Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Solving Systems of Linear Equations by Distributed Convex Optimization in the Presence of Stochastic Uncertainty ⋆

Jing Wang and Nicola Elia *,1

* *Dept. of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA (e-mail: jingwang@alumni.iastate.edu nelia@iastate.edu).*

Abstract: In this paper, we propose distributed optimization methods to solve systems of linear equations. We provide convergence analysis for both continuous and discrete time computation models based on linear systems theory. It is shown that the proposed computation approaches work for very general linear equations, scalable with data sets and can be implemented in distributed or parallel fashion. Furthermore, we show that the discrete time algorithm admits constant update step size in the presence of additive uncertainties. This robustness feature makes the approach computationally efficient and supplementary to the existing approaches to deal with uncertainties such as stochastic (sub-)gradient methods and sample averaging.

Keywords: Systems of linear equations, distributed optimization, additive uncertainties, noises, stochastic programming, distributed and parallel computation.

1. INTRODUCTION

The fast spread of networking applications has boosted the research interest towards developing computation algorithms that exhibit scalability in face of big data sets and can be implemented over individual nodes that are connected through communication links. In this paper, we will try to develop such algorithms for a fundamental problem that finds applications in many modern engineering and scientific disciplines, namely, solving systems of linear equations of the form:

$$Ax = b, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ and b is a vector in $\mathbb{R}^n$ [2]. We assume that the system (1) is consistent, i.e., it has at least one solution.

Solving a set of linear equations has a long history and the famous approach in early days is by Gaussian elimination. While this *direct* approach may be preferred for problems of small dimension, it may not be appropriate for problems with big size. In the latter case, considering computation and storage cost, it is desirable to use *iterative* methods that generate a sequence of points with one only depending on its predecessor, that converges to the solution asymptotically. Our focus in this paper is to develop (distributed) iterative computation algorithms for solving linear equations with large dimension and possibly sparse structure.

⋆ This research has been supported under NSF grant CNS-1239319.
[1] We would like to thank Prof. P.G. Voulgaris and Prof. S. Salapaka for introducing us to the problem and the stimulating discussions on alternative approaches.
[2] Our approach also works for more general rectangular matrices A. Here we assume A to be a square matrix since it is easy to elaborate our approach and represents an important class of problems.

It is difficult to provide a comprehensive account of the vast literature on iterative methods. Herein we discuss some of the well known methods. We refer the reader to Bertsekas and Tsitsiklis (1989) for the discussion of classical iterative approaches, like the Jacobi and Gauss-Seidel, Hestenes et al. (1952) for Conjugate Gradient (CG) method and Shental et al. (2008); Moallemi and Van Roy (2009) for message passing algorithms. Many methods, such as CG and the message passing algorithms, require A to be symmetric and positive definite, and may fail to converge if A is singular. The Jacobi algorithm works for diagonal dominant matrix and the Gauss-Seidel method requires A either to be positive definite and symmetric or diagonal dominant.

A large class of iterative algorithms focus on first order discrete time-invariant dynamics in the form of

$$x_{k+1} = x_k - \gamma G(Ax_k - b), \tag{2}$$

where γ is a positive scalar and G is an $n \times n$ matrix. For example, the Jacobi algorithm can be obtained from the above algorithm when G is diagonal with $g_{ii} = \frac{1}{a_{ii}}$. To ensure the above iteration converges when A is singular, some assumptions need to be imposed on matrix G and A, see, e.g., Dax (1990) and references therein. The choice of G based on those assumptions usually involves matrix inverse calculation which introduces additional computation and makes the distributed implementation of the algorithm even more difficult.

There are algorithms to solve the general A case, mostly based on variations of the CG, see, e.g., Hestenes et al. (1952) and Choi (2006) and reference therein. Those algorithms require different steps like preconditioning and (or) matrix transformations, and their applicability in networked distributed settings is questionable and not much investigated.

In this paper, differently from the above methods, our proposed algorithms are second order, which lead to important benefits, and are inspired by the early work on dynamic systems for solving saddle point problems Arrow et al (1958) and the more recent work Wang and Elia (2011) where it was shown that there is a natural feedback dynamic system for solving convex optimization problems with equality constraints.

