Locality in Network Optimization

Patrick Rebeschini, Sekhar Tatikonda

***Abstract*—In probability theory and statistics notions of correlation among random variables, decay of correlation, and bias-variance trade-off are fundamental. In this work we introduce analogous notions in optimization, and we show their usefulness in a concrete setting. We propose a general notion of correlation among variables in optimization procedures that is based on the sensitivity of optimal points upon (possibly finite) perturbations. We present a canonical instance in network optimization (the min-cost network flow problem) that exhibits locality, i.e., a setting where the correlation decays as a function of the graph-theoretical distance in the network. In the case of warm-start reoptimization, we develop a general approach to localize a given optimization routine in order to exploit locality. We show that the localization mechanism is responsible for introducing a bias in the original algorithm, and that the bias-variance trade-off that emerges can be exploited to minimize the computational complexity required to reach a prescribed level of error accuracy. We provide numerical evidence to support our claims.**

***Index Terms*—sensitivity of optimal points, decay of correlation, bias-variance, network flow, Laplacian, Green's function.**

I. Introduction

Many problems in machine learning, networking, control, and statistics can be naturally posed in the framework of network-structured convex optimization. Given the huge problem size involved in modern applications, a lot of efforts have been devoted to the development and analysis of algorithms where the computation and communication are distributed over the network. A crucial challenge remains that of identifying the structural amount of information that each computation node needs to receive from the network to yield an approximate solution with a certain accuracy.

The literature on distributed algorithms in network optimization is gigantic, with much of the earlier seminal work contained in the textbook [1]. More recent work that explicitly relate the convergence behavior of the algorithms being considered to the network topology and size include — but it is certainly not limited to — [2], [3], [4] for first-order methods, and [5], [6] for second-order methods. Distributed algorithms are iterative in nature. In their synchronous implementations, at every iteration of the algorithm each computation node processes local information that come from its neighbors. Convergence analysis yields the number of iterations that guarantee these algorithms to meet a prescribed level of error accuracy. In turns, these results translate into bounds on the total amount of information processed by each node, as k iterations of the algorithm means that each node receives information coming from nodes that are at most k-hops away in the network. As different algorithms yield different rates of convergence, the bounds on the propagation of information that are so-derived are algorithm-dependent (and also depend on the error analysis being considered). As such, they do not capture structural properties of the optimization problem.

The main aim of this paper is to propose a general notion of "correlation" among variables in network optimization procedures that fundamentally characterizes the propagation of information across the network. This notion is based on the sensitivity of the optimal solution of the optimization problem upon perturbations of parameters locally supported on the network. This notion can be used to investigate "locality," by which we mean problem instances where the correlation decays as a function of the natural distance in the network, so that, effectively, information only propagates across local portions of it. The phenomenon of locality characterizes situations where local perturbations are confined inside small regions of the network. How small the affected regions are, it depends on the particular problem instance at hand, and on the underlying graph topology. If locality is present, one could hope to exploit it algorithmically. In the case of localized perturbations, for instance, one could hope to localize known optimization routines so that only the affected regions are updated, hence yielding computational savings. While such a localization mechanism would introduce a structural "bias" with respect to the original algorithm — which updates every node in the network and not only the ones that are mostly affected by the perturbation — one could hope to exploit the "reduction in the variance" to achieve a prescribed level of error accuracy with a lower computational complexity. In the following we make all of this precise.

In this paper we define general notions of correlation, decay of correlation (locality), and bias-variance decomposition and trade-off in optimization, and we consider a concrete setting to illustrate their usefulness. The results that we present should be seen as an implementation of a general agenda that can be followed to establish and exploit locality in more general instances of network optimization.

This paper presents four main contributions discussed in separate sections, which we now summarize. Proofs are in the Appendices (see also the online version of this manuscript).

1) Sensitivity of optimal points. In Section II we present a theory on the sensitivity of optimal points for smooth convex problems with linear equality constraints upon (possibly finite) perturbations. We consider the problem of minimizing a smooth convex function $x \to f(x)$ subject to $Ax = b$, for a certain matrix A and vector $b \in \mathrm{Im}(A)$, where $\mathrm{Im}(A)$ denotes the image of A. We consider this problem as a function of

This work was supported in part by the NSF Grant ECCS-1609484.

P. Rebeschini was with the Department of Electrical Engineering, Yale University, New Haven, CT 06511, USA. He is now with the Department of Statistics, University of Oxford, Oxford, OX1 3LB, UK (e-mail: patrick.rebeschini@stats.ox.ac.uk).

S. Tatikonda is with the Department of Statistics and Data Science, Yale University, New Haven, CT 06511, USA (e-mail: sekhar.tatikonda@yale.edu).

the constraint vector b. We prove that if f is strongly convex, then the optimal point $b \to x^\star(b)$ is continuously differentiable along $\mathrm{Im}(A)$, and we explicitly characterize the effect that perturbations have on the optimal solution as a function of the objective function f, the constraint matrix A and vector b. Given a differentiable function $\varepsilon \in \mathbb{R} \to b(\varepsilon) \in \mathrm{Im}(A)$, we show that the quantity $\frac{dx^\star(b(\varepsilon))}{d\varepsilon}$ is a function of the Moore-Penrose pseudoinverse of the matrix $A\Sigma(b(\varepsilon))A^T$, where A^T is the transpose of A, and where $\Sigma(b)$ denotes the inverse of the Hessian of f evaluated at $x^\star(b)$, namely, $\Sigma(b) := \nabla^2 f(x^\star(b))^{-1}$. The literature on the sensitivity of optimal points (see Section II for a list of references) is typically only concerned with establishing infinitesimal perturbations locally, i.e., on a neighborhood of a certain $b \in \mathrm{Im}(A)$. On the other hand, the results that we present extend to *finite* perturbations as well, as we prove that the derivatives of the optimal point are continuous along $\mathrm{Im}(A)$, and hence can be integrated to deal with finite perturbations. The workhorse behind our results is Hadamard's global inverse function theorem [7], which in our setup yields necessary and sufficient conditions for the inverse of the KKT map to be continuously differentiable. Proofs are in Appendix A.

2) Notions of correlation in optimization. In Section III we provide an interpretation of the sensitivity theory previously developed in terms of notions of correlation among variables in optimization procedures, resembling analogous notions in probability theory. If the matrix A is full row rank, for instance, then the quantity $\frac{\partial x^\star(b)_i}{\partial b_a}$ is well-defined and describes how much $x^\star(b)_i$ — the i-th component of the optimal solution $x^\star(b)$ — changes upon perturbation of b_a — the a-th component of the constraint vector b. We interpret $\frac{\partial x^\star(b)_i}{\partial b_a}$ as a measure of the correlation between variables i and a in the optimization problem, and we are interested in understanding how this quantity behaves as a function of the geodesic distance between i and a. We motivate this terminology by establishing an analogy with the theory of correlations in probability, via the connection with Gaussian random variables (proofs are in Appendix B). We extend the notion of correlation beyond infinitesimal perturbations, and we show how our theory yields a first instance of comparison theorems for constrained optimization procedures, along the lines of the comparison theorems established in probability theory to capture stochastic decay of correlation and control the difference of high-dimensional distributions [8], [9].

In probability theory, decay of correlation characterizes the effective neighborhood dependency of random variables in a probabilistic network. Since the seminal work of Dobrushin [8], this concept has found many applications beyond statistical physics. Recently, it has been used to develop and prove convergence guarantees for fast distributed local algorithms for inference and decision problems on large networks in a wide variety of domains, for instance, probabilistic marginal inference [10], wireless communication [11], network learning [12], combinatorial optimization [13], and nonlinear filtering [14]. However, even in applications where the underlying problem does not involve randomness, decay of correlation is typically established upon endowing the model with a probabilistic structure, hence modifying the original problem formulation. Our results in network optimization, instead, show how locality can be described in a purely non-random setting, as a structural property of the original optimization problem. To the best of our knowledge, non-random notions of correlation in optimization have been previously considered only in [15], where the authors explicitly use the word "correlation" to denote the sensitivity of optimal points with respect to localized perturbations, and they analyze the correlation as a function of the natural distance in the graph. However, in their work correlation is regarded as a tool to prove convergence guarantees for the specific algorithm at hand (Min-Sum message-passing to solve *unconstrained* convex problems), and no general theory is built around it.

3) Locality: decay of correlation. As a paradigm for network optimization, in Section IV we consider the problem of computing network flows. This is a fundamental problem that has been studied in various formulations by different communities. The min-cost network flow variant (where the objective function is typically chosen to be linear, although there are non-linear extensions, and the constraints also include inequalities) has been essential in the development of the theory of polynomial-times algorithms for optimizations (see [16] and references therein, or [17] for a book reference). In the case of quadratic functions, the problem is equivalent to computing electrical flows, which is a fundamental primitive related to Laplacian solvers that has been extensively studied in computer science (see [18]) with also applications to learning problems ([19], [20], for instance). In the formulation we consider, a directed graph $\vec{G} = (V, \vec{E})$ is given (arrows indicate directed graphs/edges), with its structure encoded in the vertex-edge incidence matrix $A \in \mathbb{R}^{V \times \vec{E}}$. To each edge $e \in \vec{E}$ is associated a flow x_e with a cost $f_e(x_e)$, and to each vertex $v \in V$ is associated an external flow b_v. The network flow problem consists in finding the flow $x^\star(b) \in \mathbb{R}^{\vec{E}}$ that minimizes the cost $f(x) := \sum_{e \in \vec{E}} f_e(x)$ and satisfies the conservation law $Ax = b$.

In this setting, the general sensitivity theory that we have previously developed allows to characterize the derivatives of the optimal flow in terms of graph Laplacians; in fact, in this case the matrix $A\Sigma(b)A^T$ corresponds to the Laplacian of an undirected weighted graph associated to $\vec{G}$, where each directed edge $e \in \vec{E}$ is given a weight $\Sigma(b)_{ee} > 0$. Exploiting a general connection between the Moore-Penrose pseudoinverse of graph Laplacians and the Green's function of random walks on weighed graphs — which we present as standalone in Appendix C — we express the correlation term $\frac{dx^\star(b(\varepsilon))_e}{d\varepsilon}$ in terms of *differences* of Green's functions. Different graph topologies yield different decaying behaviors for this quantity, as a function of the geodesic distance $d(e, Z)$ between edge e and the set of vertices $Z \subseteq V$ where the perturbation is localized, namely, $Z := \{z \in V : \frac{db(\varepsilon)_z}{d\varepsilon} = 0\}$. In the case of expanders, we derive spectral bounds that show an exponential decay, with rate given by the second largest eigenvalue in magnitude of the diffusion random walk. In this case we establish various types of decay of correlation bounds, point-to-set and set-to-point. Appendix D contains the proofs of these results.

For grid-like graphs, based on numerical simulations and on asymptotic results for the behavior of the Green's function in infinite grids [21], we expect the correlation term to decay polynomially instead of exponentially.

4) Localized algorithms and bias-variance. In Section V we investigate applications of the framework that we propose to develop scalable computationally-efficient algorithms. To illustrate the main principle behind our reasoning, we consider the case when the solution $x^\star(b)$ is given (also known as *warm-start* scenario) and we want to compute the solution $x^\star(b+p)$ for the perturbed flow $b+p$, where p is a perturbation supported on a small localized subset $Z \subseteq V$. This setting belongs to the class of reoptimization problems typically studied in computer science and operations research (we could not to find previous literature on the setting that we consider; the case considered more frequently involves discrete problems and changes in the graph structure, as in [22]).
The decay of correlation property structurally exhibited by the min-cost network flow problem encodes the fact that when the external flow b is locally perturbed it suffices to recompute the solution only for the part of the network that is "mostly affected" by this perturbation, i.e., the set of edges that have a distance at most r from the perturbation set Z. Here, the radius r is tuned to meet the desired level of error tolerance given the size of the perturbation and given the graph topology being investigated. This allows us to show that it is possible to develop localized versions of canonical optimization algorithms that can exploit locality by only updating the edges in a subgraph of $\vec{G}$. We investigate regimes where localized algorithms can achieve computational savings against their global counterpart. The key behind these savings is the bias-variance decomposition that we give for the error of localized algorithms. This decomposition conveys the idea that by introducing some bias in a given optimization routine (in our setting, by truncating the problem size restricting the algorithm to a subgraph) one can diminish its variance and obtain faster convergence rates. We illustrate this phenomenon theoretically and with numerical simulations for localized projected gradient descent. Proofs are in Appendix E.

