Network Utility Maximization Revisited: Three Issues and Their Resolution

P T Akhil[a,*], Rajesh Sundaresan[b]

[a]*Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore 560012, India.*
[b]*Department of Electrical Communication Engineering and Robert Bosch Centre for Cyber Physical Systems, Indian Institute of Science, Bangalore 560012, India.*

Abstract

Distributed and iterative network utility maximization algorithms, such as the primal-dual algorithms or the network-user decomposition algorithms, often involve trajectories where the iterates may be infeasible, convergence to the optimal points of relaxed problems different from the original, or convergence to local maxima. In this paper, we highlight the three issues with iterative algorithms. We then propose a distributed and iterative algorithm that does not suffer from the three issues. In particular, we assert the feasibility of the algorithm's iterates at all times, convergence to global maximum of the given problem (rather than to global maximum of a relaxed problem), and avoidance of any associated spurious rest points of the dynamics. A benchmark algorithm due to Kelly, Maulloo and Tan (1998) [Rate control for communication networks: shadow prices, proportional fairness and stability, Journal of the Operational Research society, 49(3), 237-252] involves fast user updates coupled with slow network updates in the form of additive-increase multiplicative-decrease of suggested user flows. The proposed algorithm may be viewed as one with fast user updates and fast network updates that keeps the iterates feasible at all times. Simulations suggest that the convergence rate of the ordinary differential equation (ODE) tracked by our proposed algorithm's iterates is comparable to that of the ODE for the aforementioned benchmark algorithm.

Keywords: Convex programming, Network utility maximization, Kelly

---

[*]Corresponding author
Email address: akhilpt@gmail.com (P T Akhil)

decomposition, Distributed interior point method.

1. Introduction and the main result

1.1. Background

We revisit the classic setting of decentralised congestion control as addressed by Kelly et al. [1]. Consider a network with m directed link resources. Let $c(l)$ be the capacity of the link l. There are n users and each has a single fixed path. Each user sends data along its associated path with the first vertex of the path being the source of the user's data and the last vertex being its terminus. Let A be the $m \times n$ matrix with $A_{le} = 1$ if the path e uses link l and $A_{le} = 0$ otherwise. Let $[n]$ denote the set $\{1, 2, \ldots, n\}$ of users and let $[m]$ denote the set $\{1, 2, \ldots, m\}$ of links. Let $w_e : \mathbb{R}_+ \to \mathbb{R}, \; e \in [n]$ be the utility functions of the users. User e derives a utility $w_e(x(e))$ when sending a flow of rate $x(e)$. The functions $w_e, e \in [n]$ are assumed to be strictly concave and increasing. Let $x = (x(e), e \in [n])$, $w = (w_e, e \in [n])$, and $c = (c(l), l \in [m])$. Let $\mathcal{A} = \{x | x \geq 0 \text{ and } Ax \leq c\}$. Throughout, we make the standing assumption that $\mathcal{A}$ has an interior feasible point, i.e., there exists a point for which all inequalities are strict. The system optimal operating point solves the problem:

$$\text{System}(w, A, c) : \quad \max_{x \in \mathcal{A}} \; W(x) := \sum_{e=1}^{n} w_e(x(e)). \tag{1}$$

The important decentralization concerns are that the network operator does not know the utility functions of the users, and the users know neither the rate choices of the other users nor the flow constraints on the network.

Kelly [2] proposed the decomposition of the above problem into two subproblems, one to be solved by each user, and the other to be solved by the network. Let λ_e be the cost per unit rate to user e set by the network, and let p_e be the price user e is willing to pay. The maximization problem solved by user e is

$$\text{User}(w_e; \lambda_e) : \quad \max_{p_e : p_e \geq 0} \; w_e\left(\frac{p_e}{\lambda_e}\right) - p_e. \tag{2}$$

If $p = (p_e, e \in [n])$ is known to the network, its optimization problem is

$$\text{Network}(A, c; p) : \max_{x \in \mathcal{A}} \sum_{e=1}^{n} p_e \log(x(e)). \tag{3}$$

The solution to $\text{Network}(A, c; p)$ is well-known to satisfy the so-called proportional fairness criterion: if $\mu_l, l \in [m]$ are the optimal dual price variables associated with the dual to $\text{Network}(A, c; p)$, then

$$x(e) = \frac{p_e}{\sum_{l:l \in e} \mu_l}, e \in [n], \tag{4}$$

is the optimal solution to the network problem. Kelly [2] showed that there exist costs per unit rate $(\lambda_e^\star, e \in [n])$, prices $(p_e^\star, e \in [n])$, and flows $(x^\star(e), e \in [n])$, satisfying $p_e^\star = \lambda_e^\star \cdot x^\star(e)$ for $e \in [n]$ such that $p_e^\star$ solves $\text{User}(w_e; \lambda_e^\star)$ for $e \in [n]$ and $(x^\star(e), e \in [n])$ solves $\text{Network}(A, c; p^\star)$; furthermore, $(x^\star(e), e \in [n])$ is the unique solution to $\text{System}(w, A, c)$. The costs per unit rate satisfy $\lambda_e^\star = \sum_{l:l \in e} \mu_l^\star$ for some dual price variables.

In order to ensure operation at $x^\star$, taking the information asymmetry constraints into account, Kelly et al. [1] proposed the following *fast user adaptation* dynamics:

$$p_e(t) = x(e, t) \cdot w_e'(x(e, t)), \ e \in [n], \tag{5}$$

$$\frac{d}{dt} x(e, t) = \kappa \cdot \left(p_e(t) - x(e, t) \cdot \sum_{l:l \in e} \mu_l(t) \right), \ e \in [n], \tag{6}$$

$$\mu_l(t) = \psi_l \left(\sum_{e:e \ni l} x(e, t) \right), \quad l \in [m], \tag{7}$$

where $\psi_l(y)$ is a penalty[1] or cost per unit flow when the total flow in the link is y. It signifies the level of congestion in that link. Thus $\mu_l(t)$ in (7) is the cost per unit flow through link l, and may be interpreted as a dual variable of the network problem. The optimal dual variables for $\text{Network}(A, c; p)$ are such that the net cost of user e flow matches the price p_e paid by that user; see (4). The network, adapts the flow $x(e, t)$ using an additive-increase

---

[1]Kelly et al. [1] suggest two functions as examples, one of which is $\psi_l(y) = (y - c(l) + \varepsilon)^+ / \varepsilon^2$ for some $\varepsilon > 0$.

multiplicative-decrease scheme as in (6), perhaps the first mathematical justification for the scheme already then in use in TCP/IP congestion control schemes. The network attempts to equalize, albeit slowly, the instantaneous net cost of user e flow, $x(e,t) \cdot \sum_{l:l\in e} \mu_l(t)$, to the instantaneous price paid by that user, $p_e(t)$. On the other hand, if we differentiate (2) with respect to p_e and use the relation $p_e(t) = \lambda_e(t) \cdot x(e,t)$, we get that $p_e(t)$ in (5) maximizes User$(w_e; \lambda_e(t))$. So the users adapt *instantaneously* (in comparison to the network's slower speed of adaptation) to the congestion signal. Kelly et al. [1] provided a Lyapunov function for the dynamical system defined by (5)-(7). The stable equilibrium point of the dynamical system maximizes a relaxation of the system problem, as determined by the choice of $\psi_l(\cdot)$ in (7).

The papers Kelly [2] and Kelly et al. [1] are landmark papers for three reasons.

1. They provided perhaps the first mathematical justification for the additive-increase and multiplicative-decrease scheme then already in use for TCP/IP congestion.

2. They firmly rooted the idea of proportional fairness in the minds of network engineers.

3. They also provided the general framework to study other notions of fairness via utility functions and network utility maximization.

1.2. Three issues and the motivation for our work

Despite the popularity of this approach, there are three issues we would like to highlight.

- $x(t)$ may not remain feasible at all times t.

- $x(t)$ converges to the optimal value of a relaxation of the system problem.

- There are multiple fixed points for the dynamics.

The first issue was highlighted in Johansson et al. [3]. The dynamics (5)-(7) cannot then be used in systems where feasibility has to be ensured at

all times. Take for example identification of optimal flow parameters in a software defined communication network with a centralized controller. Flows may go through links of *fixed* capacities, and these *must* be respected at all times, even during the learning and exploration phases that may ensue before arrival at the final optimal flow values. This may also be required in mobile offloading settings [4], [5] under additional assumptions of strict service level guarantees, in smart grid energy routing settings [6], or in road traffic settings where one simply cannot have more traffic than the road's capacity at any time, be it during the exploratory phase or otherwise.

The second issue is often circumvented via iterative algorithms where the Lagrange multipliers or penalty functions are also adapted over time, in some examples at a slower time scale; see for example, Arrow and Hurwicz [7], Low and Lapsley [8], Chiang et al. [9], Palomar and Chiang [10], and the more recent works of Gao et al. [11] and [4]. Such approaches either assume knowledge of the utility functions at the network end or may encounter infeasible iterates, or both.

The third issue is about multiple spurious rest points, other than the global optimum, for the iterative dynamics. Indeed, if $x(e,t) = 0$ for user e at some time t_0, and if[2] $\lim_{r \to 0} r w'_e(r) = 0$, then from (5), the user's willingness to pay $p_e(t_0) = 0$, and from (6) one gets $x(e,t) \equiv 0, t \geq t_0$, i.e., the iterates never exit the facet defined by $x_e = 0$. See Figure 1. When there is no stochasticity, there is no exit from this facet, and the iterates converge to a rest point for the dynamics different from the global optimum. Further, there is no a priori guarantee that these rest points are not attracting, and so there may be no exit even under stochasticity if the iterates start sufficiently close to these rest points.

