A primal-dual method for conic constrained distributed optimization problems

Authored by:

Necdet Serhat Aybat
Erfan Yazdandoost Hamedani

Abstract

We consider cooperative multi-agent consensus optimization problems over an undirected network of agents, where only those agents connected by an edge can directly communicate. The objective is to minimize the sum of agent-specific composite convex functions over agent-specific private conic constraint sets; hence, the optimal consensus decision should lie in the intersection of these private sets. We provide convergence rates in sub-optimality, infeasibility and consensus violation; examine the effect of underlying network topology on the convergence rates of the proposed decentralized algorithms; and show how to extend these methods to handle time-varying communication networks.

1 Paper Body

Let G = (N , E) denote a connected undirected graph of N computing nodes, where N , {1, . . . , N } and E ? N ? N denotes the set of edges ? without loss of generality assume that (i, j) ? E implies i ¡ j. Suppose nodes i and j can exchange information only if (i, j) ? E, and each node i ? N has a private (local) cost function ?i : Rn ? R ? {+?} such that ?i (x) , ?i (x) + fi (x),

(1)

where ?i : Rn ? R ? {+?} is a possibly non-smooth convex function, and fi : Rn ? R is a smooth convex function. We assume that fi is differentiable on an open set containing dom ?i with a Lipschitz continuous gradient ?fi , of which Lipschitz constant is Li ; and the prox map of ?i ,

prox?i (x) , argmin ?i (y) + y?Rn

1 2

ky ? xk2 ,

(2)

is efficiently computable for i ? N , where k.k denotes the Euclidean norm. Let Ni , {j ? N : (i, j) ? E or (j, i) ? E} denote the set of neighboring nodes of i ? N , and di , —Ni — is the degree of node i ? N . Consider the following minimization problem: min

x?Rn
X
i?N
?i (x)
s.t.
Ai x ? bi ? Ki ,
?i ? N ,
(3)

where Ai ? Rmi ?n , bi ? Rmi and Ki ? Rmi is a closed, convex cone. Suppose that projections onto Ki can be computed efficiently, while the projection onto the preimage A?1 i (Ki +bi) is assumed to be impractical, e.g., when Ki is the positive semidefinite cone, projection to preimage requires solving an SDP. Our objective is to solve (3) in a decentralized fashion using the computing nodes N and exchanging information only along the edges E. In Section 2 and Section 3, we consider (3) when the topology of the connectivity graph is static and time-varying, respectively. This computational setting, i.e., decentralized consensus optimization, appears as a generic model for various applications in signal processing, e.g., [1, 2], machine learning, e.g., [3, 4, 5] and statistical inference, e.g., [6]. Clearly, (3) can also be solved in a ?centralized? fashion by communicating all the private functions ?i to a central node, and solving the overall problem at this 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

node. However, such an approach can be very expensive both from communication and computation perspectives when compared to the distributed algorithms which are far more scalable to increasing problem data and network sizes. In particular, suppose (Ai , bi) ? Rm?(n+1) and 2 ?i (x) = ? kxk1 + kAi x ? bi k for some given ? ¿ 0 for i ? N such that m ? n and N ? 1. Hence, (3) is a very large scale LASSO problem with distributed data. To solve (3) in a centralized fashion, the data {(Ai , bi) : i ? N } needs to be communicated to the central node. This can be prohibitively expensive, and may also violate privacy constraints ? in case some node i does not want to reveal the details of its private data. Furthermore, it requires that the central node has large enough memory to be able to accommodate all the data. On the other hand, at the expense of slower convergence, one can completely do away with a central node, and seek for consensus among all the nodes on an optimal decision using ?local? decisions communicated by the neighboring nodes. From computational perspective, for certain cases, computing partial gradients locally can be more computationally efficient when compared to computing the entire gradient at a central node. With these considerations in mind, we propose decentralized algorithms that can compute solutions to (3) using only local computations without explicitly requiring the nodes to communicate the functions {?i : i ? N }; thereby, circumventing all privacy, communication and memory issues. Examples of constrained machine learning problems that fit into our framework include multiple kernel learning [7], and primal linear support vector machine (SVM) problems. In the numerical section we implemented the proposed algorithms on the primal SVM problem. 1.1

