} \ge \left(1 + \frac{d_{\mathrm{m}}}{d_{\mathrm{M}}}\right)^2$, we obtain $\|X_{t+1} - X_t^\star\|_F^2 \le \rho\left\|X_t - X_t^\star\right\|_F^2$. The extension to $r > 1$ is straightforward. □

By exploiting the contractivity, the following result can be proven. Let $\Delta_t := \|X_t^\star - X_{t-1}^\star\|_F$.

Lemma 4. *For each $t = 1, \dots, T$,*

$$(a)\quad \sum_{t=1}^{T} \|X_t - X_t^\star\|_F \le c_1 + c_2 \sum_{t=1}^{T} \Delta_t$$
$$(b)\quad \sum_{t=1}^{T} \|X_t - X_t^\star\|_F^2 \le c_3 + c_4 \sum_{t=2}^{T} \Delta_t^2 + c_5 \sum_{t=1}^{T} \Delta_t$$

with constants $c_i > 0$, $i = 1, \dots, 5$.

The proof is a straightforward extension of the proof of [5, Lemma 2], and omitted for brevity.

Lemma 5. *For each $t = 1, \dots, T$,*

$$F_{t-1}(X_t) - F_{t-1}(X_{t-1}^\star) \le \frac{1}{\tau}\left(\frac{d_\mathrm{M}}{d_\mathrm{m}} + 1\right) \left\|X_{t-1} - X_{t-1}^\star\right\|_F^2 .$$

Proof. Let us consider the intermediate variables $\mathring{X}_h$, $h = 0, \dots, r$, between $t-1$ and t, starting from $\mathring{X}_0 = X_{t-1}$. Since F_{t-1} is decreased by O-DISTA, and given (23),

$$\begin{aligned} F_{t-1}(X_t) &\le F_{t-1}(\mathring{X}_1) = F_{t-1}^s(\mathring{X}_1, \overline{\mathring{X}}_1, \mathring{X}_1) \\ &\le F_{t-1}^s(\mathring{X}_1, \overline{X}_{t-1}, X_{t-1}) \le F_{t-1}^s(X_{t-1}^\star, \overline{X}_{t-1}, X_{t-1}). \end{aligned}$$

Therefore, by applying the Cauchy-Schwarz inequality,

$$\begin{aligned} &F_{t-1}(X_t) - F_{t-1}(X_{t-1}^\star) \\ &\le F_{t-1}^s(X_{t-1}^\star, \overline{X}_{t-1}, X_{t-1}) - F_{t-1}(X_{t-1}^\star) \\ &= \sum_{v\in\mathcal{V}} \left[\frac{1}{2\tau d_\mathrm{M}} \sum_{w\in\mathcal{N}_v} \left\|\overline{x}_{w,t-1} - x_{v,t-1}^\star\right\|_2^2 + \right. \\ &\left. + \frac{1}{2}(x_{v,t-1} - x_{v,t-1}^\star)^T R_{v,t-1}^\tau (x_{v,t-1} - x_{v,t-1}^\star) \right] \\ &\le \frac{1}{\tau}\left(\frac{d_\mathrm{M}}{d_\mathrm{m}} + 1\right) \sum_{v\in\mathcal{V}} \left\|x_{v,t-1} - x_{v,t-1}^\star\right\|_2^2 . \end{aligned}$$

□

For any $X \in \mathbb{R}^{n,|\mathcal{V}|}$, let us define

$$\begin{aligned} D_t(X) &:= F_t(X) - F_{t-1}(X) \\ \Delta_{Q_{v,t}} &:= \|Q_{v,t} - Q_{v,t-1}\|_2 \\ \Delta_{\phi_{v,t}} &:= \|\phi_{v,t} - \phi_{v,t-1}\|_2 . \end{aligned} \tag{25}$$

Assumption 3. *We assume that* $\sup_{v,t} \Delta_{Q_{v,t}}$ *and* $\sup_{v,t} \Delta_{\phi_{v,t}}$ *are bounded. Moreover, if $\Delta_{Q_{v,t}} \neq 0$ for at least one v, we assume that* $\sup_t \|x_t^\star\|$ *is bounded.*

We notice that this assumption is weaker than Assumption 1: $\phi_{v,t}$ does not need to be bounded; if $Q_{v,t}$ is constant in time, $x_t^\star$ does not need to be bounded. A possible application is discussed Section VII-B. Briefly, O-DISTA requires a weaker boundedness assumption than O-DR, as for O-DISTA the sequence $F_t(X_t)$ is monotone decreasing, and in particular Lemma 5 holds, which is not guaranteed for O-DR.

Lemma 6. *For any $X \in \mathbb{R}^{n,|\mathcal{V}|}$ and $t = 1, \dots, T$,*

$$D_t(X) - D_t(X_t^\star) \le \gamma_1 \|X - X_t^\star\|_F + \gamma_2 \|X - X_t^\star\|_F^2 \tag{26}$$

where $\gamma_1 = \sqrt{|\mathcal{V}|}\sup_{v,t}(\Delta_{\phi_{v,t}} + \Delta_{Q_{v,t}}\|x_t^\star\|_2)$ *and* $\gamma_2 = \frac{1}{2}\sup_{v,t} \Delta_{Q_{v,t}}$.

Proof. Since

$$D_t(X) = \sum_{v\in\mathcal{V}} \left[\frac{1}{2} x_v^T (Q_{v,t} - Q_{v,t-1}) x_v + (\phi_{v,t} - \phi_{v,t-1})^T x_v \right],$$

we have:

$$\begin{aligned} &D_t(X) - D_t(X_t^\star) = \\ &= \sum_{v\in\mathcal{V}} \left[\frac{1}{2}(x_v - x_{v,t}^\star)^T (Q_{v,t} - Q_{v,t-1})(x_v + x_{v,t}^\star) \right] \\ &+ \sum_{v\in\mathcal{V}} (\phi_{v,t} - \phi_{v,t-1})^T (x_v - x_{v,t}^\star) \\ &\le \sum_{v\in\mathcal{V}} \left\|x_v - x_{v,t}^\star\right\|_2 \left[\frac{1}{2}\Delta_{Q_{v,t}} \left\|x_v + x_{v,t}^\star\right\|_2 + \Delta_{\phi_{v,t}} \right] \\ &\le \sum_{v\in\mathcal{V}} \frac{1}{2} \left\|x_v - x_{v,t}^\star\right\|_2^2 \Delta_{Q_{v,t}} + \\ &+ \sum_{v\in\mathcal{V}} \left(\Delta_{\phi_{v,t}} + \Delta_{Q_{v,t}} \|x_{v,t}^\star\|_2 \right) \left\|x_v - x_{v,t}^\star\right\|_2 . \end{aligned}$$

Then, the thesis is obtained by applying Cauchy-Schwarz. □

Given these intermediate lemmas, we can now evaluate the dynamic regret.