The purpose of this paper is to provide an algorithm that circumvents these three issues.

---

[2]This is the case when the marginal utility for user e at supplied rate $r = 0$, $w'_e(0+)$, is finite. The quantity $\lim_{r \to 0} r w'_e(r)$ can be nonzero only when $w'_e(0+) = \infty$, for e.g., $w_e(r) = \log r$.
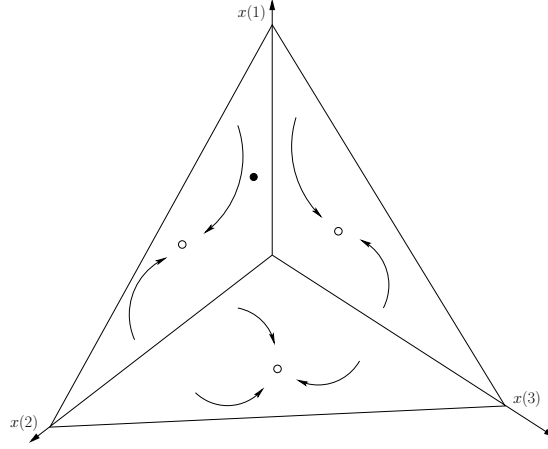
Figure 1: There are many rest points for the dynamics, with the spurious ones indicated by open circles on the facets. The solid circle is the global optimum and is on the interior. If the iteration starts on one of the facets, it remains on the facet when there is no stochasticity, and we do not have convergence to the global optimum point. Indeed, each open circle rest point is attracting for dynamics within that facet.

1.3. Most relevant related works

The literature on network utility maximization is so vast that we will not be able to do justice to twenty years of literature on the topic. However, we will focus on works where the iterates remain feasible at all times. There are three works, Hochbaum [12], Mo and Walrand [13] and La and Anantharam [14], that are very relevant to our contribution which we bring to the reader's attention. A greedy algorithm proposed by Hochbaum [12] can be adapted to solve the system problem with iterates remaining feasible at all times and without full knowledge of the utility functions at the network side. Though the algorithm circumvents the issues highlighted above, it works only when the set of feasible flows forms the "independent set of a polymatroid". This is the case when the network has, for example, a single source and multiple sinks or when the network has multiple sources but a single sink.

Mo and Walrand [13] proposed a window-based rate control mechanism that converges to the solution to Network$(A, c; p)$ for a fixed p. The window update rule of [13] uses only delay information provided to the user (propagation and round-trip delays).

La and Anantharam [14] proposed two algorithms that solve the system

problem using the decomposition of Kelly et al. [1]. The first algorithm incorporates the solution to the user problem into the window update rule of [13]. The second algorithm of La and Anantharam [14] explicitly finds the solution to the user problem and the network problem in each iteration. Although their simulations showed the convergence of the algorithm for general networks, a rigorous proof was given only for the case of a network with a single link. Their algorithm additionally imposes more stringent conditions on the utility functions than those assumed in this paper.

1.4. Our main result

In this paper, we propose a discrete-time algorithm (see Algorithm 1 below) that (1) remains feasible at all times, (2) converges to the desired global maximum of $\text{System}(w, A, c)$ (rather than to global maximum of a relaxed problem), and (3) therefore avoids spurious traps. The corresponding continuous time dynamics also shares the same properties. In comparison to [14], our algorithm applies to more general networks and a larger class of objective functions.

We now set up the notation to describe the algorithm. For a set of flows, abusing notation, write $p_e(x(e)) := x(e) \cdot w'_e(x(e))$ as per (5), and set $p(x) = (p_e(x(e)), e \in [n])$. Write

$$T(x) := \arg\max_{y \in \mathcal{A}} \sum_{e=1}^{n} p_e(x(e)) \log(y(e)) \tag{8}$$

for the solution to $\text{Network}(A, c; p(x))$. If $p_e(x(e)) = 0$ for some e, then the objective function in (8) is not strictly concave over $\mathcal{A}$. The optimization problem (8) may then have multiple solutions, and so $T(x)$ is to be viewed as a set-valued mapping whose values are convex and compact subsets of $\mathcal{A}$. Define $a_k := \frac{1}{k+1}, k = 0, 1, 2, \ldots$.

Algorithm 1. *1. Initialize $x^{(0)} \in \mathcal{A}$ such that $x^{(0)}(e) > 0$, $e \in [n]$. Initialize $k = 0$.*

2. User update:

$$p_e^{(k)} = x^{(k)}(e) \cdot w'_e(x^{(k)}(e)), \ e \in [n]. \tag{9}$$

3. *Network update:*

Find a point $v \in T(x^{(k)})$ and set:
$$x^{(k+1)} = x^{(k)} + a_{k+1}(v - x^{(k)}). \tag{10}$$

4. *Set $k \leftarrow k + 1$ and go to step 2.*

Our main result is the following theorem.

Theorem 1. *Assume that $\mathcal{A}$ has an interior feasible point. The iterates $x^{(k)}$ of Algorithm 1 converge to $x^{\star}$, the optimal solution to the system problem, i.e., $x^{(k)} \rightarrow x^{\star}$ as $k \rightarrow \infty$.*

1.5. The three issues are now resolved

Recall that our objective is to address the three main issues in the dynamics of (5)-(7), viz., the non-feasibility of the iterates, their convergence to a different solution – the solution to some relaxed problem, or their convergence to local maxima traps on one of the facets. We now argue that these issues disappear for Algorithm 1.

Observe that, in Algorithm 1, the users exhibit the same fast adaptation as in the dynamics (5). But in the network update, iterate $x^{(k+1)}$ is a convex combination of $x^{(k)}$ and $v \in T(x^{(k)})$ which, by induction, remains in the feasible set for all k. This resolves the feasibility issue that plagues the dynamics (5)-(7). In the proof we will argue that the iterates track the differential inclusion

$$\frac{d}{dt}x(t) \in T(x(t)) - x(t); \tag{11}$$

we will in fact see that the solution to this differential inclusion also remains feasible at all times.

Theorem 1 asserts that the iterates converge to the global optimum of the system problem. This resolves the issue that the dynamics (5)-(7) converge to the solution to a relaxed problem different from the original system problem.

The assertion that there is convergence to the global optimum resolves the third issue as well of avoidance of spurious rest points on the facets. This is particularly interesting since there is no stochasticity in our algorithm. See Figure 2. We must however start in the interior, but any arbitrary interior feasible point will work.
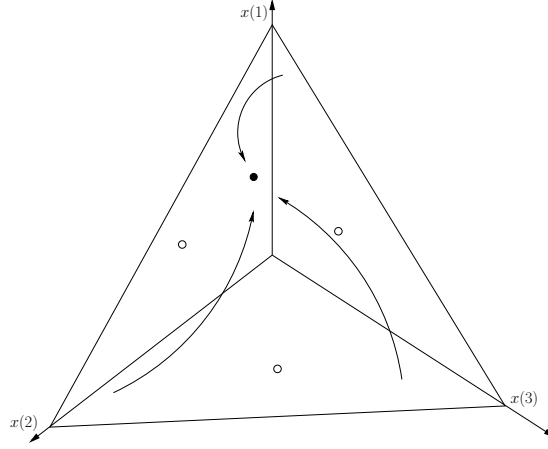
Figure 2: Dynamics of Algorithm 1. Even if we start close to the facet, so long as the initial point is on the interior, there is convergence to the global optimum.

1.6. Nontriviality of our contribution

1) *Slowdown is essential*: It is worthwhile to ask whether the convergence of Algorithm 1 could be sped up using a constant step size $a_k \equiv \gamma$, where $0 < \gamma < 1$, in the network update. In fact, Hou and Kumar [15] proposed a variant of Algorithm 1 with constant step size in the network update rule. This was done in the context of delay constrained throughput maximization in wireless networks. Step sizes determine the sizes of exploration. Larger step sizes involve faster exploration but also increased variance. In Appendix A, we provide a counterexample to show that the algorithm of Hou and Kumar with a constant step size *may not converge*. The choice of step sizes is therefore a delicate matter. A sufficient condition on step sizes $a_k, k \geq 1$, for the algorithm to converge is the usual conditions typical in stochastic approximation literature.

$$\lim_{k \to \infty} a_k = 0, \text{ and } \sum_{k=1}^{\infty} a_k \to \infty. \qquad (12)$$

The second condition ensures that there is enough exploration while the first condition gradually brings the exploration or learning rate to 0.

2) *Technical issues in the proof of convergence*: The main technical issues to surmount in showing the convergence of Algorithm 1 are (a) the dynamics in (11) also have multiple fixed points and it is *nontrivial* to show

convergence to the global optimum; (b) $T(x)$ is not necessarily a continuous function of x; see Section 2.2 and Appendix B. One must then study a differential inclusion, and the solution space may not be unique in general. An additional issue is the lack of stochasticity. We must then show that local maxima traps can yet be escaped. The main technical contribution is that these issues can be surmounted at least in this NUM problem with sufficient convexity structure.

3) *Wide applicability*: Iterative procedures for network utility maximization have wide applicability. Road traffic control, software defined network controllers with strict service level guarantees, offloading of data-traffic to WiFi providers and femto cell providers ([4] and [5]) with strict call handling requirements, and smart grid energy routing [6] are all settings where feasibility should be met at all times and where convergence to global optimality is desirable. Our algorithm is applicable in all these settings.

4) *Decentralized implementation*: At first glance, Algorithm 1 *appears* to need a central entity that computes the solution to Network$(A, c; p)$, i.e. $T(x)$, and Section 3 describes algorithms to compute $T(x)$ efficiently for some class of networks. However, the central entity is *not essential* because we can use Mo and Walrand's algorithm [13] to find $T(x)$ for a fixed p; that algorithm uses only the information available at the user end. The p can then be adapted (user updates) at a slower time scale. This enables the potential use of Algorithm 1 in a *distributed setting* in large scale networks.

