Proceedings of Machine Learning Research vol 242:1–23, 2024

Safe Dynamic Pricing for Nonstationary Network Resource Allocation

Berkay Turan BTURAN@UCSB.EDU
University of California, Santa Barbara

Spencer Hutchinson SHUTCHINSON@UCSB.EDU
University of California, Santa Barbara

Mahnoosh Alizadeh ALIZADEH@UCSB.EDU
University of California, Santa Barbara

Abstract

This paper introduces the Safe Pricing for Network Utility Maximization with Gradual Variations (SPNUM-GV) algorithm, addressing challenges in pricing-based distributed resource allocation for safety-critical systems with non-stationary utility functions. Focusing on domains where 1) users' optimal demand can only be induced through posted prices, 2) real-time two-way communication with the users is not available, 3) the induced demand must always belong to an arbitrarily shaped convex and compact feasible set in spite of price response uncertainty, and 4) the users' response to prices are evolving over time, we design SPNUM-GV to generate prices that ensure stage-wise safety of the induced demand while achieving sublinear regret. SPNUM-GV ensures safety by determining a "desired demand" within a shrunk feasible set using a projected gradient method and updating the prices to induce a demand close to the desired demand by leveraging an estimate of the users' price response function. By tuning the amount of shrinkage to account for the error between the desired and the induced demand, we prove that the induced demand always belongs to the feasible set. In addition, we prove that the regret incurred by the induced demand is $\mathcal{O}(\sqrt{T(1+V_T)})$ after T iterations, where V_T is an upper bound on the total gradual variations of the users' utility functions. Numerical simulations demonstrate the efficacy of SPNUM-GV and support our theoretical findings.

Keywords: Distributed Optimization, Network Utility Maximization, Safe Optimization

1. Introduction

As contemporary multi-user optimization paradigms expand into domains with stringent safety requirements, the need for mechanisms that can provide such safety guarantees has grown significantly. One such domain is multi-user resource allocation over networks, which classically has had applications in power distribution systems Samadi et al. (2010), congestion control in data networks Kelly et al. (1998), wireless cellular networks Chiang and Bell (2004), and congestion control in urban traffic networks Mehr et al. (2017). The common optimization goal is to solve the underlying *Network Utility Maximization* (NUM) problem, which aims to find the utility-maximizing resource allocation while meeting the safety-critical constraints of the system.

In this work, we are interested in multi-period NUM problems, where the utility functions of the users are gradually changing over time. Our work is motivated by resource allocation applications where two-way real-time communication with the users is not possible and the resource demand can only be impacted through posted prices. In spite of the general popularity of the NUM framework, the framework studied in this paper presents a unique combination of challenges not addressed by the existing literature. Firstly, the users' resource demand can only be determined by the users according to their own profit-maximizing price-response function, with the actual demand only becoming observable ex-post. Secondly, we allow no negotiations between the central coordinator and the users before posting prices. We assume that there is no private information revealing communication from the users to the central coordinator and the only information the central coordinator observes is the induced demand in response to prices. Thirdly, the systems in question have safety-critical hard constraints forming arbitrary convex and compact sets and the induced demand must meet these constraints at all times for the safe operation of the system, in spite of uncertainty about the users price response. Lastly, the utility functions of the users and therefore their response to prices are changing over time, which adds a layer of uncertainty to the problem. These challenges are however warranted, given that problems of this form appear in many real-world applications. For example, in price-based demand response, users determine their own electricity consumption in response to prices that must be set such that the realized demand does not violate the power flow constraints of the grid Vardakas et al. (2014). Power flow constraints are nonlinear and nonconvex in general (Molzahn et al. (2017)) and often solved with (nonlinear) convex relaxations (e.g. Bai et al. (2008); Farivar and Low (2013)).

Our algorithm, called Safe Pricing for NUM with Gradual Variations (SPNUM-GV), addresses the aforementioned challenges in pricing-based resource allocation frameworks with non-stationary users. SPNUM-GV determines a *desired demand* by moving the current demand vector along the direction of the current price vector and projecting it onto a shrunk feasible set, which behaves as a projected gradient method. Then, using estimates of the price response functions of the users around the current prices, it determines the updated prices that would induce a demand close to the desired demand. By carefully adjusting the amount of shrinkage to account for the estimation errors as well as the gradual variations of the utility functions, it ensures safety.

Related Work: Scholars have extensively studied pricing algorithms for NUM with *stationary* utility functions. Restricting the relevant literature to the first-order methods that are closest to our setup, the majority of studies focus on linear constraints Nedić and Ozdaglar (2009); Beck et al. (2014); Necoara and Nedelcu (2013, 2015), or on non-linear constraints with the assumption of separability and full user knowledge of these constraints Simonetto and Jamali-Rad (2016); Falsone et al. (2017); Notarnicola and Notarstefano (2019). However, none of the aforementioned studies propose an iterative pricing algorithm that induces resource demand satisfying the hard constraints of the problem *during* the iterative optimization process. Instead, these studies only provide bounds on the infeasibility amount of the resource demand (e.g., Beck et al. (2014); Necoara and Nedelcu (2015)). Lastly, Turan et al. (2023b) proposes a stage-wise safe pricing-based method for NUM problems with general convex feasible sets, however, is restricted to stationary utility functions.

