

Distributed Stochastic Multicommodity Flow Optimization

Nikolaos Chatzipanagiotis and Michael M. Zavlanos

Abstract—In this paper we are concerned with a class of stochastic multicommodity network flow problems, the so called capacity expansion planning problems. We consider a two-stage stochastic optimization formulation that incorporates uncertainty in the problem parameters. To address the computational complexity of these stochastic models, we propose a decomposition method to divide the original problem into smaller, tractable subproblems that are solved in parallel at the network nodes. Unlike relevant techniques in existing literature that decompose the problem with respect to the possible realizations of the random parameters, our approach can be applied to networked systems that lack a central processing unit and require autonomous decision making by the network nodes. Our method relies on the recently proposed Accelerated Distributed Augmented Lagrangians (ADAL) algorithm, a dual decomposition technique with regularization, which achieves very fast convergence rates.

I. INTRODUCTION

Multicommodity flow problems (MFP) [1]–[3] emerge in cases where several different commodities need to travel from origins to specific destinations along the arcs of an underlying network, subject to supply restrictions, arc capacity restrictions, and flow conservation conditions. Typical areas of MFP applications involve transportation, communication and data networks, and product distribution and logistics systems. In most of these real-world applications, the problem parameters, such as the flow demands and the arc travel attributes, are random, evolve over time, or get affected by disturbances. Hence, it is desirable to develop stochastic formulations in which decisions are evaluated against a variety of future scenarios that represent alternative outcomes of the MFP’s parameters. The most common approach involves representing the uncertain quantities as random variables [4].

In this paper, we focus on the two-stage stochastic model for nonlinear network capacity expansion problems [5]–[8], which are a popular, specialized class within the family of MFPs. Compared to multi-stage models, two-stage models simplify the stochastic optimization problem and reduce its size by assuming that all uncertain parameters are revealed at the same time, following the selection of first-stage decisions [4]. However, even for two-stage models, the emerging problem size is still enormous for most real-world applications. This motivates the use of decomposition methods as the only effective alternative to address the problem complexity [9]–[15]. These methods render the problem tractable typically by decomposing it over the set of realizations of the random variables, also called *scenarios*. Instead, in this paper we propose a new distributed optimization technique that decomposes the network problem with respect to the nodes, so that every node

computes only the capacity and flow variables that are local to itself. The advantage of our method is that it can be widely applied to problems in distributed communications, sensor, and robotic networks, where the nodes operate independently and autonomously in the absence of a central processing unit. Our method retains the benefit of dividing the large centralized problem into tractable, smaller subproblems.

Our proposed method utilizes *augmented Lagrangians* (AL), a regularization technique that relies on adding a quadratic penalty term to the ordinary Lagrangian [16]. AL methods converge very fast and do not require strict convexity of the objective function, as is the case in the classical *dual decomposition* method that relies on the ordinary Lagrangian [16]. However, they lack the decomposability properties of the ordinary Lagrangian, which calls for the development of specialized AL decomposition techniques, such as the Alternating Directions Method of Multipliers (ADMM) [17, 18] and the Diagonal Quadratic Approximation (DQA) method [19]. In this paper, we utilize the recently proposed *Accelerated Distributed Augmented Lagrangians* (ADAL) algorithm [20]. ADAL is a fully distributed optimization algorithm with very fast convergence, as suggested by numerical analysis. We compare our method to the aforementioned popular state-of-the-art distributed algorithms and show that we obtain non-negligible performance gains.

II. PROBLEM FORMULATION

In this paper, we specifically consider the two-stage stochastic nonlinear network capacity expansion problem described in [15, Example 4, p.14]. At the first stage, capacities are allocated to the arcs of the network, followed by the observation of random demands for network traffic. Then, at the second stage, a flow plan is determined utilizing the available network capacity so that the observed demand is satisfied. The objective is to minimize the combined cost of capacity allocations and the expected cost of all possible optimal flow plans.