Previous Work

There has been active research [8, 9, 10, 11, 12] on solving convex-concave saddle point problems minx maxy L(x, y). In [9] primal-dual proximal algorithms are proposed for convex-concave problems with known saddle-point structure minx maxy Ls (x, y) , ?(x) + hT x, yi ? h(y), where ? and h are convex functions, and T is a linear map. These algorithms converge with rate O(1/k) for the primal-dual gap, and they can be modified to yield a convergence rate of O(1/k 2) when either ? or h is strongly convex, and O(1/ek) linear rate, when both ? and h are strongly convex. More recently, in [11] Chambolle and Pock extend their previous work in [9], using simpler proofs, to handle composite convex primal functions, i.e., sum of smooth and (possibly) nonsmooth functions, and to deal with proximity operators based on Bregman distance functions. P Consider minx?Rn { i?N ?i (x) : x ? ?i?N Xi } over G = (N , E). Although the unconstrained consensus optimization, i.e., Xi = Rn , is well studied ? see [13, 14] and the references therein, the constrained case is still an immature, and recently developing area of active research [13, 14, 15, 16, 17, 18, 19]. Other than few exceptions, e.g., [15, 16, 17], the methods in literature require that each node compute a projection on the privately known set Xi in addition to consensus and (sub)gradient steps, e.g., [18, 19]. Moreover, among those few exceptions that do not use projections onto Xi when ?Xi is not easy to compute, only [15, 16] can handle agent-specific constraints without assuming global knowledge of the constraints by all agents. However, no rate results in terms of suboptimality, local infeasibility, and consensus violation exist for the primaldual distributed methods in [15, 16] when implemented for the agent-specific conic constraint sets Xi = {x : Ai x ? bi ? Ki } studied in this paper. In [15], a consensus-based distributed primaldual perturbation (PDP) algorithm using a square summable but not summable step-size sequence is proposed. The objective is to minimize a composition of a global network function (smooth) with the summation of local objective functions (smooth), subject to local compact sets and inequality constraints on the summation of agent specific constrained functions. They showed that the local primal-dual iterate sequence converges to a global optimal primal-dual solution; however, no rate result was provided. The proposed PDP method can also handle non-smooth constraints with similar convergence guarantees. Finally, while we were preparing this paper, we became aware of a very recent work [16] related to ours. The authors proposed a distributed algorithm on time-varying communication network for solving saddle-point problems subject to consensus constraints. The algorithm can also be applied to solve consensus optimization problems with inequality constraints that can be written as summation of local convex functions of local and global variables. Under some assumptions, it is shown that using a carefully selected decreasing step-size sequence, the ? ergodic average of primal-dual sequence converges with O(1/ k) rate in terms of saddle-point evaluation error; however, when applied to constrained optimization problems, no rate in terms of either suboptimality or infeasibility is provided.

2

Contribution. We propose primal-dual algorithms for distributed optimization subject to agent specific conic constraints. By assuming composite convex structure on the primal functions, we show that our proposed algorithms converge with O(1/k) rate where k is the number of consensus iterations. To the best of our knowledge, this is the best rate result for our setting. Indeed, ?-optimal and ?-feasible solution can be computed within O(1/?) consensus iterations for the static topology, and within O(1/?1+1/p) consensus iterations for the dynamic topology for any rational p ? 1, although O(1) constant gets larger for large p. Moreover, these methods are fully distributed, i.e., the agents are not required to know any global parameter depending on the entire network topology, e.g., the second smallest eigenvalue of the Laplacian; instead, we only assume that agents know who their neighbors are. Due to limited space, we put all the technical proofs to the appendix. 1.2

Preliminary

Let X and Y be finite-dimensional vector spaces. In a recent paper, Chambolle and Pock [11] proposed a primal-dual algorithm (PDA) for the following convex-concave saddle-point problem: min max L(x, y) , ?(x) + hT x, yi ? h(y),

x?X y?Y

where ?(x) , ?(x) + f (x),

(4)

? and h are possibly non-smooth convex functions, f is a convex function and has a Lipschitz continuous gradient defined on dom ? with constant L, and T is a linear map. Briefly, given x0 , y0 and algorithm parameters ?x , ?y ¿ 0, PDA consists of two proximal-gradient steps: E D E D 1 xk+1 ? argmin ?(x) + f (xk) + ?f (xk), x ? xk + T x, yk + Dx (x, xk) ? x x D E 1 Dy (y, yk), yk+1 ? argmin h(y) ? T (2xk+1 ? xk), y + ?y y

(5a) (5b)

where Dx and Dy are Bregman distance functions corresponding to some continuously differentiable strongly convex ?x and ?y such that dom ?x ? dom ? and dom ?y ? dom h. In particu?) , ?x (x) ? ?x (? ? i, and Dy is defined similarly. In [11], a simple x) ? h??x (? x), x ? x lar, Dx (x, x proof for the ergodic convergence is provided for (5); indeed, it is shown that, when the convexity 2 (T), then modulus for ?x and ?y is 1, if ?, ? ¿ 0 are chosen such that (?1x ? L) ?1y ? ?max ?K) ? L(? xK , y) ? L(x, y

1 K

?K , for all x, y ? X ? Y, where x

1 1 Dx (x, x0) + Dy (y, y0) ? T (x ? x0), y ? y0 , ?x ?y

1 K

PK

k=1

?K , xk and y

1 K

PK

k=1

(6)

yk .