Theorem 2. *The dynamic regret for O-DISTA (Algorithm 5) has the following bound:*

$$\mathbf{Reg}_T^d \le \alpha_0 + \alpha_1 \sum_{t=1}^{T} \Delta_t + \alpha_2 \sum_{t=1}^{T} \Delta_t$$

where $\alpha_i > 0$, $i = 0, 1, 2$.

Proof. We consider the difference of losses $F_t(X_t) - F_t(X_t^\star)$ and we add and subtract $F_{t-1}(X_t)$ to it. By using that $F_{t-1}(X_t^\star) \ge F_{t-1}(X_{t-1}^\star)$, we obtain the following bound:

$$\begin{aligned} &F_t(X_t) - F_t(X_t^\star) \le \\ &\le F_t(X_t) - F_t(X_t^\star) \pm F_{t-1}(X_t) + F_{t-1}(X_t^\star) - F_{t-1}(X_{t-1}^\star) \\ &= D_t(X_t) - D_t(X_t^\star) + F_{t-1}(X_t) - F_{t-1}(X_{t-1}^\star). \end{aligned}$$

Then, by applying Lemma 5 and Lemma 6, the last expression is upper bounded by $\gamma_1\|X_t - X_t^\star\|_F + \gamma_2\|X_t - X_t^\star\|_F^2 + \frac{1}{\tau}\left(\frac{d_\mathrm{M}}{d_\mathrm{m}} + 1\right)\left\|X_{t-1} - X_{t-1}^\star\right\|_F^2$. The thesis is obtained by summing over $t = 1, \dots, T$, and by applying Lemma 4. □

VII. Numerical results

In this section, we numerically analyze the algorithms O-IST, O-DR, and O-DISTA in two time-varying Elastic-net experiments.[1] The first problem is an instance of online identification of time-varying linear systems; the second problem is a practical example of moving target tracking based on the received signal strength (RSS), which has applications, e.g., in indoor monitoring and surveillance.

[1] The code to reproduce the proposed experiments is available at https://github.com/sophie27/Sparse-Time-Varying-Optimization.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

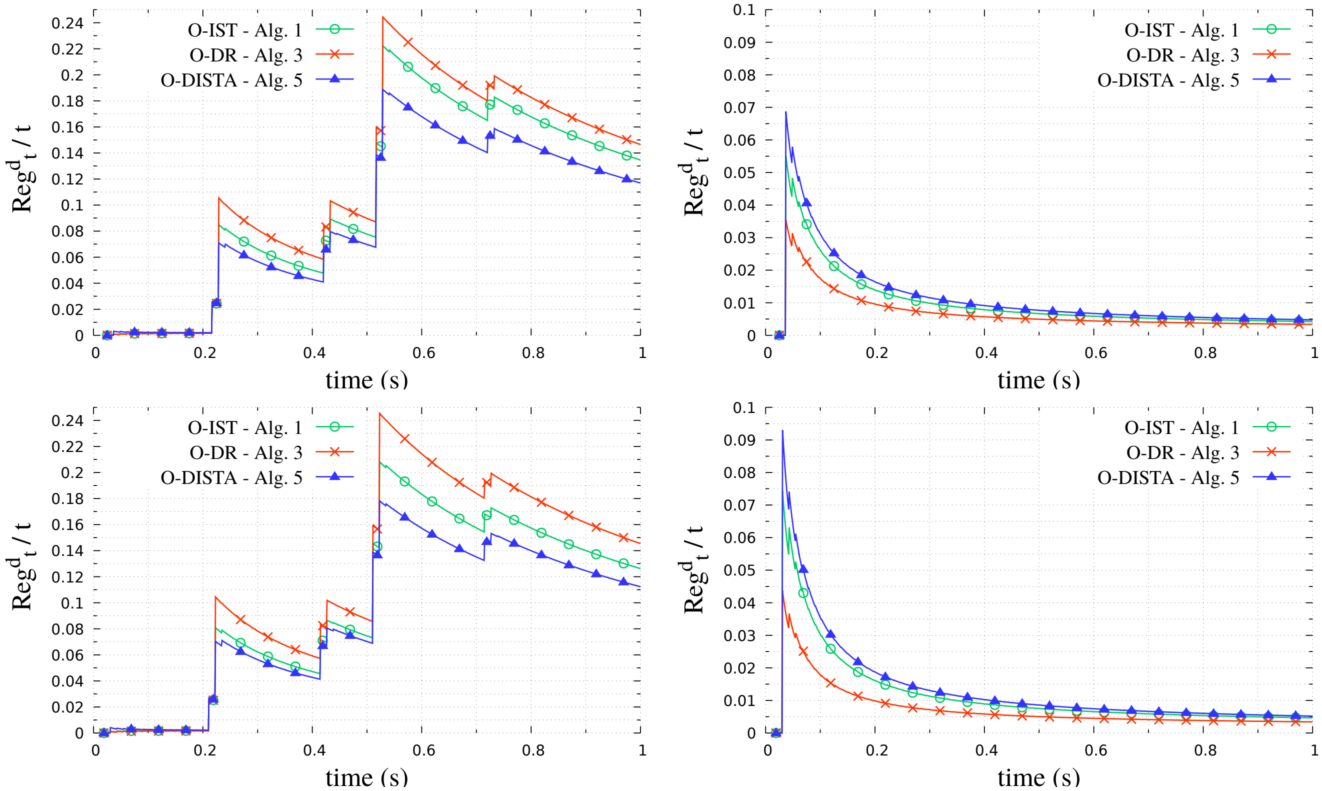

Figure 1: Dynamic regret. Left column: Experiment 1; right column: Experiment 2. From top to bottom: (1) $t_r = 12$ ms, SNR $= 25$dB; (2) $t_r = 6$ ms, SNR $= 25$dB.

A. Online compressed system identification

Online compressed system identification refers to the online estimation of the parameters of a time-varying system from compressed measurements. Specifically, we consider a time-varying autoregressive model with an exogenous input (TVARX), whose input-output relationship is as follows: $y_t = \sum_{p=1}^{P} a_{p,t} y_{t-p} + \sum_{q=1}^{Q} b_{q,t} u_{t-q} + e_t$, where $u_t, y_t \in \mathbb{R}$ respectively are the measurable input and output; $e_t \in \mathbb{R}$ is the measurement error; $a_{p,t}, b_{q,t} \in \mathbb{R}$ are the time-varying parameters to be estimated.

The dimensions P and Q are assumed to be unknown, therefore we initially set sufficiently large bounds $\widehat{P}$ and $\widehat{Q}$ for them and then we look for a parsimonious model using the ℓ_1-norm to promote sparsity.

As in [5], we iteratively collect groups of m measurements $\mathbf{y}_t := (y_t, \dots, y_{t+m})^T$. The measurements are compressed, that is, we choose $m < \widehat{P} + \widehat{Q}$; a Gaussian measurement noise with SNR=25dB is added.