Specifically, consider a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, with node set \mathcal{N} and arc set \mathcal{A} . Denote the directed arc from node i to node j by (i, j) . The capacity of each arc $(i, j) \in \mathcal{A}$ is a first-stage decision variable designated by x_{ij} . The cost of installing capacity on arc (i, j) is denoted by $c_{ij}(x_{ij})$. For each pair of nodes $(m, n) \in \mathcal{N} \times \mathcal{N}$, we require a random commodity flow d^{mn} , which is defined as the traffic that must be sent from node m to node n . We denote the flow from m to n sent through arc (i, j) by y_{ij}^{mn} , which is part of the second stage decisions. The cost of routing a flow y_{ij}^{mn} through arc (i, j) is denoted by $q_{ij}(y_{ij}^{mn})$. Moreover, let $\mathcal{A}_i^- \subseteq \mathcal{A}$ and $\mathcal{A}_i^+ \subseteq \mathcal{A}$ denote the sets of incoming and outgoing arcs for node $i \in \mathcal{N}$, respectively. Then, the second stage problem is

This work is partially supported by the NSF award CNS#1261828.

Nikolaos Chatzipanagiotis and Michael M. Zavlanos are with the Dept. of Mechanical Engineering and Materials Science, Duke University, Durham, NC, 27708, USA {n.chatzip, michael.zavlanos}@duke.edu.

the following multicommodity network flow problem

$$\begin{aligned} \min_{y_{ij}} \quad & \sum_{m,n \in \mathcal{N}} \sum_{\{(i,j) \in \mathcal{A}\}} q_{ij}(y_{ij}^{mn}) \\ \text{s.t.} \quad & \sum_{m,n \in \mathcal{N}} y_{ij}^{mn} \leq x_{ij}, \quad \forall i, j : (i, j) \in \mathcal{A}, \end{aligned} \quad (1a)$$

$$y_{ij}^{mn} \geq 0, \quad \forall i, j : (i, j) \in \mathcal{A}, m, n \in \mathcal{N}, \quad (1b)$$

$$\sum_{\{j:(i,j) \in \mathcal{A}_i^+\}} y_{ij}^{mn} - \sum_{\{j:(j,i) \in \mathcal{A}_i^-\}} y_{ji}^{mn} = \begin{cases} d^{mn}, & \text{if } i = m, \\ -d^{mn}, & \text{if } i = n, \\ 0, & \text{otherwise,} \end{cases} \quad (1c)$$

where we require that the sum of all commodity flows on an arc does not exceed its allocated capacity (1a), impose nonnegativity constraints on the flows and capacities for each arc (1b), and include commodity flow balance constraints for each node (1c).

Denote the optimal value of (1) by $Q(\mathbf{x}, \mathbf{d})$, where $\mathbf{x} \in \mathbb{R}^{|\mathcal{A}|}$ and $\mathbf{d} \in \mathbb{R}^{N(N-1)}$ are the vectors of all capacity allocations x_{ij} and demands d^{mn} , respectively, and N is the cardinality of the node set \mathcal{N} , i.e. $|\mathcal{N}| = N$. Then, the first stage problem has the form $\min_{\mathbf{x} \geq 0} \sum_{\{(i,j) \in \mathcal{A}\}} c_{ij}(x_{ij}) + \mathbb{E}[Q(\mathbf{x}, \mathbf{d})]$. We assume that all matrices and cost vectors are deterministic, except for the flow demand entries d^{mn} in the second stage equality constraints (1c), which are random. Note that exact knowledge of the probability distribution of the uncertain demand vector \mathbf{d} is not possible in most applications. Nevertheless, it is sufficient to know or at least have a good estimate of a finite number of possible realizations of \mathbf{d} , which are called *scenarios*. Then, if we denote the set of all available scenarios by \mathcal{S} , we have that $\mathbb{E}[Q(\mathbf{x}, \mathbf{d})] = \sum_{s \in \mathcal{S}} p_s Q(\mathbf{x}, \mathbf{d}_s)$, where \mathbf{d}_s denotes the specific realization of \mathbf{d} for scenario $s \in \mathcal{S}$ and p_s is its respective probability of occurring. Assuming that S scenarios are available, the two-stage stochastic nonlinear network capacity expansion problem can be formulated as the deterministic optimization problem