Formulating the problem of solving linear equations as an unconstrained quadratic programming problem has been considered in the literature, see, e.g., Bertsekas and Tsitsiklis (1989) Section 3.2.1. This formulation usually leads to a first order system whose convergence strongly depends on the property of the matrix A, e.g., diagonal dominant, positive definite, etc. In contrast, we propose to solving the linear equations by solving an optimization problem with equality constraint where the equality constraint is exactly the linear equation.

This optimization problem formulation first seems uneconomical since it propose to solve a more complicated problem. However, there are several desirable features that distinguish this approach with existing methods, such as Jacobi, Gauss-Seidel, Richardson iterative methods, messages passing, etc. First of all, the approach can treat very general linear equations, the only requirement is that the linear equations has at least one solution. Thus, it can be implemented even when A is singular, non-symmetric, and not diagonal dominant. This feature makes the algorithm well suitable in situations when the matrix A is of big dimension and its properties (like singularity) is not able to be detected a prior. Secondly, as we show in the paper, the proposed algorithm is robust to additive uncertainties in that the state of the algorithm will not diverge with additive uncertainty even we use constant step size. This feature may enable us to compute the solution more efficiently than the first order stochastic gradient algorithm Robbins and S. Monro (1951), in which vanishing step size is usually used. Thirdly, the proposed algorithm can be implemented in a distributed way in that each node deals with one component of the solution and thus scalable to the problem size. Although the third feature may be shared by available algorithms, the combination of the three makes this approach unique and promising for real implementation.

The rest of the paper is organized as follows. In Section 2, we propose the continuous time system to solve (1) and analyze its convergence property by standard linear systems theory. In Section 3, we investigate the discrete time algorithm and provide convergence analysis in the absence and presence of additive noise. We discuss the issues related to distributed implementation of the algorithm in Section 4 and provide some numerical examples in Section 5. Finally, we conclude the paper in Section 6.

2. A CONTINUOUS-TIME OPTIMIZATION SYSTEM

To solve equations (1), we propose to solve instead the following convex optimization problem:

$$\begin{aligned} p^* &= \min_{x\in\mathbb{R}^n} \|x\|_2^2 \\ &s.t.\ Ax = b. \end{aligned} \tag{3}$$

Problem (3) is called the least norm problem. It seeks to find the one with least Euclidean norm among all vectors that satisfy (1). This problem has found many applications in engineering, for example, the optimal control problems in systems and control field. Clearly, solving this problem will solve (1). We want to point out that it is possible to choose other cost functions in (3). Here we choose the Euclidean norm since the cost function is completely decomposable with respect to its components x_i and thus facilitate distributed implementation and least norm problem itself is an important problem.

At first, our formulation may seem to be uneconomical and not very useful; as a conventional approach to solve problem (3) (and thus (1)) would be by performing one Newton step. However, this step requires solving a bigger set of KKT equations, than (1). However, as we show subsequently, this formulation would allow us to generate algorithms that have all the desirable features that are discussion in Section 1.

The classical formulation to solve linear equations using convex optimization techniques is to minimize the function $f(x) = \frac{1}{2}x^TAx - x'b$, see, e.g., Bertsekas and Tsitsiklis (1989) Section 3.2.1. However, the success of the gradient decent algorithm for the mentioned unconstrained optimization problem would require A to be positive definite (to ensure that the cost function is convex). By formulating the problem as an equality constrained optimization problem would not have this limitation and the only requirement is that the linear equation is consistent .

Following Wang and Elia (2011), we construct the Lagrangian of Problem (3) as

$$L(x,\nu) = x'x + \nu'(Ax - b).$$

The associated dual problem of (3) is given by:

$$\max_{\nu\in\mathbb{R}^n} \min_{x\in\mathbb{R}^n} L(x,\nu). \tag{4}$$

We propose the following continuous time system for solving (3) and hence (1).

$$\begin{aligned} \dot{x} &= -2x - A'\nu, \\ \dot{\nu} &= Ax - b. \end{aligned} \tag{5}$$

Let x^* be the solution to Problem (3) and ν^* is the solution to the dual problem (4). Since we assume the linear equations (5) is consistent, x^* always exists. The existence of ν^* can be followed from Boyd and Vandenberghe (2004) pp. 141 and Chap. 5. Now, we define $\tilde{x}(t) = x(t) - x^*$, $\tilde{\nu}(t) = \nu(t) - \nu^*$ and let $\mathbf{z}(t) = (\tilde{x}(t), \tilde{\nu}(t))$, the evolution of $\mathbf{z}$ then can be derived from (5) as $\dot{\mathbf{z}} = \mathcal{A}\mathbf{z}$ where

$$\mathcal{A} = \begin{pmatrix} -2I & -A' \\ A & 0 \end{pmatrix}$$

The block diagram of the dynamical system (5) is shown in figure 2. Since it is an LTI system, we can understand its convergence and disturbance rejection properties from linear systems theory.