Notation. *For a given matrix M, let M^T be the transpose, M^{-1} be the inverse, and M^+ be the Moore-Penrose pseudoinverse. Let $\mathrm{Ker}(M) := \{x : Mx = 0\}$ and $\mathrm{Im}(M) := \{y : y = Mx \text{ for some } x\}$ be the kernel and image of M, respectively. Given an index set $\mathcal{I}$ and subsets $K, L \subseteq \mathcal{I}$, if $M \in \mathbb{R}^{\mathcal{I}\times\mathcal{I}}$, let $M_{K,L} \in \mathbb{R}^{K\times L}$ denote the submatrix corresponding to the rows and columns of M indexed by K and L, respectively. Let I be the identity matrix, $\mathbb{1}$ the all-one vector (or matrix), and $\mathbb{0}$ the all-zero vector (or matrix), whose sizes are implied by the context. Given a vector $x \in \mathbb{R}^{\mathcal{I}}$, let $x_i \in \mathbb{R}$ be the component associated to $i \in \mathcal{I}$, and $x_K := (x_i)_{i\in K} \in \mathbb{R}^K$ the components associated to $K \subseteq \mathcal{I}$. Let $\|x\| := (\sum_{i\in\mathcal{I}} x_i^2)^{1/2}$ be the ℓ_2-norm of x and $\|x\|_K := (\sum_{i\in K} x_i^2)^{1/2}$ be the localized ℓ_2-norm on K. Clearly, $\|x\|_K = \|x_K\|$ and $\|x\|_{\mathcal{I}} = \|x\|$. We use the notation $|K|$ to denote the cardinality of K. If $\vec{G} = (V, \vec{E})$ is a directed graph with vertex set V and edge set $\vec{E}$, let $G = (V, E)$ be the undirected graph associated to $\vec{G}$, namely, $\{u, v\} \in E$ if and only if either $(u, v) \in \vec{E}$ or $(v, u) \in \vec{E}$.*

II. Sensitivity of optimal points

Let $\mathcal{V}$ be a finite set — to be referred to as the "variable set" — and let $f : \mathbb{R}^{\mathcal{V}} \to \mathbb{R}$ be a strictly convex function, twice continuously differentiable. Let $\mathcal{F}$ be a finite set — to be referred to as the "factor set" — and let $A \in \mathbb{R}^{\mathcal{F}\times\mathcal{V}}$. Consider the following optimization problem over $x \in \mathbb{R}^{\mathcal{V}}$:

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & Ax = b, \end{aligned} \tag{1}$$

for $b \in \mathrm{Im}(A) \subseteq \mathbb{R}^{\mathcal{F}}$, so that the feasible region is not empty. Throughout this paper we think of the function f and the matrix A as fixed, and we consider the solution of the optimization problem above as a function of the vector $b \in \mathrm{Im}(A)$. By strict convexity, this problem has a unique optimal solution, that we denote by $x^\star(b) := \mathrm{argmin}\{f(x) : x \in \mathbb{R}^{\mathcal{V}}, Ax = b\}$.

Theorem 1 below provides a characterization of the way a perturbation of the constraint vector b on the subspace $\mathrm{Im}(A)$ affects the optimal solution $x^\star(b)$ in the case when the function f is strongly convex. The proof is given in Appendix A.

Theorem 1 (Sensitivity optimal point)**.** *Let $f : \mathbb{R}^{\mathcal{V}} \to \mathbb{R}$ be a strongly convex function, twice continuously differentiable. Let $A \in \mathbb{R}^{\mathcal{F}\times\mathcal{V}}$. For $b \in \mathrm{Im}(A)$, let $\Sigma(b) := \nabla^2 f(x^\star(b))^{-1}$ and $D(b) := \Sigma(b)A^T(A\Sigma(b)A^T)^+$. Then, $x^\star$ is continuously differentiable along the subspace $\mathrm{Im}(A)$, and given a differentiable function $\varepsilon \in \mathbb{R} \to b(\varepsilon) \in \mathrm{Im}(A)$ we have*

$$\frac{dx^\star(b(\varepsilon))}{d\varepsilon} = D(b(\varepsilon))\frac{db(\varepsilon)}{d\varepsilon}.$$

Most of the literature on the sensitivity of optimal points for nonlinear programs investigates *local* results for infinitesimal perturbations, establishing the existence of a local neighborhood of the perturbed parameter(s) where the optimal point(s) has(have) some analytical properties such as continuity, differentiability, etc. Classical references are [23], [24], [25], [26]. Typically, the main tool used to establish this type of results is the implicit function theorem applied to the first order optimality conditions (KKT map), which is a local statement (as such, it is flexible and it can be applied to a great variety of optimization problems). The sensitivity result that we present in Theorem 1 in the case of strongly convex functions, on the other hand, is based on Hadamard's global inverse function theorem [7]. This is a *global* statement, as it shows that the optimal point $x^\star$ is continuously differentiable along the *entire* subspace $\mathrm{Im}(A)$. This fact allows the use of the fundamental theorem of calculus to deal with finite perturbations, which is instrumental for the results developed in this paper (in particular, for the connection with comparison theorems in probability, Section III-B, and for the results in Section V). In the case of quadratic programs with linear constraints (the problem we consider can be thought of as an extension of this setting to strongly convex functions) there are many papers in parametric programming investigating the sensitivity of optimal points with respect to changes in the objective functions and constraints (see [27] and references therein). While most of these results are again local, some of them are closer to our approach and also address finite perturbations [28].

Theorem 1 characterizes the behavior of $x^\star(b)$ upon perturbations of b along $\mathrm{Im}(A) \subseteq \mathbb{R}^{\mathcal{F}}$. If the matrix A is full row rank, i.e., $\mathrm{Im}(A) = \mathbb{R}^{\mathcal{F}}$, then the optimal point $x^\star$ is everywhere continuously differentiable, and we can compute its gradient. We have the following immediate corollary.

Corollary 1 (Sensitivity optimal point, full rank case)**.** *Consider the setting of Theorem 1, with the matrix $A \in \mathbb{R}^{\mathcal{F}\times\mathcal{V}}$ having full row rank, i.e., $\mathrm{Im}(A) = \mathbb{R}^{\mathcal{F}}$. Then, the function $b \in \mathbb{R}^{\mathcal{F}} \to x^\star(b) \in \mathbb{R}^{\mathcal{V}}$ is continuously differentiable and*

$$\frac{dx^\star(b)}{db} = D(b) = \Sigma(b)A^T(A\Sigma(b)A^T)^{-1}.$$

Remark 1. *Our goal is to present the simplest setting of interest where we can establish locality and illustrate the computational savings achieved by localized algorithms. The computational advantages provided by localization are already appreciable in the well-conditioned setting that we consider, as we explain in Section V below. For this reason, we are satisfied with the assumption of strong convexity in Theorem 1. The main tool behind Theorem 1 is Hadamard's global inverse function theorem, which provides* necessary and sufficient *conditions for the inverse of a continuously differentiable function to be continuously differentiable. These conditions involve cohercitivity of the function of interest, along with a non-degeneracy condition on the determinant of the function (see the online version of the manuscript). To relax the assumptions that we give in Theorem 1, one needs to consider a more refined application of Hadamard's theorem to the KKT map.*

III. Notions of correlation in optimization

The sensitivity analysis presented in Section II suggests a natural notion of correlation between variables and factors in optimization, resembling notions of correlation among random variables in probability theory. If the matrix A is full row rank, then the quantity $\frac{\partial x^\star(b)_i}{\partial b_a}$ is well-defined and it captures the interaction between variable $i \in \mathcal{V}$ and factor $a \in \mathcal{F}$ in the optimization procedure, and the quantity $D(b)_{ia}$ in Corollary 1 characterizes this correlation as a function of the constraint matrix A, the objective function f, and the optimal solution $x^\star(b)$. Theorem 1 allows us to extend the notion of correlation between variables and factors to the more general case when the matrix A is not full rank. As an example, let $b, p \in \mathrm{Im}(A)$, and assume that p is supported on a subset $Z \subseteq \mathcal{F}$, namely, $p_a \neq 0$ if and only if $a \in Z$. Define $b(\varepsilon) := b + \varepsilon p$. Then, the quantity $\frac{dx^\star(b(\varepsilon))_i}{d\varepsilon}$ measures how much a perturbation of the constraints in Z affects the optimal solution at $i \in \mathcal{V}$, hence it can be interpreted as a measure of the correlation between variable i and the factors in Z, which is characterized by the term $(D(b(\varepsilon))\frac{db(\varepsilon)}{d\varepsilon})_i = \sum_{a\in Z} D(b(\varepsilon))_{ia}p_a$ in Theorem 1.

We now discuss how these notions of correlation relate to analogous notions in probability.

A. Connection with Gaussian random variables

The resemblance between the notion of correlations in optimization and in probability is made explicit via the analogy to the theory of Gaussian random variables. Recall the following result (proofs of the results here discussed are given for completeness in Appendix B).

Proposition 1 (Conditional mean of Gaussian random variables)**.** *Let $\mathcal{V}, \mathcal{F}$ be two finite sets. Let $X \in \mathbb{R}^{\mathcal{V}}$ be a Gaussian random vector with mean $\mu \in \mathbb{R}^{\mathcal{V}}$ and covariance $\Sigma \in \mathbb{R}^{\mathcal{V}\times\mathcal{V}}$, possibly singular. Let $A \in \mathbb{R}^{\mathcal{F}\times\mathcal{V}}$ be given. Given a differentiable function $\varepsilon \in \mathbb{R} \to b(\varepsilon) \in \mathrm{Im}(A)$, we have $\frac{d\mathbf{E}[X|AX=b(\varepsilon)]}{d\varepsilon} = \Sigma A^T(A\Sigma A^T)^+\frac{db(\varepsilon)}{d\varepsilon}$. If Σ is invertible and A is full row rank, then for each $b \in \mathbb{R}^{\mathcal{F}}$ we have $\frac{d\mathbf{E}[X|AX=b]}{db} = \Sigma A^T(A\Sigma A^T)^{-1}$.*

Proposition 1 shows that for $i \in \mathcal{V}, a \in \mathcal{F}$, the quantity $\frac{\partial \mathbf{E}[X_i|AX=b]}{\partial b_a} = (\Sigma A^T(A\Sigma A^T)^{-1})_{ia}$ can be interpreted as a measure of correlation between the random variables X_i and $(AX)_a$, as it describes how much a perturbation of $(AX)_a$ impacts X_i, upon conditioning on AX. A similar interpretation can be given in optimization for the quantities in Corollary 1, with the difference that typically these quantities depend on $b \in \mathrm{Im}(A)$, as they are functions of $x^\star(b)$.

The sensitivity results that we derived in Section II also yield notions of correlation in optimization between variables. The following lemma, an immediate application of Corollary 1, shows that the local behavior of the optimal solution of the optimization problem (1) when we freeze some coordinates, upon perturbation of these coordinates, is analogous to the behavior of the conditional mean of a non-degenerate Gaussian random vector upon changing the coordinates we condition on. Recall (see the proof of Proposition 1) that if $X \in \mathbb{R}^{\mathcal{V}}$ is a Gaussian vector with mean $\mu \in \mathbb{R}^{\mathcal{V}}$ and positive definite covariance $\Sigma \in \mathbb{R}^{\mathcal{V}\times\mathcal{V}}$, then for $I \subseteq \mathcal{V}$ and $B := \mathcal{V} \setminus I$ $\frac{d\mathbf{E}[X_I|X_B=x_B]}{dx_B} = \Sigma_{I,B}(\Sigma_{B,B})^{-1}$.

Lemma 1 (Sensitivity with respect to boundary conditions)**.** *Let $f : \mathbb{R}^{\mathcal{V}} \to \mathbb{R}$ be a strongly convex function, twice continuously differentiable. Let $I \subseteq \mathcal{V}$ be a nonempty set, and let $B := \mathcal{V} \setminus I$ not empty. Define the function $x_I^\star : x_B \in \mathbb{R}^B \longrightarrow x_I^\star(x_B) := \mathrm{argmin}\left\{f(x_I x_B) : x_I \in \mathbb{R}^I\right\}$. For $x_B \in \mathbb{R}^B$, let $H(x_B) := \nabla^2 f(x_I^\star(x_B)x_B)$ and $\Sigma(x_B) := H(x_B)^{-1}$. Then, $x_I^\star$ is continuously differentiable and $\frac{dx_I^\star(x_B)}{dx_B} = \Sigma(x_B)_{I,B}(\Sigma(x_B)_{B,B})^{-1} = -(H(x_B)_{I,I})^{-1}H(x_B)_{I,B}$.*

B. Comparison theorems

The connection between Theorem 1 and the theory of correlations in probability extends beyond infinitesimal perturbations. As previously discussed, Theorem 1 can be used to deal with finite perturbations, and so it can be interpreted as a comparison theorem to capture *uniform* correlations in optimization, along the lines of the comparison theorems in probability theory to capture stochastic decay of correlation and control the difference of high-dimensional distributions (see the seminal work in [8], and [9] for generalizations).