$$\begin{aligned} \min_{x_{ij}, y_{ij}} \quad & \left[\sum_{\{(i,j) \in \mathcal{A}\}} c_{ij}(x_{ij}) + \right. \\ & \left. \sum_{s \in \mathcal{S}} p_s \left(\sum_{m,n \in \mathcal{N}} \sum_{\{(i,j) \in \mathcal{A}\}} q_{ij}(y_{ij}^{mn}) \right) \right] \\ \text{s.t.} \quad & \sum_{m,n \in \mathcal{N}} y_{ij}^{s,mn} \leq x_{ij}, \quad \forall (i, j) \in \mathcal{A}, s \in \mathcal{S} \quad (2) \\ & y_{ij}^{s,mn} \geq 0, \quad \forall (i, j) \in \mathcal{A}, m, n \in \mathcal{N}, s \in \mathcal{S}, \\ & \sum_{\{j:(i,j) \in \mathcal{A}_i^+\}} y_{ij}^{s,mn} - \sum_{\{j:(j,i) \in \mathcal{A}_i^-\}} y_{ji}^{s,mn} = \begin{cases} d^{s,mn}, & \text{if } i = m, \\ -d^{s,mn}, & \text{if } i = n, \\ 0, & \text{otherwise,} \end{cases} \end{aligned}$$

where the superscript s now denotes the scenario index. Clearly, even if only few of the commodity demands are nonzero and they have discrete distributions, the size of the emerging optimization problem (2) is enormous for most real world applications.

III. DISTRIBUTED OPTIMIZATION USING AUGMENTED LAGRANGIANS

Existing techniques reduce the complexity of two-stage stochastic capacity expansion problems by decomposing the

original centralized problem (2) with respect to scenario bundles [11]–[15]. This approach still requires the presence of a central unit that will gather information from the network, solve the individual subproblems for the scenario bundles, and then send back the decision variables to the corresponding nodes. Instead, in this paper we present a novel method to decompose (2) into N subproblems that are solved in parallel at the node locations, using the recently proposed ADAL method [20]. The necessary coordination among subproblems is achieved using only information that is available within a local communication neighborhood of the corresponding nodes. The advantage of our proposal is that it retains the feature of reducing the problem's size, while at the same time it eliminates the need for a central coordinating and processing unit.

In order to develop the proposed distributed algorithm to solve (2), we first express (2) in an alternative, equivalent form. To this end, let \mathcal{C} be the set of commodities for a given problem, with cardinality $|\mathcal{C}| = C$, and, also, recall that \mathcal{N} and \mathcal{S} denote the node and scenario sets with cardinalities N and S , respectively. Moreover, define $\mathbf{d}^{s,c} \in \mathbb{R}^N$ to be the vector of flow demands for commodity $c \in \mathcal{C}$ at scenario $s \in \mathcal{S}$. Note that $\mathbf{d}^{s,c}$ has only two nonzero entries, corresponding to the nodes that constitute the source-sink pair for commodity c . Also, let $\mathbf{y}_i^{s,c} = [y_{i1}^{s,c}, y_{i2}^{s,c}, \dots, y_{iN}^{s,c}]^T \in \mathbb{R}^N$ be the vector of flows of commodity c that node i routes to all other nodes at scenario s , where $y_{ij}^{s,c}$ denotes the flow of commodity c that node i routes towards node j when scenario s is considered. Then, it is not hard to verify that we can define appropriate matrices $\mathbf{A}_i \in \mathbb{R}^{N \times N}$ for each node i , such that (2) can be equivalently written as

$$\begin{aligned} \min \quad & \left[\sum_{i \in \mathcal{N}} \sum_{\{j:(i,j) \in \mathcal{A}_i^+\}} c_{ij}(x_{ij}) + \right. \\ & \left. \sum_{s \in \mathcal{S}} p_s \left(\sum_{c \in \mathcal{C}} \sum_{i \in \mathcal{N}} \sum_{\{j:(i,j) \in \mathcal{A}_i^+\}} q_{ij}(y_{ij}^{s,c}) \right) \right] \\ \text{s.t.} \quad & \sum_{i \in \mathcal{N}} \mathbf{A}_i \mathbf{y}_i^{s,c} = \mathbf{d}^{s,c}, \quad \forall s \in \mathcal{S}, c \in \mathcal{C}, \\ & \sum_{c \in \mathcal{C}} y_{ij}^{s,c} \leq x_{ij}, \quad \forall (i, j) \in \mathcal{A}, s \in \mathcal{S}, \\ & y_{ij}^{s,c} \geq 0, \quad \forall (i, j) \in \mathcal{A}, s \in \mathcal{S}, c \in \mathcal{C}. \end{aligned} \quad (3)$$