It is easy to check that we can define $A_t \in \mathbb{R}^{m, \widehat{P}+\widehat{Q}}$ as follows:

$$\begin{pmatrix} y_{t-1} & \cdots & y_{t-P} & u_{t-1} & \cdots & u_{t-Q} \\ y_t & \cdots & y_{t-P+1} & u_t & \cdots & u_{t-Q+1} \\ \vdots & & & & & \vdots \\ y_{t+m-1} & \cdots & y_{t+m-P} & u_{t+m-1} & \cdots & u_{t+m-Q} \end{pmatrix}.$$

We revisit the TVARX(1,1) example considered in [68], [5]. The input-output equation is $y_t = a_{1,t} y_{t-1} + b_{1,t} u_{t-1} + e_t$, with $P = Q = 1$. Assuming P and Q unknown, we initially overestimate them as $\widehat{P} = \widehat{Q} = 10$. In other terms, our goal is to track a time-varying parameter vector $\widetilde{x}_t = (a_{1,t}, \dots, a_{10,t}, b_{1,t}, \dots, b_{10,t}) \in \mathbb{R}^n$, $n = 20$, with sparsity $k = 2$ and constant support, given linear observations, as in (3). Specifically, $\widetilde{x}_t = (a_{1,t}, 0, \dots, 0, b_{1,t}, 0, \dots, 0)$, then we refer to components 2,. . .,10,12,. . .,20 as to null parameters. The Elastic-net model (4) is efficient to tackle this problem, as shown in [5]. By cross-validation, we set $\lambda = 10^{-2}$ and $\mu = 10^{-6}$. We remark that, for stability purpose, it makes sense to assume A_t and $\widetilde{x}_t$ bounded, according to Assumption 1. No prior information on sparsity and support is exploited in the estimation. We consider a time horizon of 1 second and sampling frequency of 1000 Hz. Two experiments are conducted. In the first one, $a_{1,t}$ and $b_{1,t}$ are step-wise constant with few abrupt changes, see [68]. Specifically, we set:

Experiment 1:

$$a_1(t) = \begin{cases} -0.9 \text{ if } t < 0.5 \\ 0.9 \text{ otherwise;} \end{cases} \quad b_1(t) = \begin{cases} 0.7 \text{ if } t < 0.2 \\ -0.8 \text{ if } 0.2 \leq t < 0.4 \\ 0.8 \text{ if } 0.4 \leq t < 0.7 \\ -0.7 \text{ otherwise.} \end{cases} \tag{27}$$

In the second experiment, instead, we test a case of smoothly time-varying parameters.

Experiment 2:

$$a_1(t) = 0.8\left(1 + \frac{1}{\sqrt{t}}\right); \quad b_1(t) = 0.9 + 0.1 \sin(2 \log t).$$

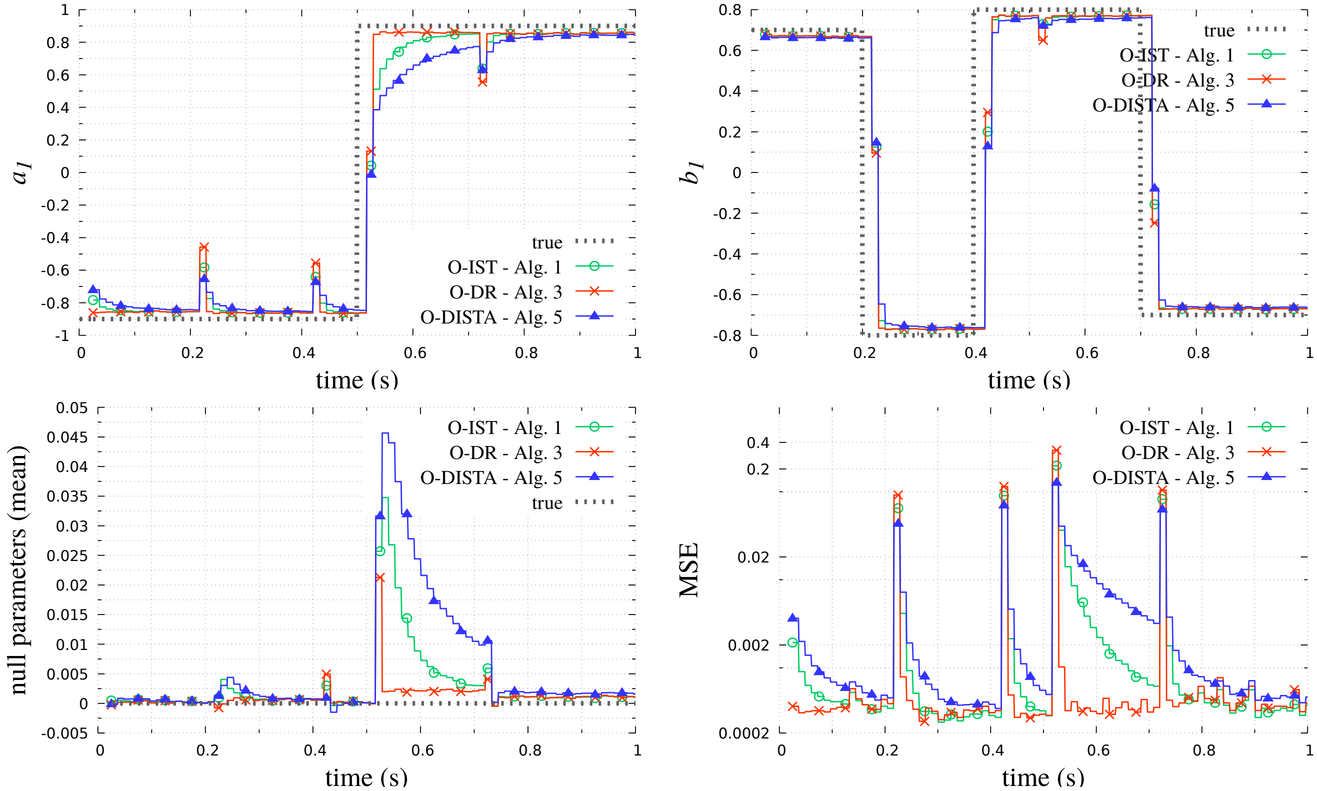

Figure 2: Experiment 1, SNR $= 25$dB, $t_r = 12$ ms. From left to right, averaged estimates of $a_{1,t}$, $b_{1,t}$, null parameters; mean square error.