To see this analogy, let us consider a simplified version of the Dobrushin comparison theorem that can be easily derived from the textbook version in [29], Theorem 8.20. Let I be a finite set, and let $\Omega := \prod_{i\in I}\Omega_i$ where Ω_i is a finite set for each $i \in I$. Define the projections $X_i : x \mapsto x_i$ for $x \in \Omega$ an $i \in I$. For any probability distribution μ on Ω, define the marginal $\mu_i(y) := \mu(X_i = y)$, and the conditional

distribution $\mu_i^x(y) := \mu(X_i = y | X_{I\setminus\{i\}} = x_{I\setminus\{i\}})$. Define the total variation distance between two distributions ν and $\tilde{\nu}$ on Ω_i as $\|\nu - \tilde{\nu}\|_T := \frac{1}{2}\sum_{y\in\Omega_i} |\nu(y) - \tilde{\nu}(y)|$.

Theorem 2 (Dobrushin comparison theorem)**.** *Let $\mu, \tilde{\mu}$ be probability distributions on Ω. For each $i, j \in I$, define $\mathsf{C}_{ij} := \sup_{x,z\in\Omega: x^{I\setminus\{j\}} = z^{I\setminus\{j\}}} \|\mu_i^x - \mu_i^z\|_T$ and $\mathsf{b}_j := \sup_{x\in\Omega} \|\mu_j^x - \tilde{\mu}_j^x\|_T$, and assume that the Dobrushin condition holds: $\max_{i\in I}\sum_{j\in I}\mathsf{C}_{ij} < 1$. Then the matrix sum $\mathsf{D} := \sum_{t\geq 0}\mathsf{C}^t$ is convergent, and for any $i \in I$, $y \in \Omega_i$, we have $\|\mu_i - \tilde{\mu}_i\|_T \leq \sum_{j\in I}\mathsf{D}_{ij}\mathsf{b}_j$.*

The Dobrushin coefficient C_{ij} is a (uniform) measure of the degree to which a perturbation of site j *directly* affects site i under the distribution μ. However, perturbing site j might also indirectly affect site i: it could affect another site k which in turn affects i, etc. The *aggregate* effect of a perturbation of site j on site i is captured by D_{ij}. The quantity b_j is a comparison term that measures the local difference at site j between μ and $\tilde{\mu}$ (in terms of the conditional distributions $\mu_j^{\cdot}$ and $\tilde{\mu}_j^{\cdot}$).

The formal analogy between the Dobrushin comparison theorem and the sensitivity results of Theorem 1 for the optimization problem (1) is made explicit by the fundamental theorem of calculus. This connection is easier to make if we assume that the matrix A has full row rank, and we consider the results in Corollary 1. In this setting, the optimal point $x^\star$ is everywhere continuously differentiable, and for each $i \in \mathcal{V}$, $b, \tilde{b} \in \mathbb{R}^{\mathcal{F}}$, $b \neq \tilde{b}$, we have $x^\star(b)_i - x^\star(\tilde{b})_i = \int_0^1 \frac{dx^\star(\theta b + (1-\theta)\tilde{b})_i}{d\theta} d\theta = \sum_{a\in\mathcal{F}} D(b,\tilde{b})_{ia}(b_a - \tilde{b}_a)$, where $D(b,\tilde{b})_{ia} := \int_0^1 (\Sigma(b_\theta)A^T(A\Sigma(b_\theta)A^T)^{-1})_{ia} d\theta$ with $b_\theta := \theta b + (1-\theta)\tilde{b}$. If for each $i \in \mathcal{V}$ and $a \in \mathcal{F}$ we have $\sup_{b\in\mathbb{R}^{\mathcal{F}}} |(\Sigma(b)A^T(A\Sigma(b)A^T)^{-1})_{ia}| \leq D_{ia}$, then from the previous expression we find $|x^\star(b)_i - x^\star(\tilde{b})_i| \leq \sum_{a\in\mathcal{F}} D_{ia}|b_a - \tilde{b}_a|$, whose structure resembles the statement in Theorem 2. The quantity D_{ia} represents a uniform measure of the aggregate impact that a perturbation of the a-th component of the constraint vector b has to the i-th component of the optimal solution $x^\star$, so the matrix D takes the analogous role of the matrix D in Theorem 2 (and as we will see below in a concrete application, see Theorem 3, suitable series expansions of D yield the analogous of C). The quantity $|b_a - \tilde{b}_a|$ is a comparison term that measures the local difference at factor a between b and $\tilde{b}$, resembling the role of b_j in Theorem 2.

In the next section we investigate the notion of correlation just introduced in the context of network optimization, in a concrete instance when the constraints naturally reflect a graph structure, and we investigate the behavior of the correlations as a function of the natural distance in the graph.

IV. Locality: decay of correlation

As a paradigm for network optimization, we consider the network flow problem that has been widely studied in various fields (see introduction). Consider a directed graph $\vec{G} := (V, \vec{E})$, with vertex set V and edge set $\vec{E}$, with no self-edges and no multiple edges. Let $G = (V, E)$ be the undirected graph naturally associated with $\vec{G}$, that is, $\{u, v\} \in E$ if and only if either $(u, v) \in \vec{E}$ or $(v, u) \in \vec{E}$. Without loss of generality, assume that G is connected (otherwise we can treat each connected component on its own). For each $e \in \vec{E}$ let x_e denote the flow on edge e, with $x_e > 0$ if the flow is in the direction of the edge, $x_e < 0$ if the flow is in the direction opposite the edge. For each $v \in V$ let b_v be a given external flow on the vertex v: $b_v > 0$ represents a source where the flow enters the vertex, whereas $b_v < 0$ represents a sink where the flow leaves the vertex. Assume that the total of the source flows equals the total of the sink flows, that is, $\mathbb{1}^T b = \sum_{v\in V} b_v = 0$, where $b = (b_v)_{v\in V} \in \mathbb{R}^V$ is the flow vector. We assume that the flow satisfies a conservation equation so that at each vertex the total flow is zero. This conservation law can be expressed as $Ax = b$, where $A \in \mathbb{R}^{V\times\vec{E}}$ is the *vertex-edge incidence matrix* defined as $A_{ve} := 1$ if e leaves node v, $A_{ve} := -1$ if e enters node v, and $A_{ve} := 0$ otherwise.

For each edge $e \in \vec{E}$ let $f_e : \mathbb{R} \to \mathbb{R}$ be its associated cost function, assumed to be strongly convex and twice continuously differentiable. The network flow problem reads as problem (1) with $f(x) := \sum_{e\in\vec{E}} f_e(x_e)$. It is easy to see that since G is connected $\mathrm{Im}(A)$ consists of all vectors orthogonal to the vector $\mathbb{1}$, i.e., $\mathrm{Im}(A) = \{y \in \mathbb{R}^V : \mathbb{1}^T y = 0\}$. Henceforth, for each $b \in \mathbb{R}^V$ with $\mathbb{1}^T b = 0$, let $x^\star(b)$ be the optimal flow.

We first apply the sensitivity theory developed in Section II to characterize the correlation between vertices (i.e., factors) and edges (i.e., variables) in the network flow problem. Then, we investigate the behavior of these correlations in terms of the natural distance on the graph G.

A. Correlation, graph Laplacians and Green's functions

In the setting of the network flow problem, Theorem 1 immediately allows us to characterize the derivatives of the optimal point $x^\star$ along the subspace $\mathrm{Im}(A)$ as a function of the graph Laplacian [30]. For $b \in \mathbb{R}^V$ such that $\mathbb{1}^T b = 0$, let $\Sigma(b) := \nabla^2 f(x^\star(b))^{-1} \in \mathbb{R}^{\vec{E}\times\vec{E}}$, which is a diagonal matrix with entries given by $\sigma(b)_e := \Sigma(b)_{ee} := (\frac{\partial^2 f_e(x^\star(b)_e)}{\partial x_e^2})^{-1} > 0$. Each term $\sigma(b)_e$ is strictly positive as f_e is strongly convex by assumption. Let $W(b) \in \mathbb{R}^{V\times V}$ be the symmetric matrix defined, for each $u, v \in V$, as $W(b)_{uv} := \sigma(b)_e$ if $e = (u, v) \in \vec{E}$ or $e = (v, u) \in \vec{E}$, and $W(b)_{uv} := 0$ otherwise. Let $D(b) \in \mathbb{R}^{V\times V}$ be the diagonal matrix with entries given by $d(b)_v := D(b)_{vv} := \sum_{u\in V} W(b)_{vu}$, for $v \in V$. Let $L(b) := D(b) - W(b)$ be the graph Laplacian of the undirected weighted graph $(V, E, W(b))$, where to each edge $e = \{u, v\} \in E$ is associated the weight $W(b)_{uv}$. A direct application of Theorem 1 (upon choosing variable set $\mathcal{V} := \vec{E}$ and factor set $\mathcal{F} := V$, and noticing that $A\Sigma(b)A^T = L(b)$) shows that the derivatives of the optimal point $x^\star$ along $\mathrm{Im}(A)$ can be expressed in terms of the Moore-Penrose pseudoinverse of $L(b)$. The connection between $L(b)^+$ and the Green's function of random walks with transition matrix $P(b) := D(b)^{-1}W(b)$ allows us to derive the following result (proofs are in Appendix C).

Theorem 3 (Sensitivity optimal flow)**.** *Given $b \in \mathbb{R}^V$ with $\mathbb{1}^T b = 0$, let $D(b) := \Sigma(b)A^T L(b)^+$. The optimal network flow $x^\star$ is continuously differentiable along $\mathrm{Im}(A)$, and given*

a differentiable function $\varepsilon \in \mathbb{R} \to b(\varepsilon) \in \mathrm{Im}(A)$ *we have* $\frac{dx^\star(b(\varepsilon))}{d\varepsilon} = D(b(\varepsilon))\frac{db(\varepsilon)}{d\varepsilon}$. *For* $e = (u,v) \in \vec{E}$, *we have*

$$\frac{dx^\star(b(\varepsilon))_e}{d\varepsilon} = \sigma(b)_e \sum_{z \in V} \frac{1}{d(b)_z} \frac{db(\varepsilon)_z}{d\varepsilon} \sum_{t=0}^{\infty} (P(b)^t_{uz} - P(b)^t_{vz}).$$

Let $b, p \in \mathbb{R}^V$ such that $\mathbb{1}^T b = \mathbb{1}^T p = 0$, and assume that p is supported on a subset $Z \subseteq V$, namely, $p_v \neq 0$ if and only if $v \in Z$. Define $b(\varepsilon) := b + \varepsilon p$. Then, as discussed in Section III, the quantity $\frac{dx^\star(b(\varepsilon))_e}{d\varepsilon}$ can be interpreted as a measure of the correlation between edge $e \in \vec{E}$ and the vertices in Z in the network flow problem. How does the correlation behave with respect to the graph distance between e and Z? Theorem 3 shows that the correlation is controlled by the *difference* of the Green's function $\sum_{t=0}^{\infty} P(b)^t_{uz}$ with respect to two neighboring starting points u and v (note that the Green's function itself is infinite, as we are dealing with finite graphs). Different graph topologies yield different decaying behaviors for this quantity. In the case of expanders, we now derive spectral bounds that decay exponentially, with rate given by the second largest eigenvalue in magnitude of the diffusion random walk. For grid-like topologies, based on simulations and on asymptotic results for the behavior of the Green's function in infinite grids [21], we expect the correlation to decay polynomially rather than exponentially.

B. Decay of correlation for expanders

Let $n := |V|$ be the cardinality of V, and for each $b \in \mathrm{Im}(A)$ let $-1 \leq \lambda_n(b) \leq \lambda_{n-1}(b) \leq \cdots \leq \lambda_2(b) < \lambda_1(b) = 1$ be the real eigenvalues of $P(b)$.[1] Define $\lambda(b) := \max\{|\lambda_2(b)|, |\lambda_n(b)|\}$ and $\lambda := \sup_{b \in \mathrm{Im}(A)} \lambda(b)$. For each $v \in V$, let $\mathcal{N}(v) := \{w \in V : \{v,w\} \in E\}$ be the set of node neighbors of v in the graph G. Let d be the graph-theoretical distance between vertices in the graph G, namely, $d(u,v)$ is the length of the shortest path between vertices $u, v \in V$. For subset of vertices $U, Z \subseteq V$, define $d(U,Z) := \min\{d(u,z) : u \in U, z \in Z\}$. For each subset of edges $\vec{F} \subseteq \vec{E}$, let $V_{\vec{F}} \subseteq V$ be the vertex set of the subgraph $(V_{\vec{F}}, \vec{F})$ of $\vec{G}$ that is induced by the edges in $\vec{F}$. Recall the definition of the localized ℓ_2-norm from Section I. The following result attests that the correlation for the network flow problem is upper-bounded by a quantity that decays exponentially as a function of the distance in the graph, with rate given by λ. For graphs where λ does not depend on the dimension, i.e., expanders, Theorem 4 can be interpreted as a first manifestation of the decay of correlation principle (i.e., locality) in network optimization. The proof is given in Appendix D.