We observe that (3) consists of: (i) a separable objective function with respect to the nodes, (ii) constraints that are local to every node i , whose feasible set we denote by $\mathcal{Z}_i = \{ \mathbf{x}_i \in \mathbb{R}^{|\mathcal{A}_i^+|}, \mathbf{y}_i \in \mathbb{R}^{|\mathcal{A}_i^+|SC} : \sum_{k \in \mathcal{K}} y_{ij}^{s,c} \leq x_{ij}, y_{ij}^{s,c} \geq 0, \forall j : (i, j) \in \mathcal{A}_i^+, s \in \mathcal{S}, c \in \mathcal{C} \}$, where $\mathbf{x}_i, \mathbf{y}_i$ are the vectors stacking the capacity and flow decision variables for node i , respectively, and (iii) SC affine, flow balance constraints $\sum_{i \in \mathcal{N}} \mathbf{A}_i \mathbf{y}_i^{s,c} = \mathbf{d}^{s,c}$. The latter constraint set introduces coupling between the flow decision variables of the different nodes and requires decomposition techniques to obtain efficient distributed solutions.

To address this challenge, in this paper, we employ distributed optimization methods with regularization, as they converge fast and can handle problems with objective functions that are not necessarily strictly convex, unlike standard dual decomposition methods. A widely used such regularization

technique is the *Method of Multipliers* (MoM), which utilizes the notion of an augmented Lagrangian, obtained by adding a quadratic penalty term to the ordinary Lagrangian [16]. The MoM is an excellent general purpose method for constrained optimization, however the quadratic penalty term introduces cross products between the decision variables, making distributed implementations challenging. For this reason, specialized AL decomposition algorithms for general convex problems have been developed in literature [17]–[19, 21]. The approach presented here is based on the ADAL algorithm [20], which exhibits very fast convergence. Application of ADAL leads to a fully distributed solution of (3), with guaranteed convergence to the optimal point, provided that the cost functions are convex.

A. Accelerated Distributed Augmented Lagrangians

ADAL is an iterative procedure that alternates between updating the primal and dual variables until convergence to the optimal solution is achieved. To apply ADAL to the problem under consideration, let k denote the iteration index and define $\lambda^k = [\lambda^{k,1,1}, \dots, \lambda^{k,1,C}, \lambda^{k,2,1}, \dots, \lambda^{k,S,C}]^T \in \mathbb{R}^{NSC}$ to be the vector of Lagrange multipliers at iteration k , associated with the coupling flow balance constraints in (3), where $\lambda^{k,s,c} = [\lambda_1^{k,s,c}, \dots, \lambda_N^{k,s,c}]^T \in \mathbb{R}^N$ for every scenario $s \in \mathcal{S}$ and commodity $c \in \mathcal{C}$. Let also

$$\begin{aligned} \Lambda_i(\mathbf{x}_i, \mathbf{y}_i, \mathbf{y}_j^k, \lambda^k) = & \sum_{j:(i,j) \in \mathcal{A}_i^+} \left[c_{ij}(x_{ij}) + \right. \\ & \left. \sum_{s \in \mathcal{S}} p_s \sum_{c \in \mathcal{C}} q_{ij}(y_{ij}^{s,c}) \right] + \sum_{s \in \mathcal{S}} p_s \sum_{c \in \mathcal{C}} (\lambda^{k,s,c})^T \mathbf{A}_i \mathbf{y}_i^{s,c} + \\ & \sum_{s \in \mathcal{S}} p_s \sum_{c \in \mathcal{C}} \frac{\rho}{2} \|\mathbf{A}_i \mathbf{y}_i^{s,c} + \sum_{j \neq i} \mathbf{A}_j \mathbf{y}_j^{k,s,c} - \mathbf{d}^{s,c}\|^2, \end{aligned} \quad (4)$$