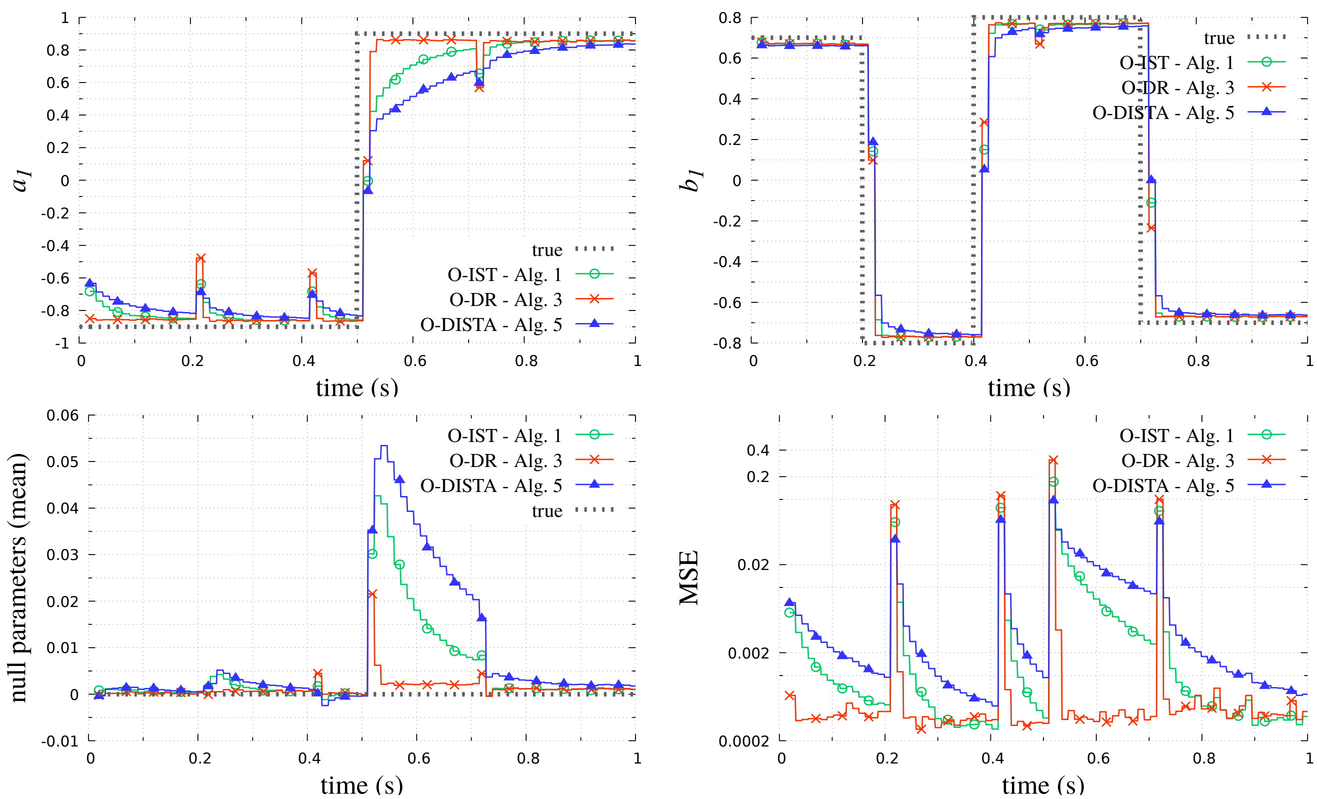

Figure 3: Experiment 1, SNR $= 25$dB, $t_r = 6$ ms. From left to right, averaged estimates of $a_{1,t}$, $b_{1,t}$, null parameters; mean square error.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

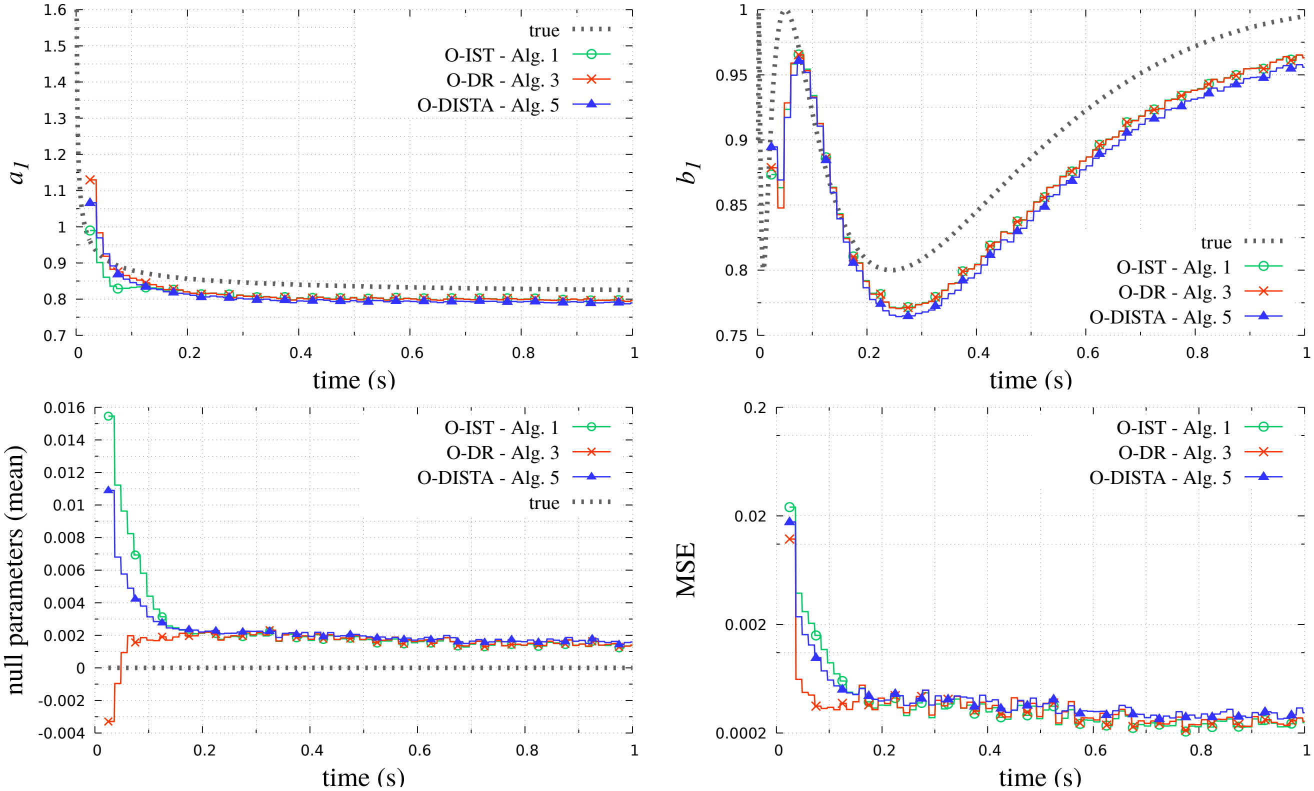

Figure 4: Experiment 2, SNR = 25dB, $t_r = 12$ ms. From left to right, averaged estimates of $a_{1,t}$, $b_{1,t}$, null parameters; mean square error.