1.7. Organization

The rest of the paper is organized as follows. In Section 2, we prove Theorem 1. In Section 3, we address the complexity of identifying the proportionally fair solution point for the network problem. We provide an example of a network where flows aggregate into a 'main branch', reminiscent of traffic from the suburbs flowing into an arterial highway leading to the downtown of a large city, for which the complexity to solve the network problem is $\mathcal{O}(n)$. We also argue that this complexity is manageable ($\mathcal{O}(n \log n)$ plus computations for feasibility checks) in situations where the feasible set is a polymatroid, for example, when all flows either originate or terminate at a single vertex. We also see in simulations that the dynamics

in (11) converge to the equilibrium at a faster rate than the dynamics of (5)-(7) for identical speed parameters κ. In Section 4, we end the paper with some concluding remarks.

2. Proof of Convergence

The update equation in step 3 of Algorithm 1 is a standard stochastic approximation scheme but without the stochasticity. A common method to analyze the asymptotic behavior of such schemes is the dynamical systems approach based on the theory of ordinary differential equations (ODE). But $T(x)$ being a set valued map necessitates the use of differential inclusions.

The outline of the proof is as follows. We will first characterize the fixed points of the mapping T. We will then argue that the system optimal point is one of the finitely many fixed points of the mapping T. We will next show that the solution to the differential inclusion in (11) models the asymptotic behavior of the iterates $x^{(k)}$. Following this, We will show that every solution to the differential inclusion converges to one of the fixed points of T via Lyapunov theory. Finally, though there may be many fixed points, we will prove that the fixed point to which the solution to the differential inclusion converges as $t \to \infty$ is the system optimal point.

2.1. Characterization of the fixed points of $T(x)$

Definition 1. *A point x is a fixed point of the set valued map T if $x \in T(x)$.*

Let $S \subset [n]$. Let $\mathcal{A}|_S$ be the subset of $\mathcal{A}$ whose points have support contained within S. Define a subproblem of the system problem as

$$\text{Subsystem}(w, A, c, S) : \max_{y \in \mathcal{A}|_S} \sum_{e \in S} w_e(y(e)). \tag{13}$$

Lemma 1. *Let $\overline{x}$ be a fixed point of the mapping T. Let $S = \{e : \overline{x}(e) > 0\}$. Then $\overline{x}$ is the unique optimal solution to the Subsystem(w, A, c, S).*

Proof We have $\overline{x}(e) = 0$ for all $e \in S^c$. If $\lim_{x(e)\downarrow 0} x(e) \cdot w_e'(x(e)) > 0$ for some $e \in S^c$, then any element $y \in T(\overline{x})$ has $y(e) > 0$ which contradicts

the fact that $\overline{x}$ is a fixed point. Hence $p_e(\overline{x}(e)) = \overline{x}(e) \cdot w_e'(\overline{x}(e)) = 0$ for all $e \in S^c$, and we may write

$$\overline{x} \in T(\overline{x}) = \arg \max_{y \in \mathcal{A}} \; \sum_{e \in S} p_e(\overline{x}(e)) \log(y(e)). \tag{14}$$

Since $\overline{x} \in \mathcal{A}|_S \subset \mathcal{A}$, we also have that $\overline{x}$ maximizes (14) over $\mathcal{A}|_S$, i.e.,

$$\overline{x} = \arg \max_{y \in \mathcal{A}|_S} \; \sum_{e \in S} p_e(\overline{x}(e)) \log(y(e)). \tag{15}$$

Let $\mu_l, l \in [m]$ and $\eta_e, e \in [n]$ be the optimal dual variables for the network subproblem (15). Its Karush-Kuhn-Tucker (KKT) conditions are

$$\frac{p_e(\overline{x}(e))}{\overline{x}(e)} = \sum_{l:l \in e} \mu_l - \eta_e, \; e \in S, \tag{16}$$

$$\mu_l \cdot \left(\sum_{e:e \ni l} \overline{x}(e) - c(l) \right) = 0, \; l \in [m], \tag{17}$$

$$\eta_e \cdot \overline{x}(e) = 0, \; e \in [n], \tag{18}$$

$$\eta_e \geq 0, \; e \in S, \; \mu_l \geq 0, \; l \in [m] \text{ and } \overline{x} \in \mathcal{A}|_S. \tag{19}$$

Since $\frac{p_e(\overline{x}(e))}{\overline{x}(e)} = w_e'(\overline{x}(e))$, it is easy to see that equations (16-19) are the KKT conditions of Subsystem(w, A, c, S) as well. Since $\overline{x}/2$ is an interior feasible point of $\mathcal{A}|_S$, KKT conditions are sufficient for optimality in (13) and $\overline{x}$ is the optimal solution to Subsystem(w, A, c, S). Uniqueness follows from the strict concavity of $w_e, e \in S$. ∎

Observe that there are only a finite number of sub-problems of the form Subsystem(w, A, c, S), $S \subset [n]$. As a consequence of Lemma 1, every fixed point of T is the unique optimal solution to Subsystem(w, A, c, S) for some set $S \subset [n]$. Hence there are only finitely many fixed points of T, each corresponding to a sub-problem Subsystem(w, A, c, S) for some $S \subset [n]$.

Is every solution to Subsystem(w, A, c, S) a fixed point of the mapping T? The possibility that $\lim_{x(e) \downarrow 0} x(e) \cdot w_e'(x(e)) > 0$ for an $e \in S^c$ and the first step of the proof of Lemma 1 says this is not always true. However, we can assert the following.

Lemma 2. *The global maximum of the system problem, $x^\star$, is a fixed point of the mapping T.*

Proof $x^\star$ solves the system problem. Let $S = \{e : x^\star(e) > 0\}$. S can be a proper subset of $[n]$. Let $\mu_l, l \in [m]$ and $\eta_e, e \in [n]$ be the optimal dual variables of the system problem. We then have

$$w'_e(x^\star(e)) = \sum_{l:l\in e} \mu_l - \eta_e, \ e \in S, \tag{20}$$

$$w'_e(0) = \sum_{l:l\in e} \mu_l - \eta_e, \ e \in S^c, \tag{21}$$

$$\mu_l \cdot \left(\sum_{e:e\ni l} x^\star(e) - c(l) \right) = 0, \ l \in [m], \tag{22}$$

$$\eta_e \cdot x^\star(e) = 0, \ e \in [n], \tag{23}$$

$$\eta_e \geq 0, \ e \in [n], \ \mu_l \geq 0, \ l \in [m] \text{ and } x^\star \in \mathcal{A}. \tag{24}$$

Observe that $w'_e(0)$ is finite for an $e \in S^c$; otherwise a small increase in $x^\star(e)$ and a corresponding decrease in $x^\star(i)$ for a suitable $i \in S$ (which has finite $w'_i(x^\star(i))$) will result in a feasible flow that has a larger objective function value. Hence $p_e(x^\star(e)) = 0$ for $e \in S^c$. Since $w'_e(x^\star(e)) = \frac{p_e(x^\star(e))}{x^\star(e)}$ for all $e \in S$, it follows from (20)-(24) that $x^\star, (\mu_l, l \in [m]), (\tilde{\eta}_e = \eta_e, e \in S)$ and $(\tilde{\eta}_e = \eta_e + w'_e(0), e \in S^c)$ satisfy the KKT conditions of the problem (8). Hence $x^\star \in T(x^\star)$. ∎

The above result provides a motivation to search for the global maximum by setting up a dynamics that will converge to a fixed point of T.

2.2. Need for the theory of differential inclusions

We now describe the issues that make it necessary to use differential inclusions to study the asymptotic behavior of $x^{(k)}$. $T(x)$ is the set of points that solve (8). If $p_e(x(e)) = 0$ for some e at a point $x \in \mathcal{A}$, then the objective function in (8) is not strictly concave. Hence there can be multiple points that solve (8). A continuous selection[3] from $T(x)$ allows the use of differential equations to analyze the stochastic approximation scheme in (10). A natural question that arises is whether there is such a continuous

---

[3]A continuous selection f from the set-valued map T is a continuous function $f : \mathcal{A} \to \mathcal{A}$ with $f(x) \in T(x)$ for each $x \in \mathcal{A}$.

selection from $T(x)$. We give an example in Appendix B showing that such a selection is not always possible.

2.3. Differential Inclusions: Preliminaries

In this section, we define a differential inclusion and state relevant results from [16] that are used to show the convergence of Algorithm 1. Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a set valued map. Consider the following differential inclusion:

$$\frac{dx}{dt} \in F(x). \tag{25}$$

A solution to the differential inclusion in (25) with initial condition $x_0 \in \mathbb{R}^n$ is an absolutely continuous function $x : \mathbb{R} \to \mathbb{R}^n$ that satisfies (25) for almost every $t \in \mathbb{R}$. The following conditions are sufficient for the existence of a solution to the differential inclusion (25):

1. $F(x)$ is nonempty, convex and compact for each $x \in \mathbb{R}^n$.

2. F has a closed graph.

3. For some $K > 0$, for all $x \in \mathbb{R}^n$, F satisfies the following condition

$$\sup_{z \in F(x)} ||z|| \leq K(1 + ||x||). \tag{26}$$

The stochastic approximation scheme with iterates in $\mathbb{R}^n$ is given as

$$y^{k+1} \in y^{(k)} + a_{k+1}(F(y^{(k)}) + U^{(k+1)}), \tag{27}$$

where a_k satisfy the usual conditions:

$$\lim_{k \to \infty} a_k = 0, \ \sum_{k=1}^{\infty} a_k \to \infty, \tag{28}$$

and $U^{(k)} \in \mathbb{R}^n$ are deterministic or random perturbations.