denote the local AL of node i , where $\rho \in \mathbb{R}_+$ is a properly defined penalty coefficient, and \mathbf{y}_j^k denotes the vector of all flow decision variables of node j at iteration k that is communicated to node i , and is considered by node i as a vector of fixed parameters. Then, at every iteration k , ADAL begins with every node i minimizing its local AL (4), subject to the local constraints, i.e., $\hat{\mathbf{y}}_i^k = \arg \min_{\mathbf{x}_i, \mathbf{y}_i} \Lambda_i(\mathbf{x}_i, \mathbf{y}_i, \{\mathbf{y}_j^k\}_{j \in \mathcal{T}_i}, \{\lambda_l^k\}_{l \in \mathcal{H}_i \cup \{i\}})$. Observe that this optimization step requires only that node i has access to the, locally available, variables associated with its own flow balance constraint and those of its neighbors. More formally, if we define the neighborhood \mathcal{H}_i of node i as the set of all nodes j that share an arc with i , i.e. $\mathcal{H}_i = \{j \in \mathcal{N} : (i,j) \cup (j,i) \in \mathcal{A}\}$, then, it is not hard to verify that node i only needs access to the the Lagrange multipliers λ_l^k of its neighbors' flow constraints $l \in \mathcal{H}_i$ and the flow variables \mathbf{y}_j^k from the set $\mathcal{T}_i = \{j \in \mathcal{N} : \mathcal{H}_i \cap \mathcal{H}_j \neq \emptyset\}$ of its two-hop neighbors.

After calculating the minimizers $\hat{\mathbf{y}}_i^k$ of the local ALs (4), every node $i \in \mathcal{N}$ updates its primal variables \mathbf{y}_i^k (that will be communicated to its neighbors) according to $\mathbf{y}_i^{k+1} = \mathbf{y}_i^k + \tau(\hat{\mathbf{y}}_i^k - \mathbf{y}_i^k)$, where τ is a stepsize, that plays a critical role in the convergence of the ADAL method. Specifically,

Algorithm 1 Accelerated Distributed AL (ADAL)

Require: Set $k = 1$ and define initial Lagrange multipliers λ^1 and primal variables \mathbf{y}_i^1 for every $i \in \mathcal{N}$.

- 1: For fixed λ^k , \mathbf{y}_i^k calculate for all nodes $i \in \mathcal{N}$ the $\hat{\mathbf{y}}_i^k$ as the solution of

$$\begin{aligned} \hat{\mathbf{y}}_i^k = & \arg \min_{\mathbf{x}_i, \mathbf{y}_i} \Lambda_i(\mathbf{x}_i, \mathbf{y}_i, \{\mathbf{y}_j^k\}_{j \in \mathcal{T}_i}, \{\lambda_l^k\}_{l \in \mathcal{H}_i \cup \{i\}}) \\ \text{s.t. } & \mathbf{x}_i, \mathbf{y}_i \in \mathcal{Z}_i \end{aligned}$$

- 2: If $\sum_{i \in \mathcal{N}} \mathbf{A}_i \mathbf{y}_i^{s,c} = \mathbf{d}^{s,c}$ for all $s \in \mathcal{S}$ and $c \in \mathcal{C}$, then stop (optimal solution found). Otherwise, for every $i \in \mathcal{N}$ set

$$\mathbf{y}_i^{k+1} = \mathbf{y}_i^k + \tau(\hat{\mathbf{y}}_i^k - \mathbf{y}_i^k)$$

and communicate \mathbf{y}_i^{k+1} to every $j \in \mathcal{H}_i$.

- 3: For every $i \in \mathcal{N}$, $s \in \mathcal{S}$ and $c \in \mathcal{C}$ set

$$\begin{aligned} \lambda_i^{k+1,s,c} = & \lambda_i^{k+1,s,c} + \rho \tau \left(\sum_{j:(i,j) \in \mathcal{A}_i^+} y_{ij}^{k+1,s,c} - \right. \\ & \left. \sum_{j:(j,i) \in \mathcal{A}_i^-} y_{ji}^{k+1,s,c} - d_i^{s,c} \right), \end{aligned}$$

communicate λ_i^{k+1} to every $j \in \mathcal{H}_i$, increase k by 1 and go to Step 1.

convergence of ADAL is guaranteed if $0 \leq \tau \leq a/q$, where $a \in [1, 2)$ is a relaxation factor and $q = \max_{i \in \mathcal{N}} |\mathcal{H}_i|$ [20]. In other words, τ is determined by the density of the network and, in particular, by its maximum degree.