First, we define the notation used throughout the paper. Next, in Theorem 1.1, we discuss a special case of (4), which will help us prove the main results of this paper, and also allow us to develop decentralized algorithms for the consensus optimization problem in (3). The proposed algorithms in this paper can distribute the computation over the nodes such that each node?s computation is based on the local topology of G and the private information only available to that node.

Notation. Throughout the paper, k.k denotes the Euclidean norm. Given a convex set S, let ?S (.) denote its support function, i.e., ?S (?) , supw?S h?, wi, let IS (?) denote the indicator function of S, i.e., IS (w) = 0 for w ? S and equal to +? otherwise, and let PS (w) , argmin{kv ? wk : v ? S} denote the projection onto S. For a closed convex set S, we define the distance function as dS (w) , kPS (w) ? wk. Given a convex cone K ? Rm , let K? denote its dual cone, i.e., K? , {? ? Rm : h?, wi ? 0 ?w ? K}, and K? , ?K? denote the polar cone of K. Note that for a given cone K ? Rm , ?K (?) = 0 for ? ? K? and equal to +? if ? 6? K? , i.e., ?K (?) = IK? (?) for all ? ? Rm . Cone K is called proper if it is closed, convex, pointed, and it has a nonempty interior. Given a convex function g : Rn ? R ? {+?}, its convex conjugate is defined as g ? (w) , sup??Rn hw, ?i ? g(?). ? denotes the Kronecker product, and In is the n ? n identity matrix. Definition 1. Let X , ?i?N Rn and X ? x = [xi]i?N ; Y , ?i?N Rmi ? Rm0 , Y ? y = P [? ? ??]? and ? = [?i]i?N ? Rm , where m , i?N mi , and ? denotes the Cartesian product. Given parameters ? ¿ 0, ?i , ?i ¿ 0 for i ? N , let D? , ?1 Im0 , D? , diag([?1i Imi]i?N), and D? , diag([?1i In]i?N). Defining ?x (x) , 12 x? D? x and ?y (y) , 12 ? ? D? ? + 12 ?? D? ? ?) = 12 kx ? x ? k2D? , and Dy (y, y ?) = leads to the following Bregman distance functions: Dx (x, x

1 2 2 1 1 ? ? + ? ? ? ? , where the Q-norm is defined as kzk , (z Qz) 2 for Q ? 0. ??? 2

D?

2

Q

D?

3

Theorem 1.1. Let X , Y, and Bregman functions Dx , Dy be defined as in Definition 1. Suppose P P ?(x) , i?N ?i (xi), and h(y) , h0 (?) + i?N hi (?i), where {?i }i?N are composite convex functions defined as in (1), and {hi }i?N are closed convex with simple prox-maps. Given A0 ? ? Rm0 ?n—N — and {Ai }i?N such that Ai ? Rmi ?n , let T = [A? A? 0] , where A , diag([Ai]i?N) ? Rm?n—N — is a block-diagonal matrix. Given the initial point (x0 , y0), the PDA iterate sequence according to {xk , yk }k?1 , generated ? (5a) and (5b) when ?x = ?y = 1, satisfies (6) for all K ? 1 ? ?? D

? if Q(A, A0) , ? ?A

?A0

?A? D? 0

?A? 0 ? ? , diag([(1 ? Li)In]i?N). Moreover, if a 0 ? 0, where D ?i D?

? saddle point exists for (4), and Q(A, A0) ? 0, then {xk , yk }k?1 converges to a saddle point of (4); k ?k hence, {? x , y }k?1 converges to the same point.

Although the proof of Theorem 1.1 follows from the lines of [11], we provide the proof in the appendix for the sake of completeness as it will be used repeatedly to derive our results. Next we discuss how (5) can be implemented to compute an ?-optimal solution to (3) in a distributed way using only O(1/?) communications over the communication graph G while respecting nodespecific privacy requirements. Later, in Section 3, we consider the scenario where the topology of 1 the connectivity graph is time-varying, and propose a distributed algorithm that requires O(1/?1+ p) communications for any p ? 1. Finally, in Section 4 we test the proposed algorithms for solving the primal SVM problem in a decentralized manner. These results are shown under Assumption 1.1. Assumption 1.1. The duality gap for (3) is zero, and a primal-dual solution to (3) exists. A sufficient condition for this is the existence of a Slater point, i.e., there exists x ? ? relint(dom ?) ? ? bi ? int(Ki) for i ? N , where dom ? = ?i?N dom ?i . such that Ai x

2

Static Network Topology

Let xi ? Rn denote the local decision vector of node i ? N . By taking advantage of the fact that G is connected, we can reformulate (3) as the following distributed consensus optimization problem: min

xi ?Rn , i?N

(

X

i?N

?i (xi) — xi = xj : ?ij , ?(i, j) ? E,

Ai xi ? bi ? Ki : ?i , ?i ? N

)