On the other hand, the online convex optimization (OCO) literature handles non-stationary utility functions we are interested in this work. For instance, Zinkevich (2003); Jadbabaie et al. (2015); Hazan et al. (2016); Guo et al. (2022); Chaudhary and Kalathil (2022) ensure stage-wise safety through projection-based methods, which optimize the primal variables directly. However, a key challenge in our setup is that users exclusively control the demand, i.e., the primal variables. While a feasible demand can be determined using a projected gradient method, the corresponding prices for inducing such demand remain unknown due to the privacy of utility functions. Accordingly, the distributed OCO literature, e.g., Mahdavi et al. (2012); Chen et al. (2017); Yuan et al. (2017); Li et al. (2020); Yu and Neely (2020), pursue optimization through pricing-type signals in multi-user settings. However, their safety guarantees are not strict and stage-wise, but in the form of sublinear cumulative constraint violation. Additionally, they adopt primal-dual optimization methods to solve the Lagrangian dual problem, which restricts the users to follow a primal update method that cannot be enforced in the resource allocation framework where users only care about maximizing their own profit. As such, our setup brings unique challenges not addressed by the prior art.

Notation and Basic Definitions: For vectors, $\|\cdot\|$ denotes the standard Euclidean norm. Given a positive integer $n > 0$, $[n]$ denotes the set of integers $\{1, 2, \ldots, n\}$. Given a vector $x \in \mathbb{R}^n$, $x_i \in \mathbb{R}$ denotes the i'th entry of x. For a matrix $A \in \mathbb{R}^{m \times n}$, A_j denotes the j'th row of A. For two vectors $x, y \in \mathbb{R}^n$, $\langle x, y \rangle$ denotes the inner product of x and y and $x \leq y$ implies element-wise inequality. Given a function $f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$ (including $n = 1$), ∇f denotes the gradient of f, $\nabla^k f$ denotes the k'th order gradient of f, and $\text{dom} f$ denotes the domain $\mathcal{X}$ of f. Given a set $\mathcal{X} \subset \mathbb{R}^n$, $\mathcal{X}^{\text{int}}$ denotes the interior of $\mathcal{X}$. Given a convex and compact set $\mathcal{X} \subset \mathbb{R}^n$ and a point $x \in \mathbb{R}^n$, $\Pi_{\mathcal{X}}(x)$ denotes the Euclidian projection of x onto $\mathcal{X}$. We denote the closed Euclidean ball with radius r centered at origin as $\bar{\mathcal{B}}(r)$.

Definition 1 *A differentiable function $f(\cdot)$ is said to be **μ-strongly concave** over the domain $\mathcal{X}$ if there exists $\mu > 0$ such that the following holds for all $x_1, x_2 \in \mathcal{X}$:*

$$\langle \nabla f(x_2) - \nabla f(x_1), x_1 - x_2 \rangle \geq \mu \|x_1 - x_2\|^2. \tag{1}$$

Definition 2 *A differentiable function $f(\cdot)$ is said to be **L-smooth** over the domain $\mathcal{X}$ if there exists $L > 0$ such that the following holds for all $x_1, x_2 \in \mathcal{X}$:*

$$\|\nabla f(x_1) - \nabla f(x_2)\| \leq L \|x_1 - x_2\|. \tag{2}$$

Definition 3 *A function $f(\cdot)$ is said to be **M-Lipschitz continuous** over the domain $\mathcal{X}$ if there exists $M > 0$ such that the following holds for all $x_1, x_2 \in \mathcal{X}$:*

$$\|f(x_1) - f(x_2)\| \leq M \|x_1 - x_2\|. \tag{3}$$

2. Network Utility Maximization with Gradual Variations

2.1. Problem Setup

We study the online version of the standard NUM problem Kelly et al. (1998), where the goal is to allocate resources to n users subject to a set of coupling constraints such that the total utility of the users over a horizon of T is maximized. It can be formulated as the following optimization problem:

$$\max_{\{x^t \in \text{dom} f^t \subseteq \mathbb{R}^n, \forall t \in [T]\}} \sum_{t=1}^{T} f^t(x^t) = \sum_{t=1}^{T} \sum_{i=1}^{n} f_i(x_i^t) \tag{4a}$$

$$\text{subject to} \qquad x^t \in \mathcal{X},\ \forall t \in [T], \tag{4b}$$

where $f_i^t(\cdot)$ is the concave utility function of user i at time t that is a function of the resource demand x_i^t and $\mathcal{X} \subset \mathbb{R}^n$ is the convex and compact set of feasible resource allocations.

For all $i \in [n]$, we define $\underline{x}_i = \min_{x \in \mathcal{X}} x_i$ and $\bar{x}_i = \max_{x \in \mathcal{X}} x_i$, which indicate the minimum and the maximum values of user i's resource demand can take in the feasible region, and let $\mathcal{X}_i = [\underline{x}_i, \bar{x}_i]$. Note that, if $x \in \mathcal{X}$, then $x_i \in \mathcal{X}_i$ and if $x \in \mathcal{X}^{\text{int}}$, then $x_i \in \mathcal{X}_i^{\text{int}}$ hold by definition. We make the following assumptions on the feasible set $\mathcal{X}$, and on the utility functions over $\mathcal{X}_i$, $\forall i \in [n]$.

Assumption 1 *Let* $\operatorname{dom} f = \bigcap_{t=1}^{T} \operatorname{dom} f^t$*. The feasible set* $\mathcal{X}$ *is a subset of* $\operatorname{dom} f$*, i.e.,* $\mathcal{X} \subseteq \operatorname{dom} f$*. The diameter of the feasible set* $\mathcal{X}$ *is bounded by* R*, i.e.,* $\|x - y\| \leq R$*,* $\forall x, y \in \mathcal{X}$*. There exists a vector* $\tilde{x}$ *in the interior of* $\mathcal{X}$ *such that* $\tilde{x} \in \mathcal{X}^{\text{int}}$.

Assumption 2 *For all* $i \in [n]$ *and* $t \in [T]$*, the utility function* $f_i^t(\cdot)$ *is* μ*-strongly concave,* L*-smooth,* M*-Lipschitz continuous, and has* β*-smooth gradient over* $\mathcal{X}_i$.