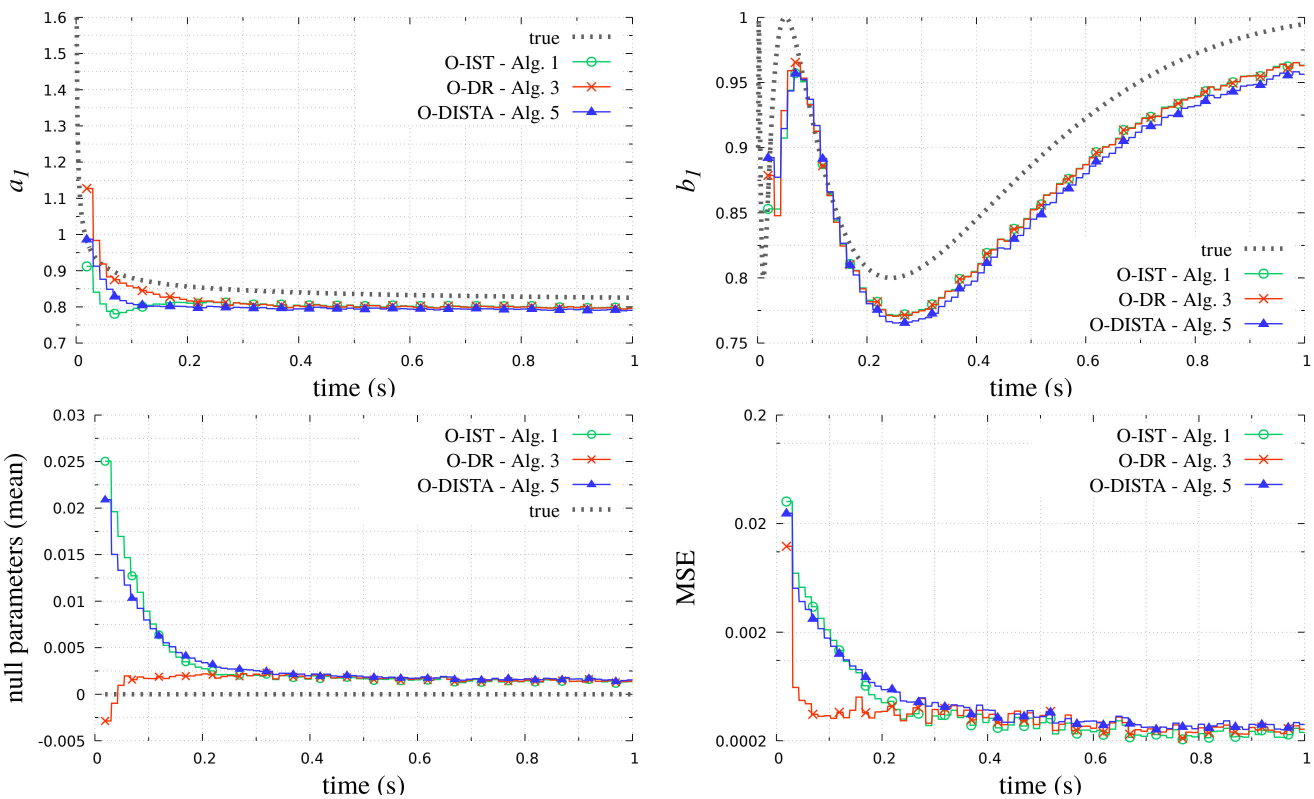

Figure 5: Experiment 2, SNR = 25dB, $t_r = 6$ ms. From left to right, averaged estimates of $a_{1,t}$, $b_{1,t}$, null parameters; mean square error.

These parameters are chosen so that $a_1(t)$ is decreasing to zero, with convergent path length $\sum_{t=1}^{T} |a_1(t+1) - a_1(t)|$, while $b_1(t)$ is oscillating, with sublinear path length $\sum_{t=1}^{T} |b_1(t+1) - b_1(t)|$, of order $\log T$. We notice that the path length was defined above on the optimal points $x_t^\star$, while here we are evaluating it on $\widetilde{x}_t$; however, $x_t^\star$ is expected to be a good approximation of $\widetilde{x}_t$, then the two path lengths are somehow equivalent, see [5, Corollary 1]. The input components are drawn from a standard Gaussian distribution, and are periodic with period m. We set $m = 12$, which corresponds to a rate compression $\frac{m}{n} = \frac{3}{5}$. This implies a delay of 12 ms to acquire the of set measurements plus the run time. To prevent an accumulation of delay, the run time of the algorithm must not exceed 12 ms, so that the algorithm processes the acquired measurements while the successive measurements are being acquired. Then, we iterate the algorithms until a prefixed maximum run time $t_r \leq m$ ms (notice that this approach is different from that of [5], where a maximum number of iterations was set). In our simulations, we test $t_r = 12$ ms and $t_r = 6$ ms.

For O-IST and O-DISTA, sufficient conditions on the parameter τ have been theoretically provided; specifically, for each t, $\tau \leq \|A_t\|_2^{-2}$ for O-IST (see Assumption 1 in [5]), and $\tau \leq \min_{v \in \mathcal{V}} \|A_{v,t}\|_2^{-2}$ for O-DISTA, see Lemma 3. In these experiments, we assume to ignore these lower bounds, and we set τ at each time step: specifically, we use $\tau = 2\|A_t\|_2^{-2}$ for O-IST, while for O-DISTA, each node computes its own τ as $2\|A_{v,t}\|_2^{-2}$. These values are observed to preserve the convergence properties of the algorithms in practice. For O-DISTA, we consider a 3-regular ring topology: there are 4 nodes, each of them taking 3 measurements and communicating with 2 neighbors.

The results shown in figures 1-5 are averaged over 200 random runs. In Figure 1, we show the evolution of the mean dynamic regret $\mathbf{Reg}_t^d/t$. Based on our theoretical results, we expect that $\mathbf{Reg}_t^d/t$ decreases to zero (that is, the dynamic regret is sublinear), when the system is static or when it evolves with sublinear path length. This behavior is confirmed by numerical simulations and can be appreciated in Figure 1. In figures 2-5, we illustrate more details for each experiment. Specifically, we provide four graphs, respectively depicting the tracking of $a_1(t)$, $b_1(t)$, and null parameters, and the mean square error, defined as MSE=$\frac{1}{P+Q}\sum_{s=1}^{T/m} \|\widetilde{x}_{sm} - \widehat{x}_{sm}\|_2^2$, where $\widetilde{x}_t = (a_{1,t}, \dots, a_{P,t}, b_{1,t}, \dots, b_{Q,t})^T$ and $\widehat{x}_t$ is the estimate (the mean estimate depicted for O-DISTA).