,

(7)

where ?ij ? Rn and ?i ? Rmi are the corresponding dual variables. Let x = [xi]i?N ? Rn—N — . The consensus constraints xi = xj for (i, j) ? E can be formulated as M x = 0, where M ? Rn—E—?n—N — is a block matrix such that M = H ? In where H is the oriented edge-node incidence matrix, i.e., the entry H(i,j),l , corresponding to edge (i, j) ? E and l ? N , is equal to 1 if l = i, ?1 if l = j, and 0 otherwise. Note that M T M = H T H ? In = ? ? In , where ? ? R—N —?—N — denotes the graph Laplacian of G, i.e., ?ii = di , ?ij = ?1 if (i, j) ? E or (j, i) ? E, and equal to 0 otherwise. ? For any closed convex set S, we have ?S? (?) = IS (?); therefore, using the fact that ?K = IKi for i i ? N , one can obtain the following saddle point problem corresponding to (7),

min max L(x, y) , x

y

X

i?N

?i (xi) + h?i , Ai xi ? bi i ? ?Ki (?i) + h?, M xi, where y = [? ? ??]? for ? = [?ij](i,j)?E ? Rn—E— , ? = [?i]i?N ? Rm , and m ,

P

i?N

(8)

mi .

Next, we study the distributed implementation of PDA in (5a)-(5b) to solve (8). Let ?(x) , P P i?N ?i (xi), and h(y) , i?N ?Ki (?i) + hbi , ?i i. Define the block-diagonal matrix A , m?n—N — diag([Ai]i?N) ? R and T = [A? M ?]? . Therefore, given the initial iterates x0 , ? 0 , ?0 and parameters ? ¿ 0, ?i , ?i ¿ 0 for i ? N , choosing Dx and Dy as defined in Definition 1, and setting ?x = ?y = 1, PDA iterations in (5a)-(5b) take the following form: h i xk+1 ? argminh?k , M xi + x

X

i?N

?i (xi) + h?f (xki), xi i + hAi xi ? bi , ?ik i +

1 kxi ? xki k2 , 2?i

1 k?i ? ?ik k2 , i ? N 2?i n o 1 ? argmin ? hM (2xk+1 ? xk), ?i + k? ? ?k k2 = ?k + ?M (2xk+1 ? xk). 2? ?

(9a)

?ik+1 ? argmin ?Ki (?i) ? hAi (2xk+1 ? xki) ? bi , ?i i + i

(9b)

?k+1

(9c)

?i

4

Since Ki is a cone, prox?i ?K (.) = PKi? (.); hence, ?ik+1 can be written in closed form as i

?ik+1

= PK?i ?ik + ?i Ai (2xk+1 ? xki) ? bi , i

i ? N.

Using recursion in (9c), we can write ?k+1 as a partial summation of primal iterates {x? }k?=0 , i.e., Pk?1 Pk ?k = ?0 + ? ?=0 M (2x?+1 ? x?). Let ?0 ? ?M x0 , s0 ? x0 , and sk , xk + ?=1 x? for k ? 1; hence, ?k = ?M sk . Using the fact that M ? M = ? ? In , we obtain hM x, ?k i = ? hx, (? ? In)sk i = ?

P

i?N hxi ,

P

k j?Ni (si

? skj)i.

Thus, PDA iterations given in (9) for the static graph G can be computed in a decentralized way, via the node-specific computations as in Algorithm DPDA-S displayed in Fig. 1 below. Algorithm DPDA-S (x0 , ? 0 , ?, {?i , ?i }i?N)

Initialization: s0i ? x0i , i ? N Step k: (k ? 0)

P k k k , 1. xk+1 ? prox?i ?i xki ? ?i ?fi (xki) + A? i ?i + ? i j?Ni (si ? sj) Pk+1 ? k+1 ? x + x , i ? N 2. sk+1 i i ?=1 i

3. ?ik+1 ? PK?i ?ik + ?i Ai (2xk+1 ? xki) ? bi , i ? N i

i?N

Figure 1: Distributed Primal Dual Algorithm for Static G (DPDA-S)

The convergence rate for DPDA-S, given in (6), follows from Theorem 1.1 with the help of following ? technical lemma which provides a sufficient condition for Q(A, A0) ? 0. Lemma 2.1. Given {?i , ?i }i?N and ? such that ? ¿ 0, and ?i , ?i ¿ 0 for i ? N , let A0 = M and ? , Q(A, ? A0) 0 if {?i , ?i }i?N and ? are chosen such that A , diag([Ai]i?N). Then Q

1 ? Li ? 2?di ?i

1 2 ? ?max (Ai), ?i

? i ? N,

(10)