Theorem 4 (Decay of correlation for expanders)**.** *Consider the setting defined above. Let* $\varepsilon \in \mathbb{R} \to b(\varepsilon) \in \mathrm{Im}(A)$ *be a differentiable function such that for any* $\varepsilon \in \mathbb{R}$ *we have* $\frac{db(\varepsilon)_v}{d\varepsilon} \neq 0$ *if and only if* $v \in Z$, *for a given* $Z \subseteq V$. *Then, for any subset of edges* $\vec{F} \subseteq \vec{E}$ *and any* $\varepsilon \in \mathbb{R}$, *we have*

$$\left\| \frac{dx^\star(b(\varepsilon))}{d\varepsilon} \right\|_{\vec{F}} \leq c \, \frac{\lambda^{d(V_{\vec{F}}, Z)}}{1-\lambda} \left\| \frac{db(\varepsilon)}{d\varepsilon} \right\|_Z,$$

with $c := \sup_{b \in \mathrm{Im}(A)} \frac{\max_{v \in V_{\vec{F}}} \sqrt{2|\mathcal{N}(v) \cap V_{\vec{F}}|}}{\min_{v \in V_{\vec{F}}} d(b)_v} \max_{u,v \in V_{\vec{F}}} W(b)_{uv}$.

Recall that $\|\frac{dx^\star(b(\varepsilon))}{d\varepsilon}\|_{\vec{F}} \equiv (\sum_{e \in \vec{F}} (\frac{dx^\star(b(\varepsilon))_e}{d\varepsilon})^2)^{1/2}$ and $\|\frac{db(\varepsilon)}{d\varepsilon}\|_Z \equiv (\sum_{v \in Z} (\frac{db(\varepsilon)_v}{d\varepsilon})^2)^{1/2}$. The bound in Theorem 4 controls the effect that a localized perturbation supported on a subset of vertices $Z \subseteq V$ has on a subset of edges $\vec{F} \subseteq \vec{E}$, as a function of the distance between $\vec{F}$ and Z, i.e., $d(V_{\vec{F}}, Z)$ (note that we only defined the distance among vertices, not edges, and that this distance is with respect to the unweighted graph G). A key feature of Theorem 4 — which is essential for the results in Section V below — is that the bound presented does not depend on the cardinality of $\vec{F}$.

Theorem 4 controls the effect that a *single* localized perturbation (supported on multiple vertices, as it has to be that $|Z| \geq 2$ for the function $\varepsilon \in \mathbb{R} \to b(\varepsilon)$ to be on $\mathrm{Im}(A)$) has on a *collection* of edges for the optimal solution, independently of the number of edges being considered. We refer to this type of decay of correlation as *set-to-point*. Analogously, it is possible to control the effect that *multiple* localized perturbations have on a *single* edge for the optimal solution, independently of the number of perturbations being considered. We refer to this type of decay of correlation as *point-to-set*. To illustrate in more detail these two types of decay of correlation, and for the sake of simplicity, we consider perturbations that are supported on exactly two vertices, corresponding to the endpoints of edges. Given $b \in \mathbb{R}^V$ such that $\mathbb{1}^T b = 0$, and $e = (u,v) \in \vec{E}$, we define the directional derivative of $x^\star$ along edge e evaluated at b as $\nabla_e x^\star(b) := \frac{dx^\star(b + \varepsilon(e_u - e_v))}{d\varepsilon}|_{\varepsilon=0}$, where for each $v \in V$, $e_v \in \mathbb{R}^V$ is the vector defined as $(e_v)_w = 0$ if $w \neq v$ and $(e_v)_v = 1$. Then, we immediate have the following corollary of Theorem 4.

Corollary 2 (Set-to-point decay of correlation)**.** *For* $b \in \mathbb{R}^V$ *with* $\mathbb{1}^T b = 0$, $\vec{F} \subseteq \vec{E}$ *and* $e \in \vec{E}$, *we have*

$$\|\nabla_e x^\star(b)\|_{\vec{F}} \equiv \sqrt{\sum_{f \in \vec{F}} (\nabla_e x^\star(b)_f)^2} \leq \sqrt{2} c \, \frac{\lambda^{d(V_{\vec{F}}, V_{\{e\}})}}{1-\lambda},$$

where c *is as in Theorem 4.*

The key feature of the bound in Corollary 2 is that it does not depend on the cardinality of $\vec{F}$. Exploiting the symmetry of the identities involving the graph Laplacian, it is also easy to establish the following analogous result. The proof is given in Appendix D.

Lemma 2 (Point-to-set decay of correlation)**.** *For* $b \in \mathbb{R}^V$ *with* $\mathbb{1}^T b = 0$, $\vec{F} \subseteq \vec{E}$ *and* $f \in \vec{E}$, *we have*

$$\sqrt{\sum_{e \in \vec{F}} (\nabla_e x^\star(b)_f)^2} \leq \sqrt{2} c' \, \frac{\lambda^{d(V_{\vec{F}}, V_{\{f\}})}}{1-\lambda},$$

with $c' = \sup_{b \in \mathrm{Im}(A)} \frac{W(b)_{wz} \max_{v \in V_{\vec{F}}} \sqrt{2|\mathcal{N}(v) \cap V_{\vec{F}}|}}{\sqrt{\min\{d(b)_w, d(b)_z\}} \min_{v \in V_{\vec{F}}} \sqrt{d(b)_v}}$.

V. Localized algorithms and bias-variance

Let us consider the network flow problem defined in the previous section, for a certain external flow $b \in \mathbb{R}^V$ such that $\mathbb{1}^T b = 0$. Let $Z \subseteq V$ and choose $p \in \mathbb{R}^V$ supported

[1] This characterization of eigenvalues for random walks on connected weighted graphs follows from the Perron-Frobenius theory. See [31].

on Z (i.e., $p_v \neq 0$ if and only if $v \in Z$) with $\mathbb{1}^T p = 0$. Assume that we perturb the external flow b by adding p. We want to address the following question: given knowledge of the solution $x^\star(b)$ for the unperturbed problem, what is a computationally efficient algorithm to compute the solution $x^\star(b+p)$ of the perturbed problem? The main idea that we want to exploit is that when locality holds and we can prove a decay of correlation property such as the one established in Theorem 4 for expander graphs, then a localized perturbation of the external flow will affect more the components of $x^\star(b)$ that are close to the perturbed sites on Z. Hence, we expect that only a subset of the components of the solution around Z needs to be updated to meet a prescribed level of error tolerance, yielding savings on the computational complexity.

A. Local problem and localized algorithms

To formalize the argument given above, henceforth let $\vec{G}' = (V', \vec{E}')$ be a subgraph of $\vec{G} = (V, \vec{E})$ such that $Z \subseteq V'$. Let $G' = (V', E')$ be the undirected graph associated to $\vec{G}'$ (see notation in Section I), and assume that G' is connected. Define $V'^C := V \setminus V'$ and $\vec{E}'^C := \vec{E} \setminus \vec{E}'$. Let us consider the localized version of the network flow problem supported on the subgraph $\vec{G}'$. Let $A' := A_{V', \vec{E}'}$ be the submatrix of A corresponding to the rows indexed by V' and the columns indexed by $\vec{E}'$. For $b' \in \mathrm{Im}(A') \subseteq \mathbb{R}^{V'}$, let $x'^\star(b') \in \mathbb{R}^{\vec{E}'}$ denote the solution of the following problem over $x' \in \mathbb{R}^{\vec{E}'}$:

$$\begin{aligned} \text{minimize} \quad & f'(x') := \sum_{e \in \vec{E}'} f_e(x'_e) \\ \text{subject to} \quad & A'x' = b'. \end{aligned} \tag{2}$$

If locality holds and the subgraph $\vec{G}'$ is large enough, we expect that the solution of the perturbed problem $x^\star(b+p)$ will be close to the solution of the unperturbed problem $x^\star(b)$ on $\vec{E}'^C$, and it will be substantially different on $\vec{E}'$. For this reason we investigate the performance of local iterative algorithms that operate only on the subgraph $\vec{G}'$, leaving the components supported on $\vec{E}'^C$ unchanged.

Definition 1 (Local algorithm)**.** *Given $b \in \mathbb{R}^V$ with $\mathbb{1}^T b = 0$, let $\mathcal{X}'_b := \{u \in \mathbb{R}^{\vec{E}} : (Au)_{V'^C} = b_{V'^C}\}$. A map $T'_b : \mathcal{X}'_b \to \mathcal{X}'_b$ defines a* local algorithm *on the subgraph $\vec{G}' = (V', \vec{E}')$ if the following two conditions hold for any choice of $x \in \mathcal{X}'_b$:*

(i) $\lim_{t\to\infty} T'^t_b(x)_{E'} = x'^\star(b')$, $b' = b_{V'} - A_{V', \vec{E}'^C} x_{\vec{E}'^C}$;

(ii) $T'_b(x)_{\vec{E}'^C} = x_{\vec{E}'^C}$.

This definition ensures that a local algorithm T'_b only updates the components of $x \in \mathcal{X}'_b$ supported on $\vec{E}'$ and there converges to the solution of problem (2) with $b' = b_{V'} - A_{V', \vec{E}'^C} x_{\vec{E}'^C}$. The components that are left invariant on $\vec{E}'^C$ play the role of boundary conditions. The algorithm that we propose to compute $x^\star(b+p)$ given knowledge of $x^\star(b)$ amounts to running a local algorithm on $\vec{G}'$ for t iterations with boundary conditions $x^\star(b)_{\vec{E}'^C}$, i.e., $T'^t_{b+p}(x^\star(b))$. By definition

$$\lim_{t\to\infty} T'^t_{b+p}(x^\star(b))_e = \begin{cases} x'^\star(b_{V'} + p_{V'} - A_{V', \vec{E}'^C} x^\star(b)_{\vec{E}'^C})_e & \text{if } e \in \vec{E}', \\ x^\star(b)_e & \text{if } e \in \vec{E}'^C. \end{cases}$$

Designing a local algorithm to compute $x^\star(b+p)$ is easy. Given any algorithmic procedure (for instance, a first-order method, a second-order method, or a primal-dual method), we can define its localized version by applying the algorithm to problem (2) with $b' = b_{V'} + p_{V'} - A_{V', \vec{E}'^C} x^\star(b)_{\vec{E}'^C}$. We now provide a general analysis of the comparison between the performance of a given algorithm and its localized version.

B. Bias-variance decomposition and trade-off

The error committed by a chosen local algorithm T'_{b+p} on the subgraph $\vec{G}'$ after $t \geq 1$ iterations is given by

$$\mathrm{Error}(\vec{G}', t) := x^\star(b+p) - T'^t_{b+p}(x^\star(b)).$$

The analysis that we give is based on the error decomposition

$$\mathrm{Error}(\vec{G}', t) = \mathrm{Bias}(\vec{G}') + \mathrm{Var}(\vec{G}', t),$$

with

$$\mathrm{Bias}(\vec{G}') := x^\star(b+p) - \left\{ \lim_{t\to\infty} T'^t_{b+p}(x^\star(b)) \right\}, \tag{3}$$

$$\mathrm{Var}(\vec{G}', t) := \left\{ \lim_{t\to\infty} T'^t_{b+p}(x^\star(b)) \right\} - T'^t_{b+p}(x^\star(b)). \tag{4}$$

This decomposition resembles the bias-variance decomposition in statistics, which motivates the choices of the terminology we use. In statistics, the term "bias" typically refers to the *approximation* error that is made from the simplifying assumptions built into the learning method, while the term "variance" refers to the *estimation* error that is made from the fluctuations of the learning method (trained on a given sample size) around its mean. Analogously, in our setting the bias term (3) represents the error that is made from the model restriction that we consider, namely, the localization of the chosen algorithmic procedure. The variance term (4) represents the error that is made by the deviation of the estimate given by the localized algorithm (at a given time t) from the optimal solution of the localized model. The bias term is algorithm-independent and it characterizes the error that we commit by localizing the optimization procedure per se, as a function of the subgraph $\vec{G}'$. The variance term depends on the algorithm that we run on $\vec{G}'$ and on time.

We now show that in some regimes the bias introduced by localization can be exploited to lower the computational complexity associated to the variance term and yield savings for local algorithms. This bias-variance trade-off further motivates our analogy with statistics and the terminology we use.