The asymptotic convergence property of (5) can be summarized as follows.

Theorem 1. Consider system (5), for any initial values $x(0)$ and $\nu(0)$, we have $\lim_{t\to\infty} x(t) = x^*$ and $\lim_{t\to\infty} \nu(t) = \nu^*$. Furthermore, $\nu^* = \nu_1 + \nu_2$ with $\nu_1 \in \mathcal{R}(A)$ and ν_2 is the component of $\nu(0)$ that lies in the null

space of A^T. Finally, the asymptotic convergence rate is exponentially fast and lower bounded by 2, i.e.,

$$||\mathbf{z}(t)|| \geq ||\mathbf{z}(0)|| \exp(-2t).$$

The proof extends the one given in Wang and Elia (2011) to the case of A not full column rank and is based on linear system arguments.

Proof. Since (5) is an LTI system, we proceed to use linear system theory with some basics of linear algebra for the convergence analysis. Let the derivative of x and ν to be zero, we have that

$$\begin{aligned} -2x - A'\nu &= 0, \\ Ax - b &= 0. \end{aligned} \tag{6}$$

Note that there might be many ν^* satisfying the above equation, we can just pick an arbitrary one and form $\tilde{\nu}$.

Using Schur formula, we have

$$\det(\lambda I - \mathcal{A}) = \det((\lambda+2)I) \det\left(\lambda I + \frac{1}{\lambda+2} AA'\right)$$

$= \det(\lambda^2 + 2\lambda I + AA')$. Let $AA' = U\Sigma^2 U'$, with U unitary, then $\det(\lambda I - \mathcal{A}) = \det(\lambda^2 + 2\lambda I + \Sigma^2) = \prod_{i=1}^{n}(\lambda^2 + 2\lambda + \sigma_i^2)$. When A is non-singular, the eigenvalues of AA' are all positive real numbers and $\mathcal{A}$ is Hurwitz. Then, the KKT conditions only have unique solution. This implies the convergence of $x(t)$ and $\nu(t)$ to the optimal primal and dual solutions..

We next consider the case when A is singular. In that case, the eigenvalues of AA' are all positive or zero. Thus, all the eigenvalues of $\mathcal{A}$ are on the strict left half plane or the origin, and the algebraic multiplicity of the zero eigenvalue of $\mathcal{A}$ is equal to that of AA'. To understand convergence of the system, we further investigate the structure property of null space of $\mathcal{A}$.

We next show that any right and left eigenvector e_r and e_l associated with the zero eigenvalue of $\mathcal{A}$ has the structure $[0_{1\times n}, e_{r2}']'$ and $[0_{1\times n}, e_{l2}']$ where e_{r2} is in the null space of A' and e_{l2} is such that $e_{l2}'A = 0$, and $0_{1\times n}$ denotes a row vector of all zeros of dimension n.

Let $e_r = [e_{r1}, e_{r2}]$ be an arbitrary right eigenvector associated with the zero eigenvalue of $\mathcal{A}$, then from $\mathcal{A}e_r = 0$, we have

$$-2e_{r1} = A'e_{r2}, \quad Ae_{r1} = 0.$$

Therefore, $e_{r1}'e_{r1} = -\frac{1}{2}e_{r2}'Ae_{r1} = 0$, which implies $e_{r1} = 0$. Furthermore $A'e_{r2} = 0$, which implies that e_{r2} is in the null space of A' and the dimension of the null space of $\mathcal{A}$ and A' are the same. This conclusion follows similarly for the structure of the left eigenvector of $\mathcal{A}$.