Example 1 (Utility function) *For instance, take* $f_i^t(x_i) = f_\alpha(x_i)$ *to be an* α*-fair utility function (see Mo and Walrand (2000)) and let* $\mathcal{X}_i = [\underline{x}_i, \bar{x}_i]$ *with* $\underline{x}_i > 0$*. We have that* $\nabla f_i^t(x_i) \leq 1/\underline{x}_i^\alpha$*,* $-\alpha/\underline{x}_i^{\alpha+1} \leq \nabla^2 f_i^t(x_i) \leq -\alpha/\bar{x}_i^{\alpha+1}$*, and* $\alpha(\alpha+1)/\bar{x}_i^{\alpha+2} \leq \nabla^3 f_i^t(x_i) \leq \alpha(\alpha+1)/\underline{x}_i^{\alpha+2}$*,* $\forall x \in \mathcal{X}_i$*. Therefore,* $f_i(x_i)$ *is* $\alpha/\bar{x}_i^{\alpha+1}$*-strongly concave,* $\alpha/\underline{x}_i^{\alpha+1}$*-smooth, and* $1/\underline{x}_i^\alpha$*-Lipschitz continuous, and has* $\alpha(\alpha+1)/\underline{x}_i^{\alpha+2}$*-smooth gradient over* $\mathcal{X}_i$.

In non-stationary environments, achieving meaningful findings often involves bounding temporal changes using diverse metrics. Within the realm of online convex optimization, the consideration typically revolves around the bounded total variation of the functions, the optimal solutions, or the function gradients over a horizon of T. In this paper, we adopt a stage-wise bounded variation in the gradients, which is formalized as follows:

Assumption 3 *For all users* $i \in [n]$ *and for all* $t \in [T]$*, there exists a known bound* V^t *on the change in the gradients:*

$$\sup_{x_i \in \mathcal{X}_i} \|\nabla f_i^t(x_i) - \nabla f_i^{t+1}(x_i)\| \leq V^t. \tag{5}$$

This type of variation in the gradients was first introduced in Chiang et al. (2012), however, using a bound on the total variation over a horizon of T (i.e., $\sum_{t\in[T]} V^t$ in our setup). The known bound on the stage-wise gradual variation assumption we impose is more restrictive, however, it is necessary to ensure the stage-wise safety of any pricing algorithm. For brevity of notation, we define $V_T = \sum_{t\in[T]} V^t$ be the total gradual variation over T periods.

Since $f_i^t(\cdot)$ are private to the users, (4) cannot be solved centrally. Therefore, pricing-based distributed optimization methods have been proposed in the literature (e.g., Palomar and Chiang (2006) for the case when $\mathcal{X}$ is a polytope and $f^t(x) = f(x)$, $\forall t \in [T]$, but without stage-wise safety) in order to incentivize selfish users with private utility functions to follow the optimal global solution. The common high-level idea is to divide the main problem into subproblems that the individual users can solve upon observing a pricing signal, and iteratively design prices $\{p^0, p^1, \dots\}$ to achieve near-optimal resource allocation. In this framework, upon observing a price $p_i \in \mathbb{R}$, each user $i \in [n]$ determines their own decision variable, i.e., resource demand, according to their own profit maximization problem:

$$g_i^t(p_i) = \underset{x_i \in \operatorname{dom} f_i^t}{\arg\max} f_i^t(x_i) - \langle p_i, x_i \rangle. \tag{6}$$

We call $g_i^t(\cdot)$ the price response function of user i at time t and let $g^t(p) = [g_1^t(p_1),\ g_2^t(p_2), \dots,\ g_n^t(p_n)]$ be the concatenated vector of price responses given a price vector $p \in \mathbb{R}^n$.

2.2. Performance and Safety Metrics

Regarding the performance of induced resource demand $\{x^t\}_{t\in[T]} = \{g^t(p^t)\}_{t\in[T]}$ in response to online prices $\{p^t\}_{t\in[T]}$, we adopt the static regret as a metric. The static regret measures the difference between the utility gain of an online pricing algorithm and that of the best fixed resource allocation in hindsight. It can be written as

$$R(T) = \sum_{t=1}^{T} f^t(x^\star) - f^t(x^t), \tag{7}$$

where

$$x^\star = \arg\max_{x\in\mathcal{X}} \sum_{t=1}^{T} f^t(x) \tag{8}$$

Under Assumption 2, the optimization problem (8) is strongly concave with coefficient μ and therefore has a unique solution denoted by $x^\star$.

Remark 4 *Although the term static refers to $x^\star$ being the static solution for all time periods, it necessitates a dynamic pricing scheme to implement the solution. This is because the users' demand is dictated by (6) and according to the first-order optimality condition, $\nabla f_i^t(x_i^\star) = p_i^t$. Therefore, even if $x^\star$ is static, the actual actions (prices $\{p^t\}_{t\in[T]}$) to implement this solution are dynamic. On the other hand, one could compare the performance of the online pricing algorithm to that of the best static prices. However, in Appendix I of the full online version of this paper Turan et al. (2023a), we show that a static pricing policy is in general infeasible, i.e., not safe. Additionally, in Appendix G of the full online version Turan et al. (2023a), we extend our analysis to bound the dynamic regret performance of our algorithm.*

In addition to the regret as a performance metric, we require that the demand induced by the prices remain in the feasible set of the problem. This can formally be stated as

$$g^t(p^t) \in \mathcal{X},\ \forall t \in [T]. \tag{9}$$

This is a natural requirement for the static regret to be a valid choice of performance metric for NUM problems with hard constraints, which is common for multi-user resource allocation frameworks with physical limitations. The definition for static regret in (7), measures the cumulative sum of instantaneous regrets $f^t(x^\star) - f^t(x^t)$ for $t \in [T]$, where the instantaneous regret at time t quantifies the difference between utility gained by the resource demand $x^\star$ and that of x^t. Accordingly, for the users to gain the utility $f^t(x^t)$ at time t, the demand x^t should be realized, which must meet the hard constraints for the safe operation of the physical system.