Assume that we want to compare the computational complexity of a global algorithm versus its localized counterpart to achieve a prescribed error accuracy $\varepsilon > 0$. Let $t' := \min\{t > 0 : \|\mathrm{Var}(\vec{G}', t)\| \leq \varepsilon\}$ be the minimal number of iterations that allows the localized algorithm $T'_{b+p}(x^\star(b))$ to achieve a variance error (measured in the ℓ_2-norm) less than ε. Let $\kappa(\vec{G}', 1/\varepsilon)$ be the computational cost needed to run the localized algorithm for t' iterations. Typically, $\kappa(\vec{G}', 1/\varepsilon)$ scales polynomially with $|V'|$ and $|\vec{E}'|$; it scales polynomially, logarithmically, or double-logarithmically with $1/\varepsilon$, depending on the regularity assumptions for the optimization problem and on the algorithmic procedure being used. For the sake of illustration, we assume that $\kappa(\vec{G}', 1/\varepsilon)$ is only a function of

$|V'|$ and $1/\varepsilon$, so we write $\kappa(|V'|, 1/\varepsilon)$. The global algorithm ($\vec{G}' = \vec{G}$) has zero bias, so the computational cost $\kappa(|V|, 1/\varepsilon)$ guarantees to achieve $\|\operatorname{Error}(\vec{G}, t)\| \leq \varepsilon$. The localized algorithm, on the other hand, has a non-zero bias due to localization. However, it will have a smaller computational cost for the variance term, as the algorithm runs on a subgraph of $\vec{G}$. To investigate the regime in which the added bias yields computational savings for the overall error term, we proceed as follows. By the triangle inequality, we have

$$\|\operatorname{Error}(\vec{G}', t)\| \leq \|\operatorname{Bias}(\vec{G}')\| + \|\operatorname{Var}(\vec{G}', t)\|.$$

Assume that we can prove a bound of the following form:

$$\|\operatorname{Bias}(\vec{G}')\| \leq \varphi \frac{1}{|V'|^{\theta}}, \tag{5}$$

for given universal constants $\varphi, \theta > 0$. Then, the choice $|V'| \geq (2\varphi/\varepsilon)^{1/\theta}$ guarantees that $\|\operatorname{Bias}(\vec{G}')\| \leq \varepsilon/2$. By requiring $\|\operatorname{Var}(\vec{G}', t)\| \leq \varepsilon/2$ we find that the computational cost $\kappa((2\varphi/\varepsilon)^{1/\theta}, 2/\varepsilon)$ will guarantee that the localized algorithm also achieves $\|\operatorname{Error}(\vec{G}', t)\| \leq \varepsilon$. This argument shows that the localized algorithm is preferable when $\kappa((2\varphi/\varepsilon)^{1/\theta}, 2/\varepsilon) < \kappa(|V|, 1/\varepsilon)$, for instance. The regime where localized algorithms yield computational savings depends on a variety of factors: the regularity assumptions, the algorithm being chosen, the dimension of the original graph, and the required error tolerance. In the following we provide concrete settings where we can establish (5) both theoretically and numerically, and investigate the computational savings due to the bias-variance trade-off. The ideas here illustrated suggest a general framework to study the trade-off between accuracy and complexity for local algorithms in network optimization.

C. Well-conditioned setting and projected gradient descent

We now consider one of the simplest settings where we can establish locality and illustrate the computational savings that can be achieved by localizing a given optimization procedure. Henceforth, let each function f_e be α-strongly convex and β-smooth for some given parameters $\alpha, \beta \in (0, \infty)$, i.e., $0 < \alpha \leq \frac{d^2 f_e(x)}{dx^2} \leq \beta < \infty$ for any $x \in \mathbb{R}, e \in \vec{E}$. In this well-conditioned case it is well-known that gradient descent converges with a number of iterations that scales logarithmically with $1/\varepsilon$. Even in this favorable case, however, the computational savings achieved by a local algorithm can be considerable, as we now show.

For the sake of illustration, let us consider projected gradient descent. Let $\Pi_{\mathcal{X}}$ be the projection operator on a set $\mathcal{X}$, defined as $\Pi_{\mathcal{X}}(x) := \operatorname{argmin}_{u \in \mathcal{X}} \|x - u\|$. The localized projected gradient descent on $\vec{G}'$ is defined as follows.

Definition 2 (Localized projected gradient descent)**.** *Given $x \in \mathcal{X}'_b$, define the set $\mathcal{X}'_b(x) := \{u \in \mathbb{R}^{\vec{E}'} : A_{V',\vec{E}'} u_{\vec{E}'} = b_{V'} - A_{V',\vec{E}'^C} x_{\vec{E}'^C}\}$. Localized projected gradient descent with step size $\eta > 0$ is the local algorithm defined by*

$$T'_b(x)_e = \begin{cases} \Pi_{\mathcal{X}'_b(x)}(x_{\vec{E}'} - \eta \nabla f'(x_{\vec{E}'}))_e & \text{if } e \in \vec{E}', \\ x_e & \text{if } e \in \vec{E}'^C. \end{cases}$$

When $\vec{G}' = \vec{G}$, this algorithm recovers the global algorithm applied to the whole graph $\vec{G}$. In the setting we consider, a classical result yields that projected gradient descent with step size $\eta = 1/\beta$ converges to the optimal solution for any starting point. This corresponds to condition (i) in Definition 1. The algorithm converges exponentially fast, with $\|T'^t_b(x)_{\vec{E}'} - x'^{\star}(b')_{\vec{E}'}\| \leq e^{-t/(2Q)} \|x_{\vec{E}'} - x'^{\star}(b')_{\vec{E}'}\|$, where $Q = \beta/\alpha$ is the so-called *condition number* (see [32][Theorem 3.6], for instance). Under the (common) assumption that $\|x_{\vec{E}'} - x'^{\star}(b')_{\vec{E}'}\| \leq R$, for a certain universal constant $R > 0$, the above convergence rate tells us that in order to reach a prescribed level of error accuracy ε, it is sufficient to run projected gradient descent for a number of iterations that does not depend on the dimension (nor on the topology) of the subgraph $\vec{G}'$ it is applied to, and that only scales logarithmically in $1/\varepsilon$.

The bias-variance tradeoff exploits the fact that the computational cost per iteration *does* depend on the dimension, even if the number of iterations is dimension-free. In the case of projected gradient descent the cost per iteration is dominated by the cost of computing the projection step, and in general this cost scales polynomially with the graph size. To be precise, it can be seen that

$$\begin{aligned} T'_{b+p}(x)_{\vec{E}'} = {} & (I - A'^T L'^{+} A')(x_{\vec{E}'} - \eta \nabla f'(x_{\vec{E}'})) \\ & + A'^T L'^{+} (b_{V'} + p_{V'} - A_{V',\vec{E}'^C} x_{\vec{E}'^C}), \end{aligned} \tag{6}$$

where $L' := A' A'^T$ is the graph Laplacian of the subgraph G'. Computing the pseudoinverse of L' exactly has a cost that scales like $O(|V'|^{\omega})$, where $\omega > 2$ is the so-called matrix multiplication constant.[2] In this case we have that $\kappa(\vec{G}', 1/\varepsilon)$ scales like $O(|V'|^{\omega} + |\vec{E}'| \log(1/\varepsilon))$. If we consider graphs that have largest degree bounded above by a universal constant k, we have $|\vec{E}'| \leq (k/2)|V'|$ and we can use the argument given at the end of Section V-B to state the following result.

Proposition 2 (Computational cost, global versus localized)**.** *Under the assumptions given in this section, assuming that* (5) *holds, projected gradient descent is guaranteed to compute a solution with accuracy ε with a cost that scales as:*

- *Global algorithm: $O(|V|^{\omega} + |V| \log(1/\varepsilon))$;*
- *Localized algorithm: $O((1/\varepsilon)^{\omega/\theta} + (1/\varepsilon)^{1/\theta} \log(1/\varepsilon))$.*

This result shows that the computational savings achieved by localized algorithms can be substantial even in a well-conditioned setting, in the case when $|V| \gg (1/\varepsilon)^{1/\theta}$, i.e., when the graph $\vec{G}$ is very large compared to the inverse of the error tolerance $1/\varepsilon$. In the next two sections we investigate this regime both in theory (for expanders) and in simulations (for expanders and grids).

D. Expander graphs

In this section we assume that G is an expander graph. Using the decay of correlation property in Theorem 4, we

[2] If we relax the requirement of performing an exact projection, we can consider efficient quasi-linear solvers to approximately compute the inverse of L' up to precision δ [33]. These solvers have a complexity that scales like $\widetilde{O}(|\vec{E}'| \log |V'| \log(1/\delta))$. Even in this case, however, the savings of localized algorithms are still considerable in appropriate regimes (essentially, the same argument that we provide in the main text holds with $\omega \approx 1$).

provide upper bounds for the bias and variance terms defined in (3) and (4) as a function of the subgraph $\vec{G}'$ and time t.

Let define the *inner boundary* of $\vec{G}'$ as $\Delta(\vec{G}') := \{v \in V' : \mathcal{N}(v) \cap V'^C \neq \varnothing\}$, which represents the subset of vertices in V' that have at least one vertex neighbor outside V' (in the undirected graph G). Let $B \in \mathbb{R}^{V \times V}$ be the *vertex-vertex adjacency matrix* of the undirected graph $G = (V, E)$, which is the symmetric matrix defined as $B_{uv} := 1$ if $\{u, v\} \in E$, $B_{uv} := 0$ otherwise. Being real and symmetric, the matrix B has $n := |V|$ real eigenvalues which we denote by $\mu_n \leq \mu_{n-1} \leq \cdots \leq \mu_2 \leq \mu_1$. Let $\mu := \max\{|\mu_2|, |\mu_n|\}$ be the second largest eigenvalue in magnitude of B. The next theorem yields bounds for the bias and variance error terms in the ℓ_2-norm. The proof of this theorem is in Appendix E.

Theorem 5 (Error localized algorithm)**.** *Let k_- and k_+ be, respectively, the minimum and maximum degree of G, $Q = \beta/\alpha$, and $\rho := \frac{Qk_+}{k_-} - 1 + \frac{Q}{k_-}\mu$. If $\rho < 1$, then*

$$\|\operatorname{Bias}(\vec{G}')\| \leq \|p\|\, \gamma\, \frac{\rho^{d(\Delta(\vec{G}'), Z)}}{(1-\rho)^2}\, \mathbf{1}_{\vec{G}' \neq \vec{G}} \qquad \textit{(algorithm-free)}$$

$$\|\operatorname{Var}(\vec{G}', t)\| \leq \|p\|\, c\, \frac{e^{-t/(2Q)}}{1-\rho} \quad \textit{(projected gradient descent)}$$

where $\gamma := c(1 + c\sqrt{k_+ - 1})$ and $c := \frac{\sqrt{2k_+}}{k_-}Q$.

The bound for the bias decays exponentially with respect to the graph-theoretical distance (i.e., the distance in the undirected graph G) between the inner boundary of $\vec{G}'$, i.e., $\Delta(\vec{G}')$, and the region where the perturbation p is supported, i.e., $Z \subseteq V$. The rate is governed by the eigenvalue μ, the condition number Q, and the maximum/minimum degree of the graph. The bound for the variance decays exponentially with respect to the running time, with rate proportional to $1/Q$. We highlight that the constants appearing in the bounds in Theorem 5 do not depend on the choice of the subgraph $\vec{G}'$ of $\vec{G}$, but depend only on μ, Q, k_+, and k_- (as the proofs in Appendix E attest, a more refined analysis can yield better constants that do depend on the choice of $\vec{G}'$). In particular, the same constants apply for the analysis of the global algorithm. In this case, the bias term is zero, so that the error equals the variance component (see the indicator function in Theorem 5).

To investigate the computational savings achieved by localized gradient descent in the case of expanders, we will use Theorem 5 to derive a bound for the bias term as in (5) and then invoke Proposition 2. To this end, for the sake of simplicity, let G be a k-regular graph, where each vertex has k neighbors ($k_+ = k_- = k$). Let us introduce a collection of subgraphs that are centered on a given vertex and are parametrized by their radii. Namely, fix a vertex $v \in V$, let $V_r' := \{w \in V : d(v, w) \leq r\}$ denote the ball of radius $r > 0$ around vertex $v \in V$, and let $\vec{G}_r' := (V_r', \vec{E}_r')$ be the subgraph of $\vec{G}$ that has vertex set V_r' and induced edge set $\vec{E}_r'$. Consider a perturbation vector $p \in \mathbb{R}^V$ that is supported on $Z := V_z'$, for a fixed $z \geq 1$. If we run the localized algorithm on $\vec{G}_r'$, with $r > z$, using the trivial bound $|V_r'| \leq k^r$, Theorem 5 yields bound (5) with $\varphi = \|p\|\gamma\frac{1}{(1-\rho)^2\rho^z}$ and $\theta = \log(1/\rho)/\log k$. As $|E_r'| \leq (k/2)|V_r'|$, we have that the global projected gradient descent achieves error tolerance ε with a cost that scales like $O(|V|^\omega + (k/2)|V|\log(1/\varepsilon))$, while its localized counterpart achieves the same accuracy with a cost that scales like $O((2\varphi/\varepsilon)^{\omega/\theta} + (k/2)(2\varphi/\varepsilon)^{1/\theta}\log(2/\varepsilon))$.