? ? 0 if (10) holds with strict inequality, where Q(A, ? and Q A0) is defined in Theorem 1.1. ?1 2 (Ai) for any ci ¿ 0 satisfies (10). Remark 2.1. Choosing ?i = (ci + Li + 2?di) , ?i = ci /?max Next, we quantify the suboptimality and infeasibility of the DPDA-S iterate sequence. Theorem 2.2. Suppose Assumption 1.1 holds. Let {xk , ? k , ?k }k?0 be the sequence generated by Algorithm DPDA-S, displayed in Fig. 1, initialized from an arbitrary x0 and ? 0 = 0. Let step-sizes {?i , ?i }i?N and ? be chosen satisfying (10) with strict inequality. Then {xk , ? k , ?k }k?0 converges to {x? , ? ? , ?? }, a saddle point of (8) such that x? = 1 ? x? and (x? , ? ?) is a primal-dual optimal solution to (3); moreover, the following error bounds hold for all K ? 1: ?K k + k?? k kM x

X

i?N

?K k?i? k dKi (Ai x i ? bi) ? ?1 /K,

—?(? xK) ? ?(x?)— ? ?1 /K,

h i

2 P ?K , where ?1 , ?2 k?? k2 ? ?2 M x0 + i?N 2?1i kx?i ? x0i k2 + ?4i k?i? k2 , and x

3

1 K

PK

k=1

xk .

Dynamic Network Topology

In this section we develop a distributed primal-dual algorithm for solving (3) when the communication network topology is time-varying. We assume a compact domain, i.e., let Di , maxxi ,x?i ?dom ?i kx ? x? k and B , maxi?N Di ¡ ?. Let C be the set of consensus decisions: C , {x ? Rn—N — : xi = x ?, ?i ? N for some x ? ? Rn s.t. k? xk ? B},

then one can reformulate (3) in a decentralized way as follows: min max L(x, y) , x

y

X

i?N

?i (xi) + h?i , Ai xi ? bi i ? ?Ki (?i) + h?, xi ? ?C (?),

where y = [? ? ??]? such that ? ? Rn—N — , ? = [?i]i?N ? Rm , and m , 5

P

i?N

mi .

(11)

P Next, we consider the implementation of PDA in (5) to solve (11). Let ?(x) , i?N ?i (xi), and P h(y) , ?C (?) + i?N ?Ki (?i) + hbi , ?i i. Define the block-diagonal matrix A , diag([Ai]i?N) ? Rm?n—N — and T = [A? In—N —]? . Therefore, given the initial iterates x0 , ? 0 , ?0 and parameters ? ¿ 0, ?i , ?i ¿ 0 for i ? N , choosing Dx and Dy as defined in Definition 1, and setting ?x = ?y = 1, PDA iterations given in (5) take the following form: Starting from ?0 = ?0 , compute for i ? N xk+1 ? argmin ?i (xi) + h?f (xki), xi i + hAi xi ? bi , ?ik i + hxi , ?ki i + i x

? xki) ? bi , ?i i + ?ik+1 ? argmin ?Ki (?i) ? hAi (2xk+1 i ?i

?k+1 ? argmin ?C (?) ? h2xk+1 ? xk , ?i + ?

1 kxi ? xki k22 , 2?i

1 k?i ? ?ik k22 , 2?i

1 k? ? ?k k22 , 2?

(12a) (12b)

?k+1 ? ?k+1 .

(12c)

Using extended Moreau decomposition for proximal operators, ?k+1 can be written as 1 k? ? (?k + ?(2xk+1 ? xk))k2 = prox??C (?k + ?(2xk+1 ? xk)) 2? 1

= ?k + ?(2xk+1 ? xk) ? ? PC (13) ?k + 2xk+1 ? xk . ?

?k+1 = argmin ?C (?) + ?

B Let 1 ? R—N — be the vector all ones, B0 , {x ? Rn : kxk ? B}. Note PB0 (x) = x min{1, kxk }.

For any x = [xi]i?N ? Rn—N — , PC (x) can be computed as PC (x) = 1 ? p(x),

where p(x) , argmin ??B0

X

i?N

k? ? xi k2 = argmin k? ? ??B0

1 X xi k 2 . —N — i?N

(14)

Let B , {x : kxi k ? B, i ? N } = ?i?N B0 . Hence, we can write PC (x) = PB ((W ? In)x) where W , —N1 — 11? ? R—N —?—N — . Equivalently, P PC (x) = PB (1 ? p?(x)) , where p?(x) , —N1 — i?N xi . (15)