Concerning Experiment 1 (figures 2-3), in general, all the implemented algorithms are able to track the true parameters. When the parameters jump between different values (this occurs at time instants 0.2, 0.4, 0.5, 0.7), all the estimates are affected by a sudden perturbation. O-DR is observed to converge faster than O-IST and O-DISTA when the parameters are constant. After jumps, O-DR adapts faster to the new parameters. On the other hand, O-DR is locally more sensitive to jumps: its peaks in correspondence of jumps are more marked. As expected, the distributed nature of O-DISTA makes it a bit less prompt than the centralized algorithms.

Moving from $t_r = 12$ ms to $t_r = 6$ ms, we obtain a slight worsening for all the algorithms, while the response delay after jumps is reduced. O-DR performance is almost equal for $t_r = 12$ ms and $t_r = 6$ ms, which suggests that O-DR almost achieves convergence to the optimal point in 6 ms.

Concerning Experiment 2 (figures 4-5), similar considerations can be drawn. In addition, we observe that O-DR and O-IST have similar performance when the parameters are slowly varying (namely, for $t \geq 200$ ms), while O-DR is more precise when the path length is higher ($t < 200$ ms).

B. Moving target tracking

In the last decade, indoor localization of moving objects has been gaining attention for purposes such as monitoring and surveillance, tracking of products in manufacturing industrial lines, control of unmanned vehicles, and location-based services. While outdoor tracking is mature, due to satellites technologies, indoor tracking is still challenging, and a variety of methodologies are proposed for it, see [69] for a complete overview.

A possible approach to indoor tracking is based on the distance estimation via RSS, which can be implemented in low-cost systems such as wireless sensor networks. RSS-positioning is often associated with CS techniques to obtain an accurate localization from few measurements, see [70], [71].

In this experiment, we consider the CS model proposed in [70] and we extend it to the dynamic case, by using an Elastic-net model. Specifically, we aim to track a moving target in a 25×25 m^2 indoor area. The area is assumed to be subdivided into square cells of side 1 m; the target is well localized when the cell where it lies is identified. This is a sparse problem since only one cell over $n = 625$ is occupied at each time step. The distance is measured with 36 sensor nodes, deployed according to a regular grid over the area, represented by the yellow points in Figure 6. Measurements are linearly obtained as $y_t = A\widetilde{x}_t+$ noise, $y_t \in \mathbb{R}^m$, through a dictionary $A \in \mathbb{R}^{m,n}$ built in a training phase. In the runtime phase, each sensor nodes takes 4 measurements of the signal emitted by the target, for a total of $m = 144 \ll n$ measurements. As to the transmission, we consider the indoor model defined by the IEEE 802.15.4 standard, as reported in [70, Equation 11]. A measurement noise corresponding to an SNR of 25dB is added. The online tracking is performed with O-IST, O-DR, and O-DISTA, which are run for 50 ms at each iteration, this time being sufficiently small such that, in the next measurement, the target is in the same cell or in adjacent cell. As to O-IST and O-DR, the data from sensors are processed in a centralized fusion center, while for O-DISTA the processing is performed in-network, with local communications determined by the grid topology depicted in the third graph of Figure 6; specifically, each node can communicate with nodes at a maximum distance of 4.5 m. In Figure 6, we show the path of the target in the 25×25 m^2 area, and the corresponding tracking. We see that the three algorithms are substantially able to track the path. O-DR is the most precise and responsive, while O-DISTA is a bit less accurate. The performance is better visible in Figure 7, where the distance $\|x_t - \widetilde{x}_t\|_2$ and the cumulative distance

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2020.3010242, IEEE Transactions on Automatic Control

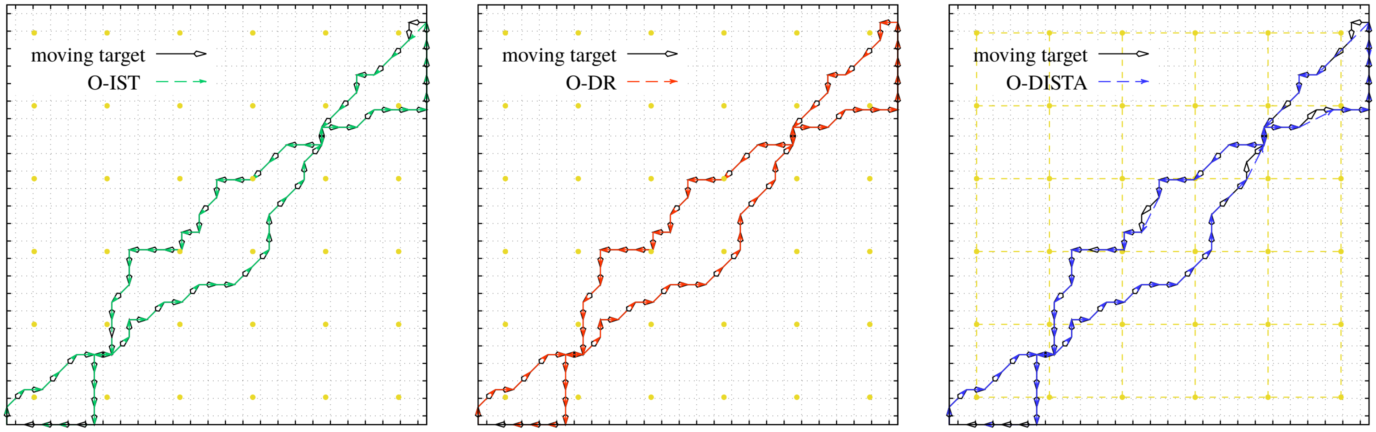

Figure 6: Indoor tracking of a moving target from RSS compressed measurements, in a 25×25 m^2 area. The path of the target and the corresponding online estimations are depicted. The yellow points denote the sensor nodes.

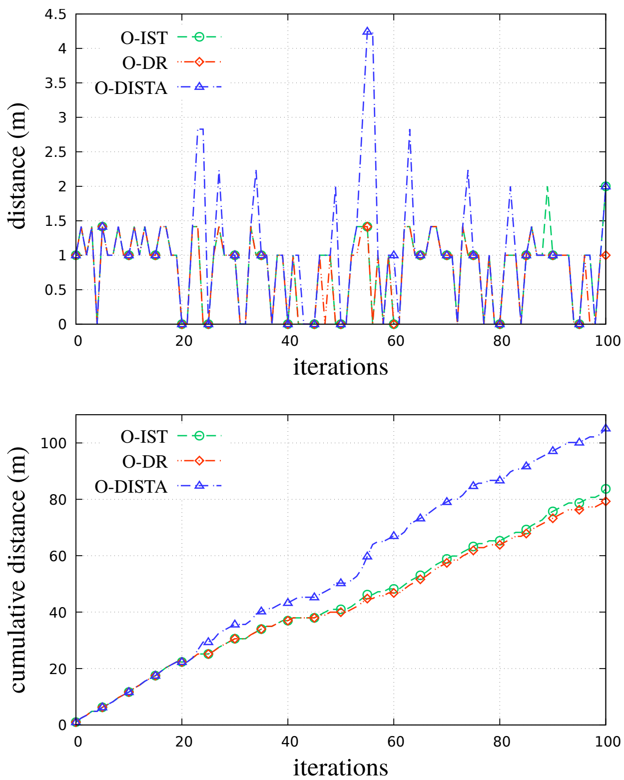

Figure 7: Indoor tracking of a moving target from RSS compressed measurements: instantaneous distance and cumulative distance.