Let $t(0) = 0, t(k) = \sum_{i=1}^{k} a_i$. Let $r_y : \mathbb{R}_+ \to \mathbb{R}^n$ be a continuous piecewise linear function formed by the interpolation of $y^{(k)}$ as in

$$r_y(t) = y^{(k)} + \frac{y^{(k+1)} - y^{(k)}}{t(k+1) - t(k)} \cdot (t - t(k)), \quad \forall \, t \in [t(k), t(k+1)). \tag{29}$$

Definition 2. *(A perturbed solution to (25)). Let $U : \mathbb{R}_+ \to \mathbb{R}^n$ be locally integrable function such that*

$$\lim_{t \to \infty} \sup_{0 \leq v \leq T} \left\| \int_t^v U(s)ds \right\| = 0.$$

Let $\delta : [0, \infty) \to \mathbb{R}$ be a function such that $\delta(t) \to 0$ as $t \to \infty$. Define

$$F^\epsilon(x) = \{y \in \mathbb{R}^m : \exists z : ||z - x|| < \epsilon, d(y, F(z)) < \epsilon\}, \tag{30}$$

where $d(y, F(z)) = \inf\{||y - q|| : q \in F(z)\}$. An absolutely continuous function $y : [0, \infty) \to \mathbb{R}^n$ is a perturbed solution to (25) if there exists $U : \mathbb{R}_+ \to \mathbb{R}^n$ and $\delta : \mathbb{R}_+ \to \mathbb{R}^n$ as above such that

$$\frac{dy}{dt} - U(t) \in F^{\delta(t)}(y(t)). \tag{31}$$

for almost every $t \in \mathbb{R}_+$.

The following lemma, taken from [16], gives conditions on $y^{(k)}$ and $U^{(k)}$ for r_y to be a perturbed solution to (25).

Lemma 3. *[16, Prop. 1.3] Suppose $y^{(k)}$ is bounded, i.e., $\sup_k ||y^{(k)}|| < M < \infty$, and for all $T > 0$,*

$$\lim_{s \to \infty} \sup\left\{ \left\| \sum_{k=s}^{i-1} a_{k+1} U^{(k+1)} \right\| : i = s + 1, s + 2, \ldots, m(t(s) + T) \right\} = 0, \tag{32}$$

where $m(t) = \sup\{m : t(m) \leq t\}$. Then $r_y(t)$ is a perturbed solution of the differential inclusion (25).

Definition 3. *A compact set L is an internally chain transitive set if for any $x, y \in L$ and every $\epsilon > 0, T > 0$, there exists $l \in \mathbb{N}$, solutions $x_1, x_2, \ldots, x_l$ to (25) and $t_i > T, \forall i$, that satisfy the following.*

1. *$x_i(t) \in L$ for all $0 \leq t \leq t_i$ and for all $i \in [l]$,*

2. *$||x_i(t_i) - x_{i+1}(0)|| \leq \epsilon$ for all $i \in [l - 1]$,*

3. *$||x_1(0) - x|| \leq \epsilon$ and $||x_l(t_l) - y|| \leq \epsilon$.*

We shall call the sequence $(x_1, x_2, \ldots, x_l)$ as an (ϵ, T) chain in L from x to y.

The following lemma, again taken from [16], characterizes the limit set of a perturbed solution.

Lemma 4. *[16, Thm. 3.6] Let r be a perturbed solution to (25). Then the limit set of $r(\cdot)$ $L(r) := \bigcap_{t \geq 0} \{r(s) : s \geq t\}$ is internally chain transitive.*

2.4. Convergence analysis

We proceed to prove the convergence of $x^{(k)}$, the iterates put out by Algorithm 1, to the optimal solution to the system problem. Observe that T maps points in $\mathcal{A}$ to itself. We show that $x^{(k)}$ asymptotically tracks the solution to the differential inclusion

$$\frac{dx}{dt} \in F(x), \tag{33}$$

where

$$F(x) := T(P_{\mathcal{A}}(x)) - x, \tag{34}$$

for $x \in \mathbb{R}^n$. $P_{\mathcal{A}}(x)$ is the projection of x onto the set $\mathcal{A}$. We then find a Lyapunov function for the dynamics in (33) to show its convergence to the system optimal point $x^\star$. The following lemma establishes that the set-valued map F defined in (34) has some good properties; it turns out that these are sufficient for the differential inclusion (25) to have a solution.

Lemma 5. *For each $x \in \mathbb{R}^n$, $F(x)$ is nonempty, convex and compact. Furthermore, F has the closed graph property and satisfies (26).*

Proof The objective function of the network problem is continuous and the constraint set $\mathcal{A}$ is compact. The maximum exists due to the Weierstrass theorem. Also, the set of maximizers is closed and convex. Thus $T(P_{\mathcal{A}}(x))$ is nonempty, convex and compact, and hence so is $F(x)$.

We next prove the closed graph property of F. A function has the closed graph property if it is upper hemicontinuous. The objective function in (8), $\sum_{e=1}^{n} p_e(x(e)) \log(y(e))$, is jointly continuous[4] in x and y. Also, the constraint set of the network problem does not vary with x. By Berge's maximum theorem [17, p. 116], $T(x)$ is upper hemicontinuous. Since $P_{\mathcal{A}}(x)$, the projection onto the convex set $\mathcal{A}$, is continuous, the composition $T(P_{\mathcal{A}}(x))$ is upper hemicontinuous. Consequently, $F(x)$ is upper hemicontinuous and

---

[4]Here we may take $\log(y(e))$ to be continuous at $y(e) = 0$ with $\log 0 := -\infty$ because then $\lim_{x(e)\downarrow 0} \log(y(e)) = \log 0$ and this sequential continuity is all that is needed to apply Berge's maximum theorem [17, p. 116]

hence has the closed graph property. Finally,

$$\sup_{z \in F(x)} ||z|| = \sup_{z \in T(P_{\mathcal{A}}(x))} ||z - x|| \leq \sup_{z \in \mathcal{A}} ||z - x|| \leq \sup_{z \in \mathcal{A}} ||z|| + ||x||$$

$$\leq K + ||x|| \leq K(1 + ||x||), \text{ where } K > \max\{\sup_{z \in \mathcal{A}} ||z||, 1\}.$$

Lemma 6. *Let $r_x(t)$ be obtained by the linear interpolation of $x^{(k)}$ as given in (29). Then $r_x(t)$ is a perturbed solution to the differential inclusion (25) with F defined as in (34).*

Proof We first show that $x^{(k)} \in \mathcal{A}$. Observe that $x^{(0)} \in \mathcal{A}$. Assume $x^{(k-1)} \in \mathcal{A}$. Since $T(x^{(k-1)}) \in \mathcal{A}$ and $x^{(k)}$ is a convex combination of $x^{(k-1)}$ and $T(x^{(k-1)})$, we have $x^{(k)} \in \mathcal{A}$. It follows that

$$F(x^k) = T(P_{\mathcal{A}}(x^{(k)})) - x^{(k)} = T(x^{(k)}) - x^{(k)}.$$

We now see that the update equation in (10) is the same as the stochastic approximation scheme in (27) with $U^{(k)} = 0$ for all k. Observe that $x^{(k)}$ is bounded because $x^{(k)} \in \mathcal{A}$ for all k and $\mathcal{A}$ is compact; since $U^{(k+1)} = 0$, the condition in (32) is trivially satisfied. Hence, by Lemma 3, $r_x(t)$ is a perturbed solution. ∎

We restrict our attention to solutions of (25) with initial condition $x(0) \in \mathcal{A}$. Since $T(x) \in \mathcal{A}$, $x(t)$ lies in $\mathcal{A}$ for all t. Define

$$\Phi_t(x_0) := \{x(t) : x \text{ solves } (25), \ x(0) = x_0\}.$$

Definition 4. *Let Λ be a subset of $\mathcal{A}$. Let $V : \mathcal{A} \to \mathbb{R}$ be a continuous function such that $V(y) < V(x), \ y \in \Phi_t(x), x \in \mathcal{A}\backslash\Lambda$ and $V(y) \leq V(x), \ y \in \Phi_t(x), x \in \Lambda$. Then V is called a Lyapunov function for Λ.*

Define $V : \mathcal{A} \to \mathbb{R}$ as

$$V(x) := \sum_{e=1}^{n} w_e(x^\star(e)) - \sum_{e=1}^{n} w_e(x(e)). \tag{35}$$

Lemma 7. *Let Λ be the set of fixed points of T. The function V in (35) is a Lyapunov function for Λ.*

Proof Let $x \in \mathcal{A}$ and $v \in T(x)$. We have, from the definition of $T(x)$ in (8), that

$$\sum_{e=1}^{n} p_e(x(e)) \log(v(e)) \geq \sum_{e=1}^{n} p_e(x(e)) \log(x(e)) \qquad (36)$$

because $v \in T(x)$ maximizes the network problem.

If $p_e(x(e)) > 0$ for all e, then the network problem has unique solution. Therefore, equality holds in (36) if and only if $v = x$, i.e, x is a fixed point of the mapping T. Thus we have

$$0 \leq \sum_{e=1}^{n} p_e(x(e)) \log \frac{v(e)}{x(e)} \overset{(a)}{\leq} \sum_{e=1}^{n} p_e(x(e)) \left(\frac{v(e)}{x(e)} - 1 \right)$$
$$= \sum_{e=1}^{n} w'_e(x(e))(v(e) - x(e)) = \nabla W(x) \cdot (v - x), \qquad (37)$$

where (a) uses the inequality $\log y \leq y - 1$.