Now the algebraic multiplicity of the zero eigenvalue of $\mathcal{A}$ is equal to that of AA', which is also equal to the geometric multiplicity of AA' since AA' is diagonalizable. Since AA' and A' share the same null space, and the geometric multiplicity of the zero eigenvalue of $\mathcal{A}$ is equal to the null space of A' from previous argument, the zero eigenvalues of $\mathcal{A}$ have the same algebraic and geometric multiplicity. Thus, the mode decomposition from linear systems theory yields that

$$\lim_{t\to\infty} \mathbf{z}(t) = \lim_{t\to\infty} \mathbf{e}^{-\mathcal{A}t}\mathbf{z}(0) = \sum_{i=1}^{\kappa} e_{r,i}e_{l,i}'\mathbf{z}(0),$$

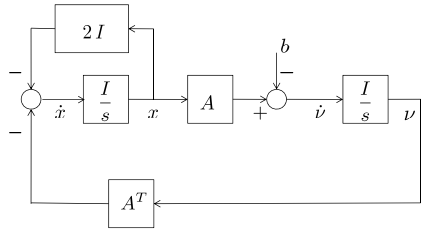

Fig. 1. The block diagram of the LTI system (5) for solving system of linear equations (1).

where κ is the dimension of the null space of $\mathcal{A}$. From the structure of $e_{r,i}$ and $e_{l,i}$, $x(t)$ converges to the solution of Problem (3) and $\nu(t)$ converges to some constant value depending only on the initial value of $\nu(t)$.

For the convergence rate analysis, we only consider the case when $\mathcal{A}$ is positive definite (i.e., A is non-singular). The more general case follows by projecting the dynamics of $\mathbf{z}(t)$ to the non-null space of $\mathcal{A}$. Let $V(t) = \frac{1}{2}||\mathbf{z}(t)||_2^2$, then

$$\begin{aligned} \dot{V}(t) &= \mathbf{z}(t)^T \mathcal{A}\mathbf{z}(t) = \mathbf{z}(t)^T \left(\frac{\mathcal{A}+\mathcal{A}^T}{2}\right)\mathbf{z}(t) \\ &= -2||\tilde{x}(t)||^2 \geq -2||\mathbf{z}(t)||^2 \end{aligned}$$

The convergence rate result thus follows. □

The above proof shows that the system will converge to the optimal solution whenever A is singular or not. When A is singular, the solutions to the dual problem is not unique and there are many ν^* that satisfy the KKT condition. In that case, the convergence value of $\nu(t)$ will depend on its initial value $\nu(0)$. Moreover, we provide an lower bound of the convergence rate that only depends on the cost function (through the scalar 2 as we define the cost function as $||x||^2$) and not the matrix A.

3. DISCRETE ALGORITHM AND ROBUSTNESS TO ADDITIVE UNCERTAINTIES

In this section, we will focus on discrete algorithm and its robustness to additive uncertainties. The analysis of the continuous time system provides many insights on the properties of the system that can be carry over for analysis of its discrete time counterpart.

We can obtain a simple discrete algorithm from Euler discretization of (5) as follows:

$$\begin{aligned} x(k+1) &= (1-2\gamma)x(k) - \gamma A'\nu(k), \\ \nu(k+1) &= \nu(k) - \gamma(Ax(k) - b), \end{aligned} \tag{7}$$

where $\gamma > 0$ is some positive constant. Define $\tilde{x}(k) = x(k) - x^*$, $\tilde{\nu}(k) = \nu(k) - \nu^*$ and let $\mathbf{z}(k) = (\tilde{x}(k)), \tilde{\nu}(k))$, (7) can be written compactly as $\mathbf{z}(k+1) = W\mathbf{z}(k)$, where

$$W = \begin{pmatrix} I - 2\gamma & -\gamma A' \\ \gamma A & I \end{pmatrix}$$

The convergence result of (7) can be summarized as follows.

Theorem 2. Let the nonzero eigenvalues of AA' be ordered as $\sigma_{min}^2, \ldots, \sigma_{max}^2$. Consider discrete algorithm (7), for any

initial values $x(0)$ and $\nu(0)$, if $0 < \gamma < \min\{1, \frac{2}{\sigma_{max}^2}\}$, we have $\lim_{k\to\infty} x(k) = x^*$ and $\lim_{k\to\infty} \nu(k) = \nu^*$. Furthermore, the asymptotic convergence rate is given by

$$\sup_{\mathbf{z}(0)\neq 0} \lim_{k\to\infty} \left(\frac{||\mathbf{z}(k)||_2}{||\mathbf{z}(0)||_2}\right)^{\frac{1}{k}} =$$

$$\max\left\{\sqrt{1-2\gamma+\gamma^2\sigma_{max}^2}, 1-\gamma\big(1-\sqrt{1-\min\{1,\sigma_{min}^2\}}\big)\right\}$$