3. Safe Pricing for NUM with Gradual Variations

In this section, we present the Safe Pricing for NUM with Gradual Variations (SPNUM-GV) algorithm, outlining a price update method that induces feasible resource demand at every iteration. Our approach involves leveraging definitions and findings from Hutchinson et al. (2023) concerning the geometric characteristics of convex and compact sets. Although Hutchinson et al. (2023) primarily delves into a linear stochastic bandit setup, which differs significantly from the NUM setup under examination, the definitions of the shrunk set provided therein remain relevant to our current context.

Algorithm 1: Safe Pricing for NUM with Gradual Variations (SPNUM-GV)

Input: p^0, $\Delta^t = \delta^t + \epsilon^t$, γ^t, η^t.

`// Initialization Stage`

Each user $i \in [n]$ receives p_i^0, $p_i^{0,s} = p_i^0 + \eta^0$ and solves:

$$[x_i^0,\ x_i^{0,s}] = [g_i^0(p_i^0),\ g_i^0(p_i^{0,s})] \tag{10}$$

`// Safe Price Update Stage`

for $t = 0$ **to** $T-1$ **do**

Compute $\hat{x}^{t+1} = \Pi_{\mathcal{X}_{\Delta^t}}(x^t + \gamma^t p^t)$.

Set $p_i^{t+1} = p_i^t + \frac{p_i^{t,s} - p_i^t}{x_i^{t,s} - x_i^t}(\hat{x}_i^{t+1} - x_i^t)$, for all $i \in [n]$.

Each user $i \in [n]$ receives $p_i^{t+1}, p_i^{t+1,s} = p_i^{t+1} + \eta^{t+1}$ and solves

$$[x_i^{t+1},\ x_i^{t+1,s}] = [g_i^{t+1}(p_i^{t+1}),\ g_i^{t+1}(p_i^{t+1,s})] \tag{11}$$

end

3.1. Geometric Properties of the Feasible Set

The main ingredient that ensures the safety of SPNUM-GV is that it operates on a shrunk feasible set, which is formally defined as follows:

Definition 5 *For a compact set $\mathcal{X} \subset \mathbb{R}^n$ and a positive scalar $\Delta \in \mathbb{R}_+$, we define the **shrunk version** of $\mathcal{X}$ as $\mathcal{X}_\Delta := \{x \in \mathcal{X} : x + v \in \mathcal{X}, \forall v \in \bar{\mathcal{B}}(\Delta)\}$.*

Example 2 (Shrunk polytope) *Let $A \in \mathbb{R}^{m \times n}$ and $\mathcal{X} = \{x \in \mathbb{R}^n : Ax \le c\}$ be a polytope. The shrunk version of $\mathcal{X}$ is defined as $\mathcal{X}_\Delta = \{x \in \mathbb{R}^n : A_j^\top x \le c_j - \Delta\|A_j\|,\ j \in [m]\}$.*

Remark 6 *If $\mathcal{X}$ is convex and compact, then $\mathcal{X}_\Delta$ is also convex and compact.*[1]

Given the above definition of the shrunk version of a set, one can consider the maximum shrinkage that a set can withstand while still being nonempty. We introduce the *maximum shrinkage of a set* in the following definition.

Definition 7 *For a compact set $\mathcal{X} \subset \mathbb{R}^n$, we define the **maximum shrinkage** of $\mathcal{X}$, as $H_{\mathcal{X}} := \sup\{\Delta : \mathcal{X}_\Delta \neq \emptyset\}$.*

3.2. Description of the Algorithm

The algorithm, called Safe Prcing for NUM with Gradual Variations (SPNUM-GV), is outlined in Algorithm 1. The algorithm is an online projected gradient method on the primal variables at its

1. We can equivalently define $\mathcal{X}_\Delta$ using Minkowski subtraction. The Minkowski subtraction of sets $A, B \subseteq \mathbb{R}^n$ is defined as $A \ominus B := \{a - b : a \in A, b \in B\}$, or equivalently, $A \ominus B = \bigcap_{b \in B}(A - b)$. Therefore, $\mathcal{X}_\Delta = \mathcal{X} \ominus \mathcal{B}(\Delta)$ is an intersection of convex and closed sets and hence is convex and closed (Schneider, 2014, Section 3.1). By Definition 5, $\mathcal{X}_\Delta$ is a subset of $\mathcal{X}$, and therefore bounded. A closed and bounded convex set is convex and compact.

core, which determines a *desired demand* in Step 2. However, note that the primal variables, i.e., the resource demand, have to be induced through prices, and the users' price response is not known. As such, SPNUM-GV determines prices that would induce a resource demand *close* to the desired demand in Step 3, given its current knowledge of the price response. To do so, the algorithm uses a linear estimation of the price response function via two-point feedback in Step 4. Therefore, central to the algorithm are two crucial steps that go hand-in-hand for safety:

Step 1: Given $x^t \in \mathcal{X}^{\text{int}}$ and p^t, Step 2 can be seen as an online projected gradient step on $\mathcal{X}_{\Delta^t}$ to determine a desired demand $\hat{x}^{t+1}$. This is due to the first-order optimality condition for $g^t(p^t) = \arg\max_{x \in \text{dom} f^t} f^t(x) - \langle p^t, x \rangle$ necessitating $\nabla f^t(x^t) = p^t$ if $x^t \in \mathcal{X}^{\text{int}}$ (since Assumption 1 implies that $\text{dom} f^t \subseteq \mathcal{X}$). The projection is however done onto the shrunk set $\mathcal{X}_{\Delta^t}$, where Δ^t is the amount of shrinkage. This is essential to the algorithm's safety because the uncertainty in the price response functions may cause the actual induced demand x^{t+1} in response to the price vector p^{t+1} to deviate from the desired demand $\hat{x}^{t+1}$. Incorporating this margin into the feasible set $\mathcal{X}$ ensures safety if $\|x^{t+1} - \hat{x}^{t+1}\| \in \bar{\mathcal{B}}(\Delta^t)$.