Remark 2 (General graph topologies)**.** *While the theoretical analysis that we present only holds for expanders, the main idea applies to any graph topology. In general, one needs to use Theorem 3 to establish locality results in the same spirit of Theorem 4 and Theorem 5, possibly establishing different rates of decay (i.e., not exponential) for different graphs, as discussed in Section IV-A and as we will see in Section V-E.*

E. Numerical Simulations

Let us consider the quadratic problem obtained with the choice $f_e(x) = \tau_e x^2/2$ for any $e \in \vec{E}$, where each τ_e is an independent sample from the uniform distribution in $[1, 2]$. The function $f = \sum_{e \in \vec{E}} f_e$ is α-strongly convex and β-smooth with $\alpha = 1$ and $\beta = 2$, as the Hessian $\nabla^2 f$ is diagonal with entries $\frac{d^2 f_e(x)}{dx^2} = \tau_e$. In this case the optimal solution of the optimization problem can be computed analytically. If we define $\Sigma := (\nabla^2 f)^{-1}$ and let $\Sigma' := \Sigma_{V',V'}$ be the submatrix indexed by V', we have $x^\star(b) = \Sigma A^T(A\Sigma A^T)^+ b$ and $x'^\star(b') = \Sigma' A'^T(A'\Sigma' A'^T)^+ b'$. Using these expressions we can compute the bias term (3). For an arbitrary $v \in V$, we take $Z = \{w \in V : d(v, w) \leq 1\}$. We draw the components of the vector b and the non-zero components of the vector p independently from the uniform distribution in $[-1, 1]$, imposing the conditions $\mathbb{1}^T b = 0$ and $\mathbb{1}^T p = 0$ (this is done by modifying an arbitrary component of the randomly-generated vectors to impose that the sum of the components equals zero). Given $r > 0$, let $\vec{G}_r' = (V_r', \vec{E}_r')$ be the subgraph of $\vec{G}$ that includes all the vertices within a distance r from v.

We consider three graphs with 900 nodes each: a cycle, a two-dimensional square grid with periodic boundary conditions, and a 3-regular expander (uniformly sampled from the family of 3-regular graphs).

Figure 1 shows the behavior of the ℓ_2-norm of the bias term as a function of the radius r and the size $|V_r'|$ (i.e., number of vertices) of the subgraph $\vec{G}_r'$ for the three graph topologies of interest. Each topology gives rise to a different behavior. For the cycle graph, the bias error stays constant while $V_r' \neq V$ and drops to zero only when $V_r' = V$. This illustrates the fact that in the cycle there is no decay of correlation as perturbations do not dissipate. Borrowing again terminology from probability theory, we can say that the cycle represents a case of "long-range dependence." For the two-dimensional grid, the bias decays polynomially with r while the subgraph $\vec{G}_r'$ does not reach the boundaries of $\vec{G}$ (as we conjectured, see Section IV-A), and then decays at a faster rate. For the expander, the bias decays exponentially with r, as we proved in Theorem 5. With respect to $|V_r'|$, the decay of the bias for both grids and expanders aligns with the polynomial ansatz in (5). This polynomial decay for expanders was proved in Section V-D.

Figure 2 shows the behavior of the variance term for the localized projected gradient descent algorithm. As prescribed by classical results, and as we proved more precisely in the case of expanders (Theorem 5), the algorithm converges exponentially fast and the rate of convergence can be upper

bounded by a quantity that does not depend on the graph topology nor on the graph size. As described in Section V-C, the cost incurred by the algorithm to achieve a given accuracy for the variance is seen to increase polynomially with $|V_r'|$.

Finally, Figure 3 compares the computational cost of global and localized algorithms to achieve the same level of error accuracy. This plot aligns with the findings of Proposition 2 in the case of expanders. In particular, the plot shows that the computational savings achieved by localized algorithms are considerable in the regime $1/\varepsilon \ll |V|$.

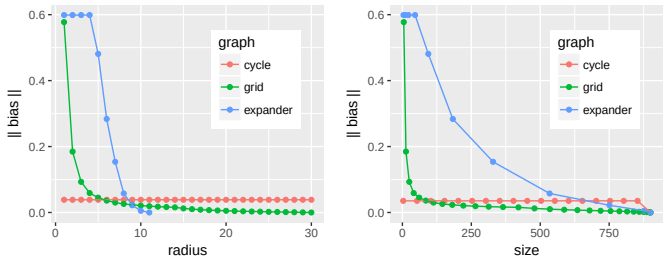


Fig. 1. Typical realizations of $\|\operatorname{Bias}(\vec{G}_r')\|$ as a function of the radius r (left) and the size $|V_r'|$ (right) of the subgraph $\vec{G}_r' = (V_r', \vec{E}_r')$. The value of the bias for the expander is zero for any $r \geq 11$, as $V_r' = V$ in this case.

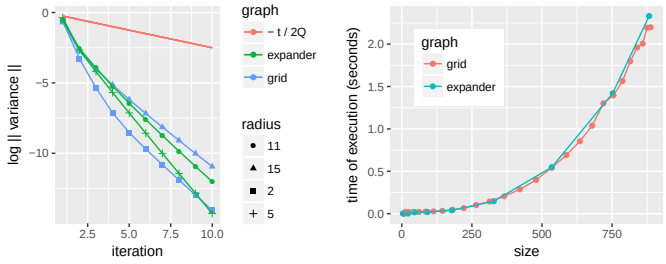


Fig. 2. (Left) Typical realizations of $\log \|\operatorname{Var}(\vec{G}_r', t)\|$ for projected gradient descent as a function of the iteration step t, for different graph topologies and different choices of the radius r. We do not plot results for the cycle as in that case the algorithm converges in one iteration as there is a unique solution to $Ax = b$. We also plot the theoretical upper bound $-t/2Q$, with $Q = \beta/\alpha = 2$. (Right) Time of execution to run localized projected gradient descent to achieve $\|\operatorname{Var}(\vec{G}_r', t)\| \leq 10^{-7}$ as a function of $|V_r'|$. We use the `ginv` function from the `MASS` library in `R` to compute the pseudoinverse of the Laplacian matrix L' and run the algorithm as described in (6).

VI. Conclusions

The main contribution of this paper is to derive a general analogy between natural concepts in probability and statistics (i.e., notions of correlation among random variables, decay of correlation, and bias-variance decomposition and trade-off) and similar notions that can be introduced in optimization. In this paper we have proposed notions of correlation that are based on the sensitivity of optimal points. We have illustrated how decay of correlation (locality) can be established in a canonical network optimization problem (min-cost network flow), and how it can be used to design local algorithms to compute the solution of a problem after localized perturbations are made to the system. In principle, the framework that we propose can be applied to *any* optimization problem. The key point is to establish decay of correlation for the problem at

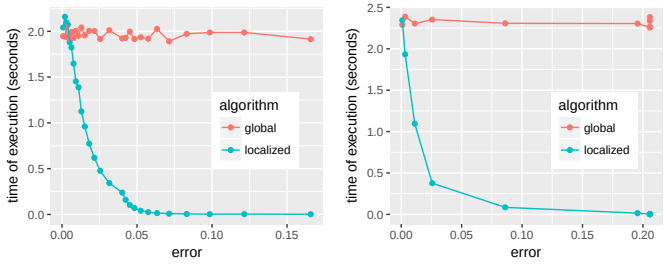


Fig. 3. Time of execution to run projected gradient descent (gobal and localized) to achieve $\|\operatorname{Error}(\vec{G}_r', t)\| \leq \varepsilon + 10^{-7}$ as a function of the error tolerance parameter ε, for the grid (left) and the expander (right). As in the well-conditioned setting that we examine the algorithm drives the variance error to zero exponentially fast (in terms of number of iterations, see Figure 2), the error is dominated by the bias term. As a consequence, for the localized algorithm we adopt the strategy to choose the smallest radius r such that $\|\operatorname{Bias}(\vec{G}_r')\| \leq \varepsilon$ and then run the localized algorithms on $\vec{G}_r'$ for the smallest number of iterations t such that $\|\operatorname{Var}(\vec{G}_r', t)\| \leq 10^{-7}$.

hand, deriving results that are analogous to the ones established in Section IV-B. In the case of the min-cost network flow problem, we showed that establishing locality reduces to bounding the discrete derivative of the Green's function of the diffusion random walk, as described in Theorem 3. We proved an exponential decay for the correlation in expander graphs (as a function of the distance from the perturbation), and we conjectured and provided numerical evidence for a polynomial decay in grid-like topologies. Once results on locality are established for the particular problem at hand, these results translate into a bound for the bias term of the error decomposition of localized algorithms, as the one that we give in Section V-D. The analysis of the variance term, on the other hand, depends on the algorithm that one wants to localize. This part is not technically difficult, as it amounts to analyzing the performance of the chosen algorithm when applied to a subgraph with frozen boundary conditions. Establishing locality in more general settings and problems, and extending the ideas here presented to *cold-start* scenarios (where one wants to compute the optimal solution of an optimization problem starting from possibly any initial condition) remain open questions for future investigation.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[2] B. Johansson, M. Rabi, and M. Johansson, "A randomized incremental subgradient method for distributed optimization in networked systems," *SIAM J. on Optimization*, vol. 20, no. 3, pp. 1157–1170, 2009.

[3] D. Mosk-Aoyama, T. Roughgarden, and D. Shah, "Fully distributed algorithms for convex optimization problems," *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3260–3279, 2010.

[4] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Automat. Contr.*, vol. 57, no. 3, pp. 592–606, 2012.

[5] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed Newton method for network utility maximization-I: Algorithm," *IEEE Transactions on Automatic Control*, vol. 58, no. 9, pp. 2162–2175, 2013.

[6] ——, "A distributed Newton method for network utility maximization-II: Convergence," *IEEE Transactions on Automatic Control*, vol. 58, no. 9, pp. 2176–2188, 2013.

[7] S. Krantz and H. Parks, *The Implicit Function Theorem: History, Theory, and Applications*, ser. The Implicit Function Theorem: History, Theory, and Applications. Birkhäuser, 2002.

[8] R. L. Dobrušin, "Definition of a system of random variables by means of conditional distributions," *Teor. Verojatnost. i Primenen.*, vol. 15, pp. 469–497, 1970.
[9] P. Rebeschini and R. van Handel, "Comparison theorems for Gibbs measures," *Journal of Statistical Physics*, vol. 157, no. 2, pp. 234–281, 2014.
[10] S. C. Tatikonda and M. I. Jordan, "Loopy belief propagation and Gibbs measures," in *Proc. UAI*, vol. 18, 2002, pp. 493–500.
[11] D. Weitz, "Counting independent sets up to the tree threshold," in *Proceedings of the Thirty-eighth Annual ACM Symposium on Theory of Computing*. ACM, 2006, pp. 140–149.
[12] G. Bresler, E. Mossel, and A. Sly, "Reconstruction of markov random fields from samples: Some observations and algorithms," in *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2008, vol. 5171, pp. 343–356.
[13] D. Gamarnik, D. A. Goldberg, and T. Weber, "Correlation decay in random decision networks," *Mathematics of Operations Research*, vol. 39, no. 2, pp. 229–261, 2014.
[14] P. Rebeschini and R. van Handel, "Can local particle filters beat the curse of dimensionality?" *Ann. Appl. Probab.*, vol. 25, no. 5, pp. 2809–2866, 10 2015.
[15] C. C. Moallemi and B. Van Roy, "Convergence of min-sum message-passing for convex optimization," *Information Theory, IEEE Transactions on*, vol. 56, no. 4, pp. 2041–2050, 2010.
[16] D. Gamarnik, D. Shah, and Y. Wei, "Belief propagation for min-cost network flow: Convergence and correctness," *Operations Research*, vol. 60, no. 2, pp. 410–428, 2012.
[17] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
[18] N. K. Vishnoi, "Lx = b," *Foundations and Trends® in Theoretical Computer Science*, vol. 8, no. 1–2, pp. 1–141, 2013.
[19] M. Belkin and P. Niyogi, "Semi-supervised learning on riemannian manifolds," *Machine Learning*, vol. 56, no. 1, pp. 209–239, 2004.
[20] R. Kyng, A. Rao, S. Sachdeva, and D. A. Spielman, "Algorithms for lipschitz learning on graphs." in *COLT*, ser. JMLR Workshop and Conference Proceedings, vol. 40, 2015, pp. 1190–1223.
[21] G. Lawler and V. Limic, *Random Walk: A Modern Introduction*, ser. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2010.
[22] C. Archetti, L. Bertazzi, and M. G. Speranza, "Reoptimizing the traveling salesman problem," *Networks*, vol. 42, no. 3, pp. 154–159, 2003.
[23] J. Guddat, "Stability in convex quadratic parametric programming," *Mathematische Operationsforschung und Statistik*, vol. 7, no. 2, pp. 223–245, 1976.
[24] S. M. Robinson, *Generalized equations and their solutions, Part I: Basic theory*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1979, pp. 128–141.
[25] A. V. Fiacco, *Introduction to sensitivity and stability analysis in nonlinear programming / Anthony V. Fiacco*. Academic Press New York, 1983.
[26] J. Kyparisis, "Uniqueness and differentiability of solutions of parametric nonlinear complementarity problems," *Mathematical Programming*, vol. 36, no. 1, pp. 105–113, 1986.
[27] H. Phu and N. Yen, "On the stability of solutions to quadratic programming problems," *Mathematical Programming*, vol. 89, no. 3, pp. 385–394, 2001.
[28] J. C. G. Boot, "On sensitivity analysis in convex quadratic programming problems," *Operations Research*, vol. 11, no. 5, pp. 771–786, 1963.
[29] H.-O. Georgii, *Gibbs measures and phase transitions*, 2nd ed., ser. de Gruyter Studies in Mathematics. Berlin: Walter de Gruyter & Co., 2011, vol. 9.
[30] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.
[31] L. Lovász, "Random walks on graphs: A survey," *Combinatorics, Paul Erdos is Eighty*, vol. 2, no. 1, pp. 1–46, 1993.
[32] S. Bubeck, "Convex optimization: Algorithms and complexity," *Foundations and Trends® in Machine Learning*, vol. 8, no. 3-4, pp. 231–357, 2015.
[33] I. Koutis, G. L. Miller, and R. Peng, "A nearly-m log n time solver for sdd linear systems," in *FOCS*. IEEE, 2011, pp. 590–598.