Although x-step and ?-step of the PDA implementation in (12) can be computed locally at each node, computing ?k+1 requires communication among the nodes. Indeed, evaluating the average operator p?(.) is not a simple operation in a decentralized computational setting which only allows for communication among neighbors. In order to overcome this issue, we will approximate p?(.) operator using multi-consensus steps, and analyze the resulting iterations as an inexact primal-dual algorithm. In [20], this idea has been exploited within a distributed primal algorithm for unconstrained consensus optimization problems. We define the consensus step as one time exchanging local variables among neighboring nodes ? the details of this operation will be discussed shortly. Since the connectivity network is dynamic, let G t = (N , E t) be the connectivity network at the time t-th consensus step is realized for t ? Z+ . We adopt the information exchange model in [21]. Assumption 3.1. Let V t ? R—N —?—N — be the weight matrix corresponding to G t = (N , E t) at the time of t-th consensus step and Nit , {j ? N : (i, j) ? E t or (j, i) ? E t }. Suppose for all t ? Z+ : (i) V t is doubly stochastic; (ii) there exists ? ? (0, 1) such that for i ? N , Vijt ? ? if j ? Nit , and Vijt = 0 if j ? / Nit ; (iii) G ? = (N , E ?) is connected where E ? , {(i, j) ? N ? N : t (i, j) ? E for infinitely many t ? Z+ }, and there exists Z ? T ? ¿ 1 such that if (i, j) ? E ? , then ? (i, j) ? E t ? E t+1 ? ... ? E t+T ?1 for all t ? 1. Lemma 3.1. [21] Let Assumption 3.1 holds, and W t,s = V t V t?1 ...V s+1 for t ? s + 1. Given s ? 0 the entries of W t,s converges to N1 as t ? ? with a geometric rate, i.e., for all i, j ? N , one

? ? ? ? has Wijt,s ? N1 ? ??t?s , where ? , 2(1+? ?T)/(1?? T), ? , (1?? T)1/T , and T? , (N ?1)T ? .

Consider the k-th iteration of PDA as shown in (12). Instead of computing ?k+1 exactly according to (13), we propose to approximate ?k+1 with the help of Lemma 3.1 and set ?k+1 to this approximation. In particular, let tk be the total number of consensus steps done before k-th iteration of PDA, and let qk ? 1 be the number of consensus steps within iteration k. For x = [xi]i?N , define Rk (x) , PB (W tk +qk ,tk ? In) x k

(16)

to approximate PC (x) in (13). Note that R (?) can be computed in a distributed fashion requiring qk communications with the neighbors for each node. Indeed, Rk (x) = [Rki (x)]i?N

such that Rki (x) , PB0 6

X

j?N

t +qk ,tk

Wijk

xj .

(17)

Moreover, the approximation error, Rk (x) ? PC (x), for any x can be bounded as in (18) due to non-expansivity of PB and using Lemma 3.1. From (15), we get for all i ? N , X t +q ,t

kRki (x) ? PB0 p?(x) k = kPB0 Wijk k k xj ? PB0 j?N

?k

X

t +q ,t Wijk k k

j?N

?

1 N
1 N
X
j?N
xj k
?
xj k ? N ??qk kxk .
(18)
Thus, (15) implies that kRk (x) ? PC (x)k ? N ??qk kxk. Next, to obtain an inexact variant of (12), we replace the exact computation in (12c) with the inexact iteration rule: ?k+1 ? ?k + ?(2xk+1 ? xk) ? ?Rk
1 k ? ?
+ 2xk+1 ? xk .
(19)
Thus, PDA iterations given in (12) can be computed inexactly, but in decentralized way for dynamic connectivity, via the node-specific computations as in Algorithm DPDA-D displayed in Fig. 2 below. Algorithm DPDA-D (x0 , ? 0 , ?, {?i , ?i }i?N , {qk }k?0)
Initialization: ?0i ? 0, i ? N Step k: (k ? 0)
k k , ri ? 1. xk+1 ? prox?i ?i xki ? ?i ?fi (xki) + A? i ?i + ? i i
k+1 k+1 k k ? PK?i ?i + ?i Ai (2xi ? xi) ? bi , i ? N 2. ?i 3. For ? = 1, . . .P , qk t +? 4. ri ? j?N tk +? ?{i} Vijk rj , i ? N
1 k ? ? i
+ 2xk+1 ? xki i
i?N
i
5. End For
6. ?k+1 ? ?ki + ?(2xk+1 ? xki) ? ?PB0 ri , i i
i?N
Figure 2: Distributed Primal Dual Algorithm for Dynamic G t (DPDA-D)
Next, we define the proximal error sequence {ek }k?1 as in (20), which will be used later for analyzing the convergence of Algorithm DPDA-D displayed in Fig. 2. ek+1 , PC
1 k ? ?
+ 2xk+1 ? xk ? Rk ?1 ?k + 2xk+1 ? xk ;
(20)
hence, ?k = ?k + ?ek for k ? 1 when (12c) is replaced with (19). In the rest, we assume ?0 = 0. The following observation will also be useful to prove error bounds for DPDA-D iterate sequence. For each i ? N , the definition of Rki in (17) implies that Rki (x) ? B0 for all x; hence, from (19), k?k+1 k ? k?ki + ?(2xk+1 ? xki)k + ?kRki i i
1 k ? ?
Thus, we trivially get the following bound on ?k :
+ 2xk+1 ? xk k ? k?ki k + 4?B.
? k?k k ? 4? N B k.
(21)