The final step of ADAL at iteration k involves the update of the dual variables $\lambda_i^{k+1,s,c}$, corresponding to the flow balance constraint of commodity c at node i for scenario s , according to $\lambda_i^{k+1,s,c} = \lambda_i^{k+1,s,c} + \rho \tau (\sum_{j:(i,j) \in \mathcal{A}_i^+} y_{ij}^{k+1,s,c} - \sum_{j:(j,i) \in \mathcal{A}_i^-} y_{ji}^{k+1,s,c} - d_i^{s,c})$. The dual updates are distributed by structure. Each node can update the dual variables corresponding to its own flow balance constraints after receiving the updated primal (flow decision) variables from its neighbors. Essentially, the constraint violation with respect to the flow decision variables defines the direction for the dual updates. The proposed method is summarized in Alg. 1.

IV. NUMERICAL ANALYSIS

Numerical experiments were performed in order to assess the performance of the ADAL method applied to the problem at hand. In order to compare our approach with other AL decomposition algorithms in the literature, we also present results for the popular ADMM method [17], which is known to exhibit fast convergence speeds in general, for small though accuracies [18].

In all simulations, the examined networks were randomly generated with the agents uniformly distributed in rectangle boxes. The source nodes were intentionally positioned one network diameter apart from their corresponding sink nodes, in order to prevent trivial problem setups. Moreover, the different scenarios were generated by taking samples from uniformly distributed random commodity demand vectors, whose entries were assumed to be independent. Note that implementation of the ADAL and ADMM algorithms necessitates appropriate selection of the penalty parameter ρ that is known to critically

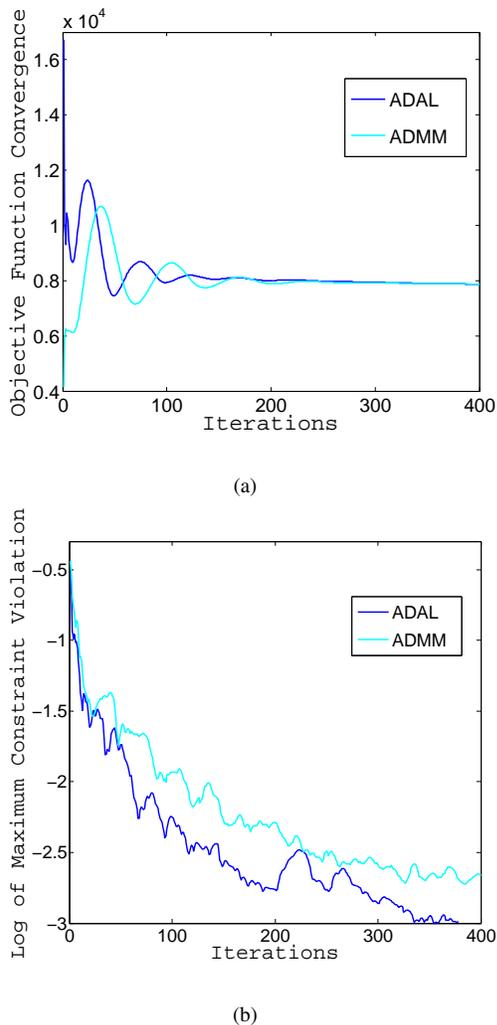


Fig. 1. Comparative convergence results between the ADAL and ADMM methods for a two-stage network capacity expansion problem with 60 nodes, 5 source-sink pairs and 100 scenarios: a) Objective function convergence and b) Maximum constraint violation evolution.

affect performance. In our simulations, we found that ADMM requires relatively larger values of ρ compared to ADAL.

Fig. 1 depicts typical convergence results obtained after applying ADAL and ADMM to a representative problem with 60 nodes, 5 commodities and 100 scenarios. We observe that ADAL converges slightly faster, both in terms of objective function and constraint violation convergence. Note that solving the centralized problem (2) directly, was not possible on our 24GB RAM workstation.