Proof. We analyze the system by investigate the characteristic function of W. We only analyze the case when A is non-singular. When A is singular, the proof can be obtained with similar arguments in the proof of Theorem 1. When A is non-singular, the eigenvalues of AA' are all positive, and We have

$$\begin{aligned}\det(\lambda I - W) &= \det((\lambda I - I)(\lambda I - I + 2\gamma I) + \gamma^2 AA') \\ &= \Pi_{i=1}^n \left[(\lambda-1)(\lambda-1+2\gamma) + \gamma^2\sigma_i^2\right]\end{aligned}$$

Let $h_i(\lambda) = \lambda^2 - 2(1-\gamma)\lambda + \gamma^2\sigma_i^2 + 1 - 2\gamma$. If the roots of $h_i(\lambda)$ are complex conjugates, then their magnitude is equal to $\gamma^2\sigma_i^2 + 1 - 2\gamma$. Since $0 < \gamma < 2/\sigma_i^2$, $\gamma^2\sigma_i^2 + 1 - 2\gamma < 1$. When the roots of $h_i(\lambda)$ are real, they are given by $1-\gamma \pm \gamma\sqrt{(1-\sigma_i^2)}$. In this case, we have $0 < \sigma_i^2 < 1$. Since $0 < \gamma < 1$, $|1-\gamma \pm \gamma\sqrt{1-\sigma_i^2}| < 1$. Thus, we have proved that if $0 < \gamma < \min\{1, \frac{2}{\sigma_{max}^2}\}$, all the roots of W have magnitude less than 1 and the algorithm will converge.

To proceed the asymptotic convergence rate analysis, we note that if $\sigma_i^2 > 1$, then the roots of $h_i(\lambda)$ are complex conjugates, and the magnitude of the roots is $\sqrt{1-2\gamma+\gamma^2\sigma_i^2}$. If $\sigma_i < 1$, the maximal magnitude of the two real roots is $1-\gamma\big(1-\sqrt{1-\min\{1,\sigma_{min}^2\}}\big)$. Thus, the convergence rate result follows from the well know fact that for any real matrix B, $\rho(B) = \lim_{k\to\infty} ||\mathbf{A}^k||^{\frac{1}{k}}$, see, e.g., Corollary 5.6.14 of Horn and Johnson (1985). □

Theorem 2 states that the step size γ should be upper bounded by the minimum of 1 and the inverse of the largest eigenvalue of AA'. This result is in consistent with the classical Richardson iterative method Richardson (1910) in which the step size should be inversely scaled with the spectral radius of matrix A to ensure convergence. Note that our algorithm does not require A to be positive definite as in Richardson method, at the expense of additional computation of the dual variable. However, the justification of using the dual variable update does not only attributed to its ability to tackle more general problems, but also to the robustness feature that are brought by its feedback control nature, which we will further clarify in the next subsection.

3.1 Analysis of Discrete Time Algorithm under Additive Uncertainties

In the applications which requires solving linear system equations, the measurement b may not be exactly known as it is often measured and thus corrupted by additive noise. Communication noise may also be present in geographically distributed implementation of the algorithm. Motivated from those considerations, we next analyze the algorithm under additive uncertainties. We consider the following algorithm:

$$\begin{aligned}x(k+1) &= (1-2\gamma)x(k) - \gamma A'\nu + w(k), \\ \nu(k+1) &= \nu(k) - \gamma(Ax(k) - b) + v(k),\end{aligned} \tag{8}$$

where we assume each component $w_i(k)$ and $v_i(k)$ for all $k = 0, 1, \ldots$ are i.i.d distributed random variables with zero mean and bounded variance. Here, for each time index k, all the components $w_i(k)$ and $v_i(k)$ may be correlated to each other, but we assume all of them are independent of the initial condition $x(0)$ and $\nu(0)$. The stochastic convergence property of (8) can be summarized as follows.