Moreover, for any ? and ? we have that
?C (?) = sup h?, xi + h? ? ?, xi ? ?C (?) + x?C
?
N B k? ? ?k.
(22)
Theorem 3.2. Suppose Assumption 1.1 holds. Starting from ?0 = 0, ? 0 = 0, and an arbitrary x0 , let {xk , ? k , ?k }?k?0 be the iterate sequence generated using Algorithm DPDA-D, displayed in p Fig. 2, using qk = k consensus steps at the k-th iteration for all k ? 1 for some rational p ? 1. Let primal-dual step-sizes {?i , ?i }i?N and ? be chosen such that the following holds: 1
?i
? Li ? ?
1 2 ¿ ?max (Ai), ?i
? i ? N.
(23)
Then {xk , ? k , ?k }k?0 converges to {x? , ? ? , ?? }, a saddle point of (11) such that x? = 1 ? x? and (x? , ? ?) is a primal-dual optimal solution to (3). Moreover, the following bounds hold for all K ? 1: k?? k dC? (? xK) +
X
i?N
?K k?i? k dKi (Ai x i ? bi) ?
?2 + ?3 (K) , K
—?(? xK) ? ?(x?)— ?
?2 + ?3 (K) , K
h i
!1 ? 0 ? 1 4 k ? 0 2 ? 2
x ? x ? + P , kx k? x , ? , 2k? k k? k+ ?x k + k 2 i i i i?N ?i k=1 ?i h ?
i ? P K k? k qk and ?3 (K) , 8N 2 B 2 ? 2?k 2 + ? + ?N B k . Moreover, supK?Z+ ?3 (K) ¡ ?; k=1 ? ?K
where x
hence,
,
1 K
PK
1 K ?3 (K)
1 = O(K).
7
Remark 3.1. Note that the suboptimality, infeasibility and consensus violation at the K-th iteration is O(?3 (K)/K), where ?3 (K) denotes the error accumulation due to approximation errors, and PK ?3 (K) can be bounded above for all K ? 1 as ?3 (K) ? R k=1 ?qk k 2 for some constant ? p ?p P? R ¿ 0. Since k=1 ? k k 2 ¡ ? for any p ? 1, if one chooses qk = k for k ? 1, then the total number of communications per node until the end of K-th iteration can be bounded above by PK 1+1/p). For large p, qk grow slowly, it makes the method more practical at the cost k=1 qk = O(K of longer convergence time due to increase in O(1) constant. Note that qk = (log(k))2 also works and it grows

very slowly. We assume agents know qk as a function of k at the beginning, hence, synchronicity can be achieved by simply counting local communications with each neighbor.

4

Numerical Section

We tested DPDA-S and DPDA-D on a primal linear SVM problem where the data is distributed among the computing nodes in N . For the static case, communication network G = (N , E) is a connected graph that is generated by randomly adding edges to a spanning tree, generated uniformly at random, until a desired algebraic connectivity is achieved. For the dynamic case, for each consensus round t ? 1, G t is generated as in the static case, and V t , I ? 1c ?t , where ?t is the Laplacian of G t , and the constant c ¿ dtmax . We ran DPDA-S and DPDA-D on the line and complete graphs as well to see the topology effect ? for the dynamic case when the topology is line, each G t is a random line graph. Let S , {1, 2, .., s} and D , {(x? , y?) ? Rn ? {?1, +1} : ? ? S} be a set of feature vector and label pairs. Suppose S is partitioned into Stest and Strain , i.e., the index sets for the test and training data; let {Si }i?N be a partition of Strain among the nodes N . Let w = [wi]i?N , b = [bi]i?N , and ? ? R—Strain — such that wi ? Rn and bi ? R for i ? N . Consider the following distributed SVM problem: min

w,b,?

n X 1 2

i?N

kwi k2 + C —N —

XX

i?N ??Si

?? :

y? (wiT x? + bi) ? 1 ? ?? , ?? ? 0, ? ? Si , i ? N , wi = wj , bi = bj (i, j) ? E

o

Similar to [3], {x? }??S is generated from two-dimensional multivariate Gaussian distribution with covariance matrix ? = [1, 0; 0, 2] and with mean vector either m1 = [?1, ?1]T or m2 = [1, 1]T with equal probability. The experiment was performed for C = 2, —N — ? = 10, s = 900 such that —Stest — = 600, —Si — = 30 for i ? N , i.e., —Strain — = 300, and qk = k. We ran DPDA-S and DPDA-D on line, random, and complete graphs, where the random graph is generated such that the algebraic connectivity is approximately 4. Relative suboptimality and relative consensus

violation, i.e., max(i,j)?E k[wi? bi]? ? [wj? bj]? k/ [w? ? b?] , and absolute feasibility violation are

plotted against iteration counter in Fig. 3, where [w? ? b?] denotes the optimal solution to the central problem. As expected, the convergence is slower when the connectivity of the graph is weaker. Furthermore, visual comparison between DPDA-S, local SVMs (for two nodes) and centralized SVM for the same training and test data sets is given in Fig. 4 and Fig. 5 in the appendix.