More generally, let $p_e(x(e)) > 0$ for $e \in S \subset [n]$ and $p_e(x(e)) = 0$ for $e \in S^c$; in particular, $x(e) = 0$ for $e \in S^c$. Define $\tilde{v}$ to be $\tilde{v}(e) = v(e)1_S(e), e \in [n]$.

The value of the objective function in (8) evaluated at v and $\tilde{v}$ are equal. Hence $\tilde{v} \in T(x)$,

$$\sum_{e \in S} p_e(x(e)) \log(\tilde{v}(e)) \geq \sum_{e \in S} p_e(x(e)) \log(x(e)), \qquad (38)$$

and $\tilde{v}$ must be the unique solution to the problem defined in (15). Therefore, (38) holds with equality if and only if $\tilde{v} = x$. Following the steps leading to (37), we have

$$\sum_{e \in S} w'_e(x(e))(\tilde{v}(e) - x(e)) \geq 0 \qquad (39)$$

which is a strict inequality if $\tilde{v} \neq x$. Since $v(e) - x(e) \geq 0$ for $e \in S^c$, this along with (39) yields

$$\nabla W(x) \cdot (v - x) \geq 0; \qquad (40)$$

since $w'_e(x(e)) = w'_e(0) > 0$ for $e \in S^c$, equality holds in (40) if and only if $v = x$. Hence

$$\frac{dV(x(t))}{dt} = -\nabla W(x(t)) \cdot (v - x(t)) \leq 0, \ \forall \ v \in T(x(t)). \qquad (41)$$

The inequality in (41) holds with an equality if and only if $v = x$. Therefore V is a Lyapunov function for Λ. ■

Lemma 8. *Let Λ be the set of fixed points of T. Every internally chain transitive set for F in (34) is a singleton that is a subset of Λ.*

Proof By Lemma 1, there are at most finitely many fixed points of the mapping T. Hence the cardinality of the set Λ is finite and $V(\Lambda)$ has empty interior. Also, by Lemma 7, V is a Lyapunov function for Λ. Proposition 3.27 of [16] states that if V is a Lyapunov function for Λ and if $V(\Lambda)$ has an empty interior, then every internally chain transitive set is a subset of Λ.

Choose ϵ small enough so that open balls of radius ϵ centered at each of the finite points Λ are disjoint. Fix $T \geq 0$. Since any (ϵ, T) chain involves remaining in Λ for all time and jumps of size at most ϵ to another point in Λ, by the disjointedness of the ϵ-balls covering Λ, there can be no (ϵ, T) chain in Λ joining two of its distinct points. It follows that the internally chain transitive subsets of Λ are singletons. ■

Lemma 9. *The iterates $x^{(k)}$ converges to a fixed point of the mapping T.*

Proof In Lemma 6, we showed that $r_x(t)$ is a perturbed solution to (25). By Lemma 4, the limit set of $r_x(t)$ is internally chain transitive. By Lemma 8, $L(r_x)$ is a singleton and $L(r_x) \subset \Lambda$. Let $\hat{x} \in L(r_x)$. Since $\mathcal{A}$ is compact and $\hat{x}$ is the only limit point of the sequence $x^{(k)}$, every subsequence of $x^{(k)}$ has a further subsequence that converges to $\hat{x}$. Hence $x^{(k)}$ converges to $\hat{x}$. ■

In the rest of this section, we show that the iterates converge to $x^\star$, the optimal solution to the system problem.

Let the dual variables of the optimization problem Network$(A, c; p)$ be $\mu_l, l \in [m]$. Kelly et al. [1] simplified the dual to this problem to be:

Dual(p, A, c) :

$$\min_{\mu_l \geq 0, l \in [m]} \left(\sum_{e=1}^{n} p_e \cdot \log \frac{1}{\sum_{l:l \in e} \mu_l} + \sum_{l=1}^{m} \mu_l c(l) \right). \qquad (42)$$

We now argue that the search for the optimal $\mu_l, l \in [m]$ may be restricted to a compact set.

Lemma 10. *The optimization problem in (42) with $p_e = p_e(x(e))$ is equivalent to the following optimization problem. For any $x \in \mathcal{A}$,*

$$\max_{0 \leq \mu_l \leq 2P/c(l)} \sum_{e=1}^{n} p_e(x(e)) \cdot \log \left(\sum_{l:l \in e} \mu_l \right) - \sum_{l=1}^{m} \mu_l c(l), \qquad (43)$$

where $P := \max_{x \in \mathcal{A}} \sum_{e=1}^{n} x(e) \cdot w'_e(x(e)) < \infty$.

Proof Define $R(\mu)$ to be the objective function in (43). For any $\mu_l > 2P/c(l)$, by reducing μ_l, we increase the objective function's value. To see this, it suffices to show that $\frac{\partial R(\mu)}{\partial \mu_l} < 0$ for any $\mu_l > 2P/c(l)$. But this is easily checked as follows:

$$\frac{\partial R(\mu)}{\partial \mu_l} = \sum_{e:e \ni l} p_e(x(e)) \frac{1}{\sum_{l':l' \in e} \mu_{l'}} - c(l) \leq \sum_{e:e \ni l} p_e(x(e)) \frac{1}{\mu_l} - c(l)$$

$$\leq \frac{1}{\mu_l} \sum_{e=1}^{n} p_e(x(e)) - c(l) \leq \frac{1}{\mu_l} \left[\max_{x \in \mathcal{A}} \sum_{e=1}^{n} p_e(x(e)) \right] - c(l)$$

$$= \frac{P}{\mu_l} - c(l) < 0, \qquad (44)$$

where the last inequality follows if $\mu_l > 2P/c(l)$. $\blacksquare$

Lemma 11. *Let $x^{(k)}$ converge to $\hat{x}$, a fixed point of the mapping T. Then $\hat{x} = x^\star$, the optimal solution to the system problem.*

Proof Let $v^{(k+1)}$ solve problem (8) with $p_e = p_e(x^{(k)}(e))$, and so $v^{(k+1)} \in T(x^{(k)})$ satisfies the KKT conditions

$$\frac{p_e(x^{(k)}(e))}{v^{(k+1)}(e)} - \sum_{l:l \in e} \mu_l^{(k)} + \eta_e^{(k)} = 0, \ e \in [n], \qquad (45)$$

$$\mu_l^{(k)} \cdot \left(\sum_{e:e \ni l} v^{(k+1)}(e) - c(l) \right) = 0, \ l \in [m], \qquad (46)$$

$$\eta_e^{(k)} \cdot v^{(k+1)}(e) = 0, \ e \in [n], \qquad (47)$$

$$\eta_e^{(k)} \geq 0, \ e \in [n], \ \mu_l^{(k)} \geq 0, \ l \in [m]. \qquad (48)$$

Let us first claim that $x^{(k)} > 0$ for all $k \geq 0$. This holds for $k = 0$, the initial point, in Algorithm 1. If, for some k, $x^{(k)} > 0$, then $p_e(x^{(k)}(e)) > 0$ for all e and so, $v^{(k+1)} > 0$ and consequently, $x^{(k+1)}$ being a convex combination of $x^{(k)}$ and $v^{(k+1)}$ also satisfies $x^{(k+1)} > 0$. The claim follows by induction.

Since $x^{(k)}(e) > 0$, we have $v^{(k+1)}(e) > 0$, and so, by (47), $\eta_e^{(k)} = 0$, $e \in [n]$. Thus (45) simplifies to

$$\frac{p_e(x^{(k)}(e))}{v^{(k+1)}(e))} = \sum_{l:l \in e} \mu_l^{(k)}. \tag{49}$$

Since we also have $p_e(x^{(k)}(e)) > 0$, and $v^{(k+1)}(e) > 0$ in (49), we have $\sum_{l:l \in e} \mu_l^{(k)} > 0$. Hence

$$v^{(k+1)}(e) = \frac{p_e(x^{(k)}(e))}{\sum_{l:l \in e} \mu_l^{(k)}} = \frac{x^{(k)}(e) \cdot w_e'(x^{(k)}(e))}{\sum_{l:l \in e} \mu_l^{(k)}}. \tag{50}$$

Suppose $\hat{x}(e) = 0$. Observe that $x^{(k)}(e) > 0$ for all k. Hence $v^{(k+1)}(e) < x^{(k)}(e)$ infinitely often, which is the same as saying

$$\frac{x^{(k)}(e) w_e'(x^{(k)}(e))}{\sum_{l:l \in e} \mu_l^{(k)}} < x^{(k)}(e) \tag{51}$$

occurs infinitely often. This implies

$$w_e'(x^{(k)}(e)) < \sum_{l:l \in e} \mu_l^{(k)} \tag{52}$$

infinitely often. Consider the subsequence that satisfies (52). Henceforth, let $x^{(k)}$ denote that subsequence. We now make the following observations. In Lemma 10, we showed that $\mu_l^{(k)}$ takes values on a compact set, and so we can find a further subsequence such that $\mu_l^{(k_\tau)} \to \mu_l^\star$ for some $\mu_l^\star$, but for all $l \in [m]$. Since (52) holds for $k = k_\tau$, by letting $\tau \to \infty$, we have

$$w_e'(\hat{x}(e)) \leq \sum_{l:l \in e} \mu_l^\star, \ \forall\, e \text{ such that } \hat{x}(e) = 0 \tag{53}$$

$$w_e'(\hat{x}(e)) = \sum_{l:l \in e} \mu_l^\star, \ \forall\, e \text{ such that } \hat{x}(e) > 0; \tag{54}$$

the latter follows by letting $\tau \to \infty$ in (50) and from $x^{(k+1)} = x^{(k)} + a_k(v^{(k+1)} - x^{(k)})$. Choose

$$\eta^\star(e) = 0 \text{ if } \hat{x}(e) > 0, \tag{55}$$

$$\eta^\star(e) = \sum_{l:l \in e} \mu_l^\star - w_e'(\hat{x}(e)) \text{ if } \hat{x}(e) = 0. \tag{56}$$

Thus, from (53),

$$\eta^\star(e) \cdot \hat{x}(e) = 0 \ \forall \ e \in [n] \text{ and } \eta^\star(e) \geq 0 \ \forall \ e \in [n]. \tag{57}$$

Since (46) and (48) are true for indices k_τ, taking limit as $\tau \to \infty$, we get

$$\mu_l^\star \cdot \left(\sum_{e:e \ni l} \hat{x}(e) - c(l) \right) = 0, \ l \in [m], \tag{58}$$

$$\mu_l^\star \geq 0, \ \forall \ l. \tag{59}$$

Equations (53)-(59) are the KKT conditions for the system problem. Hence $\hat{x} = x^\star$, the optimal solution to the system problem. ∎

Proof of Theorem 1: Theorem 1 follows from Lemma 11. ∎

3. Algorithmic Complexity and Speed of Convergence

In this section, we remark on the complexity of Algorithm 1. Each iteration of the algorithm has 1) a user update which adapts the amount a user is willing to pay to the network, and 2) a network update which adapts the rates allocated to the users.