Step 2: Upon determining the desired demand $\hat{x}^{t+1}$, the task is to establish p_i^{t+1} to ideally induce $\hat{x}_i^{t+1}$ for all $i \in [n]$. Given the lack of knowledge of the price response function $g_i^{t+1}(p)$, it is not possible to determine p_i^{t+1} such that $g_i^{t+1}(p_i^{t+1}) = \hat{x}_i^{t+1}$. Instead, the central coordinator uses a linear approximation of the price response function g_i^t at the previous timestep around p_i^t using the two-point feedback $(x_i^t, x_i^{t,s})$ to $(p_i^t, p_i^{t,s})$:

$$\hat{g}_i^t(p) = g_i^t(p_i^t) + \frac{x_i^{t,s} - x_i^t}{p_i^{t,s} - p_i^t}(p - p_i^t). \tag{12}$$

By setting $p = p_i^{t+1}$, $g_i^t(p_i^t) = x_i^t$, and $\hat{g}_i^t(p_i^{t+1}) = \hat{x}_i^{t+1}$, we get the price update rule in Step 3. Finally, the central coordinator broadcasts p^{t+1} and $p^{t+1,s}$ to get the two-point feedback to be used in the next iteration. Such two-point feedback mechanisms are common in the online optimization literature, e.g., Gao et al. (2018); Cao and Başar (2021).

Using the Taylor series expansion $g_i^t(p_i^{t+1}) = g_i^t(p_i^t) + \nabla g_i^t(p_i^t)(p_i^{t+1} - p^t) + R_1$ we can write the induced demand $x_i^{t+1} = g_i^{t+1}(p_i^{t+1})$ as:

$$g_i^{t+1}(p_i^{t+1}) = g_i^{t+1}(p_i^{t+1}) - g_i^t(p_i^{t+1}) + g_i^t(p_i^t) + \nabla g_i^t(p_i^t)(p_i^{t+1} - p_i^t) + R_1, \tag{13}$$

where R_1 includes the higher order terms. Comparing (13) and (12), we observe that for user i, the error between $\hat{x}_i^{t+1} = \hat{g}_i^t(p_i^{t+1})$ and $x_i^{t+1} = g_i^{t+1}(p_i^{t+1})$ stems from three sources: 1) the difference between the price response functions $g_i^t(p_i^{t+1})$ and $g_i^{t+1}(p_i^{t+1})$, i.e., the gradual variations, 2) the difference between the estimated derivative $(x_i^{t,s} - x_i^t)/(p_i^{t,s} - p_i^t)$ and $\nabla g_i^t(p_i^t)$, and 3) the high order terms not captured by the linear approximation, i.e., R_1. By properly choosing Δ^t to accommodate for the total error caused by those three sources, we can ensure safety. In particular, we let $\Delta^t = \epsilon^t + \delta^t$ and tune ϵ^t to handle source 1 and δ^t to handle sources 2 and 3.

For the algorithm to proceed as described above, the initial price vectors $p^0, p^{0,s}$ should induce the demand vectors $x^t, x^{t,s} \in \mathcal{X}^{\text{int}}$. Since this has to hold before getting any feedback from the users, we make the following assumption:

Assumption 4 *There exists a known price vector p^0 such that $g^0(p^0), g^0(p^{0,s}) \in \mathcal{X}^{\text{int}}$.*

Remark 8 *One way to satisfy Assumption 4 is to choose* η^0 *such that* $\mathcal{X}_{\frac{\sqrt{n}\eta^0}{\mu}}$ *is non-empty and* p^0 *such that* $g(p^0) \in \mathcal{X}_{\frac{\sqrt{n}\eta^0}{\mu}}$*, which is proven in Appendix F of the full online version Turan et al. (2023a).*

In the next section, we characterize a principled way to choose parameters Δ^t, γ^t, and η^t in order to induce feasible resource demand. Additionally, we prove that the regret incurred by the demand induced by Algorithm 1 is $\mathcal{O}(\sqrt{T(1+V_T)})$ after T iterations.

4. Safety and Regret Analysis

In order to prove the safety and the regret guarantees of our algorithm, we will need to bound the distance between a point in $x \in \mathcal{X}$ and its projection onto the shrunk set $\Pi_{\mathcal{X}_\Delta}(x)$. The following definition from Hutchinson et al. (2023) formalizes this notion called the *sharpness of a set*, which is defined as the maximum distance from any point in a set to the projection of it onto the shrunk version of that set.