Theorem 3. Consider discrete algorithm (8), for any initial values $x(0)$ and $\nu(0)$, if $0 < \gamma < \min\{1, \frac{2}{\sigma_{max}^2}\}$, we have $\lim_{k\to\infty} \mathbf{E}x(k) = x^*$, $\lim_{k\to\infty} \mathbf{E}\nu(k) = \nu^*$. Furthermore, $\lim_{k\to\infty} \mathbf{E}(\tilde{x}(k)\tilde{x}'(k))$ is bounded and converges to a fixed value, $\lim_{k\to\infty} \mathbf{E}(\tilde{\nu}(k)\tilde{\nu}'(k))$ is bounded and converges to a fixed value when A is non-singular and diverges to infinity when A is singular.

Remark 4. A similar result can be derived for the continuous time system (5) with additive noise. The continuous time setting can be used to explain how simple dynamical systems can solve (approximately) linear systems of equations collectively in the presence of noisy interconnections. The above resilience to additive noise is consistent with our previous results in Wang and Elia (2010, 2011, 2013).

Theorem 3 shows a remarkable robustness feature of the algorithm to additive uncertainties. The mean of $x(k)$ will converge to the optimal solution and the covariance of $\tilde{x}$ will be bounded and converging whenever A is singular or not, although the covariance of $\tilde{\nu}$ will diverge when A is singular[3]. This property will allow one to approximately compute the solution of the linear equations by just time averaging of $x(k)$. To see why this practical method works, we need the following result. For simplicity of exposition, we assume that A is not singular.

Compared to the first order algorithm (2), system (7) has two states in which $\nu(k)$ can be interpreted as the state of a dynamic feedback controller. In this way, (2) can be viewed as a feedback control system. Since feedback control can reduce the effect of uncertainties on system performance, it is understandable that the noise effect can now be mitigated. From linear systems theory, when A is singular, the eigenvectors associated with the marginal stable modes are in the null space of the transpose of matrix $C = [I_n, 0_{n\times n}]$[4]. Therefore, the marginal stable mode is unobservable from $\tilde{x}$ and any external bounded excitation can not destabilize it.

Lemma 5. Let $X_n = \frac{x(0)+x(1)+\ldots+x(n-1)}{n}$, if $0 < \gamma < \min\left\{1, \frac{2}{\sigma_{max}^2}\right\}$, we have

$$\lim_{n\to\infty} \mathbf{E}||X_n - x^*||^2 = 0.$$

Remark 6. Since $x(k)$ are not i.i.d., and the mean of each $x(k)$ is not equal to that of X_n, we do not have a version of weak law of large numbers. However, the above result provides a practical way of computing the

[3] There has been some remedies to prevent the divergence of $\nu(k)$ by ceasing the update of $\nu_i(k)$, see, e.g., Wang and Elia (2013).

[4] Assuming $\tilde{x}$ is the output, then $C\mathbf{z}(k)$ defines the output equation.

optimal solution by using X_n since we have proved that the variance between X_n and x^* converges to zero. In practice, the sample average of $x(k)$ can be taken beginning at some $k > 0$ for which the algorithm almost "converge". In this way, the initial generated $x(k)$s that are far from the convergence values are removed and the number of sampling points for the same approximation accuracy would be reduced.

Remark 7. We note that our approach is fundamentally different from the widely used stochastic gradient algorithms in which one need to use diminishing step size to attenuate the noise, see, e.g., Robbins and S. Monro (1951) and the sample average approximation techniques Shapiro (2003). In contrast, the algorithm we proposed converge exponentially fast, and the convergence value can be approximated with less iterations.

The similar properties of primal-dual like approaches to additive noise has been illustrated in our early work Wang and Elia (2010), where we try to solve the unconstrained distributed convex optimization problem. Since any convex cost function can be decomposed into the summation of two or more convex functions, that paper essentially demonstrated that we can use ideas from consensus and convex optimization to solve any unconstrained quadratic programming problems with additive uncertainties. Here, although we do not use the idea from dynamic consensus, the proposed second order system still exhibits the same property.

4. DISTRIBUTED SOLUTION TO SYSTEMS OF LINEAR EQUATIONS

In this section, we show how the optimization system (5) can solve systems of linear equations in a distributed or parallel fashion. The main idea also works similarly for discrete the time algorithm. We omit the presence of noise for simplicity of explanation. Note however that noise can be present in the communication links between the nodes. The resilience to noise of the distributed implementation follows directly from the results of the previous section.