Since w_e is known at the user end, w_e' is easy to obtain either numerically or analytically. Hence the user update (9) can be implemented by each user in $\mathcal{O}(1)$ steps.

The network update consists of solving the network problem (8). Its complexity depends on the network structure. We indicate the complexity of the network update for the following simple networks: a polymatroidal network with a single source and multiple sinks or multiple sources and a single sink; a flow aggregating network with the structure in Figure 3.

Polymatroidal network: Consider a network with a single source and n sinks. The source sends flows at a rate $x(1), x(2), \ldots, x(n)$ to the sinks. Megiddo [18] showed that the set of feasible flows $(x(e), e \in [n])$ forms the independent set of a polymatroid. Therefore, the network problem is a separable concave maximization over the independent set of a polymatroid. The fastest known algorithm that solves this optimization problem is a scaling based greedy algorithm proposed by Hochbaum [12]. The algorithm obtains
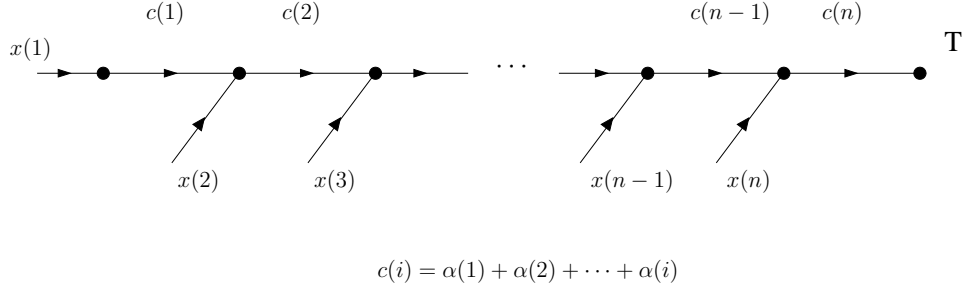
$$c(i) = \alpha(1) + \alpha(2) + \cdots + \alpha(i)$$

Figure 3: Flow aggregating network.

an "ϵ-optimal" solution to the network problem in $\mathcal{O}\left(n(\log n + F)\log\frac{B}{n\epsilon}\right)$ where F is the complexity to check whether a certain increase in one of the components of $(x(e), e \in [n])$ would make the flow infeasible. B is the total amount of resource to be allocated and is $\mathcal{O}(n)$.

Flow aggregating network: Let $(x(e), e \in [n])$ denote the flow through the network in Figure 3. The flow constraints of the network are

$$x(e) \geq 0, e \in [n],$$
$$x(1) \leq \alpha(1),$$
$$x(1) + x(2) \leq \alpha(1) + \alpha(2), \tag{60}$$
$$\vdots$$
$$x(1) + x(2) + \cdots + x(n) \leq \alpha(1) + \alpha(2) + \cdots + \alpha(n).$$

where $\alpha(e) \geq 0$ for all e. The constraints in (60) are referred to as linear ascending constraints. This problem arises as the core optimization problem in several wireless communication problems (Padakandla and Sundaresan [19], Viswanath and Anantharam [20], Lagunas et al. [21], Sanguinetti et al. [22]) and operations research problems (Clark and Scarf [23], Wang [24]). See [25] for a survey and a discussion of several algorithms. The network problem is the maximization of a so-called d-separable concave function over the linear ascending constraints. Veinott Jr. [26] mapped this problem to the geometrical problem of finding the "concave cover" of the set of points $(\sum_{e=1}^{i} \alpha(e), \sum_{e=1}^{i} p_e)$, $i \in [n]$, in $\mathbb{R}^2$. The "string algorithm" of Muckstadt and Sapra [27] finds the concave cover of a set of points in $\mathbb{R}^2$ in $\mathcal{O}(n)$ steps. See [25] for details.

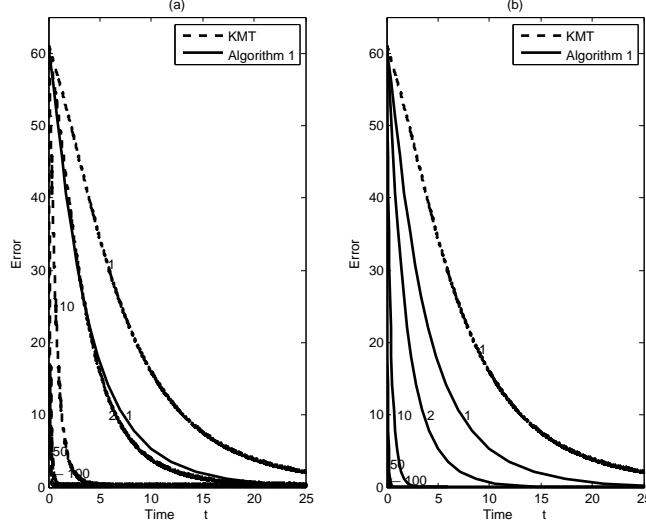*Simulations*: We now discuss some simulation studies investigating the

Figure 4: (a) Error dynamics for (5)-(7) denoted ("KMT") with $\kappa = 1, 2, 10, 50, 100$; error dynamics for (61) denoted "Algorithm 1" with $\kappa = 1$ is also plotted for comparison. (b) The roles of (5)-(7) and (61) are swapped.

speed of convergence of the ODE that our proposed algorithm will track. But we caution the reader that the ODE convergence rate does not give the full picture of convergence rate since the timescale is dictated by the step sizes. In the plots, the solid curves correspond to the error plots of the system

$$\frac{d}{dt}x(t) \in \kappa \cdot (T(x(t)) - x(t)). \tag{61}$$

The dashed curves correspond to the error plots of the system (5)-(7). The differential inclusion (61) has scaling factor κ when compared with (11) and corresponds to a scaled version of Algorithm 1. The scaling is to enable comparison of (61) with the system (5)-(7) which already has the scaling factor κ in (6).

All figures are for the flow aggregating network with $n = m = 10$ and link capacities $c(l) = 10 \times l, l \in [m]$. The utility functions are chosen as $(1/\beta(e)) \cdot x^{\beta(e)}$, for $e \in [n]$ with $\beta(e) = .09 \cdot e$. The initial point for both differential inclusions is always the lexicographically maximal point[5]. This is the most natural starting point when the network does not know the

---

[5]The lexicographically maximal point is one where the minimum allocation (across
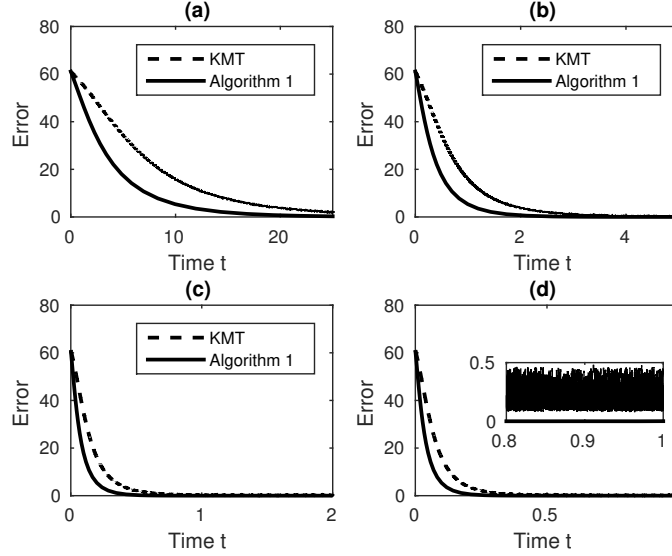
Figure 5: Comparison of error dynamics for the systems (5)-(7) ("KMT" curve) and (61) ("Algorithm 1" curve). Subplot (a) has $\kappa = 1$, (b) has $\kappa = 10$, (c) has $\kappa = 50$, and (d) has $\kappa = 100$. Note that the time axis is scaled differently in each of the subfigures. The inset picture in (d) shows an enlarged view of the error. The "KMT" curve settles at a small but positive error.

users' utility functions and considers all users to be equal. While we report the results only for this particular β and c, we have simulated several other settings, and the results are qualitatively the same. We do not repeat them here for brevity.

Figure 4(a) shows that as κ scales up, the speed of convergence of the system (5)-(7) increases. For comparison, we have included the solid curve for (61) with $\kappa = 1$.