Definition 9 *For a convex and compact set* $\mathcal{X} \subset \mathbb{R}^n$*, we define the* ***sharpness*** *of* $\mathcal{X}$ *as*

$$\mathrm{Sharp}_{\mathcal{X}}(\Delta) := \sup_{x \in \mathcal{X}} \|\Pi_{\mathcal{X}_\Delta}(x) - x\|, \tag{14}$$

for all non-negative Δ *such that* $\mathcal{X}_\Delta$ *is nonempty.*

The following proposition establishes a bound on the sharpness of convex and compact sets as a linear function of Δ:

Proposition 10 *(Hutchinson et al., 2023, Corollary 11) For a convex, compact set* $\mathcal{X} \subset \mathbb{R}^n$ *with non-empty interior, we have that* $\mathrm{Sharp}_{\mathcal{X}}(\Delta) \leq \Gamma_{\mathcal{X}}\Delta$ *where* $\Gamma_{\mathcal{X}} \geq 1$ *is a constant that depends only on the geometry and the dimension of* $\mathcal{X}$*.*

Example 3 (Sharpness of a polytope Hutchinson et al. (2023)) *Let* $\mathcal{X} = \{x \in \mathbb{R}^n : Ax \leq c\}$ *be a polytope with a nonempty interior. Define* $\mathcal{I}_A$ *to refer to the collection of all sets of* d *indices such that for each* $\{i_1, i_2, ..., i_d\} \in \mathcal{I}_A$ *the vectors* $A_{i_1}, A_{i_2}, ..., A_{i_d}$ *are linearly independent. For each* $\ell \in \mathcal{I}_A$ *where* $\ell = \{i_1, i_2, ..., i_d\}$*, we define* $A^\ell = [A_{i_1}^\top \; A_{i_2}^\top \; ... \; A_{i_d}^\top]^\top$*. We have that* $\mathrm{Sharp}_{\mathcal{X}}(\Delta) \leq \sqrt{d}K_{\mathcal{X}}\Delta$*, where* $K_{\mathcal{X}} := \max_{\ell \in \mathcal{I}_A} \kappa(A^\ell)$ *and* $\kappa(A^\ell)$ *is the condition number of* A^ℓ*.*

Example 4 (Sharpness of a ball in $\mathbb{R}^n$) *Let* $\mathcal{X} = \{x \in \mathbb{R}^n : (x - x_0)^\top (x - x_0) \leq r^2\}$ *be a ball in* $\mathbb{R}^n$ *with radius* r *centered at* x_0*. We have that* $\mathrm{Sharp}_{\mathcal{X}}(\Delta) = \Delta$*.*

Although we do not specify a closed-form expression of $\Gamma_{\mathcal{X}}$ for a general convex and compact set $\mathcal{X}$, it relates to the sharpness of polytopes that are contained in $\mathcal{X}$, which have closed-form bounds as given by Example 3. We refer the reader to Hutchinson et al. (2023) (Proposition 10) for a detailed discussion.

In the following subsections, we will first characterize the choice of algorithm parameters that guarantee primal feasibility at all iterations and then prove the regret of Algorithm 1 under this choice of parameters.

4.1. Feasibility Analysis

The following proposition characterizes the choice of the parameters Δ^t, γ^t, and η^t to induce feasible resource demand:

Proposition 11 *Suppose that*

$$V^t < \min\left\{\frac{\mu^4}{12n\beta L^2\Gamma_{\mathcal{X}}^2}, \frac{\mu H_{\mathcal{X}}}{2\sqrt{n}}\right\} \tag{15}$$

holds for all $t \in [T]$. *Let* $\Delta^t = \delta^t + \epsilon^t$ *and choose algorithm parameters satisfying*

$$\epsilon^t = 2\sqrt{n}V^t/\mu, \tag{16}$$

$$\gamma^t \leq \min\left\{\sqrt{\frac{(H_{\mathcal{X}} - \epsilon^t)\mu^3}{8\beta L^2 n^{3/2} M^2}}, \frac{\mu^3}{8\beta L^2 \Gamma_{\mathcal{X}} M n}\right\}, \tag{17}$$

$$\delta^t = \frac{8\beta L^2 n^{3/2} M^2}{\mu^3}(\gamma^t)^2, \tag{18}$$

$$\eta^t < \min\left\{\frac{L(M\sqrt{n}\gamma^t + \Delta^t\Gamma_{\mathcal{X}})}{2}, \frac{\mu\delta^{t-1}}{4\sqrt{n}}\right\}. \tag{19}$$

Then for all $t \geq 0$, $\|\hat{x}^{t+1} - x^{t+1}\| \leq 3\delta^t/4 + \epsilon^t$ *and* $\|x^{t+1} - x^{t+1,s}\| < \delta^t/4$. *Accordingly, the demand vectors* x^t *and* $x^{t,s}$ *induced by Algorithm 1 are in the strict interior of the feasible set.*

Remark 12 *The upper bound specified in (15) on the stage-wise gradual variation* V^t *is a sufficient condition for the safety of the algorithm. Note that the shrinkage* Δ^t *consists of two terms:* δ^t *and* ϵ^t. *Intuitively,* ϵ^t *accounts for the gradual variation* V^t *since* $\epsilon^t = \mathcal{O}(V^t)$ *in (16). However, to account for large variations* V^t, ϵ^t *needs to be large. The analysis shows that the error* $\|\hat{x}^{t+1} - x^{t+1}\|$ *has a dependency on* ϵ^t *as* $\mathcal{O}((\epsilon^t)^2)$, *which necessitates* ϵ^t *and therefore* V^t *to be bounded by the first argument in the minimum operator specified in (15). In addition, a large* ϵ^t *due to a large* V^t *could cause the shrunk set* $\mathcal{X}_{\Delta^t}$ *to become empty. Accordingly, the second argument in the minimum operator specified in (15) ensures that we can shrink* $\mathcal{X}$ *by an amount* ϵ^t *without causing it to become empty, i.e.,* $\epsilon^t < H_{\mathcal{X}}$. *Furthermore, since* $\delta = \mathcal{O}((\gamma^t)^2)$ *in (18), we ensure that* $\delta^t + \epsilon^t < H_{\mathcal{X}}$ *by imposing the upper bound on* γ^t *in (17) specified by the first argument of the minimum operator.*

The proof of Proposition 11 can be found in Appendix A of the full online version Turan et al. (2023a). Given that under Proposition 11, x^t for all $t \geq 1$ are feasible and therefore implementable, the static regret (7) is a valid choice of performance metric. Next, we prove that the regret of Algorithm 1 is $\mathcal{O}(\sqrt{T(1+V_T)})$.