APPENDIX A
HADAMARD'S GLOBAL INVERSE THEOREM

We prove Theorem 1. The proof relies on Hadamard's global inverse function theorem, which characterizes when a C^k function is a C^k diffeomorphism. Recall that a function from $\mathbb{R}^m$ to $\mathbb{R}^m$ is said to be C^k if it has continuous derivatives up to order k. A function is said to be a C^k *diffeomorphism* if it is C^k, bijective, and its inverse is also C^k. In the online version of the manuscript we include all the details regarding Hadamard's global inverse function theorem. The following corollary is the backbone behind Theorem 1.

Lemma 3 (Diffeomorphism for Lagrangian multipliers map)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a strongly convex function, twice continuously differentiable. Let $A \in \mathbb{R}^{m\times n}$ be a given matrix. Define the function Φ from $\mathbb{R}^n \times \mathbb{R}^m$ to $\mathbb{R}^n \times \mathbb{R}^m$ as*

$$\Phi(x,\nu) := \begin{pmatrix} \nabla f(x) + A^T\nu \\ Ax \end{pmatrix},$$

for any $x \in \mathbb{R}^n$, $\nu \in \mathbb{R}^m$. Then, the restriction of the function Φ to $\mathbb{R}^n \times \mathrm{Im}(A)$ is a C^1 diffeomorphism.

Proof. See the online version of the manuscript. □

We now present the proof of Theorem 1.

Proof of Theorem 1. The Lagrangian of the optimization problem is the function $\mathcal{L}$ from $\mathbb{R}^{\mathcal{V}} \times \mathbb{R}^{\mathcal{F}}$ to $\mathbb{R}$ defined as $\mathcal{L}(x,\nu) := f(x) + \sum_{a\in\mathcal{F}} \nu_a (A_a^T x - b_a)$, where A_a^T is the a-th row of the matrix A and $\nu = (\nu_a)_{a\in\mathcal{F}}$ is the vector formed by the Lagrangian multipliers. Let us define the function Φ from $\mathbb{R}^{\mathcal{V}} \times \mathbb{R}^{\mathcal{F}}$ to $\mathbb{R}^{\mathcal{V}} \times \mathbb{R}^{\mathcal{F}}$ as

$$\Phi(x,\nu) := \begin{pmatrix} \nabla_x \mathcal{L}(x,\nu) \\ Ax \end{pmatrix} = \begin{pmatrix} \nabla f(x) + A^T\nu \\ Ax \end{pmatrix}.$$

For any fixed $\varepsilon \in \mathbb{R}$, as the constraints are linear, the Lagrange multiplier theorem says that for the unique minimizer $x^\star(b(\varepsilon))$ there exists $\nu'(b(\varepsilon)) \in \mathbb{R}^{\mathcal{F}}$ so that

$$\Phi(x^\star(b(\varepsilon)), \nu'(b(\varepsilon))) = \begin{pmatrix} \mathbb{0} \\ b(\varepsilon) \end{pmatrix}. \tag{7}$$

As $A^T(\nu+\mu) = A^T\nu$ for each $\mu \in \mathrm{Ker}(A^T)$, the set of Lagrangian multipliers $\nu'(b(\varepsilon)) \in \mathbb{R}^{\mathcal{F}}$ that satisfies (7) is a translation of the null space of A^T. We denote the unique translation vector by $\nu^\star(b(\varepsilon)) \in \mathrm{Im}(A)$. By Hadamard's global inverse function theorem, as shown in Lemma 3, the restriction of the function Φ to $\mathbb{R}^{\mathcal{V}} \times \mathrm{Im}(A)$ is a C^1 diffeomorphism, namely, it is continuously differentiable, bijective, and its inverse is also continuously differentiable. In particular, this means that the functions $x^\star : b \in \mathrm{Im}(A) \to x^\star(b) \in \mathbb{R}^{\mathcal{V}}$ and $\nu^\star : b \in \mathrm{Im}(A) \to \nu^\star(b) \in \mathrm{Im}(A)$ are continuously differentiable along the subspace $\mathrm{Im}(A)$. Differentiating both sides of (7) with respect to ε, we get

$$\begin{pmatrix} H & A^T \\ A & \mathbb{0} \end{pmatrix} \begin{pmatrix} x' \\ \tilde{\nu} \end{pmatrix} = \begin{pmatrix} \mathbb{0} \\ \frac{db(\varepsilon)}{d\varepsilon} \end{pmatrix},$$

where $H := \nabla^2 f(x^\star(b(\varepsilon)))$, $x' := \frac{dx^\star(b(\varepsilon))}{d\varepsilon}$, $\tilde{\nu} := \frac{d\nu^\star(b(\varepsilon))}{d\varepsilon}$. As the function f is strongly convex, the Hessian $\nabla^2 f(x)$ is positive definite for every $x \in \mathbb{R}^{\mathcal{V}}$, hence it is invertible for every $x \in \mathbb{R}^{\mathcal{V}}$. Solving the linear system for x' first, from the

first equation $Hx' + A^T\tilde{\nu} = \mathbb{0}$ we get $x' = -H^{-1}A^T\tilde{\nu}$. Substituting this expression in the second equation $Ax' = \frac{db(\varepsilon)}{d\varepsilon}$, we get $L\tilde{\nu} = -\frac{db(\varepsilon)}{d\varepsilon}$, where $L := AH^{-1}A^T$. The set of solutions to $L\tilde{\nu} = -\frac{db(\varepsilon)}{d\varepsilon}$ can be expressed in terms of the pseudoinverse of L as follows $\{\tilde{\nu} \in \mathbb{R}^{\mathcal{F}} : L\tilde{\nu} = -\frac{db(\varepsilon)}{d\varepsilon}\} = -L^{+}\frac{db(\varepsilon)}{d\varepsilon} + \mathrm{Ker}(L)$. We show that $\mathrm{Ker}(L) = \mathrm{Ker}(A^T)$. We show that $L\nu = \mathbb{0}$ implies $A^T\nu = \mathbb{0}$, as the opposite direction trivially holds. In fact, let $A' := A\sqrt{H^{-1}}$, where $\sqrt{H^{-1}}$ is the positive definite matrix that satisfies $\sqrt{H^{-1}}\sqrt{H^{-1}} = H^{-1}$. The condition $L\nu = A'A'^T\nu = \mathbb{0}$ is equivalent to $A'^T\nu \in \mathrm{Ker}(A')$. At the same time, clearly, $A'^T\nu \in \mathrm{Im}(A'^T)$. However, $\mathrm{Ker}(A')$ is orthogonal to $\mathrm{Im}(A'^T)$, so it must be $A'^T\nu = \mathbb{0}$ which implies $A^T\nu = \mathbb{0}$ as $\sqrt{H^{-1}}$ is positive definite. As $\mathrm{Im}(L^{+}) = \mathrm{Ker}(L)^{\perp} = \mathrm{Ker}(A^T)^{\perp} = \mathrm{Im}(A)$, we have that $\tilde{\nu} = -L^{+}\frac{db(\varepsilon)}{d\varepsilon}$ is the unique solution to $L\tilde{\nu} = -\frac{db(\varepsilon)}{d\varepsilon}$ that belongs to $\mathrm{Im}(A)$. Substituting this expression into $x' = -H^{-1}A^T\tilde{\nu}$, we finally get $x' = H^{-1}A^TL^{+}\frac{db(\varepsilon)}{d\varepsilon}$. The proof follows as $\Sigma(b) = H^{-1}$. □

APPENDIX B
CORRELATION

In the online version of the manuscript, we provide the proofs of Proposition 1 and Lemma 1 in Section III.

APPENDIX C
GRAPH LAPLACIANS AND RANDOM WALKS

In the online version of the manuscript, we provide a self-contained appendix containing several connections between graph Laplacians and random walks on weighted graphs.

APPENDIX D
DECAY OF CORRELATION

In the online version of the manuscript, we provide the proofs of the decay of correlation properties stated in Section IV, namely, Theorem 4 (set-to-point) and Lemma 2 (point-to-set). These proofs rely on the sensitivity analysis for the network flow problem established in Theorem 3.

APPENDIX E
LOCALIZED ALGORITHM

This section is devoted to the proof of Theorem 5, which states error bounds for the localized projected gradient descent algorithm. The proof relies on the decay of correlation property established in Theorem 4 for the network flow problem. Recall that the constants appearing in the bounds in Theorem 5 do not depend on the choice of the subgraph $\vec{G}'$ of $\vec{G}$, but depend only on μ, Q, k_+, and k_-. To prove this type of bounds, we first need to develop estimates to relate the eigenvalues of weighted subgraphs to the eigenvalues of the corresponding unweighted graph.

A. Eigenvalues interlacing

Let $G = (V, E)$ be a simple (i.e., no self-loops, and no multiple edges), connected, undirected graph, with vertex set V and edge set E. Let $B \in \mathbb{R}^{V\times V}$ be the vertex-vertex adjacency matrix of the graph, which is the symmetric matrix defined as $B_{uv} := 1$ if $\{u, v\} \in E$, $B_{uv} := 0$ otherwise. If $n := |V|$, denote by $\mu_n \le \mu_{n-1} \le \cdots \le \mu_2 \le \mu_1$ the eigenvalues of B. Let $G' = (V', E')$ be a connected subgraph of G. Assume that to each edge $\{u, v\} \in E'$ is associated a non-negative weight $W_{uv} = W_{vu} > 0$, and let $W_{uv} = 0$ if $\{u, v\} \notin E$. Let D' be a diagonal matrix with entries $D'_{vv} = \sum_{w\in V'} W'_{vw}$ for each $v \in V'$. Let $P' := D'^{-1}W'$. If $m := |V'|$, denote by $\lambda'_m \le \lambda'_{m-1} \le \cdots \le \lambda'_2 \le \lambda'_1$ the eigenvalues of P'. The following proposition relates the eigenvalues of P' to the eigenvalues of B. In particular, we provide a bound for the second largest eigenvalue in magnitude of P' with respect to the second largest eigenvalue in magnitude of B, uniformly over the choice of G'.

Proposition 3 (Eigenvalues interlacing)**.** *Let $w_- \le W_{vw} \le w_+$ for any $\{v, w\} \in E$, for some constants $w_-, w_+ > 0$. Let k_- and k_+ be, resp., the min and max degree of G. Then,*

$$1 - \frac{w_+k_+}{w_-k_-} + \frac{w_+}{w_-k_-}\mu_{i+n-m} \le \lambda'_i \le 1 - \frac{w_-k_-}{w_+k_+} + \frac{w_-}{w_+k_+}\mu_i.$$

Therefore, if $\lambda' := \max\{|\lambda'_2|, |\lambda'_m|\}$ and $\mu := \max\{|\mu_2|, |\mu_n|\}$, we have $\lambda' \le \frac{w_+k_+}{w_-k_-} - 1 + \frac{w_+}{w_-k_-}\mu$.

Proof. See the online version of the manuscript. □

B. Proof of Theorem 5

We now present the proof of Theorem 5. The proof relies on repeatedly applying Theorem 4 in Section IV (which captures the decay of correlation for the network flow problem) and the fundamental theorem of calculus.