Figure 4(b) shows that as κ scales up, the rate of convergence of (61) also increases similarly. Again, for comparison, we have included the dashed curve for (5)-(7) with $\kappa = 1$.

These two subfigures show that convergence can be sped up similarly in the two systems, (5)-(7) and (61), by simply increasing κ.

---

users) is maximized among all feasible points; further the second minimum is maximized among all points with equal minimum allocation, and so on.

Figures 5(a)-5(d) compare (5)-(7) directly with (61) for identical κ. The speeding up parameter κ equals 1, 10, 50 and 100 in Figures 5(a), 5(b), 5(c), and 5(d) respectively. These figures demonstrate that convergence speed of (61) is comparable to that of (5)-(7) as long κ is identical for the two systems. We saw the same qualitative behavior across several randomly chosen problem parameters.

The inset in Figure 5(d) shows an enlarged view of the error plots after the algorithms' settlement close to their respective limiting values. We see that the error plot of system (5)-(7) settles at a small but positive value. This is consistent with the observation that KMT algorithm solves a relaxation of the original system problem.

Figure 6 reproduces the plots in Figure 5(a) but with abscissa values restricted to time interval $[0, 15]$. The dash-dotted line plots the iterates put out by Algorithm 1 with the kth iterate plotted at time instant

$$t_k = \sum_{i=1}^{k} a_i. \tag{62}$$

As expected, we see that the iterates trace the error plot of the ODE (61) for $\kappa = 1$. This justifies the comparison between the system (5)-(7) and the ODE (61).

We also compare our algorithm with a benchmark interior point algorithm, the projected gradient algorithm [28, Sec. 2.3]. The projected gradient descent algorithm is not distributed because the step-size selection according to Armijo-Goldstein rule would require the knowledge of utility functions. Hence the comparison is made in the following two ways.

We first compare the case when the stepsize is $a_k = 1/(k + 1)$ as for stochastic approximation. With these fixed stepsizes, the projected gradient descent can also be implemented in a distributed fashion, similar to ours. The network asks all users to send flows according to $x^{(k)}$ and invites these users to send gradients of their private utility functions at these points. The users follow this. With the gradient information, the network identifies a new location by employing gradient descent, projects it on the feasible set, and then asks users to send flows according to this projected $x^{(k+1)}$. The procedure then repeats.
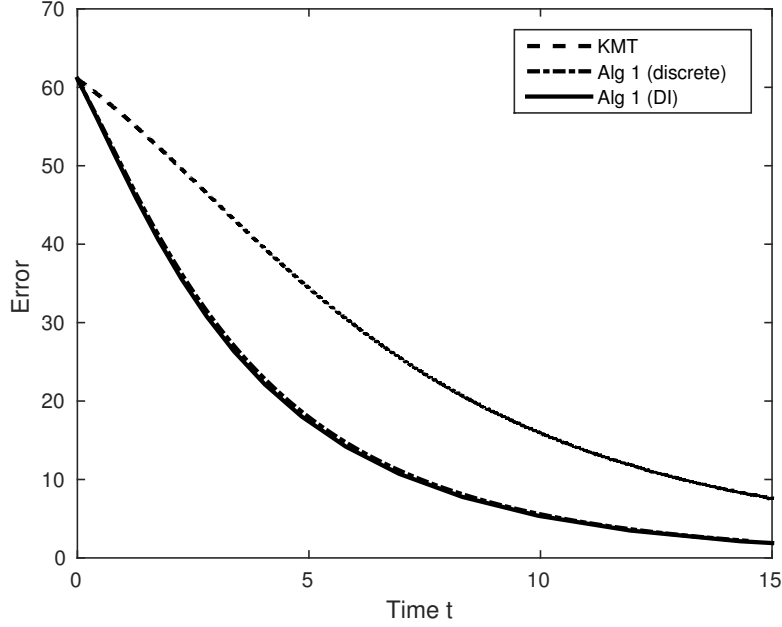
Figure 6: This figure is generated using the same problem parameters used to generate Figure 5(a). The curve KMT corresponds to system (5)-(7) and curve Alg1 (DI) plots the ODE (61) for $\kappa = 1$. Alg1 (discrete) plots the iterates of Algorithm 1 at timeinstants t_k given by (62).

We next compare the case when the stepsizes are according to the Armijo-Goldstein rule. This cannot be done in a distributed fashion since the improvement comparisons require knowledge of the private utility functions. So, for fair comparison, we too use stepsizes according to the Armijo-Goldstein rule to get a centralized variant of Algorithm 1.

As can be seen from the two new plots in Figures 7(a) and 7(b), in both cases, our algorithm does much better than projected gradient descent. In Figure 7(b), our algorithm is very close to the axes. This is quite reassuring.

4. Conclusion

We considered the network utility maximization problem in a distributed framework where the users do not know the network structure or utility functions of other users and the network does not know the users' utility functions. We decomposed the system problem into user subproblems and
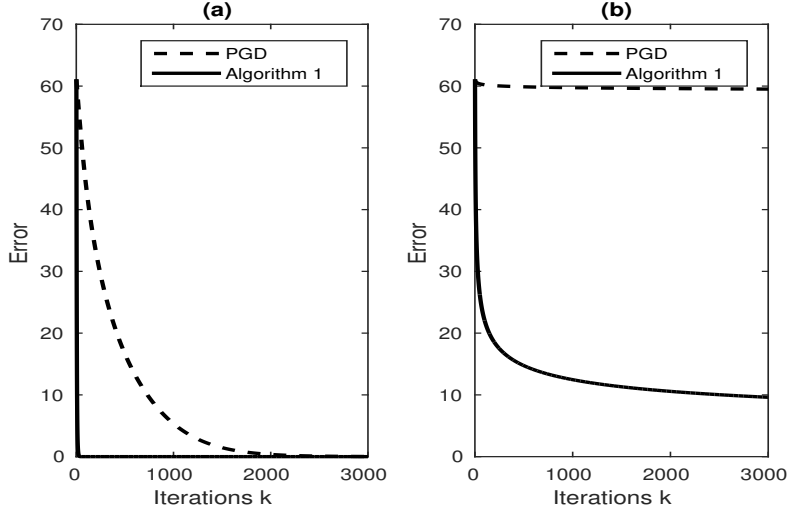
Figure 7: Figure compares the performance of Algorithm 1 with the projected gradient descent algorithm. The plot in Figure 7(a) corresponds to the case when step-size for both algorithms is calculated using Armijo-Goldstein rule whereas for the plots in Figure 7(b) step-size $a_k = \frac{1}{k+1}$ is used.

a network subproblem following the methodology of [1]. Unlike the dual decomposition iterative methods of [7], [1], [8], etc., the iterations proposed in Algorithm 1 ensure feasibility at every step. The convergence of the algorithm was shown using the theory of differential inclusions. The iterates avoid local maxima traps on the facets. Efficient methods to solve the network problem for some special networks were also described. Finally, sample simulations show that, in several examples, Algorithm 1's associated differential inclusion (11) converges faster to the system optimal point when compared with the iterates arising from the ODE (5)-(7). The ODE convergence rate however does not give the full picture of convergence rate of the iterates since the timescale is dictated by the step sizes. For the convergence rate of the iterates, a natural approach is to use the method of Borkar [29, Ch. 4] to get sample complexity bounds. However they do not directly apply since the ODE dynamics is not necessarily Lipschitz, which is a crucial assumption in [29, Ch. 4]. See Appendix B for a discontinuous T mapping. A more intricate analysis of convergence rates is therefore required and is left as future research.

Appendix A. Counterexample to the Algorithm of Hou et al. [15]

In this section, we provide an example where Algorithm 1 does not converge for a constant step size $a_k = \gamma$, $\gamma \in (0, 1)$. Consider a two user single link network. The system problem for the case is

$$\text{Maximize} \quad w_1(x(1)) + w_2(x(2))$$
$$\text{subject to} \quad x(1) + x(2) = B,$$

where B is the capacity of the link. Let $d = (d_1, d_2)$ be the initial flow through the link. Let $f = (f_1, f_2)$ be defined as

$$f = d + \gamma(T(d) - d), \tag{A.1}$$

the flow allocated in the first iteration of Algorithm 1. Without loss of generality, choose

$$d_1 < f_1 < T_1(d) < B. \tag{A.2}$$

Also, choose

$$T_1(f) = d_1 + f_1 - T_1(d) \tag{A.3}$$

The flow allocated to user 1 in the second iteration is

$$f_1 + \gamma(T_1(f) - f_1) \stackrel{(a)}{=} f_1 + \gamma(d_1 - T_1(d))$$
$$\stackrel{(b)}{=} d_1 + \gamma(T_1(d) - d) + \gamma(d - T_1(d))$$
$$= d_1, \tag{A.4}$$

where (a) and (b) are due to (A.3) and (A.1) respectively. Equations (A.1) and (A.4) imply that the flows put out by the algorithm oscillates from d to f and vice versa. It remains to be shown that there exists w_1 and w_2 that is consistent with the choices made in (A.2) and (A.3).