4.2. Regret Analysis

The following theorem establishes an upper bound on the regret incurred by Algorithm 1.

Theorem 13 *Let* $\gamma^t = \gamma$ *be a constant satisfying (17) for all* $t \in [T]$ *and choose* δ^t, ϵ^t, *and* η^t *as in Proposition 11. Then for all* $t \geq 0$, *the demand vectors induced by Algorithm 1 are feasible. Furthermore, the regret* $R(T)$ *for* $T \geq 1$ *satisfies*

$$R(T) \leq \mathcal{O}\left(\gamma T + (1 + V_T)/\gamma\right) \tag{20}$$

where $\mathcal{O}(\cdot)$ *hides other constants.*

The proof of Theorem 13 and the explicit constants of (20) can be found in Appendix B of the full online version Turan et al. (2023a). Based on Theorem 13, we can arrive at the following corollary regarding the optimal step size.

Corollary 14 *Let $\gamma = c_1\sqrt{(1+V_T)/T}$ for some c_1 such that (17) holds for all $t \in [T]$. Then*

$$R(T) \leq \mathcal{O}\left(\sqrt{T(1+V_T)}\right). \tag{21}$$

The proof of Corollary 14 and the explicit constants of (21) can be found in Appendix C of the full online version Turan et al. (2023a). Accordingly, Algorithm 1 with constant step size $\gamma^t = \gamma = \mathcal{O}(\sqrt{(1+V_T)/T})$, $\forall t \in [T]$, produces feasible solutions that achieve a regret of $\mathcal{O}(\sqrt{T(1+V_T)})$. Hence, if the total variation V_T is sublinear in T, Algorithm 1 achieves a sublinear regret.

Remark 15 *In Hazan et al. (2016) it was shown that the lower bound for static regret with strongly concave objective functions is $\Omega(\log(T))$. Compared with (21), our result has a worse dependency on T as $\sqrt{T}$, and furthermore a dependence on the total variation as $\sqrt{V_T}$. This is the trade-off for ensuring safety: The conservative approach through shrinkage of the feasible set results in an extra $\mathcal{O}(\sum_{t\in[T]} V^t/\gamma^t)$ term in the regret analysis. Therefore, we can not utilize diminishing step sizes in the form of $\gamma^t = \mathcal{O}(1/t)$ to exploit the strong concavity of the problem as common in the literature to achieve faster convergence rates.*

5. Numerical Study

In this section, we numerically demonstrate the regret and safety guarantees of SPNUM-GV on a problem with a feasible set characterized by non-linear inequalities. We select the feasible set $\mathcal{X} = \{x \in \mathbb{R}^n : \|x\| \leq 1\}$ as the unit ball in $\mathbb{R}^n$ centered at the origin, and the utility functions as $f_i^t(x_i) = -0.5(x_i - y_i)^2 - x_i - (\theta_i + \nu_i^t)\log(1 + e^{x_i})$ for all $i \in [n]$, where θ_i is sampled uniformly from $[0.1, 0.9]$ and y_i is sampled uniformly from $[-2, 2]$ at random at the beginning of each run. ν_i^t is a random variable bounded as $|\nu_i^t| \leq 0.1,\ \forall i \in [n], t \in [T]$, and sampled at each iteration to account for the gradual variations.

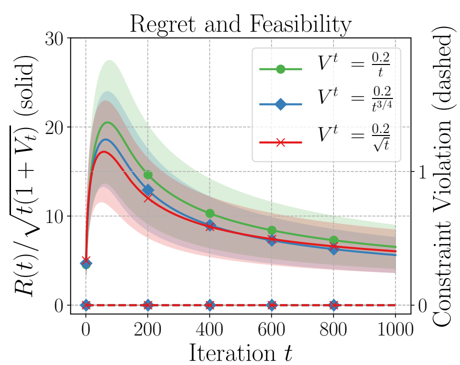

Figure 1: Results for the numerical study on SPNUM-GV. The regret divided by $\sqrt{t(1+V_t)}$ is plotted in solid lines, and constraint violation is plotted in dashed lines, where constraint violation is 0 if $x^t \in \mathcal{X}$ and 1 otherwise.

We studied 3 different configurations of ν_i^t, where at each iteration t, ν_i^t is sampled uniformly from $[-0.1, 0.1]$ and scaled by $\{\frac{1}{t}, \frac{1}{t^{1/2}}, \frac{1}{t^{3/4}}\}$. Accordingly, we set $V^t \in \{\frac{0.2}{t}, \frac{0.2}{t^{1/2}}, \frac{0.2}{t^{3/4}}\}$ to be known upper bounds on the gradual variations. For each configuration, we ran SPNUM 50 times for a horizon of $T = 1000$. The results are illustrated in Figure 1. The figure shows that **1)** the regret of SPNUM-GV grows as $\mathcal{O}((\sqrt{t(1+V_t)})$ and **2)** SPNUM-GV guarantees feasible demand at all iterations. Additional details of this numerical study and other studies can be found in Appendix H of the full online version Turan et al. (2023a).

Acknowledgments

This work was supported by NSF grant #1847096.

References

Xiaoqing Bai, Hua Wei, Katsuki Fujisawa, and Yong Wang. Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, 30(6-7): 383–392, 2008.

Amir Beck, Angelia Nedić, Asuman Ozdaglar, and Marc Teboulle. An $o(1/k)$ gradient method for network resource allocation problems. *IEEE Transactions on Control of Network Systems*, 1(1): 64–73, 2014.

Xuanyu Cao and Tamer Başar. Decentralized online convex optimization with feedback delays. *IEEE Transactions on Automatic Control*, 67(6):2889–2904, 2021.