V. CONCLUSIONS

In this paper, we considered the two-stage stochastic nonlinear network capacity expansion problem [15]. We proposed a new distributed optimization technique, which decomposes the, often intractable, centralized problem into smaller, tractable subproblems that are solved in parallel at the network nodes. Unlike existing methods in literature that decompose the problem with respect to scenarios, the advantage of our method is that it can be applied to cases where the nodes must operate independently and autonomously in the absence of a

central processing unit. Moreover, it is a first order method utilizing ALs, thus it combines low computational complexity with the robustness and convergence speed properties of regularization. We compared our method to a popular state-of-the-art distributed algorithm and showed that we obtain non-negligible performance gains.

VI. ACKNOWLEDGEMENTS

The authors would like to thank Darinka Dentcheva for valuable comments and discussions.

REFERENCES

- [1] L. R. Ford and D. R. Fulkerson, *Flows in Networks*. Princeton University Press, 1962.
- [2] A. A. Assad, "Multicommodity network flows: A survey," *Networks*, vol. 8, no. 1, pp. 37–91, 1978.
- [3] A. Ouerou, P. Mahey, and J. Vial, "A survey of algorithms for convex multicommodity flow problems," *Manage. Sci.*, vol. 46, no. 1, pp. 126–147, Jan. 2000.
- [4] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory*, Philadelphia, USA, 2009.
- [5] S.-G. Chang and B. Gavish, "Telecommunications network topological design and capacity expansion: Formulations and algorithms," *Telecommunication Systems*, vol. 1, no. 1, pp. 99–131, 1993.
- [6] M. Mollaghasemi, K. Lecroy, M. Georgiopoulos, and M. Laguna, "Applying robust optimization to capacity expansion of one location in telecommunications with demand uncertainty," *Manage. Sci.*, vol. 44, no. 11, pp. 101–110, Nov. 1998.
- [7] Z. Li and M. G. Ierapetritou, "Capacity expansion planning through augmented lagrangian optimization and scenario decomposition," *AICHE Journal*, vol. 58, no. 3, pp. 871–883, 2012.
- [8] Z.-L. Chen, S. Li, and D. Tirupati, "A scenario-based stochastic programming approach for technology and capacity planning," *Comput. Oper. Res.*, vol. 29, no. 7, pp. 781–806, Jun. 2002.
- [9] I. J. Lustig, J. M. Mulvey, and T. J. Carpenter, "Formulating two-stage stochastic programs for interior point methods," *Oper. Res.*, vol. 39, no. 5, pp. 757–770, Sep. 1991.
- [10] J. R. Birge, "Decomposition and partitioning methods for multistage stochastic linear programs," *Oper. Res.*, vol. 33, no. 5, pp. 989–1007, 1985.
- [11] R. T. Rockafellar and R. J. B. Wets, "Scenarios and policy aggregation in optimization under uncertainty," *Math. Oper. Res.*, vol. 16, no. 1, pp. 119–147, 1991.
- [12] J. M. Mulvey and A. Ruszczyński, "A new scenario decomposition method for large-scale stochastic optimization," *Oper. Res.*, vol. 43, no. 3, pp. 477–490, 1995.
- [13] S. Nielsen and S. Zenios, "Solving multistage stochastic network programs on massively parallel computers," *Mathematical Programming*, vol. 73, no. 3, pp. 227–250, 1996.
- [14] K. J. Singh, A. B. Philpott, and R. K. Wood, "Dantzig-wolfe decomposition for solving multistage stochastic capacity-planning problems," *Oper. Res.*, vol. 57, no. 5, pp. 1271–1286, Sep. 2009.
- [15] A. Ruszczyński and A. Shapiro, in *Stochastic Programming*, ser. Handbooks in Operations Research and Management Science. Elsevier, 2003.
- [16] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.
- [17] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, 1992.
- [18] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [19] A. Ruszczyński, "On convergence of an Augmented Lagrangian decomposition method for sparse convex optimization," *Mathematics of Operations Research*, vol. 20, pp. 634–656, 1995.
- [20] N. Chatzipanagiotis, D. Dentcheva, and M. Zavlanos, "Augmented Lagrangian method for distributed optimization," *Mathematical Programming*, 2013, (Submitted).
- [21] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximations," *Computers and Mathematics with Applications*, vol. 2, pp. 17–40, 1976.