Proof of Theorem 5. Consider the setting of Section V.
Bias term. Let us first bound the bias outside $\vec{E}'$. Let $n := |V|$, and for each $b \in \mathrm{Im}(A)$ let $-1 \le \lambda_n(b) \le \lambda_{n-1}(b) \le \cdots \le \lambda_2(b) < \lambda_1(b) = 1$ be the eigenvalues of $P(b)$. Let $\lambda(b) := \max\{|\lambda_2(b)|, |\lambda_n(b)|\}$ and $\lambda := \sup_{b\in\mathrm{Im}(A)} \lambda(b)$. Define $b(\varepsilon) := b + \varepsilon p$, for any non-negative real number $\varepsilon \ge 0$. If $e \in \vec{E}'^C$, then $T'_{b(\varepsilon)}(x^\star(b))_e = x^\star(b)_e$ and

$$\mathrm{Bias}(\vec{G}')_e = x^\star(b(1))_e - x^\star(b(0))_e = \int_0^1 d\varepsilon\, \frac{dx^\star(b(\varepsilon))_e}{d\varepsilon}.$$

By the triangle inequality for the ℓ_2-norm and Theorem 4,

$$\|\mathrm{Bias}(\vec{G}')\|_{\vec{E}'^C} \le \sup_{\varepsilon\in\mathbb{R}} \left\| \frac{dx^\star(b(\varepsilon))}{d\varepsilon} \right\|_{\vec{E}'^C} \le c\|p\| \frac{\lambda^{d(\Delta(\vec{G}'), Z)}}{1-\lambda},$$

where we used that $\sup_{\varepsilon\in\mathbb{R}} \|\frac{db(\varepsilon)}{d\varepsilon}\|_Z = \|p\|$, as $\frac{db(\varepsilon)_v}{d\varepsilon} = p_v$ for $v \in Z$ and $\frac{db(\varepsilon)_v}{d\varepsilon} = 0$ for $v \notin Z$, and $c := \sqrt{2k_+}Q/k_-$.

Let us now consider the bias inside $\vec{E}'$. Consider problem (2) in Section V. For any $\varepsilon > 0, \theta > 0$, define

$$b'(\varepsilon, \theta) := b(\varepsilon)_{V'} - A_{V',\vec{E}'^C} x^\star(b(\theta))_{\vec{E}'^C}. \tag{8}$$

Without loss of generality, we can index the elements of V' and $\vec{E}'$ so that the matrix A has the following structure:

$$A = \begin{pmatrix} A_{V',\vec{E}'} & A_{V',\vec{E}'^C} \\ A_{V'^C,\vec{E}'} & A_{V'^C,\vec{E}'^C} \end{pmatrix} = \begin{pmatrix} A' & A_{V',\vec{E}'^C} \\ \mathbb{0} & A_{V'^C,\vec{E}'^C} \end{pmatrix}.$$

For any x that satisfies the flow constraints on $\vec{E}'^C$ with respect to $b(\varepsilon)$, namely, $A_{V\setminus V',\vec{E}'^C} x_{\vec{E}'^C} = b(\varepsilon)_{V\setminus V'}$, we have

$$\left\{\lim_{t\to\infty} T'^t_{b+p}(x)\right\}_{\vec{E}'} = x'^\star(b(1)_{V'} - A_{V',\vec{E}'^C} x_{\vec{E}'^C}).$$

Clearly $x^\star(b)$ satisfies the flow constraints on $\vec{E}'^C$ with respect to $b(1)$, as p is supported on V' so that $b(\varepsilon)_{V'^C} = b_{V'^C}$. Recalling the definition of $b'(\varepsilon,\theta)$ in (8), we then have $(\lim_{t\to\infty} T'^t_{b+p}(x^\star(b)))_{\vec{E}'} = x'^\star(b'(1,0))$. On the other hand, as $x^\star(b(1))$ is a fixed point of the map $T'_{b(1)}$, we can characterize the components of $x^\star(b(1))$ supported on $\vec{E}'$ as

$$x^\star(b(1))_{\vec{E}'} = \left\{\lim_{t\to\infty} T'^t_{b(1)}(x^\star(b(1))\right\}_{\vec{E}'} = x'^\star(b'(1,1)).$$

It is easy to check that $b'(\varepsilon,\theta) \in \mathrm{Im}(A')$ for each value of ε and θ. In fact, as $\vec{G}'$ is connected by assumption, then $\mathrm{Im}(A')$ corresponds to the subspace of $\mathbb{R}^{V'}$ orthogonal to the all-ones vector $\mathbb{1}$. We have $\mathbb{1}^T b'(\varepsilon,\theta) = \mathbb{1}^T b_{V'} + \varepsilon \mathbb{1}^T p_{V'} - \mathbb{1}^T A_{V',\vec{E}'^C} x^\star(b(\theta))_{\vec{E}'^C}$. Note that $\mathbb{1}^T p_{V'} = 0$ by assumption. Also, $0 = \mathbb{1}^T b = \mathbb{1}^T b_{V'} + \mathbb{1}^T b_{V'^C}$ so that $\mathbb{1}^T b_{V'} = -\mathbb{1}^T b_{V'^C}$. Analogously, as $\mathbb{1}^T A = \mathbb{0}^T$, we have $\mathbb{1}^T A_{V',\vec{E}'^C} = -\mathbb{1}^T A_{V'^C,\vec{E}'^C}$. Hence,

$$\mathbb{1}^T b'(\varepsilon,\theta) = -\mathbb{1}^T b_{V'^C} + \mathbb{1}^T A_{V'^C,\vec{E}'^C} x^\star(b(\theta))_{\vec{E}'^C} = \mathbb{0}^T,$$

where the last equality follows as clearly $A_{V'^C,\vec{E}'^C} x^\star(b(\theta))_{\vec{E}'^C} = b_{V'^C}$. Therefore, for $e \in \vec{E}'$,

$$\mathrm{Bias}(\vec{G}')_e = \int_0^1 d\theta\, \frac{dx'^\star(b'(1,\theta))_e}{d\theta}.$$

For each $b' \in \mathrm{Im}(A')$, let $W'(b') \in \mathbb{R}^{V'\times V'}$ be a symmetric matrix defined as $W'(b')_{uv} = (\frac{\partial^2 f_e(x'^\star(b')_e)}{\partial x_e^2})^{-1}$ if either $e=(u,w)\in\vec{E}$ or $e=(w,u)\in\vec{E}$, and $W'(b')_{uv} := 0$ otherwise. Let $D'(b') \in \mathbb{R}^{V'\times V'}$ be a diagonal matrix with entries $D'(b')_{vv} = \sum_{w\in V'} W'(b')_{vw}$. Let $P'(b') := D'(b')^{-1} W'(b')$. If $m := |V|$, let $-1 \le \lambda'_m(b') \le \lambda'_{m-1}(b') \le \cdots \le \lambda'_2(b') < \lambda'_1(b') = 1$ be the eigenvalues of $P'(b')$ (where this characterization holds as G' is connected by assumption). Define $\lambda'(b') := \max\{|\lambda'_2(b')|, |\lambda'_m(b')|\}$ and $\lambda' := \sup_{b'\in\mathrm{Im}(A')} \lambda'(b')$. Proceeding as above, applying Theorem 4 to the optimization problem defined on $\vec{G}'$ we get

$$\|\mathrm{Bias}(\vec{G}')\|_{\vec{E}'} \le \sup_{\theta\in\mathbb{R}} \left\| \frac{dx'^\star(b'(1,\theta))}{d\theta} \right\|_{\vec{E}'},$$

which is upper-bounded by $c\,\frac{1}{1-\lambda'} \sup_{\theta\in\mathbb{R}} \|\frac{\partial b'(1,\theta)}{\partial\theta}\|_{\Delta(\vec{G}')}$, where we used that $\frac{\partial b'(\varepsilon,\theta)_v}{\partial\theta} = 0$ if $v \in V'\setminus\Delta(\vec{G}')$, and clearly $d(V',\Delta(\vec{G}')) = 0$ as $\Delta(\vec{G}') \subseteq V'$. For $v\in\Delta(\vec{G}')$ we have $\frac{\partial b'(\varepsilon,\theta)_v}{\partial\theta} = -\sum_{e\in\vec{E}'^C} A_{ve} \frac{dx^\star(b(\theta))_e}{d\theta}$. If $\vec{F}(v) := \{e \in \vec{E}'^C : e=(u,v) \text{ or } e=(v,u) \text{ for some } u \in V'^C\}$ denotes the set of edges that are connected to v but do not belong to $\vec{E}'$, we have $(\frac{\partial b'(1,\theta)_v}{\partial\theta})^2 \le (\sum_{e\in\vec{F}(v)} |\frac{dx^\star(b(\theta))_e}{d\theta}|)^2$, which by Jensen's inequality admits $|\vec{F}(v)| \sum_{e\in\vec{F}(v)} (\frac{dx^\star(b(\theta))_e}{d\theta})^2$ as a upper bound. As $\max_{v\in\Delta(\vec{G}')} |\vec{F}(v)| \le k_+ - 1$, applying Theorem 4 as above we get $\|\frac{\partial b'(1,\theta)}{\partial\theta}\|_{\Delta(\vec{G}')} \le \sqrt{k_+-1}\,\|\frac{dx^\star(b(\theta))}{d\theta}\|_{\vec{E}'^C} \le c\sqrt{k_+-1}\,\|p\|\,\frac{\lambda^{d(\Delta(\vec{G}'),Z)}}{1-\lambda}$. Therefore, $\|\mathrm{Bias}(\vec{G}')\|_{\vec{E}'} \le c^2\sqrt{k_+-1}\,\|p\|\,\frac{\lambda^{d(\Delta(\vec{G}'),Z)}}{(1-\lambda')(1-\lambda)}$. By the triangle inequality for the ℓ_2-norm, $\|\mathrm{Bias}(\vec{G}')\| \le \|\mathrm{Bias}(\vec{G}')\|_{\vec{E}'} + \|\mathrm{Bias}(\vec{G}')\|_{\vec{E}'^C}$, so we obtain

$$\|\mathrm{Bias}(\vec{G}')\| \le c(1+c\sqrt{k_+-1})\,\|p\|\,\frac{\lambda^{d(\Delta(\vec{G}'),Z)}}{(1-\lambda')(1-\lambda)}.$$

By Proposition 3 we have $\max\{\lambda,\lambda'\} \le \frac{Qk_+}{k_-} - 1 + \frac{Q}{k_-}\mu$, and the bound for the bias term follows.

Variance term. As $(\lim_{t\to\infty} T'^t_{b+p}(x^\star(b)))_{\vec{E}'} = x'^\star(b'(1,0))$, we get

$$\begin{aligned}\|\mathrm{Var}(\vec{G}',t)\|_{\vec{E}'} &= \|x'^\star(b'(1,0)) - T'^t_{b+p}(x^\star(b))_{\vec{E}'}\| \\ &\le e^{-\frac{t}{2Q}} \|x'^\star(b'(1,0)) - x'^\star(b'(0,0))\|,\end{aligned}$$

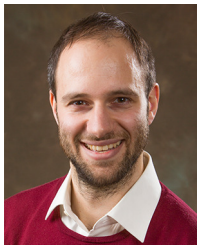
where in the last inequality we used that $x^\star(b)_{\vec{E}'} = x'^\star(b'(0,0))$. For each $e\in\vec{E}'$ we have

$$x'^\star(b'(1,0))_e - x'^\star(b'(0,0))_e = \int_0^1 d\varepsilon\, \frac{dx'^\star(b'(\varepsilon,0))_e}{d\varepsilon},$$

and using the triangle inequality for the ℓ_2-norm, applying Theorem 4 to the optimization problem defined on $\vec{G}'$,

$$e^{\frac{t}{2Q}} \|\mathrm{Var}(\vec{G}',t)\|_{\vec{E}'} \le \sup_{\varepsilon\in\mathbb{R}} \left\| \frac{dx'^\star(b'(\varepsilon,0))}{d\varepsilon} \right\|_{\vec{E}'} \le c\|p\| \frac{1}{1-\lambda'},$$

where we used that $\frac{\partial b'(\varepsilon,0)_v}{\partial\varepsilon} = \frac{db(\varepsilon)_v}{d\varepsilon} = p_v$ for $v\in Z$ and $\frac{db(\varepsilon)_v}{d\varepsilon} = 0$ for $v\notin Z$, and that $d(V',Z) = 0$ as $Z\subseteq V'$. Clearly, $\mathrm{Var}(\vec{G}',t)_e = 0$ for $e\in\vec{E}'^C$, as $T'_b(x^\star(b))_e = x^\star(b)_e$. Hence, $\|\mathrm{Var}(\vec{G}',t)\| = \|\mathrm{Var}(\vec{G}',t)\|_{\vec{E}'}$ and the proof is concluded as $\lambda' \le \frac{Qk_+}{k_-} - 1 + \frac{Q}{k_-}\mu$ by Proposition 3. □



Patrick Rebeschini received the B.S. and M.S. degrees in physics from University of Padova, Italy, in 2006 and 2009, respectively, and the Ph.D. degree in operation research and financial engineering from Princeton University in 2014. Currently, he is an associate professor in the Department of Statistics at the University of Oxford. Prior to that, he was a postdoctoral associate and then an associate research scientist in the Department of Electrical Engineering and in the Yale Institute for Network Science at Yale University, and a lecturer in the Department of Computer Science at Yale University.

Sekhar Tatikonda (S'92-M'00-SM'13) received the B.S. (1993), M.S. (1995), and Ph.D. (2000) in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology, Cambridge, MA. He was a postdoctoral fellow at the University of California, Berkeley from 2000-2002. In 2002, he joined Yale University, New Haven, CT, where he is currently an associate professor in the Department of Statistics and Data Science. His research interests are in communication theory, information theory, stochastic control, distributed optimization, statistical machine learning and inference.