Sapana Chaudhary and Dileep Kalathil. Safe online convex optimization with unknown linear safety constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6175–6182, 2022.

Tianyi Chen, Qing Ling, and Georgios B Giannakis. An online convex optimization approach to proactive network resource allocation. *IEEE Transactions on Signal Processing*, 65(24):6350–6364, 2017.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *Conference on Learning Theory*, pages 6–1. JMLR Workshop and Conference Proceedings, 2012.

Mung Chiang and Jason Bell. Balancing supply and demand of bandwidth in wireless cellular networks: utility maximization over powers and rates. In *IEEE INFOCOM 2004*, volume 4, pages 2800–2811. IEEE, 2004.

Alessandro Falsone, Kostas Margellos, Simone Garatti, and Maria Prandini. Dual decomposition for multi-agent distributed optimization with coupling constraints. *Automatica*, 84:149–158, 2017.

Masoud Farivar and Steven H Low. Branch flow model: Relaxations and convexification—part i. *IEEE Transactions on Power Systems*, 28(3):2554–2564, 2013.

Xiand Gao, Xiaobo Li, and Shuzhong Zhang. Online learning with non-convex losses and non-stationary regret. In *International Conference on Artificial Intelligence and Statistics*, pages 235–243. PMLR, 2018.

Hengquan Guo, Xin Liu, Honghao Wei, and Lei Ying. Online convex optimization with hard constraints: Towards the best of two worlds and beyond. *Advances in Neural Information Processing Systems*, 35:36426–36439, 2022.

Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Spencer Hutchinson, Berkay Turan, and Mahnoosh Alizadeh. The impact of the geometric properties of the constraint set in safe optimization with bandit feedback. In *Learning for Dynamics and Control Conference*, pages 497–508. PMLR, 2023.

Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. Online optimization: Competing with dynamic comparators. In *Artificial Intelligence and Statistics*, pages 398–406. PMLR, 2015.

Frank P Kelly, Aman K Maulloo, and David Kim Hong Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research society*, 49(3):237–252, 1998.

Xiuxian Li, Xinlei Yi, and Lihua Xie. Distributed online optimization for multi-agent networks with coupled inequality constraints. *IEEE Transactions on Automatic Control*, 66(8):3575–3591, 2020.

Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research*, 13(1): 2503–2528, 2012.

Negar Mehr, Jennie Lioris, Roberto Horowitz, and Ramtin Pedarsani. Joint perimeter and signal control of urban traffic via network utility maximization. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2017.

Jeonghoon Mo and Jean Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on networking*, 8(5):556–567, 2000.

Daniel K Molzahn, Florian Dörfler, Henrik Sandberg, Steven H Low, Sambuddha Chakrabarti, Ross Baldick, and Javad Lavaei. A survey of distributed optimization and control algorithms for electric power systems. *IEEE Transactions on Smart Grid*, 8(6):2941–2962, 2017.

Ion Necoara and Valentin Nedelcu. Rate analysis of inexact dual first-order methods application to dual decomposition. *IEEE Transactions on Automatic Control*, 59(5):1232–1243, 2013.

Ion Necoara and Valentin Nedelcu. On linear convergence of a distributed dual gradient algorithm for linearly constrained separable convex problems. *Automatica*, 55:209–216, 2015.

Angelia Nedić and Asuman Ozdaglar. Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM Journal on Optimization*, 19(4):1757–1780, 2009.

Ivano Notarnicola and Giuseppe Notarstefano. Constraint-coupled distributed optimization: a relaxation and duality approach. *IEEE Transactions on Control of Network Systems*, 7(1):483–492, 2019.

Daniel Pérez Palomar and Mung Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications*, 24(8):1439–1451, 2006.

Pedram Samadi, Amir-Hamed Mohsenian-Rad, Robert Schober, Vincent WS Wong, and Juri Jatskevich. Optimal real-time pricing algorithm based on utility maximization for smart grid. In *2010 First IEEE International Conference on Smart Grid Communications*, pages 415–420. IEEE, 2010.

Rolf Schneider. *Convex bodies: the Brunn–Minkowski theory*. Number 151. Cambridge university press, 2014.

Andrea Simonetto and Hadi Jamali-Rad. Primal recovery from consensus-based dual decomposition for distributed convex optimization. *Journal of Optimization Theory and Applications*, 168:172–197, 2016.

Berkay Turan and Mahnoosh Alizadeh. Safe dual gradient method for network utility maximization problems. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 6953–6959. IEEE, 2022.

Berkay Turan, Spencer Hutchinson, and Mahnoosh Alizadeh. Safe dynamic pricing for nonstationary resource allocation problems, 2023a. URL https://drive.google.com/file/d/1_Gn3qdUKFvYllbndJ_s8C6s1ao4jm4Wb/view.

Berkay Turan, Spencer Hutchinson, and Mahnoosh Alizadeh. A safe first-order method for pricing-based resource allocation in safety-critical networks. *arXiv preprint arXiv:2310.03808*, 2023b.

John S Vardakas, Nizar Zorba, and Christos V Verikoukis. A survey on demand response programs in smart grids: Pricing methods and optimization algorithms. *IEEE Communications Surveys & Tutorials*, 17(1):152–178, 2014.

Hao Yu and Michael J Neely. A low complexity algorithm with o $(\sqrt{T})$ regret and o (1) constraint violations for online convex optimization with long term constraints. *Journal of Machine Learning Research*, 21(1):1–24, 2020.

Deming Yuan, Daniel WC Ho, and Guo-Ping Jiang. An adaptive primal-dual subgradient algorithm for online distributed constrained optimization. *IEEE transactions on cybernetics*, 48(11):3045–3055, 2017.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.