$\sum_{i=1}^{t} \|x_i - \widetilde{x}_i\|_2$ are shown. A distance of 1 m or of $\sqrt{2}$ m is natural when the target is moving, since the delay for processing in envisaged: such distances mean that the target has moved in an adjacent (horizontal/vertical or diagonal) cell. The distance may be null when the target stops in a cell. On the basis of this observation, O-DR is optimal, while O-DIST and O-DISTA present few more delays and missed corners.

Finally, we remark that, in this experiment the dictionary A is constant, then in Assumption 3, Q_t is constant, which means that $\widetilde{x}_t$ is not required to be bounded. In practice, this means that the moving target $\widetilde{x}_t$ is not required to move within a fixed area to match the theoretical features of O-DISTA. On the other hand, if the area is not priorly fixed, a system of moving sensors should be provided.

VIII. Conclusions

In this work, we develop and analyze novel centralized and distributed strategies for sparse time-varying optimization. Specifically, we consider quadratic, strongly convex, optimization problems with ℓ_1 regularization. In this setting, we provide a rigorous analysis in terms of dynamic regret. Furthermore, numerical experiments on compressed system identification and indoor moving target tracking are presented. Future work will be devoted to extend the analysis to larger classes of composite problems.

References

[1] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 647–662, 2015.

[2] J. L. Jerez, P. J. Goulart, S. Richter, G. A. Constantinides, E. C. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *IEEE Trans. Autom. Control*, vol. 59, no. 12, pp. 3238–3251, 2014.

[3] T. Chen, Q. Ling, and G. B. Giannakis, "An online convex optimization approach to proactive network resource allocation," *IEEE Trans. Signal Process.*, vol. 65, no. 24, pp. 6350–6364, 2017.

[4] A. Mokhtari, S. Shahrampour, A. Jadbabaie, and A. Ribeiro, "Online optimization in dynamic environments: Improved regret rates for strongly convex problems," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2016.

[5] S. M. Fosson, "Online optimization in dynamic environments: a regret analysis for sparse problems," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2018.

[6] A. S. Bedi, P. Sarma, and K. Rajawat, "Tracking moving agents via inexact online gradient descent algorithm," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 202–217, 2018.

[7] A. Simonetto, "Prediction-correction dual ascent for time-varying convex programs," in *Proc. Amer. Control Conf. (ACC)*, 2018, pp. 4508–4513.

[8] E. Dall'Anese, A. Simonetto, S. Becker, and L. Madden, "Optimization and learning with information streams: Time-varying algorithms and applications," *arXiv 1910.08123*, 2019.

[9] J. M. Bravo, A. Suarez, M. Vasallo, and T. Alamo, "Slide window bounded-error time-varying systems identification," *IEEE Trans. Autom. Control*, vol. 61, no. 8, pp. 2282–2287, 2016.

[10] G. Prando, D. Romeres, and A. Chiuso, "Online identification of time-varying systems: A Bayesian approach," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2016, pp. 3775–3780.

[11] Q. Ling and A. Ribeiro, "Decentralized dynamic optimization through the alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1185–1197, 2014.

[12] N. Vaswani and J. Zhan, "Recursive recovery of sparse signal sequences from compressive measurements: A review," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3523–3549, 2016.

[13] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A class of prediction-correction methods for time-varying convex optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4576–4591, 2016.

[14] S. Rahili and W. Ren, "Distributed continuous-time convex optimization with time-varying cost functions," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1590–1605, 2017.

[15] C. Sun, M. Ye, and G. Hu, "Distributed time-varying quadratic optimization for multiple agents under undirected graphs," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3687–3694, 2017.

[16] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Trans. Autom. Control*, vol. 63, no. 3, pp. 714–725, 2018.

[17] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Found. Trends Mach. Learn.*, vol. 4, no. 2, pp. 107–194, 2012.

[18] L. Zhang, T. Yang, J. Yi, R. Jin, and Z.-H. Zhou, "Improved dynamic regret for non-degenerate functions," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 733–742.

[19] L. Zhang, T. Yang, R. Jin, and Z.-H. Zhou, "Dynamic regret of strongly adaptive methods," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 9372–9381.

[20] K. Lu, G. Jing, and L. Wang, "Online distributed optimization with strongly pseudoconvex-sum cost functions," *IEEE Trans Autom. Control*, pp. 1–1, 2019.

[21] R. Dixit, A. S. Bedi, R. Tripathi, and K. Rajawat, "Online learning with inexact proximal online gradient descent algorithms," *IEEE Trans. Signal Process.*, vol. 67, no. 5, pp. 1338–1352, 2019.

[22] A. Simonetto and E. Dall'Anese, "Prediction-correction algorithms for time-varying constrained optimization," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5481–5494, 2017.

[23] A. Simonetto, "Dual prediction-correction methods for linearly constrained time-varying convex programs," *IEEE Trans. Autom. Control*, vol. 64, no. 8, pp. 3355–3361, 2019.

[24] A. Simonetto, A. Koppel, A. Mokhtari, G. Leus, and A. Ribeiro, "Decentralized prediction-correction methods for networked time-varying convex optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5724–5738, 2017.

[25] S. Lee, A. Nedić, and M. Raginsky, "Coordinate dual averaging for decentralized online optimization with nonseparable global objectives," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 34–44, 2018.

[26] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[27] R. Tóth, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "Compressive system identification in the linear time-invariant framework," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2011, pp. 783–790.

[28] B. M. Sanandaji, T. L. Vincent, M. B. Wakin, and R. Tóth, "Compressive system identification of LTI and LTV ARX models," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2011, pp. 791–798.

[29] D. Zachariah, S. Chatterjee, and M. Jansson, "Dynamic iterative pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4967–4972, 2012.

[30] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270–5284, 2013.

[31] J. Mota, N. Deligiannis, A. C. Sankaranarayanan, V. Cevher, and M. Rodrigues, "Dynamic sparse state estimation using $\ell_1 - \ell_1$ minimization: Adaptive-rate measurement bounds, algorithms and applications," in *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2015, pp. 3332–3336.

[32] A. S. Charles, A. Balavoine, and C. J. Rozell, "Dynamic filtering of time-varying sparse signals via ℓ_1 minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5644–5656, 2016.