We have, by the definition of $T(.)$,

$$\frac{d_1 w_1'(d_1)}{d_1 w_1'(d_1) + d_2 w_2'(d_2)} = \frac{T_1(d)}{B}, \tag{A.5}$$

$$\frac{f_1 w_1'(f_1)}{f_1 w_1'(f_1) + f_2 w_2'(f_2)} = \frac{T_1(f)}{B}. \tag{A.6}$$
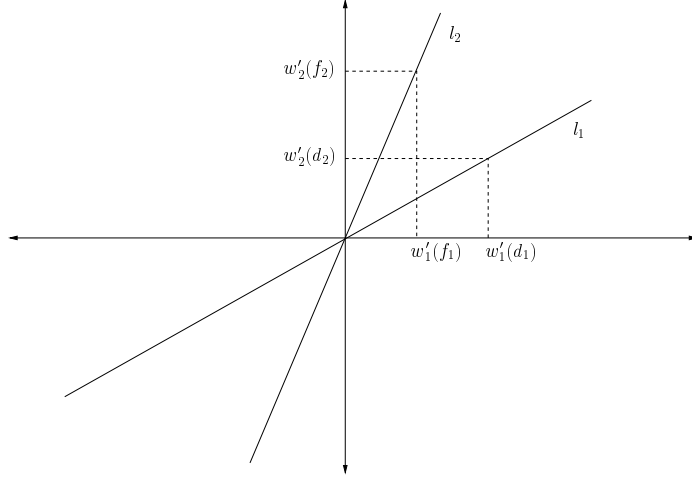
Figure A.8: Figure shows lines l_1 and l_2. $w_1'(d_1)$ and $w_1'(f_1)$ are plotted on the horizontal axis. $w_2'(d_2)$ and $w_2'(f_2)$ are plotted on the vertical axis.

We rewrite the equations (A.5) and (A.6) as

$$d_1(1 - \frac{T_1(d)}{B})w_1'(d_1) - d_2\frac{T_1(d)}{B}w_2'(d_2) = 0, \qquad (A.7)$$

$$f_1(1 - \frac{T_1(f)}{B})w_1'(f_1) - f_2\frac{T_1(f)}{B}w_2'(f_2) = 0. \qquad (A.8)$$

We now view (A.7) and (A.8) as linear equations in $w_1'(d_1), w_2'(d_2)$ and $w_1'(f_1), w_2'(f_2)$ respectively. Lines l_1 and l_2 in Figure A.8 plot (A.7) and (A.8) respectively. Since $d_1 + d_2 = f_1 + f_2 = B$, $d_1 < f_1$ implies $d_2 > f_2$. Also, from (A.4) and the fact that $d_1 < f_1$, we have $T_1(f) < d_1$. Hence, by (A.2), $T_1(f) < T_1(d)$. Since $d_1 < f_1$, $d_2 > f_2$ and $T_1(f) < T_1(d)$, the slope of l_1 is smaller than the slope of l_2.

Since $d_1 < f_1$ and $d_2 > f_2$, by the strict concavity of w_1 and w_2, we must have

$$w_1'(f_1) < w_1'(d_1) \text{ and } w_2'(d_2) < w_2'(f_2). \qquad (A.9)$$

Figure A.8 shows how to choose $w_1'(d_1), w_2'(d_2), w_1'(f_1)$ and $w_2'(f_2)$ satisfying (A.2),(A.3),(A.7) and (A.8).

Appendix B. An Example of a Discontinuous T mapping

In this Appendix, we show that there is no selection from within $T(x)$ that could make the selection a single continuous mapping. Consider a

special case of $T(x)$ as defined below. Take $w_e(\cdot) = w(\cdot)$ for some increasing and strictly concave $w(\cdot)$. Let

$$T(x) = \arg\max_y \sum_{e=1}^{3} p_e(x(e)) \cdot \log(y(e)) \qquad \text{(B.1)}$$

$$\text{subject to} \qquad y(1) \le c,$$

$$y(1) + y(2) \le 2c,$$

$$y(1) + y(2) + y(3) \le 3c.$$

where $p_e(x(e)) = x(e) \cdot w'(x(e))$, $e = 1, 2, 3$. Let $w(\cdot)$ satisfy $x(e) \cdot w'(x(e))|_{x(e)=0} = 0$. Consider $T(x)$ at $x = (c, 0, 0)$. We have

$$T(c, 0, 0) = \{(c, r, s) : r \in [0, c], s \in [0, 2c] : r + s \le 2c\}.$$

Consider a sequence $y^{(k)} \to x$ such that $y^{(k)}(1) = c$, $y^{(k)}(2) = 0$, $y^{(k)}(3) > 0$ for each k and $\lim_{k\to\infty} y^{(k)}(3) = 0$. It is easy to see that $T(y^{(k)}) = \{(c, 0, 2c)\}$ for each k, and so we must select $(c, 0, 2c)$ at $x = (c, 0, 0)$.

Now, consider another sequence $z^{(k)} \to x$. Let $z^{(k)}(1) = c$, $z^{(k)}(2) = z^{(k)}(3) > 0$ for each k and $\lim_{k\to\infty} z^{(k)}(2) = \lim_{k\to\infty} z^{(k)}(3) = 0$. We then have $T(z^{(k)}) = \{(c, c, c)\}$ for each k, and so we must now select (c, c, c) at $x = (c, 0, 0)$. Since these two selections do not match, T cannot be made continuous at $(c, 0, 0)$ by a choice of a value in $T(c, 0, 0)$.

So T has to be dealt with as a set valued mapping, which brings us to differential inclusions.

References

[1] F. P. Kelly, A. K. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research society*, vol. 49, no. 3, pp. 237–252, 1998.

[2] F. Kelly, "Charging and rate control for elastic traffic," *Eur. Trans. Telecommun.*, vol. 8, no. 1, pp. 33–37, Jan. 1997.

[3] B. Johansson, P. Soldati, and M. Johansson, "Mathematical decomposition techniques for distributed cross-layer optimization of data net-

works," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1535–1547, 2006.

[4] G. Iosifidis, L. Gao, J. Huang, and L. Tassiulas, "A double-auction mechanism for mobile data-offloading markets," *IEEE/ACM Transactions on Networking (TON)*, vol. 23, no. 5, pp. 1634–1647, 2015.

[5] K. P. Naveen and R. Sundaresan, "A double-auction mechanism for mobile data-offloading markets with strategic agents," in *Proceedings of the 16th International Symposium on Modeling and Optimization in Mobile, Adhoc, and Wireless Networks, WiOpt*, 2018.

[6] S. Shan, K. N. Narayanan, and L. Umanand, "Virtual energy routers (vers) for energy Internet," in *Proc. of the 2018 IEEE PES Innovative Smart Grid Technologies Conference (ISGT-Europe*, Sarajevo, Bosnia and Herzegovina, 2018.

[7] K. J. Arrow and L. Hurwicz, "On the stability of the competitive equilibrium, I," *Econometrica: Journal of the Econometric Society*, pp. 522–552, 1958.

[8] S. H. Low and D. E. Lapsley, "Optimization flow control-I: basic algorithm and convergence," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 6, pp. 861–874, 1999.

[9] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.

[10] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 1439–1451, 2006.

[11] L. Gao, G. Iosifidis, J. Huang, and L. Tassiulas, "Economics of mobile data offloading," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 3303–3308.

[12] D. S. Hochbaum, "Lower and upper bounds for the allocation problem and other nonlinear optimization problems," *Mathematics of Operations Research*, vol. 19, no. 2, pp. 390–409, 1994.

[13] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking (TON)*, vol. 8, no. 5, pp. 556–567, 2000.

[14] R. J. La and V. Anantharam, "Utility-based rate control in the internet for elastic traffic," *IEEE/ACM Transactions on Networking (TON)*, vol. 10, no. 2, pp. 272–286, 2002.

[15] I.-H. Hou and P. Kumar, "Utility maximization for delay constrained qos in wireless," in *INFOCOM, 2010 Proceedings IEEE*, 2010.

[16] M. Benaim, J. Hofbauer, and S. Sorin, "Stochastic approximations and differential inclusions," *SIAM Journal on Control and Optimization*, vol. 44, no. 1, pp. 328–348, 2005.

[17] C. Berge, *Topological Spaces: including a treatment of multi-valued functions, vector spaces, and convexity.* Courier Corporation, 1963.

[18] N. Megiddo, "Optimal flows in networks with multiple sources and sinks," *Mathematical Programming*, vol. 7, no. 1, pp. 97–107, 1974.

[19] A. Padakandla and R. Sundaresan, "Power minimization for CDMA under colored noise," *IEEE Transactions on Communications*, vol. 57, no. 10, pp. 3103–3112, Oct. 2009.

[20] P. Viswanath and V. Anantharam, "Optimal sequences for CDMA with colored noise: A Schur-saddle function property," *IEEE Trans. Inf. Theory*, vol. IT-48, pp. 1295–1318, Jun. 2002.

[21] D. Palomar, M. A. Lagunas, and J. Cioffi, "Optimum linear joint transmit-receive processing for MIMO channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1179–1197, May 2004.

[22] L. Sanguinetti and A. D'Amico, "Power allocation in two-hop amplify and forward MIMO systems with QoS requirements," *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2494–2507, May 2012.

[23] A. Clark and H. Scarf, "Optimal policies for a multi-echelon inventory problem," *Mangement Science*, vol. 6, pp. 475–490, 1960.

[24] Z. Wang, "On solving convex optimization problems with linear ascending constraints," *Optimization Letters*, vol. 9, no. 5, pp. 819–838, Oct. 2015.

[25] P. T. Akhil and R. Sundaresan, "A survey of algorithms for separable convex optimization with linear ascending constraints," *SADHANA - Academy Proceedings in Engineering Sciences*, `http://arxiv.org/abs/1608.08000`, (in press).

[26] A. F. Veinott Jr., "Least d-majorized network flows with inventory and statistical applications," *Management Science*, vol. 17, pp. 547–567, 1971.

[27] J. A. Muckstadt and A. Sapra, *Principles of Inventory Management*, 2nd ed. Springer, 2010.

[28] D. P. Bertsekas, *Nonlinear programming*. Athena scientific Belmont, 1999.

[29] V. S. Borkar, *Stochastic Approximation: A Dynamical Systems View-Point*. Cambridge University Press, 2008.