The operations of system (5) can be described by a graphical model. Considering a bipartite graph, where on one side the nodes are associated with the primal variables x_j, $j = 1, \dots, n$ and on the other side, the nodes are associated with the dual variables ν_i $i = 1, \dots, n$.

Each node ν_i has a simple integrator dynamics

$$\dot{\nu}_i = -b_i + v_i$$

where v_i is the input. Each node x_j has a stable dynamics

$$\dot{x}_i = -2x_i - u_i$$

where u_i is the input. If $a_{ij} \neq 0$ there is a link of weight a_{ij} from x_j to ν_i, and a link of weight a_{ij} from node ν_i to node x_j. Thus the link between ν_i and x_j is undirected. The input to each node is the sum of messages on its incoming links, namely $v_i = \sum_{j=1}^n a_{ij} x_j$ and $u_j = \sum_{i=1}^n a_{ij} \nu_i$.

For example, consider

$$A = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 5 \\ 2 \\ -3 \\ -1 \end{bmatrix}. \tag{9}$$

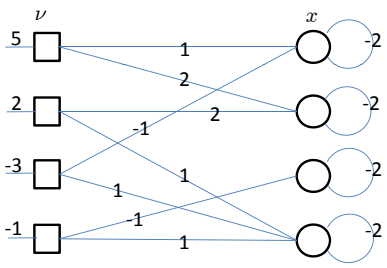

Fig. 2. Bipartite graph representation of system (5)

Figure 4 shows the corresponding bipartite graph describing the communication network intrinsic in the structure of A. Note that when A is sparse, only few links are present between the ν and the x nodes corresponding to the nonzero coefficients of A.

Alternatively, nodes x_i and ν_i can be collapsed into a second order node (x_i, ν_i). For the example above, the corresponding networked system is given by

$$\dot{\xi} = P\xi + N\xi - B$$

where $\xi_i = [x_i, \nu_i]$, and omitting the block of zeros,

$$P = \begin{bmatrix} -2 & -1 & & & & & & \\ 1 & 0 & & & & & & \\ & & -2 & -2 & & & & \\ & & 2 & 0 & & & & \\ & & & & -2 & 0 & & \\ & & & & 0 & 0 & & \\ & & & & & & -2 & -1 \\ & & & & & & 1 & 0 \end{bmatrix}$$

$$N = \begin{bmatrix} & & 0 & 0 & 0 & 1 & & \\ & & 2 & 0 & 0 & 0 & & \\ 0 & -2 & & & & & 0 & 0 \\ 0 & 0 & & & & & 1 & 0 \\ 0 & 0 & & & & & 0 & 1 \\ -1 & 0 & & & & & 1 & 0 \\ & & 0 & -1 & 0 & -1 & & \\ & & 0 & 0 & -1 & 0 & & \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 5 \\ 0 \\ 2 \\ 0 \\ -3 \\ 0 \\ -1 \end{bmatrix}$$

The derivation we have proposed in this section is general and suggests that there is an ad-hoc network architecture, which is implicit in the structure of A and allows for a distributed solution of systems of equations.[5] On the other hand, network topologies that work for large sets of A's are of interest. For example, the network topology consistent with a certain zero sparsity pattern of A, can solve any feasible problem where A has that sparsity pattern. A relevant special class consists of matrices with symmetric sparsity structures. In this case ν_i and x_i have the same set of in-neighbors. Thus the same topology is used for updating both primal and dual variables. Furthermore, when A is symmetric, ν_i and x_i have the same set of in-neighbors with the same set of weights. This extra structure in A further simplifies the required network infrastructure to perform the distributed computation of solutions.

We want to remark that each node only computes one component of the solution vector. This is fine in most situations where x_i corresponds to physical variables the agents needs to assume (e.g. position) as a solution of the

[5] This approach also applies to other convex optimization problems with separable cost functions.

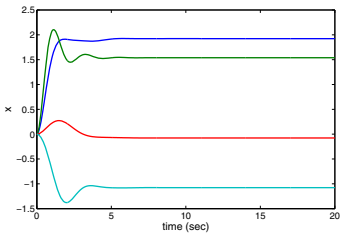

Fig. 3. Convergence of the states $x(t)$ of the continuous time algorithm (5) to the solution of $Ax = b$ with minimum 2 norm. A is singular and non-symmetric.