[33] A. Balavoine, C. J. Rozell, and J. Romberg, "Discrete and continuous-time soft-thresholding for dynamic signal recovery," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3165–3176, 2015.

[34] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. Simul.*, vol. 4, no. 4, pp. 1168–1200, 2005.

[35] M. Fornasier, "Numerical methods for sparse recovery," in *Theoretical Foundations and Numerical Methods for Sparse Recovery*, M. Fornasier, Ed. Radon Series Comp. Appl. Math., de Gruyter, 2010, pp. 93–200.

[36] P. L. Lions and B. Mercier, "Splitting algorithms for the sum of two nonlinear operators," *SIAM J. Numer. Anal.*, no. 6, pp. 964–979, 1979.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1 – 122, 2010.

[38] P. Giselsson and S. Boyd, "Linear convergence and metric selection for Douglas-Rachford splitting and ADMM," *IEEE Trans. Autom. Control*, no. 62, 2017.

[39] C. Ravazzi, S. M. Fosson, and E. Magli, "Distributed iterative thresholding for ℓ_0/ℓ_1-regularized linear inverse problems," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 2081–2100, 2015.

[40] A. Simonetto, "Time-varying convex optimization via time-varying averaged operators," *arXiv 1704.0733*, 2017.

[41] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, pp. 267–288, 1996.

[42] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. New York: Springer, 2013.

[43] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Royal Stat. Soc. B*, vol. 67, no. 2, pp. 301–320, 2005.

[44] R. J. Tibshirani, "The Lasso problem and uniqueness," *Elec. J. Stat.*, vol. 7, pp. 1456–1490, 2013.

[45] A. Khalajmehrabadi, N. Gatsis, D. J. Pack, and D. Akopian, "A joint indoor WLAN localization and outlier detection scheme using LASSO and elastic-net optimization techniques," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2079–2092, 2017.

[46] M. Gallieri, *Lasso-MPC - Predictive Control with ℓ_1-Regularised Least Squares*, 1st ed. Springer Publishing Company, Incorporated, 2016.

[47] R. P. Aguilera, G. Urrutia, R. A. Delgado, D. Dolz, and J. C. Agüero, "Quadratic model predictive control including input cardinality constraints," *IEEE Trans. Autom. Control*, vol. 62, no. 6, pp. 3068–3075, 2017.

[48] J. H. Lee, K. S. Lee, and W. C. Kim, "Model-based iterative learning control with a quadratic criterion for time-varying linear systems," *Automatica*, vol. 36, no. 5, pp. 641 – 657, 2000.

[49] T. Oomen and C. R. Rojas, "Sparse iterative learning control with application to a wafer stage: Achieving performance, resource efficiency, and task flexibility," *Mechatronics*, vol. 47, pp. 134–147, 2017.

[50] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 928–936.

[51] 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.

[52] T. Yang, L. Zhang, R. Jin, and J. Yi, "Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 449–457.

[53] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari, "Composite objective mirror descent," in *Conf. Learn. Theory (COLT)*, 2010, pp. 14–26.

[54] H. Wang and A. Banerjee, "Online alternating direction method," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2012, pp. 1119–1126.

[55] T. Suzuki, "Dual averaging and proximal gradient descent for online alternating direction multiplier method," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 28, no. 1, 2013, pp. 392–400.

[56] S. Hosseini, A. Chapman, and M. Mesbahi, "Online distributed convex optimization on dynamic networks," *IEEE Trans. Autom. Control*, vol. 61, no. 11, pp. 3545–3550, 2016.

[57] M. Akbari, B. Gharesifard, and T. Linder, "Individual regret bounds for the distributed online alternating direction method of multipliers," *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1746–1752, 2019.

[58] P. Sopasakis, N. Freris, and P. Patrinos, "Accelerated reconstruction of a compressively sampled data stream," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2016, pp. 1078–1082.

[59] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[60] J. Matamoros, S. M. Fosson, E. Magli, and C. Antón-Haro, "Distributed ADMM for in-network reconstruction of sparse signals with innovations," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 4, pp. 225–234, 2015.

[61] A. Fiandrotti, S. M. Fosson, C. Ravazzi, and E. Magli, "GPU-accelerated algorithms for compressed signals recovery with application to astro-

nomical imagery deblurring," *Int. J. Remote Sens.*, vol. 39, no. 7, pp. 2043–2065, 2018.
[62] W. M. Moursi and Y. Zinchenko, "A note on the equivalence of operator splitting methods," in *Splitting Algorithms, Modern Operator Theory, and Applications*, H. H. Bauschke, R. S. Burachik, and D. R. Luke, Eds. Springer International Publishing, 2019, pp. 331–349.
[63] Y. Zhang, E. Dall'Anese, and M. Hong, "Dynamic ADMM for real-time optimal power flow," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, 2017, pp. 1085–1089.
[64] M. Maros and J. Jaldén, "ADMM for distributed dynamic beamforming," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 2, pp. 220–235, 2018.
[65] M. Maros and J. Jaldén, "On decentralized tracking with ADMM for problems with time-varying curvature," *arXiv 1903.06492*, 2019.
[66] X. Cao and K. J. R. Liu, "Dynamic sharing through the ADMM," *IEEE Trans. Autom. Control*, p. Early Access, 2019.
[67] D. Davis and W. Yin, *Convergence Rate Analysis of Several Splitting Schemes*. Cham: Springer International Publishing, 2016, pp. 115–163.
[68] Y. Li, H. l. Wei, and S. A. Billings, "Identification of time-varying systems using multi-wavelet basis functions," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 3, pp. 656–663, 2011.
[69] D. Dardari, P. Closas, and P. M. Djurić, "Indoor tracking: Theory, methods, and technologies," *IEEE Trans. Vehic. Tech.*, vol. 64, no. 4, pp. 1263–1278, 2015.
[70] C. Feng, S. Valaee, and Z. Tan, "Multiple target localization using compressive sensing," in *IEEE Global Telecommun. Conf. (GLOBECOM)*, 2009, pp. 1–6.
[71] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Received-signal-strength-based indoor positioning using compressive sensing," *IEEE Trans. Mobile Comput.*, vol. 11, no. 12, pp. 1983–1993, 2012.

Sophie M. Fosson (M'12) received the M.Sc. degree in Applied Mathematics from Politecnico di Torino, Italy, in 2005. She received the Ph.D. degree in Mathematics for the Industrial Technologies from Scuola Normale Superiore di Pisa, Italy, in 2011. From 2012 to 2016, she was a Postdoctoral Associate at the Department of Electronics and Telecommunications, Politecnico di Torino. Currently, she is Assistant Professor at the Department of Control and Computer Engineering, Politecnico di Torino. Her main research interests are sparse optimization, compressed sensing, system identification and machine learning.