Figure 3: Static (top) and Dynamic (bottom) network topologies: line, random, and complete graphs

8

2 References

[1] Qing Ling and Zhi Tian. Decentralized sparse signal recovery for compressive sleeping wireless sensor networks. Signal Processing, IEEE Transactions on, 58(7):3816?3827, 2010. [2] Ioannis D Schizas, Alejandro Ribeiro, and Georgios B Giannakis. Consensus in ad hoc WSNs with noisy links - Part I: Distributed estimation of deterministic signals. Signal Processing, IEEE Transactions on, 56(1):350?364, 2008. [3] Pedro A Forero, Alfonso Cano, and Georgios B Giannakis. Consensus-based distributed support vector machines. The Journal of Machine Learning Research, 11:1663?1707, 2010. [4] Ryan McDonald, Keith Hall, and Gideon Mann. Distributed training strategies for the structured perceptron. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 456?464. Association for Computational Linguistics, 2010. [5] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. Knowledge and Data Engineering, IEEE Transactions on, 25(11):2483?2493, 2013. [6] Gonzalo Mateos, Juan Andr?es Bazerque, and Georgios B Giannakis. Distributed sparse linear regression. Signal Processing, IEEE Transactions on, 58(10):5262?5276, 2010. [7] Francis R Bach, Gert RG Lanckriet, and Michael I Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In Proceedings of the twenty-first international conference on Machine learning, page 6. ACM, 2004. [8] Angelia Nedi?c and Asuman Ozdaglar. Subgradient methods for saddle-point problems. Journal of optimization theory and applications, 142(1):205?228, 2009. [9] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision, 40(1):120?145, 2011. [10] Bingsheng He and Xiaoming Yuan. Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. SIAM Journal on Imaging Sciences, 5(1):119?149, 2012. [11] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal?dual algorithm. Mathematical Programming, 159(1):253?287, 2016. [12] Yunmei Chen, Guanghui Lan, and Yuyuan Ouyang. Optimal primal-dual methods for a class of saddle point problems. SIAM Journal on Optimization, 24(4):1779?1814, 2014. [13] A. Nedic and A. Ozdaglar. Convex Optimization in Signal Processing and Communications, chapter Cooperative Distributed Multi-agent Optimization, pages 340?385. Cambridge University Press, 2010. [14] A. Nedi?c. Distributed optimization. In Encyclopedia of Systems and Control, pages 1?12. Springer, 2014. [15] Tsung-Hui Chang, Angelia Nedic, and Anna Scaglione. Distributed constrained optimization by consensus-based primal-dual perturbation method. Automatic Control, IEEE Transactions on, 59(6):1524?1538, 2014. [16] David

Mateos-N?un? ez and Jorge Cort?es. Distributed subgradient methods for saddle-point problems. In 2015 54th IEEE Conference on Decision and Control (CDC), pages 5462?5467, Dec 2015. [17] Deming Yuan, Shengyuan Xu, and Huanyu Zhao. Distributed primal?dual subgradient method for multiagent optimization via consensus algorithms. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 41(6):1715?1724, 2011. [18] Angelia Nedi?c, Asuman Ozdaglar, and Pablo A Parrilo. Constrained consensus and optimization in multiagent networks. Automatic Control, IEEE Transactions on, 55(4):922?938, 2010. [19] Kunal Srivastava, Angelia Nedi?c, and Du?san M Stipanovi?c. Distributed constrained optimization over noisy networks. In Decision and Control (CDC), 2010 49th IEEE Conference on, pages 1945?1950. IEEE, 2010. [20] Albert I Chen and Asuman Ozdaglar. A fast distributed proximal-gradient method. In Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on, pages 601?608. IEEE, 2012. [21] Angelia Nedi?c and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. Automatic Control, IEEE Transactions on, 54(1):48?61, 2009. [22] Ralph Tyrell Rockafellar. Convex analysis. Princeton university press, 2015. [23] H. Robbins and D. Siegmund. Optimizing methods in statistics (Proc. Sympos., Ohio State Univ., Columbus, Ohio, 1971), chapter A convergence theorem for non negative almost supermartingales and some applications, pages 233 ? 257. New York: Academic Press, 1971.

9