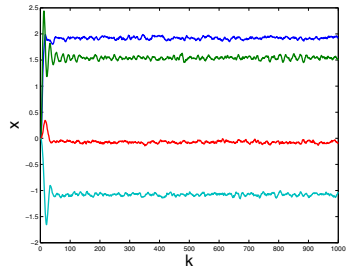

Fig. 4. Trajectory of the states $x(k)$ in the presence of noise for algorithm (8). Note that $x(k)$ does not diverge in the presence of additive noise, although we use fixed step size $\gamma = 0.1$.

problem at hand. If one is interested in the whole solution vector, a global collector of the x_i's is necessary. This would be the case of distributed solvers running on chips or multi processor architectures.

5. EXAMPLE

In this section, we use an example to illustrate the effectiveness of our algorithms. We consider the linear equations with data given by (9). Recall that here A is *not symmetric* and *singular*. We first consider continuous time optimization system (5). Figure 3 shows the convergence to a feasible solution, $x^* = [1.9231\ 1.5385\ -0.0769\ -1.0769]'$, with the $2-$norm value $||x^*||_2 = 2.689$.

Figure 4 shows the convergence of the discrete-time system (8) in the presence of noise. In this case, the maximum eigenvalue of $A'A$ is $\sigma^2_{\max} = 9.0732$, thus we choose $\gamma = 0.1$. Each difference equation is subject it IID Gaussian noise with zero mean and variance 0.01. As the noise continues to excite the system, the primal and dual variable are subject to variations as described by Theorem 3. This noise can be averaged out by each agent using a moving averaging. The average state $\bar{x} = [1.9198\ 1.5414\ -0.0733\ -1.0766]'$, is obtained by averaging the last 500 samples of the response.

6. CONCLUSIONS

In this paper, we have proposed a novel convex optimization problem formulation to solve linear equations. This new formulation has led to the application of primal dual like approaches to solve systems of linear equations. This seemingly uneconomical approach works for very general matrix A, robust to additive uncertainties, scalable to problem data and is easy to be implemented in distributed way. Since in real applications, it is difficult to evaluate if A is singular or not when A has a large dimension, we hope our approach can alleviate this difficulty and still allow distributed implementation. Moreover, the robustness feature of the algorithm offers new approach to tackle stochastic optimization problems besides the application to linear equations.

REFERENCES

D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

M. R. Hestenes and E. Stiefel, "Methods of Conjugate Gradients for Solving Linear Systems," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, pp.410-436, 1952.

O. Shental, P. H. Siegel, J. K. Wolf, D. Bickson and D. Dolev, "Gaussian belief propagation solver for systems of linear equations," *IEEE Symposium on Information Theory*, pp. 1863-1867, 2008.

C. C. Moallemi and B. Van Roy, "Convergence of Min-Sum Message Passing for Quadratic Optimization," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2413-2423, 2009.

A. Dax, "The Convergence of Linear Stationary Iterative Processes for Solving Singular Unstructured Systems of Linear Equations", SIAM Review, vol. 32, no. 4, pp. 611-635, 1990.

S.-C. Choi, *Iterative methods for singular linear equations and least-squares problems*, Ph.D. thesis, Institute for Computational and Mathematical Engineering, Stanford University, Dec. 2006.

K. J. Arrow, L. Hurwicz and H. Uzawa, *Studies in Linear and Nonlinear Programming,* Stanford University Press, Stanford, 1958.

J. Wang and N. Elia, "A control perspective for centralized and distributed convex optimization," *in IEEE Conference on Decision and Control and European Control Conference*, pp. 3800-3805, 2011.

S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambrige Univ. Press, 2004.

H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statistics*, vol. 22, pp.400-407, 1951.

A. Shapiro, "Monto Carlo Sampling methods," in *Handbook in Operations Research and Management Science*. Amsterdam: Elsevier Science, 2003, vol. 10, pp. 353-426.

J. Wang and N. Elia, "Control approach to distributed convex optimization, " in *Allerton Conference in Communication, Control and Computation*, pp. 557-561, 2010.

J. Wang and N. Elia, "Distributed averaging algorithms resilient to communication noise and dropouts," *IEEE Transaction on Signal Processing*, vol. 61, no. 9, pp. 2231-2242, 2013.

F. L. Richardson, "The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam," *Philosophical Transactions of the Royal Society A*, Issue, 210; pp. 307-357.

Horn R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge.