LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-833325

A nonsmooth nonconvex optimization algorithm for two-stage optimization problems

J. Wang, C. G. Petra

March 30, 2022

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

A simplified nonsmooth nonconvex bundle method with applications to security-constrained ACOPF problems

Jingyi WANG*and Cosmin G. PETRA

Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
Livermore, California, USA

November 8, 2023

Table of contents

Abstract

An optimization algorithm for a group of nonsmooth nonconvex problems inspired by two-stage stochastic programming problems is proposed. The main challenges for these problems include (1) the problems lack the popular lower-type properties such as prox-regularity assumed in many nonsmooth nonconvex optimization algorithms, (2) the objective can not be analytically expressed and (3) the evaluation of function values and subgradients are computationally expensive. To address these challenges, this report first examines the properties that exist in many two-stage problems, specifically upper-C^2 objectives. Then, we show that quadratic penalty method for security-constrained alternating current optimal power flow (SCACOPF) contingency problems can make the contingency solution functions upper-C^2. Based on these observations, a simplified bundle algorithm that bears similarity to sequential quadratic programming (SQP) method is proposed. It is more efficient in implementation and computation compared to conventional bundle methods. Global convergence analysis of the algorithm is presented under novel and reasonable assumptions. The proposed algorithm

*wang125@llnl.gov

therefore fills the gap of theoretical convergence for smoothed SCACOPF problems. The inconsistency that might arise in our treatment of the constraints are addressed through a penalty algorithm whose convergence analysis is also provided. Finally, theoretical capabilities and numerical performance of the algorithm are demonstrated through numerical examples.

Keywords: Optimization; Nonsmooth; Nonconvex; Upper regularity;

1 Introduction

In this report, we consider the class of nonsmooth nonconvex constrained optimization problems in the form of

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + \mathcal{R}(x) \\ \text{subject to} \quad & c(x) = c_E \\ & d^l \leq d(x) \leq d^u \\ & x^l \leq x \leq x^u, \end{aligned} \tag{1}$$

where the functions $f(\cdot) : \mathbb{R}^n \to \mathbb{R}$, $c(\cdot) : \mathbb{R}^n \to \mathbb{R}^{m_c}$, $d(\cdot) : \mathbb{R}^n \to \mathbb{R}^{m_d}$ are continuously differentiable. The entries of the bound vectors d^l and d^u are in $\mathbb{R}$. The bounds on the optimization variables x are such that $x^l, x^u \in \mathbb{R}^n$, $x^l_j < x^u_j$, for all $j \in \{1, ..., n\}$. The function $\mathcal{R}(\cdot)$ is nonsmooth and nonconvex, as in a large number of important applications. In addition, the analytical form of $\mathcal{R}(\cdot)$ might not be available, forcing a potential algorithm to rely on known points in the optimization space. Prominent problems in the form of (1) include two-stage stochastic programming problems with recourse [4, 21, 49]. While general to apply to various paradigms of two-stage optimization under uncertainty (or other nonsmooth problems), the methodology presented in this report is driven by the problem of optimal operation of large-scale electrical transmission power grids.

1.1 Power grid optimization

Electricity generation and distribution in nationwide power grid systems rely upon optimization models and tools to find the power generation injection levels and transmission power flows at each of the grid nodes so that the demand at given substations is met at the lowest generation cost and minimum transmission losses [1]. Among them, alternating current optimal power flow (ACOPF) models have been proposed, researched, and adopted in some cases in operations because they model the power grid more accurately (*e.g.*, capture reactive power and include transmission losses) than the economic dispatch models. ACOPF models are becoming increasingly challenged with the penetration of (highly intermittent) renewable sources of energy (*e.g.*, wind and solar) and ongoing shifts in demand, which are caused by the emergence of commodity solar systems, battery storage, and electric vehicles [8, 31]. To better accommodate these emerging technologies, power grid operators need to operate increasingly complex power grid systems under highly stochastic

demand and generation profiles and frequent equipment failures.

SCACOPF is one of the salient emerging optimization paradigms for increasing the reliability of the power grid and ensuring its operation [38] under various types of failures. SCACOPF extends the capabilities of ACOPF by requiring that the state of the grid is secure with respect to a comprehensive list of equipment *contingencies* (*e.g.*, failures of generators, transmission lines, and transformers) and sometimes under stochastic demand and/or generation [18, 38]. As a result, the SCACOPF mathematical optimization problem reaches extreme scale as it needs to simulate multiple ACOPF models (routinely $O(10^5)$) in order to find a secure state of the grid. An equally important challenge is given by the highly nonlinear and nonconvex nature of SCACOPF (as well as of ACOPF) problems, which makes it difficult to find global (or at least good quality) optima of the problem. On the other hand, SCACOPF models need to be solved under strict time limitations, *i.e.*, in real-time, to allow ample adjustment time for the equipment (generation ramp up or down, load shedding, transmission switching, etc.). These challenges have sparked research over the last decades to study new scalable optimization algorithms and develop parallel computer implementations for SCACOPF problems.

Parallel computing has recently shown promising results for reaching real-time solutions for SCACOPF. In [9, 40, 41, 43], the SCACOPF problem is solved in parallel by decomposing the linear algebra of interior-point methods [35] using a Schur complement technique. Alternative parallel computing approaches, such as the optimization-based decompositions from [28] and [39], break down the SCACOPF problem at the level of the formulation into base case ACOPF and contingency response ACOPF subproblems and enforce the reconciliation between subproblems' coupling variables using first-order gradient-based methods [28] or carefully chosen approximations for the coupling terms [39]. Decomposed problem formulation however often generates a nonsmooth nonconvex $\mathcal{R}(\cdot)$, posing challenge to the design of an algorithm that converges theoretically and is computationally efficient.

While effective in practice, decomposed SCACOPF algorithms such as the smoothed two-stage solver in [39] has not been fully analyzed in theory. The existing nonsmooth nonconvex optimization literature does not apply directly to the problem. In this report, we aim to contribute to the theoretical analysis of the decomposition algorithms for SCACOPF problems, and more broadly two-stage optimization problems. In addition, we provide algorithm design and implementation details that could be valuable for large-scale nonsmooth nonconvex optimization problems.

1.2 Nonsmooth optimization

Nonsmooth optimization has been researched extensively for decades. Most prominent methods include subgradient methods and bundle methods. Subgradient method takes steps in the direction of a subgradient at a given point, relying heavily on a robust step size control algorithm to achieve good rates of convergence [50]. Bundle methods are widely regarded as one of the most efficient optimization methods to address discontinuous first-order derivatives [22, 33]. The bundle method develops an approximation model for the objective with the information from previous iterations, referred to as a bundle, and solves optimization subproblems with the model [23, 30]. The solution to the subproblem is regarded as a trial step, which through a rejection criterion is either taken as a serious step or rejected but included in the bundle to improve the trial step for the next iteration. In the case with convex objectives, the linearization error between the objective function and the tangent planes that comprise the approximation model is positive, a property that is not valid for nonconvex functions. Therefore, adjustment to the approximation model is needed. A commonly used one, called the down-shift mechanism, is introduced in [30] and used in [24, 26, 48, 56]. Convergence analysis using this mechanism can be found in [2, 36]. Given additional local convexity properties, *e.g.*, lower-C^2, the slope of the tangent planes can be titled as well to generate positive linearization errors [45]. These redistributed bundle methods are shown to work in practice under less ideal conditions [19]. Constrained nonsmooth nonconvex optimization adds another layer of complexity on top for bundle methods. Convex constraints can typically be maintained as they are in the subproblems and convergence analysis would stand valid [19]. In particular, affine constraints, commonly appearing in applications do not pose extra challenge [17] in convergence analysis.

In dealing with nonsmooth nonconvex objectives with general constraints, ideas from penalty and filter methods are often applied to incorporate the constraints into the objective in bundle methods. The global convergence studies in this case typically shows the algorithm can either converge to a KKT point of the original problem or to a stationary/critical point of the constraint violation. The latter case could lead to an infeasible solution. An exact penalty merit function that measures the progress of both the objective and inequality constraint violation is used in the redistributed bundle method in [55]. Lower-C^2 and a special strong Slater condition are assumed to ensure global convergence. In [16], a progress function that is the maximum of a penalized objective reduction and constraint violation is chosen. The bundle method is applied to the subproblem whose

objective is replaced by the progress function, eliminating the general constraint. Given lower-C^1 or upper-C^1 property, convergence is proved. Similar algorithm with direct assumptions on the penalty and quadratic parameters are presented in [34]. Others have chosen a different set of penalized objective functions [29].

Outside bundle methods, [14] proposed a sequential quadratic programming (SQP) method using gradient sampling for nonconvex nonsmooth inequality constrained optimization. As conventional in SQP, the constraint is relaxed through linearization while the iterations are taken at differentiable points of the Lipschitz objectives and constraints. The global convergence result of the algorithm shows that accumulation points are the stationary points of the exact penalty function, which can be equal to the constraint violation function depending on the penalty parameter. A more efficient BFGS-SQP is proposed in [13] which shows promising numerical behaviors without theoretical guarantee of convergence. In [54], a smoothing function of the objective that satisfies gradient consistency property is used together with augmented Lagrangian constraint relaxation. Extensive discussion of constraint qualifications are presented in order to achieve convergence. Line search of the Lagrangian function is possible due to the smoothing function which converges in the limit to the nonsmooth objective. Alternating direction method of multipliers (ADMM) has also been applied to nonsmooth nonconvex problems and in particular interest to us, distributed and asynchronous parallel algorithm [20]. In [53] the convergence analysis requires the objective to be locally prox-regular with affine constraints, similar to those in bundle method literature. One of the difficulties in applying ADMM to our application is the non-existence of the analytical form of part of the objective. Recently, difference-of-convex functions have been systematically studied in [12]. In particular interest to this paper, the recourse function of linearly bi-parameterized two-stage problems with quadratic recourse is shown to have convexity-concavity property [27]. The authors then proposed an iterative algorithm with a quadratic convex subproblem that converges subsequently to generalized critical points. The presence of a nonsmooth concave function from recourse function in the objective is novel and closely related to upper-C^2 property.

The report is organized as follows. In Section 2, we describe general two-stage stochastic programming problems with an emphasis on the SCACOPF problem. In particular, we discuss the upper-type properties of $\mathcal{R}(\cdot)$ that arise from such problems, which serve as the guild lines in designing the algorithm. We also propose quadratic penalty smoothing to enable a large group of problems to possess some upper-type property that they do not have otherwise. In Section 3, our simplified bundle algorithm is proposed and its

global convergence analysis is provided given novel assumptions drawn from Section 2. The algorithm can be seen as an extension of SQP to nonsmooth nonconvex problems. We also discuss the adjustable update rules for the approximations of second-stage optimal value functions and its application to two-stage stochastic programming problems. An algorithm to address possible inconsistency due to our treatment of the constraints in the subproblems is presented in Section 3.4, whose global convergence analysis is provided as well. Numerical experiments are shown in Section 4 that illustrate both the theoretical and numerical capabilities of the proposed algorithm. We note that the proposed algorithms can be parallelized efficiently for two-stage stochastic programming problems as shown in [52], which greatly improves computational efficiency since evaluation of $\mathcal{R}(\cdot)$ and its general subgradients can be the computationally expensive. This prepares the algorithm well for large-scale SCACOPF applications.

2 Problem description, preliminaries and regularization

Two-stage stochastic optimization problems with recourse fits within the formulation of (1). Using expectation as an example, the nonsmooth nonconvex function $\mathcal{R}(\cdot)$, also referred to as the expected *recourse* function [49], can be expressed as $\mathcal{R}(x) = \mathbb{E}_\Omega[r(x,\omega)]$, where $\mathbb{E}$ is the expectation operator. The function $r(x,\omega)$ is the optimal value function of the second-stage problem parameterized by x and the random variable ω over a probability space Ω. More specifically, $r(x,\omega)$ has the following mathematical form:

$$\begin{aligned} r(x,\omega) = \underset{y}{\text{minimize}} \quad & p(y,x,\omega) \\ \text{subject to} \quad & c(y,x,\omega) = c_E(\omega) \\ & d^l(\omega) \le d(y,x,\omega) \le d^u(\omega) \\ & y^l(\omega) \le y \le y^u(\omega). \end{aligned} \tag{2}$$

In (2), the functions $p(\cdot,\cdot,\omega) : \mathbb{R}^p \times \mathbb{R}^n \times \Omega \to \mathbb{R}$, $c(\cdot,\cdot,\omega) : \mathbb{R}^p \times \mathbb{R}^n \times \Omega \to \mathbb{R}^{m_c}$, $d(\cdot,\cdot,\omega) : \mathbb{R}^p \times \mathbb{R}^n \times \Omega \to \mathbb{R}^{m_d}$ are assumed to be smooth. The entries of the bound vector $d^l(\omega)$ and $d^u(\omega)$ are in $\mathbb{R}$ and the bounds on the optimization variables y is such that $y^l(\omega) \in \mathbb{R}^p$, $y^u(\omega) \in \mathbb{R}^p$ and $y^l_j(\omega) < y^u_j(\omega)$, for all $j \in \{1,\dots,p\}$ and $\omega \in \Omega$.

SCACOPF models can be established in the form of (1)–(2), where a secure state of the power grid is found that minimizes operation cost $f(\cdot)$ plus the average monetary penalties $p(\cdot,\cdot,\cdot)$ associated with not satisfying power demand and violating grid's power flows over all contingencies. Assuming uniform distribution, the sample space of ω consists of the

set of all possible K contingencies, each taken with equal probability $\frac{1}{K}$. The first-stage optimization variables x in (1) correspond to power generation and power flow levels that are to be implemented instantly in practice; while the second stage variables y are recourse actions to be implemented should a contingency ω occur. Thus, problem (1) simplified with discrete uniform probability distribution can be written as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + \frac{1}{K}\sum_{i=1}^{K} r_i(x) \\ \text{subject to} \quad & c(x) = c_E \\ & d^l \leq d(x) \leq d^u \\ & x^l \leq x \leq x^u, \end{aligned} \tag{3}$$

where recourse functions $r_i(\cdot) : \mathbb{R}^n \to \mathbb{R}$, for all $i \in 1, \ldots, K$, are the optimal solution functions to the deterministic second-stage optimization subproblems, namely,

$$\begin{aligned} r_i(x) = \underset{y_i}{\text{minimize}} \quad & p_i(x, y_i) \\ \text{subject to} \quad & c(x, y_i) = c_E \\ & d_i^l \leq d(x, y_i) \leq d_i^u \\ & y_i^l \leq y_i \leq y_i^u. \end{aligned} \tag{4}$$

When K is relatively small, the problem can be solved as a single-stage problem through off-the-shelf optimization packages. However, it is usually difficult to satisfy the requirement of real-time solution time. If the number of contingencies K is exceedingly large, which is common in real-world power grid operations, then solution through serial optimization solvers is intractable. One approach to tackle such large-scale problems is to use memory-distributed algorithms with the help of parallel computing [7]. The evaluation of $r_i(\cdot)$ at a given x is of considerable computational cost and can reach $O(10^2)$ seconds for real-world power grids. This characteristic requires the design of the optimization algorithm to avoid evaluating $r_i(\cdot)$ as much as possible.

2.1 Preliminaries and notations

As mentioned earlier, throughout this work, we assume functions $f(\cdot), c(\cdot), d(\cdot)$ in (1) are smooth, while functions $r_i(\cdot)$ are proper (A1, [47]) and locally Lipschitz. To simplify notation, we use $r(\cdot)$ to replace $r_i(\cdot)$ and let $K = 1$. A closed ball in $\mathbb{R}^n$ centered at $\bar{x} \in \mathbb{R}^n$ with radius $\rho > 0$ is denoted as $B_\rho(\bar{x})$. In nonsmooth nonconvex optimization

literature, both Clarke subgradient [11] and lower regular subgradient (8.3, [47]) have been widely adopted in analysis. While they both enjoy the outer/upper-semicontinuity property necessary in establishing convergence (6.6, [47]), Clarke subdifferential, denoted by $\bar{\partial} r(\bar{x})$, of function $r(\cdot)$ at $\bar{x}$ is used in this work. In addition, the less common upper regular/general subgradient offers a critical view in discussing the properties of interest.

The lower regular subdifferential of $r(\cdot)$ at point $\bar{x}$, denoted as $\hat{\partial} r(\bar{x})$, is defined by

$$\hat{\partial} r(\bar{x}) = \left\{ g \in \mathbb{R}^n | \liminf_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r(x) - r(\bar{x}) - \langle g, x - \bar{x} \rangle}{\|x - \bar{x}\|} \geq 0 \right\}, \tag{5}$$

where the 2-norm $\|\cdot\|$ is used and $\langle \cdot \rangle$ is the inner product in $\mathbb{R}^n$. The notion of *f-attentive* convergence, which is crucial for the concept of general subgradient, is defined as

$$x^\nu \underset{r}{\to} \bar{x} \quad \Leftrightarrow \quad x^\nu \to \bar{x} \quad \text{with} \quad r(x^\nu) \to r(\bar{x}), \tag{6}$$

where $\{x^\nu\}$ is a sequence of points. Given the assumption of Lipschitz $r(\cdot)$, this becomes trivial. If there exists a sequence $\{x^\nu\}$ such that $x^\nu \underset{r}{\to} \bar{x}$ and $g^\nu \in \hat{\partial} r(x^\nu)$ with $g^\nu \to g$, g is called a lower general subgradient of $r(\cdot)$ at $\bar{x}$, written as $g \in \partial r(\bar{x})$. If a Lipschitz function r is lower regular (or subdifferentially regular as in 7.25, [47]), then its lower general subdifferential is the same as its lower regular subdifferential (Corollary 8.11, [47]). Due to the popularity of lower-type properties in optimization, lower general subgradient is often simply called general subgradient (8.3, [47]).

Next, as introduced in [47] and detailed in [32], upper regular subdifferential is defined through

$$\hat{\partial}^+ r(\bar{x}) \coloneqq -\hat{\partial}(-r)(\bar{x}) = \left\{ g \in \mathbb{R}^n | \limsup_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r(x) - r(\bar{x}) - \langle g, x - \bar{x} \rangle}{\|x - \bar{x}\|} \leq 0 \right\}. \tag{7}$$

The general upper subdifferential at $\bar{x}$ is given as $\partial^+ r(\bar{x}) \coloneqq -\partial(-r)(\bar{x})$. A function $r(\cdot)$ is called upper regular if $-r(\cdot)$ is lower regular [47]. Some examples of upper regular functions include all continuous concave functions and all functions strictly differentiable. The upper/lower general subdifferential is locally bounded for Lipschitz continuous function (9.13, [47]). An important relationship between an upper regular subgradient and Clarke subdifferential is established next.

Lemma 2.1. *(lower subdifferential and Clarke subdifferential) Let $-r(\cdot)$ be a Lipschitz and lower regular function at $\bar{x}$, then its Clarke subdifferential and lower general subdifferential at $\bar{x}$ are equivalent, i.e. $\partial(-r)(\bar{x}) = \bar{\partial}(-r)(\bar{x})$.*

Proof. This Lemma is readily available from the results in [47]. By 8.49 and 9.13 in [47], for a Lipschitz continuous function $-r(\cdot)$, we have its Clarke subdifferential $\bar{\partial}(-r)(\bar{x}) = \text{con}\partial(-r)(\bar{x})$. For a lower regular function $-r(\cdot)$, $\hat{\partial}(-r)(\bar{x}) = \partial(-r)(\bar{x})$ (8.11, [47]). Given the convexity of lower regular subdifferential $\hat{\partial}(-r)(\bar{x})$ (8.6, [47]), $\bar{\partial}(-r)(\bar{x}) = \text{con}\hat{\partial}(-r)(\bar{x}) = \hat{\partial}(-r)(\bar{x}) = \partial(-r)(\bar{x})$. □

Lemma 2.2. *(upper subdifferential and Clarke subdifferential) Let $r(\cdot)$ be Lipschitz and upper regular at $\bar{x}$, then its upper general subdifferential and Clarke subdifferential at $\bar{x}$ are equivalent. In particular, if $g \in \partial^+ r(\bar{x})$, then $g \in \bar{\partial} r(\bar{x})$.*

Proof. We rely on the properties of lower regular functions in Lemma 2.1. Since $r(\cdot)$ is upper regular, by definition $-r(\cdot)$ is lower regular and $-\hat{\partial}(-r)(\bar{x}) = \hat{\partial}^+ r(\bar{x}) = \partial^+ r(\bar{x})$. By the symmetry property of the Clarke subgradient for locally Lipschitz functions (2.3.1, [11]), we have $\bar{\partial} r(\bar{x}) = -\bar{\partial}(-r)(\bar{x})$. By Lemma 2.1, $-\bar{\partial}(-r)(\bar{x}) = -\hat{\partial}(-r)(\bar{x})$. Therefore, $\partial^+ r(\bar{x}) = -\hat{\partial}(-r)(\bar{x}) = \bar{\partial} r(\bar{x})$. As a result, $g \in \partial^+ r(\bar{x})$ is also a Clarke subgradient, *i.e.* $g \in \bar{\partial} r(\bar{x})$. □

Due to their equivalence, for upper regular functions, Clarke subgradient and upper general subgradient can be used interchangeably. Moving on to more restrictive assumptions than regularity, lower-C^1 functions, introduced in [51] and [47], are commonly assumed in nonsmooth optimization and have a few equivalent definitions [15]. A function $r(\cdot) : O \to \mathbb{R}$, where $O \subset \mathbb{R}^n$ is open is said to be lower-C^k on O, if on some neighborhood V of each $\bar{x} \in O$ there is a representation

$$r(x) = \max_{t \in T} r_t(x), \tag{8}$$

where the functions $r_t(\cdot)$ are of class C^k on V and the index set T is a compact space such that $r_t(\cdot)$ and all of its partial derivatives through order k are jointly continuous on $(t, x) \in T \times V$. Similarly, a function is upper-C^k on O if on a neighborhood V of $\bar{x} \in O$ we can write

$$r(x) = \min_{t \in T} r_t(x), \tag{9}$$

where $r_t(\cdot)$ are of class C^k on V. The set T is compact and $r_t(\cdot)$ and all of the partial derivatives through order k are jointly continuous on $(t, x) \in T \times V$.

A widely used T is a closed and bounded subset of $\mathbb{R}^p$. Thus, if

$$r(x) = \min_{t \in T} p(t, x) \tag{10}$$

for all $x \in O$, and $p(\cdot,\cdot)$ and its first- and second-order partial derivatives in x depend continuously on (t,x), $r(\cdot)$ is upper-C^2. Rockafellar [46] and Clarke [10] further simplified the objective in (10) for Lipschitz functions. If $r(\cdot): U \to \mathbb{R}$, where $U \subset \mathbb{R}^n$ is an open, convex and bounded set, is Lipschitz, then it is upper-C^2 if there exists $\sigma > 0$, a compact set S and continuous functions $b(\cdot): S \to \mathbb{R}^n, c(\cdot): S \to \mathbb{R}$ such that

$$r(x) = \min_{s\in S}\{\sigma \|x\|^2 - \langle b(s), x\rangle - c(s)\} \tag{11}$$

for $\forall x \in U$.

While the original definition has clear indication for two-stage optimization problems, an alternative definition based on the function and subgradient value is more useful in analysis. A function is called lower-C^1 at $\bar{x}$ if

$$\begin{aligned} \forall \epsilon > 0,\ \exists\ \rho > 0, s.t.\ \forall x, x' \in B_\rho(\bar{x}),\ g \in \partial r(x), \\ r(x') - r(x) - \langle g, x' - x\rangle \geq -\epsilon \left\|x' - x\right\|. \end{aligned} \tag{12}$$

A function is lower-C^1 on an open set O if it is lower-C^1 for all $x \in O$. By definition, a function $r(\cdot)$ is upper-C^1 if $-r(\cdot)$ is lower-C^1 at $\bar{x}$.

An intuitive, equivalent definition of a finite-valued, lower-C^2 function $r(\cdot)$ on an open set O is that for any point $\bar{x} \in O$, there exists a threshold value $\rho_0 > 0$ such that $r(\cdot) + \frac{\rho}{2}\|\cdot\|^2$ is convex on an open neighborhood of $\bar{x}$ for all $\rho > \rho_0$. It is worth noting that another popular property: prox-regularity is closely related to lower-C^2. Lower-C^2 functions are prox-regular and for Lipschitz continuous functions, prox-regularity also guarantees lower-C^2 on an open set $O \subset \mathbb{R}^n$ (13.33, [47]).

Since a function $r(\cdot)$ is called upper-C^2 if and only if $-r(\cdot)$ is lower-C^2 at $\bar{x} \in \mathbb{R}^n$, for a finite, Lipschitz and upper-C^2 function $r(\cdot)$ on an open set O with $\bar{x} \in O$, there exists $\rho > 0$, such that

$$-r(x) + r(\bar{x}) - \langle -g, x - \bar{x}\rangle \geq -\frac{\rho}{2}\|x - \bar{x}\|^2, \tag{13}$$

where $-g \in \partial(-r)(\bar{x})$ and $x, \bar{x} \in O$. Since $-r(\cdot)$ is lower-C^2 and thus lower regular, $-g \in \hat{\partial}(-r)(\bar{x})$. By definition, $g \in \partial^+(r)(\bar{x})$ and by Lemma 2.2, $g \in \bar{\partial} r(\bar{x})$. Inequality (13) is equivalent to

$$r(x) - r(\bar{x}) - \langle g, x - \bar{x}\rangle \leq \frac{\rho}{2}\|x - \bar{x}\|^2, \tag{14}$$

where $g \in \partial^+ r(\bar{x})$ and $x, \bar{x} \in O$. Notice that there exists a uniform ρ such that (14) stands for all $x \in D \subset O$ and $g \in \partial^+ r(\bar{x})$, where D is compact (10.54, [47], [45]). We refer to (14) as the upper-C^2 inequality. It is worth pointing out that upper-C^2 does not guarantee

differentiability, lower-C^1 or lower regularity. It does ensure upper regularity where the upper subdifferential is equivalent to Clarke subdifferential. A simple example is

$$f(x) = \begin{cases} x, & -1 \leq x \leq 0, \\ \frac{1}{2}x, & 0 \leq x \leq 1, \end{cases} \tag{15}$$

which is concave and not differentiable or lower regular at $x = 0$.

A large number of nonsmooth nonconvex functions satisfy some of the properties described in this section. They allow the use of Clarke subgradient in the analysis of global convergence. To obtain desired properties for the objective function, regularization techniques might be necessary.

2.2 Smoothing of the second-stage problem

In many two-stage stochastic optimization problems, the second-stage solution function $r(\cdot)$ while lacking differentiability, satisfy the conditions for upper-C^2 property. For example, if the coupling of variables are in the smooth objective only, then by (10), $r(\cdot)$ is upper-C^2. In our target application, the first-stage variable x is coupled linearly in the constraints of the second-stage problems [39]. If the coupling exists in inequality however, upper-C^2 conditions might not be satisfied. Regularization of the problem could help smooth out the non-differentiability. To see this, we first make the following assumption for this section based on observation from SCACOPF problems.

Assumption 2.3. *The problem (4) can be reformulated with uncoupled objective and coupled constraints that are linear in the first-stage variable x.*

Using non-negative slack variables $s_i^l \geq 0, s_i^u \geq 0$, the coupled inequality constraints in (4) can be converted to equality constraints with

$$\begin{aligned} d(x, y_i) - d_i^l &= s_i^l \\ d_i^u - d(x, y_i) &= s_i^u \end{aligned} \tag{16}$$

For simplicity reasons, the subscript i for the ith second-stage problem is dropped. Moreover, the slack variables s_i^l, s_i^u can be regarded as part of the optimization variables y. Clearly the relevant equality constraints are linear in s by definition. Separating the coupled and uncoupled constraints, by Assumption 2.3, we denote the coupled constraints as

$$Wx - h(y) = 0, \tag{17}$$

where the slack variables are considered part of y and W is the corresponding linear operator. Using one of the standard matrix norms, W is assumed to be bounded with $\|W\| = w$. The second-stage problem from (4) now becomes

$$\begin{aligned} r(x) = \underset{y}{\text{minimize}} \quad & p(y) \\ \text{subject to} \quad & Wx - h(y) = 0, \\ & c(y) = c_{E,2} \\ & d^l \leq d(y) \leq d^u \\ & y^l \leq y \leq y^u, \end{aligned} \tag{18}$$

where $c_{E,2}$ is used to emphasize the uncoupled constraint. Given smooth $h(\cdot)$, $r(\cdot)$ still might not be differentiable or upper-C^2. However, it is possible now to apply the quadratic penalty method [35] to achieve upper-C^2. To illustrate both points, two simple examples are presented where differentiability can be improved. Example 1 is a one-dimensional optimization problem with equality constraint given in (19)

$$\begin{aligned} r(x) = \min_{y} \quad & y \\ \text{s.t.} \quad & y^2 = x \\ & y \geq 0, \end{aligned} \tag{19}$$

where $y \in \mathbb{R}$ and $x \geq 0$. It is obvious that the solution function is $r(x) = \sqrt{x}, x \geq 0$, which is continuous yet not Lipschitz continuous at $x = 0$. Using the quadratic penalty function with coefficient μ, the optimization problem is smoothed to (20)

$$\begin{aligned} r_\mu(x) = \min_{y} \quad & y + \mu \left\| y^2 - x \right\|^2 \\ \text{s.t.} \quad & y \geq 0. \end{aligned} \tag{20}$$

The smoothed solution function $r_\mu(\cdot)$ becomes Lipschitz continuous at $x = 0$, as illustrated on the left plot in Figure 1. The value of μ is a trade-off between accurate approximation of $r(\cdot)$ and the range of the transition period close to $x = 0$. Example 2 considers an optimization problem on $y \in \mathbb{R}$ with an inequality constraint

$$\begin{aligned} r(x) = \min_{y} \quad & ay^2 + by \\ \text{s.t.} \quad & y \geq x \\ & y \geq 0. \end{aligned} \tag{21}$$

The solution function $r(\cdot)$ is differentiable but not continuously differentiable at $x = 0$. With quadratic penalty and a slack variable s for the inequality constraint, the problem transforms to

$$\begin{aligned} r_\mu(x) = \min_y \quad & ay^2 + by + \mu \|x + s - y\|^2 \\ \text{s.t.} \quad & y, s \geq 0. \end{aligned} \tag{22}$$

The function $r_\mu(\cdot)$ is then smoothed into a continuously differentiable one as seen on the right in Figure 1. The smoothed function $r_\mu(\cdot)$ in both examples are now upper-C^2.

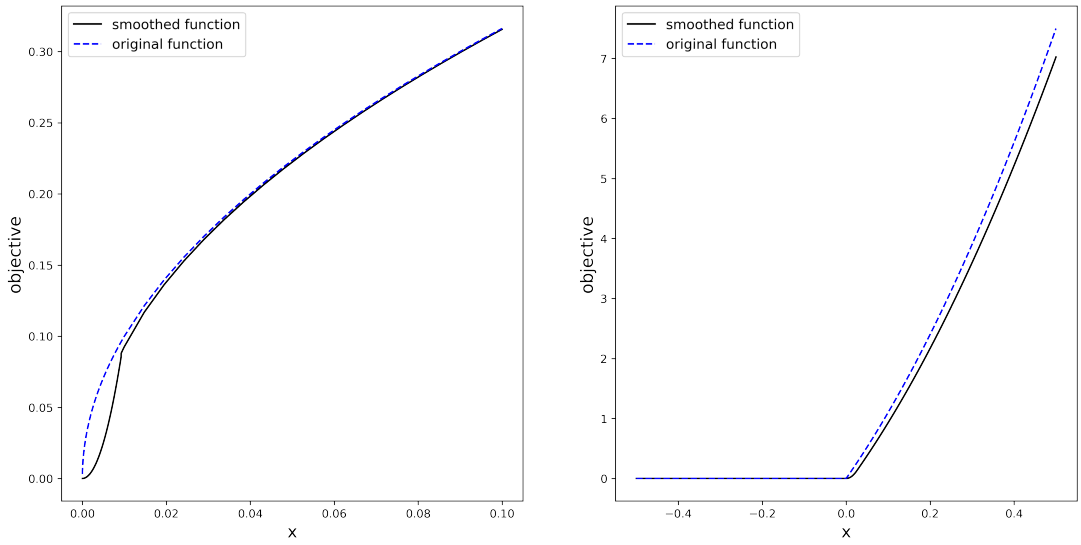

Figure 1: Quadratic penalty smoothing example: example 1 on the left, example 2 on the right

Similarly, for the more general case, it is possible to obtain desirable properties such as upper-C^2 for second-stage solution functions by incorporating the coupled constraints into the objective through a quadratic penalty such as

$$\begin{aligned} r_\mu(x) = \underset{y}{\text{minimize}} \quad & \mu \|Wx - h(y)\|^2 + p(y) \\ \text{subject to} \quad & c(y) = c_{E,2} \\ & d_l \leq d(y) \leq d_u \\ & y^l \leq y \leq y^u \end{aligned} \tag{23}$$

where μ is the penalty coefficient. As $\mu \to \infty$, the feasible accumulation points of the solutions to (23) become the solution to that of (4) [35]. It is worth pointing out that the coupling part of x is converted into a squared distance function, which has been studied extensively [42, 47]. While we focus on the linearly coupled constraint from SCACOPF,

as the goal is to pursue upper-C^2 of $r(\cdot)$, it is clear from its definition (10) that linearity is not necessary. In fact a smooth coupling in both x and y with quadratic penalty would suffice.

The properties of $r_\mu(\cdot)$ from (23) is examined next. The feasible set of y is denoted as $\Phi(x)$. An important fact that is repeatedly used is that $\Phi(x) \equiv \Phi$, independent of x due to the regularization. For $r_\mu(\cdot)$ to be continuous at x, the problem needs to have certain bounded properties. For example, it is common to assume coercivity [17] or level-bounded objective functions [16]. On the other hand, in our target applications, the variables x and y are bounded above and below by real, finite values. The slack variable s, defined through functions on x, y, are effectively bounded as well. Hence, we choose to assume bounded domain for x and compact domain for y, denoted as $X \in \mathbb{R}^n$ and $Y \in \mathbb{R}^p$, respectively for simplicity. Notice again Y is now independent of x. The optimal solution set is denoted as $S(x) \subset Y$ and the continuity result is given in Lemma 2.4 based on Chapter 4 from [5], the proof of which requires additional definitions and is left for the Appendix.

Lemma 2.4. *The optimal value function of the smoothed second-stage problem $r_\mu(\cdot)$ is continuous, and the multifunction $x \to S(x)$ is upper semicontinuous at x*

In addition, the compact domain and linear coupling of $r_\mu(\cdot)$ leads to the following Lemma.

Lemma 2.5. *The optimal solution function $r_\mu(\cdot)$ is Lipschitz continuous on its domain.*

Proof. Given that x is bounded, the domain of y is compact, and the continuous differentiability of the objective in (23), $r_\mu(x)$ is bounded. Let $M \in \mathbb{R} > 0$ be the upper bound of the absolute value of the coupled constraint, such that

$$\|Wx - h(y)\| \leq M, \quad \forall x \in X, y \in Y. \tag{24}$$

Denote by x_1, x_2 two points in the domain and $y_1 \in S(x_1), y_2 \in S(y_2)$ their corresponding optimal solutions. To simplify the notations, write $h_1 = h(y_1), h_2 = h(y_2), p_1 = p(y_1), p_2 = p(y_2)$. Since $y_1 \in S(x_1)$ and Y is independent of x, we have

$$\begin{aligned}
\mu \|Wx_1 - h_1\|^2 + p_1 &\leq \mu \|Wx_1 - h_2\|^2 + p_2 \\
&= \mu \|W(x_1 - x_2) + Wx_2 - h_2\|^2 + p_2 \\
&\leq \mu \|W(x_1 - x_2)\|^2 + 2\mu [W(x_1 - x_2)]^T (Wx_2 - h_2) \\
&\quad + \mu \|Wx_2 - h_2\|^2 + p_2.
\end{aligned} \tag{25}$$

Given $w = \|W\|$,

$$\begin{aligned}\mu \|Wx_1 - h_1\|^2 + p_1 - &\mu \|Wx_2 - h_2\|^2 - p_2 \\ &\leq \mu \|W(x_1 - x_2)\|^2 + 2\mu[W(x_1 - x_2)]^T[Wx_2 - h_2] \\ &\leq \mu \|W(x_1 - x_2)\| \left(\|W(x_1 - x_2)\| + 2\|Wx_2 - h_2\|\right) \\ &\leq \mu w \|x_1 - x_2\| \left(w\|x_1 - x_2\| + 2M\right).\end{aligned} \tag{26}$$

Similarly,

$$\begin{aligned}\mu \|Wx_2 - h_2\|^2 + p_2 - &\mu \|Wx_1 - h_1\|^2 - p_1 \\ &\leq \mu \|W(x_1 - x_2)\|^2 + 2\mu[W(x_2 - x_1)]^T[Wx_1 - h_1] \\ &\leq \mu \|W(x_1 - x_2)\| \,|\, \left(\|W(x_1 - x_2)\| + 2\|Wx_1 - h_1\|\right) \\ &\leq \mu w \|x_1 - x_2\| \left(w\|x_1 - x_2\| + 2M\right).\end{aligned} \tag{27}$$

Let $\left\|x^u - x^l\right\| = D, L = \mu w(wD + 2M)$, we have

$$\begin{aligned}|r_\mu(x_1) - r_\mu(x_2)| = &\left|\mu \|Wx_2 - h_2\|^2 + p_2 - \mu \|Wx_1 - h_1\|^2 - p_1\right| \\ &\leq \mu w \|x_1 - x_2\| \left(w\|x_1 - x_2\| + 2M\right) \\ &\leq \mu w \|x_1 - x_2\| \left(wD + 2M\right) \\ &\leq L \|x_1 - x_2\|.\end{aligned} \tag{28}$$

□

Next we show that an upper general subgradient of $r_\mu(\cdot)$ at $\bar{x}$ can be expressed as

$$g_\mu(\bar{x}) = 2\mu W^T(W\bar{x} - h(\bar{y})) \in \partial^+ r_\mu(\bar{x}). \tag{29}$$

Proposition 2.6. *The vector $g_\mu(\bar{x})$ in (29) is an upper general subgradient of $r_\mu(\cdot)$ in (23). In addition, the upper-C^2 inequality in (14) is satisfied with $g_\mu(\bar{x})$, i.e., for $x, \bar{x}$ in the domain, there exists $C > 0$ such that $r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(\bar{x})(x - \bar{x}) \leq C\|x - \bar{x}\|^2$.*

Proof. Let $p = p(y), \bar{p} = p(\bar{y}), h = h(y), \bar{h} = h(\bar{y})$, where $y \in S(x)$ and $\bar{y} \in S(\bar{x})$. The left-hand side of the inequality in (14) can be written as

$$\begin{aligned}r_\mu(x) - &r_\mu(\bar{x}) - g_\mu^T(\bar{x})(x - \bar{x}) \\ &= \mu(x^T W^T Wx - 2h^T Wx + h^T h) + p - \mu(\bar{x}^T W^T W\bar{x} - 2\bar{h}^T W\bar{x} + \bar{h}^T\bar{h}) - \bar{p} \\ &\quad - 2\mu(\bar{x}^T W^T Wx - \bar{x}^T W^T W\bar{x} - \bar{h}^T Wx + \bar{h}^T W\bar{x}) \\ &= \mu(\|W(x - \bar{x})\|^2 - 2h^T Wx + h^T h - \bar{h}^T\bar{h} + 2\bar{h}^T Wx) + p - \bar{p} \\ &= \mu(\|W(x - \bar{x})\|^2 + \|h - Wx\|^2 - \|Wx\|^2 - \left\|\bar{h} - Wx\right\|^2 + \|Wx\|^2) + p - \bar{p} \\ &= \mu\|W(x - \bar{x})\|^2 + \mu\|h - Wx\|^2 + p - \mu\left\|\bar{h} - Wx\right\|^2 - \bar{p}.\end{aligned} \tag{30}$$

Further, since $y \in S(x)$, we have

$$\begin{aligned} r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(\bar{x})(x-\bar{x}) \leq & \mu w^2 \|x-\bar{x}\|^2 + \mu \|h - Wx\|^2 + p - \mu \left\|\bar{h} - Wx\right\|^2 - \bar{p} \\ \leq & \mu w^2 \|x-\bar{x}\|^2 . \end{aligned} \tag{31}$$

Taking the limit so that $x \to \bar{x}$,

$$\limsup_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(\bar{x})(x-\bar{x})}{\|x-\bar{x}\|} \leq 0 \tag{32}$$

By definition (7), $g_\mu(\bar{x})$ is a upper regular subgradient, hence a upper general subgradient. Let $C = \mu w^2$, the upper-C^2 inequality is satisfied. □

The quadratic penalty smoothing of the second-stage problems allows the following important property to be achieved.

Proposition 2.7. *The second-stage optimal solution function $r_\mu(\cdot)$ is upper-C^2 and thus satisfies the upper-C^2 inequality in (14) on its domain.*

Proof. There are multiple ways to show this. The most straightforward one is to apply directly the definition (10). From Lemma 2.5, $r_\mu(\cdot)$ is Lipschitz continuous. From the definition of $r_\mu(\cdot)$ in (23), the feasible set of the optimization variables y is compact in $\mathbb{R}^m$ and independent of x. The coupling is now only in the objective, with the non-coupled part $p(\cdot)$ smooth in y. Further, the coupling in objective is quadratic in x, rendering it twice continuously differentiable in a neighborhood of both x and y. By definition leading to (10), $r_\mu(\cdot)$ is upper-C^2.

Intuitively, as pointed out in 10.57 of [47], squared distance function on a nonempty closed set is upper-C^2. The objective, while having an additional smooth function $p(\cdot)$, is only coupled in the squared distance function. From the viewpoint of (11), given a $x \in \mathbb{R}^n$, an open, convex and bounded neighborhood of x can be found where the objective that defines $r_\mu(\cdot)$ in (23) fits the form in (11). It is pointed out that such a neighborhood could contain infeasible points for the first-stage problem, while not affecting the property of the function $r_\mu(\cdot)$ itself. Therefore, $r_\mu(\cdot)$ is upper-C^2. The proposition then follows. □

Proposition 2.8. *If the solution set $S(x)$ is a singleton at $\bar{x}$ such that $S(\bar{x}) = \{\bar{y}\}$, then $r_\mu(\cdot)$ is differentiable at $\bar{x}$ and $g_\mu(\bar{x}) = 2\mu W^T(W\bar{x} - h(\bar{y}))$ is the gradient $\nabla r_\mu(\bar{x})$.*

Proof. Let us take $x, \bar{x} \in X$ and use shorthands $\bar{p} = p(\bar{y}), p = p(y), \bar{h} = h(\bar{y}), h = h(y)$. We can write

$$\begin{aligned}
&r_\mu(x) - r_\mu(\bar{x}) - \bar{g}_\mu^T(x - \bar{x}) \\
=&\left(\mu \|Wx - h\|^2 + p\right) - \left(\mu \left\|W\bar{x} - \bar{h}\right\|^2 + \bar{p}\right) - 2\mu(W\bar{x} - \bar{h})^T W(x - \bar{x}) \\
=&\mu(\|W(x - \bar{x})\|^2 - 2h^T Wx + h^T h - \bar{h}^T \bar{h} + 2\bar{h}^T Wx) + p - \bar{p} \\
=&\mu(\|W(x - \bar{x})\|^2 + 2h^T W\bar{x} - 2\bar{h}^T W\bar{x} - 2h^T Wx + 2\bar{h}^T Wx) \\
&+ \mu \|h - W\bar{x}\|^2 + p - \mu \left\|\bar{h} - W\bar{x}\right\|^2 - \bar{p} \\
=&\mu(\|W(x - \bar{x})\|^2 - 2\bar{h}^T W(\bar{x} - x) + 2h^T W(\bar{x} - x)) \\
&+ \mu \|h - W\bar{x}\|^2 + p - \mu \left\|\bar{h} - W\bar{x}\right\|^2 - \bar{p} \\
\geq&\mu(\|W(x - \bar{x})\|^2 - 2\bar{h}^T W(\bar{x} - x) + 2h^T W(\bar{x} - x)) \\
=&\mu(\|W(x - \bar{x})\|^2 - 2[h - \bar{h}]^T W(x - \bar{x}))
\end{aligned} \tag{33}$$

Since $\bar{h} = h(\bar{y})$ is unique at $\bar{x}$, $h \to \bar{h}$ as $x \to \bar{x}$ and

$$\begin{aligned}
\liminf_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(x - \bar{x})}{\|x - \bar{x}\|} &\geq \lim_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{\mu(-2w \left\|h - \bar{h}\right\| \|x - \bar{x}\|)}{\|x - \bar{x}\|} \\
&= \lim_{\substack{x \to x \\ \bar{x} \neq \bar{x}}} -2\mu w \left\|h - \bar{h}\right\| = 0.
\end{aligned} \tag{34}$$

In addition, from the proof of Proposition 2.6, we know

$$\limsup_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(x - \bar{x})}{\|x - \bar{x}\|} \leq 0 \tag{35}$$

Therefore,

$$\lim_{\substack{x \to \bar{x} \\ x \neq \bar{x}}} \frac{r_\mu(x) - r_\mu(\bar{x}) - g_\mu^T(x - \bar{x})}{\|x - \bar{x}\|} = 0, \tag{36}$$

and the proposition follows. □

On the other hand, if the optimal solution $\bar{h}$ is not unique, there could exist multiple upper regular subgradients and $r_\mu(\cdot)$ might not be differentiable. Indeed, uniqueness of the solution $y \in S(x)$ is not guaranteed. It is possible to put more restrictions on $h(\cdot)$ so that $r_\mu(\cdot)$ becomes continuously differentiable.

Proposition 2.9. *The optimal solution function $r_\mu(\cdot)$ is lower-C^2 on a neighborhood O of $\bar{x}$, if $h(y), y \in S(x)$, is Lipschitz continuous on O, i.e., $\forall x_1, x_2 \in O$, there exists $L_h > 0$ such that $\|h(y_1) - h(y_2)\| < L_h \|x_1 - x_2\|$ on O. Moreover, $r_\mu(\cdot)$ is continuously differentiable at $\bar{x}$.*

To summarize, while in many nonsmooth nonconvex optimization methods, lower-C^2 or prox-regularity are assumed, for some important applications such as the decomposed formulation of SCACOPF, lower regularity of the objective is not available. The upper-type properties however appears natural for two-stage problems, and has motivated us to make assumptions differently than the conventional ones and design algorithms accordingly.

There are multiple convergence definitions in nonsmooth nonconvex analysis, *e.g.*, stationary point, Karush–Kuhn–Tucker (KKT) point, (Fritz-John) critical point, etc. In this paper, the focus is on first-order optimality condition with Clarke subgradient. Without losing generality, problem (3) can be recast for simplicity as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & r(x) \\ \text{subject to} \quad & c(x) = 0 \\ & 0 \le x \le x_u, \end{aligned} \tag{37}$$

where $c(x) : \mathbb{R}^n \to \mathbb{R}^m$. As mentioned earlier in this section, transforming (3) requires the new variables x in (37) to contain slack variables, which are implicitly bounded from bound constraints on x. We opt to keep the upper bound x_u in the bound constraints explicit, as in (3) to emphasize that the feasible set of x is bounded . We point out that the bound constraints form a convex set, with the nonconvexity left to the equality constraints.

Problem (37) is assumed to be calm (6.4, [11]) at its local minimum. Calmness can be viewed as a weak constraint qualification (6.4, [11]). In particular, the widely adopted linear independence constraint qualification (LICQ) (17.2, [35]) in smooth optimization ensures calmness. Calmness guarantees the Lagrange multiplier for the objective function in Fritz-John critical point equation [11] is nonzero(6.4.4, [11]). Therefore, we can use a KKT point instead of a Fritz-John critical point in the optimality condition [11]. For problem (37), a first-order optimality condition at a local minimum $\bar{x}$ is that there exists $\bar{\lambda} \in \mathbb{R}^m$ and $\bar{\zeta}_l \ge 0$, $\bar{\zeta}_l \in \mathbb{R}^n$, $\bar{\zeta}_u \ge 0$, $\bar{\zeta}_u \in \mathbb{R}^n$ such that

$$\begin{aligned} 0 \in \bar{\partial} r(\bar{x}) + \nabla c(\bar{x})\bar{\lambda} - \bar{\zeta}_l + \bar{\zeta}_u, \\ \bar{Z}_u(\bar{x} - x_u) = 0, \bar{Z}_l\bar{x} = 0, \\ c_j(\bar{x}) = 0, j = 1, \dots, m, \\ \bar{\lambda}_j c_j(\bar{x}) = 0, j = 1, \dots, m, \\ \bar{\zeta}_l, \bar{\zeta}_u, x_u - \bar{x}, \bar{x} \ge 0. \end{aligned} \tag{38}$$

The matrix $\nabla c(\bar{x})$ is of dimension $n \times m$. The matrices $\bar{Z}_u, \bar{Z}_l$ are diagonal matrices whose diagonal values are $\bar{\zeta}_u$ and $\bar{\zeta}_l$, respectively. A point that satisfies (38) is called a KKT point

of problem (37). For upper regular functions, it is possible to establish a stronger form of optimality condition, especially the first condition in (38) as explained in [32].

3 Simplified bundle algorithm

Given the nonsmooth nonconvex nature of problem (37), we consider the bundle methods which have proven to be one of the most successful methods in solving such problems [25]. Bundle methods utilize information generated through previous iterative steps to form an approximation of the objective. Typically such an approximation model is a supportive one that produces smaller function value than the real function. Meanwhile, many such algorithms rely on the quadratic coefficient in the approximation to avoid line search [37, 45]. Another feature is the existence of a robust rejection mechanism to ensure the approximation is reasonable, similar to trust-region methods [35]. A solution to an iterative subproblem generates a trial step that is either accepted or rejected. A trial point is called a serious point if it is accepted. Convergence analysis for bundle methods typically require the objective to have properties such as lower-C^2 and lower-C^1. For large-scale problems such as SCACOPF problems, a clear drawback is that the complex update rule for the bundle and the large number of bundle points needed in the approximation could increase computing time considerably.

The proposed algorithm simplifies the bundle method while retain many of its features. Motivated by the properties exhibited from two-stage stochastic optimization problems discussed in Section 2, we make the assumption that the objective $r(\cdot)$ is upper-C^2, formalized below.

Assumption 3.1. *The Lipschitz continuous objective function $r(\cdot)$ in problem (37) is upper-C^2.*

In particular, the inequality (14) is satisfied and since x is bounded, there exists a ρ for the entire domain. In general, upper regularity, which is closely related to concavity, is less explored for optimization problems. We point out that [16, 36] have studied bundle methods and to our knowledge were the first to prove convergence for upper-C^1 objective and constraints. However, in that case the parameters of the approximation model are not guaranteed to be finite, besides the aforementioned challenges in applying bundle method. To take full advantage of the smooth constraints $c(\cdot)$, we assume uniform boundedness on their Hessian, a common assumption in literature [14].

Assumption 3.2. *The constraints $c(\cdot)$ are twice differentiable. There exists a constant H_u^c such that the Hessian of constraints $c(\cdot)$ satisfy $\frac{1}{2}x^T\nabla^2 c_j(x)x \leq H_u^c \|x\|^2$ for any $x \in \mathbb{R}^n$ and $1 \leq j \leq m$.*

3.1 Algorithm description

The simplified bundle algorithm is an iterative method with approximated objective at each iteration. It bears similarity to SQP methods in the treatment of constraints and can be viewed as its extension. Compared to conventional bundle methods, the convex quadratic approximation $\phi_k(\cdot)$ to the objective $r(\cdot)$ in (37) is dependent only on the current serious point instead of a bundle of points. More specifically, at iteration k and its serious step x_k, the local approximation model $\phi_k(\cdot)$ is

$$\phi_k(x) = r(x_k) + g_k^T(x - x_k) + \frac{1}{2}\alpha_k \|x - x_k\|^2, \tag{39}$$

where $g_k \in \bar{\partial} r(x_k)$, and $\alpha_k > 0$ is a scalar quadratic coefficient. Equivalently, denoting $d = x - x_k$, $\phi_k(x)$ can be reformulated as $\Phi_k(d)$ such that

$$\Phi_k(d) = r_k + g_k^T d + \frac{1}{2}\alpha_k \|d\|^2, \tag{40}$$

where $r_k = r(x_k)$. The function value and subgradient at x_k are exact, *i.e.*, $\Phi_k(0) = r_k, \nabla\Phi_k(0) = g_k$. Furthermore, the smooth constraints in (37) are linearized. The subproblem to be solved at iteration k is

$$\begin{aligned} \underset{d}{\text{minimize}} \quad & \Phi_k(d) \\ \text{subject to} \quad & c(x_k) + \nabla c(x_k)^T d = 0, \\ & d_l^k \leq d \leq d_u^k, \end{aligned} \tag{41}$$

where $d_l^k = -x_k, d_u^k = x_u - x_k$. As in SQP algorithms, it is possible that the linearized constraints cause the problem (41) to be infeasible. There are multiple ways to address this issue, one of which will be presented in Section 3.4. In this section, we operate under the assumption that (41) can be solved and its solution is denoted as d_k. To measure progress in both the objective and the constraints, the l_1 merit function is adopted:

$$\phi_{1\theta_k}(x) = r(x) + \theta_k \|c(x)\|_1, \tag{42}$$

where $\|\cdot\|_1$ is the 1-norm and $\theta_k > 0$ is a penalty parameter. A line search step on the constraints is needed in order to ensure progress in the merit function (42). The predicted

change on the objective is defined as

$$\delta_k = \Phi_k(0) - \Phi_k(d_k) = -g_k^T d_k - \frac{1}{2}\alpha_k \|d_k\|^2 . \tag{43}$$

To measure whether the approximation model $\Phi_k(\cdot)$ of the objective formed at x_k is still valid at the trial step $x_k + d_k$, we define ratio ρ_k as

$$\rho_k = \begin{cases} r(x_k) - r(x_k + d_k) - \eta_l^+ \delta_k, & \delta_k \geq 0, \\ r(x_k) - r(x_k + d_k) - \eta_l^- \delta_k, & \delta_k < 0, \end{cases} \tag{44}$$

where $0 < \eta_l^+ \leq 1$ and $\eta_l^- \geq 1$ are two parameters of the algorithm. If $\rho_k > 0$, the model is valid and the algorithm proceeds to line search. Otherwise, the trial step $x_k + d_k$ is rejected and the parameter α_k is updated to find a different trial step. This process draws inspiration from trust-region methods.

The change in the model objective δ_k is not necessarily positive. Therefore, the corresponding threshold η_l^+ and η_l^- differ based on the sign of δ_k. In both cases, the actual change in the objective $r(x_k) - r(x_k + d_k)$ is allowed to be slightly worse than the predicted change. This means that if δ_k is non-negative, the actual decrease can be smaller than the predicted decrease δ_k, though a fraction η_l^+ of δ_k is required. If δ_k is negative, the actual increase in objective value can be slightly larger than the predicted increase value $-\delta_k$, the degree to which is governed by $\eta_l^- \geq 1$.

Let the line search parameter be $\beta_k \in (0, 1]$. Then, the serious step taken is given as $x_{k+1} = x_k + \beta_k d_k$. Let $\delta_k^\beta = \Phi_k(0) - \Phi_k(\beta_k d_k)$, we have

$$\delta_k^\beta = \Phi_k(0) - \Phi_k(\beta_k d_k) = -\beta_k g_k^T d_k - \frac{1}{2}\beta_k^2 \alpha_k \|d_k\|^2 . \tag{45}$$

Similar to ρ_k, the ratio between predicted and actual change in objective at x_{k+1} is denoted as ρ_k^β, whose definition is

$$\rho_k^\beta = \begin{cases} r(x_k) - r(x_{k+1}) - \eta_\gamma^+ \delta_k^\beta, & \delta_k^\beta \geq 0, \\ r(x_k) - r(x_{k+1}) - \eta_\gamma^- \delta_k^\beta, & \delta_k^\beta < 0. \end{cases} \tag{46}$$

The parameter η_γ^+ and η_γ^- can have different values than η_l^+ and η_l^- to increase flexibility

of the algorithm. The first-order optimality conditions of the subproblem (41) are

$$\begin{aligned} g_k + \alpha_k d_k - \nabla c(x_k)\lambda^{k+1} - \zeta_l^{k+1} + \zeta_u^{k+1} = 0, \\ Z_u^{k+1}(d_k - d_u^k) = 0, Z_l^{k+1}(d_k - d_l^k) = 0, \\ \Lambda^{k+1}\left[c(x_k) + \nabla c(x_k)^T d_k\right] = 0, \\ \zeta_u^{k+1}, \zeta_l^{k+1}, d_k - d_l^k, d_u^k - d_k \geq 0, \\ c(x_k) + \nabla c(x_k)^T d_k = 0, \end{aligned} \tag{47}$$

where $\lambda^{k+1} \in \mathbb{R}^m$ is the Lagrange multiplier for $c(\cdot)$, and $\zeta_u^{k+1}, \zeta_l^{k+1} \in \mathbb{R}^n$ are the Lagrange multipliers for the bound constraints. The matrices $\Lambda^{k+1}, Z_u^{k+1}, Z_l^{k+1}$ are diagonal matrices whose diagonal values are $\lambda^{k+1}, \zeta_u^{k+1}$ and ζ_l^{k+1}, respectively. An equivalent form of the complementarity conditions of bound constraints based on x_u instead of d_l^k, d_u^k are

$$\begin{aligned} Z_u^{k+1}(x_k + d_k - x_u) = 0, Z_l^{k+1}(x_k + d_k) = 0, \\ \zeta_u^{k+1}, \zeta_l^{k+1}, x_k + d_k, x_u - x_k - d_k \geq 0. \end{aligned} \tag{48}$$

The line search conditions are given as follows

$$\begin{aligned} \theta_k \left\|c(x_k)\right\|_1 + \beta_k(\lambda^{k+1})^T c(x_k) \geq \theta_k \left\|c(x_{k+1})\right\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \left\|d_k\right\|^2, \\ \theta_k \left\|c(x_k)\right\|_1 + \eta_\gamma^+\beta_k(\lambda^{k+1})^T c(x_k) \geq \theta_k \left\|c(x_{k+1})\right\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \left\|d_k\right\|^2, \\ \theta_k \left\|c(x_k)\right\|_1 + \eta_\gamma^-\beta_k(\lambda^{k+1})^T c(x_k) \geq \theta_k \left\|c(x_{k+1})\right\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \left\|d_k\right\|^2. \end{aligned} \tag{49}$$

The differences between the conditions are the parameters η_γ^+ and η_γ^- in the second and third inequalities, which stem from the unknown sign of δ_k and δ_k^β. For simplicity in implementation and analysis, we use the following alternative condition for line search that encompasses all three

$$\theta_k \left\|c(x_k)\right\|_1 - \eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right| \geq \theta_k \left\|c(x_{k+1})\right\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \left\|d_k\right\|^2. \tag{50}$$

We will show that condition (50) implies conditions in (49) in Lemma 3.6. The simplified bundle method is presented in Algorithm 1, where $\|\cdot\|_\infty$ is the infinity norms. The items involving consistency restoration such as π_{k-1} are explained in Section 3.4.

3.2 Convergence analysis

If the algorithm terminates in a finite number of steps, stopping test in step 4, which can be modified if needed, is satisfied with the error tolerance ϵ and the solution is considered

Algorithm 1: Simplified bundle method

1 Initialize x_0, α_0, stopping error tolerance ϵ, and $k = 1$. Choose scalars $0 < \eta_l^+ \leq 1$, $0 < \eta_\beta < \eta_\gamma^+ \leq 1$, $\eta_l^- \geq 1$,$\eta_\gamma^- \geq 1$, $\eta_\alpha > 1$ and $\gamma > 0$. Evaluate the function value $r(x_0)$ and subgradient $g(x_0)$.

2 **for** $k = 0, 1, 2, ...$ **do**

3 Form the quadratic function Φ_k in (40) and solve subproblem (41) to obtain d_k and Lagrange multiplier λ^{k+1}. (If inconsistent constraints are encountered, enter consistency restoration and go back to step 2 with $k = k + 1$.)

4 **if** $\|d_k\| \leq \epsilon$ **then**

5 Stop the iteration and exit the algorithm.

6 Evaluate function value $r(x_k + d_k)$. Compute δ_k in (43) and ρ_k in (44).

7 Set the merit function parameter θ_k so that $\theta_k = \max\{\theta_{k-1}, \eta_\gamma^- \left\|\lambda^{k+1}\right\|_\infty + \gamma\}$. If feasibility restoration is called for iteration $k - 1$, let $\theta_k = \max\{\frac{1}{\pi_{k-1}}, \eta_\gamma^- \left\|\lambda^{k+1}\right\|_\infty + \gamma\}$.

8 **if** $\rho_k > 0$ **then**

9 Find the line search parameter $\beta_k > 0$ using backtracking, starting at $\beta_k = 1$ and halving if too large, such that the conditions in (50) are satisfied. Evaluate $r(x_{k+1})$ and compute ρ_k^β in (46).

10 **if** $\rho_k^\beta < 0$ **then**

11 Break and go to line 14.

12 Take the step $x_{k+1} = x_k + \beta_k d_k$.

13 **else**

14 Reject the trial step.

15 Call the chosen α_k update rules to obtain $\alpha_{k+1} = \eta_\alpha \alpha_k$.

found. Let $\epsilon = 0$, based on step 4, $\|d_k\| = 0$. As d_k solves (41), optimality conditions in (47) are satisfied, of which the first equation reduces to

$$g_k - \nabla c(x_k)\lambda^{k+1} - \zeta_l^{k+1} + \zeta_u^{k+1} = 0. \tag{51}$$

Given $g_k \in \bar{\partial} r(x_k)$, we have $0 \in \bar{\partial} r(x_k) - \nabla c(x_k)\lambda^{k+1} - \zeta_l^{k+1} + \zeta_u^{k+1}$. In addition, by $c(x_k) + \nabla c(x_k)^T d_k = 0$ from (47), we have $c(x_k) = 0$. So x_k is feasible in terms of the equality constraints. Together with the bound constraints that are enforced in the subproblem (41), the rest of the equations in (38) are also satisfied. Therefore, x_k satisfies (38) and is a KKT point for (37) as the algorithm exits. In what follows, the convergence analysis is carried out for the case with an infinite number of steps, *i.e.*, $\|d_k\| > 0$. We start by showing that the parameter α_k in Algorithm 1 eventually stabilizes, *i.e.*, becomes constant.

Lemma 3.3. *Given the assumption (3.1), Algorithm 1 produces a finite number of rejected steps. As a consequence, the quadratic coefficient α_k is bounded above and remains constant for k large enough.*

Proof. From the upper-C^2 property (14), we have

$$r(x_k + d) - r_k - g_k^T d \leq C \|d\|^2 \tag{52}$$

for a fixed constant $C > 0$. In the first part of the proof we show that if at some iteration k, α_k satisfies

$$\alpha_k > 2C, \tag{53}$$

then no rejected steps can occur in Algorithm 1 after iteration k. This means steps 8 and 10 of the algorithm $\rho_t > 0$ and $\rho_t^\beta > 0$ will hold for all iterations $t \geq k$. The inequalities (52) and (53) imply

$$\begin{aligned} r_k - r(x_k + d_k) \geq & - g_k^T d_k - C \|d_k\|^2 \\ > & - g_k^T d_k - \frac{1}{2}\alpha_k \|d_k\|^2 \\ = & \Phi_k(0) - \Phi_k(d_k). \end{aligned} \tag{54}$$

As in the definition (44) of ρ_k, we distinguish between two cases based on the sign of δ_k. If $\delta_k = \Phi_k(0) - \Phi_k(d_k) \geq 0$, then since $0 < \eta_l^+ \leq 1$, (54) gives

$$\begin{aligned} r_k - r(x_k + d_k) > & \Phi_k(0) - \Phi_k(d_k) \\ \geq & \eta_l^+ \left[\Phi_k(0) - \Phi_k(d_k)\right]. \end{aligned} \tag{55}$$

If $\delta_k = \Phi_k(0) - \Phi_k(d_k) < 0$, given $\eta_l^- \geq 1$, we can also write based on (54) that

$$\begin{aligned} r_k - r(x_k + d_k) >& \Phi_k(0) - \Phi_k(d_k), \\ \geq& \eta_l^- \left[\Phi_k(0) - \Phi_k(d_k)\right]. \end{aligned} \tag{56}$$

As a consequence, by definition (44), we conclude $\rho_k > 0$. Similar inequalities hold for $x_{k+1} = x_k + \beta_k d_k$ since one can write based on (52) that

$$\begin{aligned} r(x_k) - r(x_{k+1}) \geq& -\beta_k g_k^T d_k - C\beta_k^2 \|d_k\|^2 > -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 \\ =& \Phi_k(0) - \Phi_k(\beta_k d_k). \end{aligned} \tag{57}$$

Same steps that lead to (55) and (56) for $\eta_\gamma^+, \eta_\gamma^-$ imply $\rho_k^\beta > 0$. Therefore, for $t \geq k$, $\rho_t > 0$ and $\rho_t^\beta > 0$ and thus $\alpha_t = \alpha_k$. Equivalently, no rejected steps occur once (53) holds. Since α_k is increased monotonically with a ratio $\eta_\alpha > 1$ whenever a rejected step is encountered, only a finite number of rejected steps are needed to reach $\alpha_k > 2C$, which are followed by serious steps.

For the second part of the proof, suppose now $\alpha_k \leq 2C$ for all k. Then, only no or a small number of rejected steps can be taken by the algorithm. The monotonically increasing α_k ensures that there exists k such that $\alpha_t = \alpha_k \leq 2C$ for all $t \geq k$. This completes the proof. □

Remark 3.4. *For simplicity, we choose to increase α_k monotonically in the algorithm. In practice, we encourage that α_k be reduced if $\rho_k > 0$ and $\eta_l^+ > \eta_u^+$ where η_u^+ is an upper threshold for the parameter η_l^+. In other words, if the actual decrease in objective is bigger than a certain ratio of the predicted decrease, then $\Phi_k(\cdot)$ is a good approximation and we reduce the quadratic coefficient to encourage larger step size. From the convergence analysis point of view, the upper-C^2 constant C is not uniform in the entire domain. A decrease in α_k allows the algorithm to adjust better to the local upper-C^2 constant that could be relatively small compared to C, which could result in improved convergence in practice.*

Lemma 3.5. *Given Assumption 3.2, the line search process of Algorithm 1 is well-defined, in that a $\beta_k \in (0, 1]$ that satisfies the line search conditions in (50) exists and can be found in a finite number of steps through backtracking step 9 as long as the Lagrange multipliers λ^{k+1} from (41) remain finite.*

Proof. If λ^{k+1} remains finite throughout the algorithm, then a finite θ_k is guaranteed as well based on how it is chosen in Algorithm 1 step 7 as it stops increasing for k large

enough. Since $c(\cdot)$ is smooth, we apply Taylor expansion to the jth equality constraint, $j = 1, ..., m$, at x_k for $x_{k+1} = x_k + \beta_k d_k$ to obtain

$$c_j(x_{k+1}) = c_j(x_k) + \beta_k \nabla c_j(x_k)^T d_k + \frac{1}{2}\beta_k^2 d_k^T H_{k\beta}^j d_k, \tag{58}$$

where $H_{k\beta}^j$ is the Hessian $\nabla^2 c_j(\cdot)$ at a point on the line segment determined by x_k and x_{k+1}. Given d_k as the solution to (41), we have that $c_j(x_k) + \nabla c_j(x_k)^T d_k = 0$ and as a consequence, we can write based on (58) that

$$c_j(x_{k+1}) = (1 - \beta_k)c_j(x_k) + \frac{1}{2}\beta_k^2 d_k^T H_{k\beta}^j d_k.$$

By Assumption 3.2, $|c_j(x_{k+1})| \leq |(1 - \beta_k)c_j(x_k)| + \beta_k^2 H_u^c \|d_k\|^2$, which in turn implies that

$$\|c(x_{k+1})\|_1 \leq (1 - \beta_k) \|c(x_k)\|_1 + m\beta_k^2 H_u^c \|d_k\|^2 . \tag{59}$$

Applying simple norm inequalities, we have

$$\beta_k \left|(\lambda^{k+1})^T c(x_k)\right| \leq \beta_k \left\|\lambda^{k+1}\right\|_\infty \|c(x_k)\|_1 . \tag{60}$$

Since step 7 of the algorithm chooses $\theta_k \geq \eta_\gamma^- \left\|\lambda^{k+1}\right\|_\infty + \gamma$, where η_γ^- and γ are positive constants, we can write based on (59) and (60) that

$$\begin{aligned}
&\theta_k \|c(x_k)\|_1 - \eta_\gamma^- \beta_k \left|(\lambda^{k+1})^T c(x_k)\right| - \theta_k \|c(x_{k+1})\|_1 \\
&\quad \geq (\theta_k - \eta_\gamma^- \beta_k \left\|\lambda^{k+1}\right\|_\infty) \|c(x_k)\|_1 - \theta_k(1 - \beta_k) \|c(x_k)\|_1 - \theta_k m\beta_k^2 H_u^c \|d_k\|^2 \\
&\quad = (\theta_k\beta_k - \eta_\gamma^- \beta_k \left\|\lambda^{k+1}\right\|_\infty) \|c(x_k)\|_1 - \theta_k m H_u^c \beta_k^2 \|d_k\|^2 \\
&\quad \geq \beta_k \left(\gamma \|c(x_k)\|_1 - \theta_k m H_u^c \beta_k \|d_k\|^2\right) .
\end{aligned} \tag{61}$$

Therefore, if β_k is reduced by the line search step through backtracking in Algorithm 1 to satisfy

$$0 < \beta_k \leq \frac{\eta_\beta \alpha_k}{2H_u^c \theta_k m}, \tag{62}$$

then

$$\theta_k \|c(x_k)\|_1 - \eta_\gamma^- \beta_k \left|(\lambda^{k+1})^T c(x_k)\right| - \theta_k \|c(x_{k+1})\|_1 \geq -\frac{1}{2}\eta_\beta \alpha_k \beta_k \|d_k\|^2 .$$

Using ceiling function $\lceil \cdot \rceil$, which returns the least integer greater than the input, we can write

$$\beta_k \geq \frac{1}{2}^{\lceil \log_{\frac{1}{2}} \frac{\eta_\beta \alpha_k}{2H_u^c \theta_k m} \rceil} . \tag{63}$$

We remark that both the denominator and numerator in (62) are positive and independent of the line search. Further, by Lemma 3.3 and finite λ^{k+1}, all terms in (62) remain finite. Therefore, the backtracking stops in finite steps. This completes the proof. □

Lemma 3.6. *The $\beta_k \in (0,1]$ that meets the line search condition in (50) also satisfies the conditions from (49), or equivalently*

$$\begin{aligned}
\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\theta_k \|c(x_k)\|_1 + \theta_k \|c(x_{k+1})\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \|d_k\|^2, \\
\eta_\gamma^+\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\theta_k \|c(x_k)\|_1 + \theta_k \|c(x_{k+1})\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \|d_k\|^2, \\
\eta_\gamma^-\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\theta_k \|c(x_k)\|_1 + \theta_k \|c(x_{k+1})\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \|d_k\|^2.
\end{aligned} \tag{64}$$

Proof. From simple absolute value inequality, we have

$$\begin{aligned}
\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|, \\
\eta_\gamma^+\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\eta_\gamma^+\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|, \\
\eta_\gamma^-\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|.
\end{aligned} \tag{65}$$

Given that $0 < \eta_\gamma^+ \leq 1 \leq \eta_\gamma^-$,

$$-\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right| \leq -\beta_k \left|(\lambda^{k+1})^T c(x_k)\right| \leq -\eta_\gamma^+\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|. \tag{66}$$

Therefore, from (65) and (66)

$$\begin{aligned}
\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|, \\
\eta_\gamma^+\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|, \\
\eta_\gamma^-\beta_k(\lambda^{k+1})^T c(x_k) &\geq -\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right|.
\end{aligned} \tag{67}$$

From the line search condition (50), we have

$$-\eta_\gamma^-\beta_k \left|(\lambda^{k+1})^T c(x_k)\right| \geq -\theta_k \|c(x_k)\|_1 + \theta_k \|c(x_{k+1})\|_1 - \eta_\beta \frac{1}{2}\alpha_k\beta_k \|d_k\|^2. \tag{68}$$

Combined with (67) the proof is completed. □

Lemma 3.7. *The step $x_{k+1} = x_k + \beta_k d_k$ is a decreasing step for the merit function (42) if β_k satisfies the line search condition. Further, if the Lagrange multipliers λ^k is finite for all k, the speed of decrease satisfies $\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) > c_\phi \|d_k\|^2$ for some constant c_ϕ.*

Proof. For a serious step x_{k+1} to be taken, step 8 and 10 are satisfied so that $\rho_k > 0, \rho_k^\beta > 0$. We distinguish three cases based on the value of α_k and sign of δ_k^β. The first case is $\alpha_k > 2C$. By upper-C^2 property in (52), as shown in (57), we have

$$\begin{aligned}
r(x_k) - r(x_{k+1}) >& -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k\beta_k^2 \|d_k\|^2 \\
=& \Phi_k(0) - \Phi_k(\beta_k d_k) = \delta_k^\beta.
\end{aligned}$$

In the second case, $\alpha_k \leq 2C$ and $\delta_k^\beta \geq 0$. From the definition of ρ_k^β in (46),

$$r(x_k) - r(x_{k+1}) > \eta_\gamma^+ \left[-\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 \right]. \tag{69}$$

The third case is when $\alpha_k \leq 2C$ and $\delta_k^\beta < 0$, and we have

$$r(x_k) - r(x_{k+1}) > \eta_\gamma^- \left[-\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 \right]. \tag{70}$$

Rearranging the first equation in optimality condition (47), we have

$$g_k + \alpha_k d_k = \nabla c(x_k)\lambda^{k+1} + \zeta_l^{k+1} - \zeta_u^{k+1}. \tag{71}$$

Then, taking the inner product with $-d_k$ and using the last equation from (47) we have

$$\begin{aligned} -g_k^T d_k - \alpha_k \|d_k\|^2 &= -(\lambda^{k+1})^T \nabla c(x_k)^T d_k - d_k^T \zeta_l^{k+1} + d_k^T \zeta_u^{k+1} \\ &= (\lambda^{k+1})^T c(x_k) - (d_k - d_l^k + d_l^k)^T \zeta_l^{k+1} + (d_k - d_u^k + d_u^k)^T \zeta_u^{k+1} \\ &= (\lambda^{k+1})^T c(x_k) - (d_l^k)^T \zeta_l^{k+1} + (d_u^k)^T \zeta_u^{k+1} \\ &= (\lambda^{k+1})^T c(x_k) + x_k^T \zeta_l^{k+1} + (x_u - x_k)^T \zeta_u^{k+1} \\ &\geq (\lambda^{k+1})^T c(x_k). \end{aligned} \tag{72}$$

The third equality of (72) comes from the complementarity conditions $Z_l^{k+1}(d_k - d_l^k) = 0$ and $Z_u^{k+1}(d_k - d_u^k) = 0$ in (47). The inequality can be obtained from bound constraints in (48) where $x_k \geq 0$, $x_u - x_k \geq 0$, $\zeta_l^{k+1} \geq 0$ and $\zeta_u^{k+1} \geq 0$ for the current and previous iteration. Next, multiplying both sides of (72) by β_k and then subtracting $\frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2$ leads to

$$\begin{aligned} -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 &\geq \alpha_k \beta_k \|d_k\|^2 - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 + \beta_k (\lambda^{k+1})^T c(x_k) \\ &\geq \frac{1}{2}\alpha_k \beta_k \|d_k\|^2 + \beta_k (\lambda^{k+1})^T c(x_k), \end{aligned} \tag{73}$$

where the second inequality makes use of $\beta_k \in (0, 1]$. Notice that the left-hand side of (73) is δ_k^β and is not guaranteed to be positive. Multiplying both sides of (73) by η_γ^+ and η_γ^- respectively, we obtain

$$\begin{aligned} -\eta_\gamma^+ \beta_k g_k^T d_k - \frac{1}{2}\eta_\gamma^+ \alpha_k \beta_k^2 \|d_k\|^2 &\geq \frac{1}{2}\eta_\gamma^+ \alpha_k \beta_k \|d_k\|^2 + \eta_\gamma^+ \beta_k (\lambda^{k+1})^T c(x_k), \\ -\eta_\gamma^- \beta_k g_k^T d_k - \frac{1}{2}\eta_\gamma^- \alpha_k \beta_k^2 \|d_k\|^2 &\geq \frac{1}{2}\eta_\gamma^- \alpha_k \beta_k \|d_k\|^2 + \eta_\gamma^- \beta_k (\lambda^{k+1})^T c(x_k). \end{aligned} \tag{74}$$

Finally, we can examine the merit function $\phi_{1\theta_k}(\cdot)$. If $\alpha_k > 2C$, combine the inequality in (57), (73) and the first inequality from Lemma 3.6, we have

$$\begin{aligned}\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) =& r(x_k) - r(x_{k+1}) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ >& -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\alpha_k \beta_k \|d_k\|^2 + \beta_k (\lambda^{k+1})^T c(x_k) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\alpha_k \beta_k \|d_k\|^2 - \eta_\beta \beta_k \frac{1}{2}\alpha_k \|d_k\|^2 \\ =& (1-\eta_\beta)\frac{1}{2}\alpha_k \beta_k \|d_k\|^2 .\end{aligned} \tag{75}$$

Otherwise, with $\alpha_k \leq 2C$ and $\delta_k^\beta \geq 0$, we apply in order (69), (74) and the second inequality from Lemma 3.6 to obtain

$$\begin{aligned}\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) =& r(x_k) - r(x_{k+1}) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ >& -\eta_\gamma^+ \beta_k g_k^T d_k - \eta_\gamma^+ \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\eta_\gamma^+ \alpha_k \beta_k \|d_k\|^2 + \eta_\gamma^+ \beta_k (\lambda^{k+1})^T c(x_k) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\eta_\gamma^+ \alpha_k \beta_k \|d_k\|^2 - \eta_\beta \beta_k \frac{1}{2}\alpha_k \|d_k\|^2 \\ =& (\eta_\gamma^+ - \eta_\beta)\frac{1}{2}\alpha_k \beta_k \|d_k\|^2 ,\end{aligned} \tag{76}$$

with $\eta_\gamma^+ - \eta_\beta > 0$. Similarly, when $\delta_k^\beta \leq 0$, applying in order (70), (74) and the third inequality from Lemma 3.6, we have

$$\begin{aligned}\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) =& r(x_k) - r(x_{k+1}^\beta) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ >& -\eta_\gamma^- \beta_k g_k^T d_k - \eta_\gamma^- \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\eta_\gamma^- \alpha_k \beta_k \|d_k\|^2 + \eta_\gamma^- \beta_k (\lambda^{k+1})^T c(x_k) + \theta_k \|c(x_k)\|_1 - \theta_k \|c(x_{k+1})\|_1 \\ \geq& \frac{1}{2}\eta_\gamma^- \alpha_k \beta_k \|d_k\|^2 - \eta_\beta \beta_k \frac{1}{2}\alpha_k \|d_k\|^2 \\ =& (\eta_\gamma^- - \eta_\beta)\frac{1}{2}\alpha_k \beta_k \|d_k\|^2 ,\end{aligned} \tag{77}$$

where $\eta_\gamma^- - \eta_\beta > 0$.

Therefore, in all cases, a serious step $x_{k+1} = x_k + \beta_k d_k$ is a decreasing direction for the merit function $\phi_{1\theta_k}(\cdot)$. If λ^k is finite for all k, θ_k will stay constant for k large enough from step 7 of Algorithm 1. Let $\bar{\theta}$ be the constant value for k large enough so that $\theta_k \leq \bar{\theta}$ for

all k. Then, by (63),

$$\beta_k \geq \frac{1}{2}^{\lceil \log_{\frac{1}{2}} \frac{\eta_\beta \alpha_0}{2H_u^c \theta_m} \rceil} := \bar{\beta}, \tag{78}$$

due to the monotonicity of α_k and θ_k. In other words, β_k is bounded below by $\bar{\beta}$ for all k. From (75), (76), (77), $\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) > (\eta_\gamma^+ - \eta_\beta)\frac{1}{2}\alpha_0\bar{\beta}\,\|d_k\|^2$. Or simply, there exists c_ϕ such that $\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1}) > c_\phi\,\|d_k\|^2$. □

In order to obtain a stabilized λ^k, a constraint qualification is necessary for the constraints $c(x) = 0$ in (37). In Section 2, we discussed calmness as the weak constraint qualification that would ensure a KKT point instead of a Fritz-John critical point in our nonsmooth upper-C^2 setup. Here, we resort to the stronger LICQ [35] to prove stabilization of Lagrange multipliers for our proposed algorithm. A topic of further research will be to derive the results of this section under a weak constraint qualification such as calmness.

Lemma 3.8. *If LICQ of the constraints in (37) are satisfied at every accumulation points $\bar{x}$ of serious steps $\{x_k\}$ generated by the algorithm, then the sequence of Lagrange multipliers for the solutions to problem (41) $\{\zeta_u^{k+1}\}, \{\zeta_l^{k+1}\}$ and $\{\lambda^{k+1}\}$ are bounded. Thus, there exists k, such that $\left\|\lambda^t\right\|_\infty \leq \lambda^U$, $\zeta_u^t \leq \zeta_u^U$ and $\zeta_l^t \leq \zeta_l^U$ for all $t \geq k$, where $\lambda^U > 0, \zeta_u^U > 0, \zeta_l^U > 0$ are the upper bounds. Further, this means there exists $\bar{\theta}$ such that $\theta_t = \bar{\theta}$ for all $t \geq k$.*

Proof. We rewrite the first equation in optimality condition in (47) as

$$g_k + \alpha_k d_k - \sum_{j=1}^{m} \lambda_j^{k+1} \nabla c_j(x_k) - \sum_{i=1}^{n} (\zeta_l^{k+1})_i e_i + \sum_{i=1}^{n} (\zeta_u^{k+1})_i e_i = 0, \tag{79}$$

where $e_i \in \mathbb{R}^n$ is a vector such that $(e_i)_i = 1$ and $(e_i)_k = 0, k \neq i$. Since $(\zeta_l^{k+1})_i (\zeta_u^{k+1})_i = 0$, the bound constraints Lagrange multipliers are combined into $\zeta^{k+1} = \zeta_l^{k+1} - \zeta_u^{k+1}$. A component of ζ_l^{k+1} or ζ_u^{k+1} is unbounded if and only if the corresponding component in ζ^{k+1} is unbounded. Let I be the index set of the active bound constraints, hence

$$g_k + \alpha_k d_k = \sum_{j=1}^{m} \lambda_j^{k+1} \nabla c_j(x_k) + \sum_{i \in I} (\zeta^{k+1})_i e_i. \tag{80}$$

Since $\{x_k\}, \{g_k\}$ are bounded ($r(\cdot)$ being Lipschitz continuous on a bounded domain) and $\{\alpha_k\}$ is finite by Lemma 3.3, the left-hand side of the equation stays bounded throughout the iterations. From LICQ at $\bar{x}$, we know that $\nabla c_j(\bar{x}) \in \mathbb{R}^n$ and $e_i, i \in I$ are linearly independent and bounded vectors. Without losing generality, suppose $\lambda_j^{k+1}, j \in [1, m]$ is not bounded as $k \to \infty$. Then, we have $\left\|\lambda^k\right\|_\infty \to \infty$. Passing on to a subsequence if necessary, we can assume $x_k \to \bar{x}$ as $k \to \infty$, where $\bar{x}$ is an accumulation point. Regardless

of the behavior of $\{\zeta^{k+1}\}$, the right-hand side of (80) will be unbounded due to linear independence among the vectors. This is a contradiction. Same process can be repeated for $\zeta_j^{k+1}, j \in [1, m]$.

Therefore, there exist $\lambda^U, \zeta_l^U \geq 0, \zeta_u^U \geq 0$ such that $\left\|\lambda^t\right\|_\infty \leq \lambda^U$, $\zeta_u^t \leq \zeta_u^U$ and $\zeta_l^t \leq \zeta_l^U$ for all k. Since θ_k is determined by λ^k (step 7 in Algorithm 1), there exists k and $\bar{\theta}$ such that $\theta_t = \bar{\theta}$ for all $t \geq k$. □

Theorem 3.9. *Given the Assumptions 3.1 and 3.2, if the constraints in (37) satisfy the conditions in Lemma 3.8, then every accumulation point of the solution steps $\{x_k\}$ generated from Algorithm 1 is a KKT point of the problem (37). That is, there exists a subsequence of $\{x_k\}$ that converges to $\bar{x}$, and $\bar{\lambda} \in \mathbb{R}^m$, $\bar{\zeta}_u \in \mathbb{R}^n$, $\bar{\zeta}_l \in \mathbb{R}^n$ such that the first-order optimality conditions are satisfied at $\bar{x}$*

$$\begin{aligned} 0 \in \bar{\partial} r(\bar{x}) - \nabla c(\bar{x})\bar{\lambda} - \bar{\zeta}_l + \bar{\zeta}_u, \\ \bar{Z}_l \bar{x} = 0, \;\; \bar{Z}_u(\bar{x} - x_u) = 0, \\ c(\bar{x}) = 0, \\ \bar{\zeta}_l, \bar{\zeta}_u, \bar{x}, x_u - \bar{x} \geq 0. \end{aligned} \tag{81}$$

Proof. By Lemma 3.3, there exists $k_0 > 0$ such that for all $t > k_0$, $\alpha_t = \alpha_{k_0} = \bar{\alpha}$ and all following steps are serious steps. By Lemma 3.8, there exists $k_1 > 0$ such that for $t > k_1$, the Lagrange multipliers are bounded above and $\theta_t = \theta_{k_1} = \bar{\theta}$. We say k is large enough if $k \geq \max(k_0, k_1)$, in which case the parameters of the algorithm stabilizes at $\alpha_t = \bar{\alpha}$ and $\theta_t = \bar{\theta}$ for $t \geq k$.

Since the domain of x is bounded and $r(\cdot)$ is Lipschitz, the serious steps sequence $\{x_k\}$ as well as the subgradient sequence $\{g_k\}$ are bounded. Therefore, there exists at least one accumulation point for $\{x_k\}$. Let $\bar{x}$ be an accumulation point of $\{x_k\}$ and $\{x_{k_s}\}$ be a subsequence of $\{x_k\}$ such that $x_{k_s} \to \bar{x}$.

From Lemma 3.5, line search terminates successfully and by Lemma 3.7, for k large enough, $\{\phi_{1\theta_k}(x_k)\}$ is a decreasing and bounded sequence with a fixed parameter $\bar{\theta}$. Thus, $\phi_{1\theta_k}(x_k)$ converges. Let $\lim_{k\to\infty} \phi_{1\theta_k}(x_k) \to \bar{\phi}_{1\bar{\theta}}$, *i.e.*, $\lim_{k\to\infty} r(x_k) + \bar{\theta}\left\|c(x_k)\right\|_1 \to \bar{\phi}_{1\bar{\theta}}$. From the proof of Lemma 3.7, (75), (76) and (77), we know that $\phi_{1\theta_k}(x_k) - \phi_{1\theta_k}(x_{k+1})$ is bounded below in the order of $\|d_k\|^2$. Therefore, $\lim_{k\to\infty} \|d_k\| \to 0$. In particular, $\lim_{s\to\infty} \|d_{k_s}\| \to 0$. By the last equation in (47), $c(x_{k_s}) \to 0$. Thus, $\bar{x}$ satisfies the equality constraints $c(\cdot)$. Given that the bound constraints are satisfied by all x_k, $0 \leq \bar{x} \leq x_u$.

Passing on to a subsequence if necessary, we let $g_{k_s} \to \bar{g}$, $\lambda_{k_s} \to \bar{\lambda}$, $\zeta_u^{k_s} \to \bar{\zeta}_u$, $\zeta_l^{k_s} \to \bar{\zeta}_l$.

From the first equation in the optimality conditions (47), we have

$$0 = \bar{g} - \nabla c(\bar{x})\bar{\lambda} - \bar{\zeta}_l + \bar{\zeta}_u. \tag{82}$$

By the outer semicontinuity of Clarke subdifferential, with $g_{k_s} \in \bar{\partial} r(x_{k_s})$, we have $\bar{g} \in \bar{\partial} r(\bar{x})$. As a result, $0 \in \bar{\partial} r(\bar{x}) - \nabla c(\bar{x})\bar{\lambda} - \bar{\zeta}_l + \bar{\zeta}_u$. The complementarity conditions of bound constraints from (47) leads to $\bar{Z}_u(\bar{x} - x_u)$, $\bar{Z}_l\bar{x} = 0$. Together with the equality constraints $c(\bar{x}) = 0$, the first-order optimality conditions (81) of problem (37) at $\bar{x}$ are satisfied.

□

While the line search is only conducted on the less computationally expensive and analytically known smooth constraints $c(\cdot)$, it is possible to avoid it altogether. In bundle methods, it has been shown that a convex feasible set for x can make the algorithm converge without line search [19]. Similarly, if the constraints form a convex feasible set, they do not need to be linearized and the simplified bundle algorithm converges without line search.

Proposition 3.10. *If the equality constraint $c(\cdot)$ and bound constraints in (37) form a convex and bounded set in $\mathbb{R}^n$, then instead of (41) we solve subproblem*

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \phi_k(x) \\ \text{subject to} \quad & c(x) = 0, \\ & 0 \le x \le x_u. \end{aligned} \tag{83}$$

And the line search step can be skipped with $x_{k+1} = x_k + d_k$, d_k being the solution to (83). The convergence properties are maintained.

3.3 Application to two-stage stochastic optimization problem

The algorithm and convergence analysis can be readily extended to two-stage stochastic programming problems, where the quadratic approximation function $\phi_k(\cdot)$ is needed only for the nonsmooth nonconvex second-stage solution functions. Problem (3) is approximated locally as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) + \phi_k(x) \\ \text{subject to} \quad & c(x) = c_E \\ & d^l \le d(x) \le d^u \\ & x^l \le x \le x^u. \end{aligned} \tag{84}$$

The first-stage objective $f(\cdot)$, which is continuously differentiable, is kept as it is. As a result, we can take advantage of the sparsity structure arising from $f(\cdot)$ since the Hessian of $\phi_k(\cdot)$ is diagonal.

The update rule of α_k is critical and problem dependent. It is a trade-off between robust convergence behavior (large α_k) and fast but potentially unstable convergence (small α_k). The α_k in Algorithm 1 does not bear much meaningful structure from the objective function since the Hessian itself might not exist. Nevertheless, for problems with better differentiability, it is possible to explore ways to extract more second-order information.

One option is to use the Barzilai-Borwein (BB) gradient method [3], which can be interpreted as an approximation to the secant equation. The update rule for α_k is

$$\alpha_k = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}, \tag{85}$$

where $s_{k-1} = x_k - x_{k-1}, y_{k-1} = g_k - g_{k-1}$. This choice of α_k can in practice increase the convergence rate if the objective $r(\cdot)$ have more favorable properties [44]. Alternatively, α_k can be viewed as a measure of the inverse of a trust-region radius. The larger α_k is, the smaller the step size will be. Hence, α_k can be updated based on how accurate the previous approximation is, as in trust-region methods [35]. This view is adopted in the proposed algorithm. A simple multiplication rule where $\alpha_{k+1} = \eta_\alpha \alpha_k, \eta_\alpha > 1$ could be effective when α_k is increased. In all cases, problem specific α_{max} and α_{min} can be assigned to make the algorithm more efficient and robust. This is the area of the algorithm that is rich for experimentation.

It is also possible to gauge α_k based on function value, in addition to the trust-region ratio ρ_k. If the function value range of $r(\cdot)$ is known, such rules might provide better estimate of α_k. For example, we can find α_k by requiring the minimum value of $\phi_k(\cdot)$ over a chosen subset of domain $X' \subset X$ to be larger than certain ratio of the function value at x_k, *i.e.*,

$$\underset{x \in X'}{\text{minimize}}\ \phi_k(x) \geq \eta_k r_k \tag{86}$$

where η_k is the chosen ratio.

The same ρ_k is computed and if $\rho_k > 0$ is not satisfied as in step 8 and 14 of Algorithm 1, η_k is increased by the fixed increase ratio η_α with $\eta_{k+1} = \eta_\alpha \eta_k$. Using η_k as an intermediate parameter, α_k is then obtained as the minimum value that would hold (86) true. The choice of α_k thus depends on local function value r_k and subgradient g_k as well as η_k and will no longer stay monotonic throughout the iterations as in Algorithm 1.

More importantly in practice, α_k can be reduced when ρ_k behaves well, *e.g.*, is close to 1. In our experience, reducing α_k helps to achieve convergence faster while the algorithm remains robust due to the mechanism of rejecting a step. To further speed up convergence, scalar α_k can also be replaced by a diagonal matrix with varying values. One way of specifying the diagonal values is to take into account the distance between a component of x and its upper and lower bounds. It is possible that multiple components of the optimization variable x reach their upper/lower bounds. Since they are more likely to stay at the bounds, it is reasonable to assign them larger corresponding diagonal values of the matrix α_k to encourage movement of other components of x. For a first-order algorithm, this could make a difference in convergence and proves to be so in the SCACOPF application.

3.4 Consistency restoration in linearized constraint

As mentioned previously, the linearized constraints of the model subproblem (41) can become infeasible even when the original problem (37) is feasible, a phenomenon referred to as inconsistency, which is also present in SQP methods. In this section we propose a supplemental consistency restoration algorithm to tackle this difficulty. This algorithm solves, instead of (41), a penalized subproblem where the constraints are incorporated into the objective in hope of generating a new serious point with consistent linearized constraints. As is common with penalty methods, the accumulation points might not be feasible KKT points. For the update rule of the penalty parameter, we borrow an idea from a sequential linear-quadratic programming (SLQP) method in [6] that relies on a feasibility problem solution.

Whenever problem (41) has inconsistent linearized constraints, the following penalty problem is formulated:

$$\begin{aligned} \underset{d}{\text{minimize}} \quad & \pi_k \Phi_k(d) + \left\| c(x_k) + \nabla c(x_k)^T d \right\|_1 \\ \text{subject to} \quad & d_l^k \leq d \leq d_u^k, \end{aligned} \tag{87}$$

where $\pi_k \geq 0$ is the penalty parameter. While (87) is straightforward, to avoid the difficulties with nonsmooth objective, as conventional in SQP methods, the following equivalent

quadratic programming problem is solved instead

$$\begin{aligned} \underset{d,v,w}{\text{minimize}} \quad & \pi_k \Phi_k(d) + \sum_{j=1}^{m} (v_j + w_j) \\ \text{subject to} \quad & c_j(x_k) + \nabla c_j(x_k)^T d = v_j - w_j, \ j = 1, ...m, \\ & d_l^k \leq d \leq d_u^k, \\ & 0 \leq v, w, \end{aligned} \tag{88}$$

where $v, w \in \mathbb{R}^m$ are slack variables. Denoting d_k, v^k, w^k as the solutions to (88), the first-order optimality conditions of problem (88) involving d are

$$\begin{aligned} \pi_k \left[g_k + \alpha_k d_k \right] + \sum_{i=1}^{m} \lambda_j^{k+1} \nabla c_j(x_k) - \zeta_l^{k+1} + \zeta_u^{k+1} = 0, \\ \lambda_j^{k+1} \left[c_j(x_k) + \nabla c_j(x_k)^T d_k - v_j^k + w_j^k \right] = 0, \ j = 1, \ldots, m, \\ c_j(x_k) + \nabla c_j(x_k)^T d_k - v_j^k + w_j^k = 0, \ j = 1, \ldots, m, \\ Z_u^{k+1}(d_k - d_u^k) = 0, Z_l^{k+1}(d_k - d_l^k) = 0, \\ \zeta_u^{k+1}, \zeta_l^{k+1}, d_k + x_k, x_u - x_k - d_k \geq 0. \end{aligned} \tag{89}$$

Here, $\lambda^{k+1} \in \mathbb{R}^m$,$\zeta_u^{k+1}, \zeta_l^{k+1} \in \mathbb{R}^n$ are the Lagrange multipliers for the constraints on d. The matrices Z_u^{k+1}, Z_l^{k+1} are diagonal matrices whose diagonal values are ζ_u^{k+1} and ζ_l^{k+1}, respectively. The remaining optimality conditions on slack variables v and w are

$$\begin{aligned} 1 - \lambda_j^{k+1} - p_j^{k+1} = 0, \ j = 1, ...m, \\ 1 + \lambda_j^{k+1} - q_j^{k+1} = 0, \ j = 1, ...m, \\ P^{k+1} v^k = 0, Q^{k+1} w^k = 0, \\ v^k, w^k, p^{k+1}, q^{k+1} \geq 0, \end{aligned} \tag{90}$$

where $p^{k+1}, q^{k+1} \in \mathbb{R}^m$ are the Lagrange multipliers for v^k, w^k. The matrices P^{k+1}, Q^{k+1} are diagonal matrices whose diagonal values are p^{k+1} and q^{k+1}, respectively.

Based on whether the slack variable bound constraints are active, the relations between Lagrange multipliers can be simplified. To see that, define the sign function $\sigma_j^k : \mathbb{R}^n \to \mathbb{R}, j = 1, \ldots, m$ of d such that

$$\sigma_j^k(d) = \begin{cases} -1, & c_j(x_k) + \nabla c_j(x_k)^T d < 0 \\ 0, & c_j(x_k) + \nabla c_j(x_k)^T d = 0 \\ 1, & c_j(x_k) + \nabla c_j(x_k)^T d > 0 \end{cases}. \tag{91}$$

In addition, we divide the constraints into two sets based on the value of $c_j(x_k)+\nabla c_j(x_k)^T d_k$. For simplicity, the two sets are referred to as the set of active and inactive equality constraints as we do without slack variables. More specifically, the active equality constraint set is defined at d_k as

$$A_k = \{1 \leq j \leq m | c_j(x_k) + \nabla c_j(x_k)^T d_k = 0\}, \tag{92}$$

and the inactive equality constraint set is

$$V_k = \{1 \leq j \leq m | c_j(x_k) + \nabla c_j(x_k)^T d_k \neq 0\}. \tag{93}$$

We can now integrate the optimality conditions (90) into (89) in the following Lemma.

Lemma 3.11. *For inactive equality constraints $c_j(\cdot), j \in V_k$, $\lambda_j^{k+1} = \sigma_j^k(d_k)$. For active equality constraints, i.e., $j \in A_k$, $-1 \leq \lambda_j^{k+1} \leq 1$.*

Proof. Note first that for any $1 \leq j \leq m$, the slack variable solutions satisfy $v_j^k w_j^k = 0$. This is due to the bound constraints on v, w and their presence in the objective. Next, we consider the three cases given the value of $c_j(x_k) + \nabla c_j(x_k)^T d_k$, which corresponds to the three values of $\sigma_j^k(d_k)$ for $j = 1, \ldots, m$. The first two cases both have $j \in V_k$. If $\sigma_j^k(d_k) = 1$, then by the third equation in (89), $v_j^k > 0, w_j^k = 0$. From the complementarity equations in (90), the corresponding Lagrange multiplier to v_j^k is 0, *i.e.*, $p_j^{k+1} = 0$. By the first equation in (90), $\lambda_j^{k+1} = 1$. If $\sigma_j^k(d_k) = -1$, then similarly using the third equation in (89), $v_j^k = 0, w_j^k > 0$ and the corresponding Lagrange multiplier $q_j^{k+1} = 0$. By the second equation in (90), $\lambda_j^{k+1} = -1$. The first part of the Lemma is proven.

In the last case, $j \in A_k$, *i.e.*, $\sigma_j^k(d_k) = 0$. By the third equation in (89), we have $v_j^k = 0, w_j^k = 0$. Combine the first two equations in (90) through summation and subtraction, we obtain

$$\begin{aligned} \lambda_j^{k+1} &= \frac{1}{2}(q_j^{k+1} - p_j^{k+1}), \\ 2 &= p_j^{k+1} + q_j^{k+1}. \end{aligned} \tag{94}$$

Applying the bound constraints on p^{k+1}, q^{k+1} to the second equation in (94), we have $0 \leq p_j^{k+1} \leq 2$, $0 \leq q_j^{k+1} \leq 2$ and therefore from the first equation in (94) $-1 \leq \lambda_j^{k+1} \leq 1$. □

Similar to (43), we define $\delta_k^{\pi_k}$ to be the change in objective of the penalty subproblem (88) with penalty π_k, which based on (87) is

$$\delta_k^{\pi_k} = \pi_k \left(-g_k^T d_k - \frac{1}{2}\alpha_k \|d_k\|^2 \right) + \|c(x_k)\|_1 - \left\|c(x_k) + \nabla c(x_k)^T d_k\right\|_1. \tag{95}$$

The ratios ρ_k and ρ_k^β are again used to address the nonsmoothness of $r(\cdot)$, whose definitions are given in (44) and (46). The algorithm also requires line search given d_k to obtain a serious step $x_{k+1} = x_k + \beta_k d_k$, $\beta_k \in (0, 1]$. To simplify the analysis, we adopt in this section $\eta_\gamma^- = \eta_\gamma^+ = 1$ and $\eta_l^- = \eta_l^+ = 1$, making the definition of ρ_k, ρ_k^β in (44) and (46) identical across branches. Thus for a serious step, regardless of the value of α_k, the same expression between predicted and actual change in $r(\cdot)$ is satisfied. That is, for a serious step, $r(x_k) - r(x_{k+1}) > \Phi_k(0) - \Phi_k(\beta_k d_k)$ and $r(x_k) - r(x_k + d_k) > \Phi_k(0) - \Phi_k(d_k)$.

The renewed merit function and line search conditions are

$$\phi_{1\pi_k}(x) = r(x) + \frac{1}{\pi_k} \|c(x)\|_1 , \tag{96}$$

and

$$\frac{1}{\pi_k} \|c(x_k)\|_1 + \frac{\beta_k}{\pi_k} (\lambda^{k+1})^T \nabla c(x_k)^T d_k \geq \frac{1}{\pi_k} \|c(x_{k+1})\|_1 - \eta_\beta \frac{1}{2} \alpha_k \beta_k \|d_k\|^2 , \tag{97}$$

respectively.

To update the penalty parameter, the following feasibility problem is also solved:

$$\begin{aligned} \underset{d}{\text{minimize}} \quad & \left\| c(x_k) + \nabla c(x_k)^T d \right\|_1 \\ \text{subject to} \quad & d_l^k \leq d \leq d_u^k. \end{aligned} \tag{98}$$

Denote by d_k^f the solution to (98) and δ_k^f its predicted decrease, whose form is

$$\delta_k^f = \|c(x_k)\|_1 - \left\| c(x_k) + \nabla c(x_k)^T d_k^f \right\|_1 . \tag{99}$$

Notice that $\delta_k^f \geq 0$. This value is compared against $\delta_k^{\pi_k}$.

The consistency restoration algorithm is given in Algorithm 2. It is called upon by Algorithm 1 when inconsistency occurs at step 3 and exits after one serious step iteration in step 14 of Algorithm 2. However, it is possible that the linearized constraints remain inconsistent and Algorithm 2 is called repeatedly. In this case, the update rule of the penalty parameter ensures that the algorithm converges toward critical points for linearized constraint violations. A point $\bar{x}$ is called a critical point of the linearized constraint violation of $c(\cdot)$ if $\delta_k^f = 0$ at $\bar{x}$. Notice such a critical point can be either feasible or infeasible to the original problem (37).

While Algorithm 2 solves a penalized subproblem instead, it includes all the elements in Algorithm 1 to deal with the nonsmoothness of $r(\cdot)$, including the update rule for α_k. Thus, we can reuse many of the same conclusions from Section 3.2 and only provide rigorous proofs if necessary. Since the acceptance and rejection of a trial step is based on ρ_k and

Algorithm 2: Simplified bundle method: consistency restoration

Given x_k, α_k, $r(x_k)$, $g(x_k)$, θ_k and other parameters such as ϵ from Algorithm 1, choose the update coefficient $0 < \eta_\pi, \eta_f < 1$ for π_k and error tolerance $\epsilon^f \geq 0$.

If π_{k-1} does not exist, let $\pi_k = \frac{1}{\theta_k}$. Otherwise let $\pi_k = \min\left(\pi_{k-1}, \frac{1}{\theta_k}\right)$. Solve (88) with π_k and obtain d_k.

Solve the feasibility problem (98) to obtain solution d_k^f and compute δ_k^f from (99).

if $\delta_k^f < \epsilon^f$ **then**

    Stop the iteration and exit the algorithm.

while $\delta_k^{\pi_k} < \eta_f \delta_k^f$ **do**

    Reduce π_k through $\pi_k = \eta_\pi \pi_k$ and re-solve (88) with the updated π_k.

Obtain the solution d_k and Lagrange multipliers λ^{k+1} given π_k. Evaluate $r(x_k + d_k)$ and compute δ_k in (43) and ρ_k in (44).

if $\rho_k > 0$ **then**

    Find the line search parameter $\beta_k > 0$ using backtracking, starting at $\beta_k = 1$ and halving if too large, such that (97) is satisfied. Compute ρ_k^β in (46).

    if $\rho_k^\beta < 0$ **then**

        Break and go to 17.

    Take the serious step $x_{k+1} = x_k + \beta_k d_k$.

    Exit consistency restoration. Go back to Algorithm 1 and start a new iteration.

else

    Reject the trial step and update α_k with $\alpha_{k+1} = \eta_\alpha \alpha_k$.

    Go back to step 2.

ρ_k^β, which in turn solely relies on properties of $r(\cdot)$, Lemma 3.3 holds true, as claimed in the following Lemma.

Lemma 3.12. *Under the Assumption (3.1) of upper-C^2 property for the objective $r(\cdot)$, the consistency restoration Algorithm 2 produces a finite number of rejected steps. Consequently, the parameter α_k of Algorithm 2 stabilizes, i.e., there exists k such that $\alpha_t = \alpha_k$ for all $t \geq k$.*

Proof. Since Algorithm 2 has identical mechanism for rejecting steps and increasing α_k to Algorithm 1, which only relies on the property of $r(\cdot)$, the proof of Lemma 3.3 can be directly applied here. That is, only a finite number of rejected steps are needed to achieve $\alpha_k > 2C$, which guarantees $\rho_k > 0$, $\rho_k^\beta > 0$, and produces a serious step. If $\alpha_k \leq 2C$ for all k, then only finite number of rejected steps are generated, which also ensures $\rho_k > 0, \rho_k^\beta > 0$ for k large enough. As a result, there exists a k such that $\alpha_t = \alpha_k$ for $t \geq k$ (see proof of Lemma 3.3), with a finite number of rejected steps produced. □

The following lemma shows that the update rule for π_k in Algorithm 2 is well-defined.

Lemma 3.13. *The steps 6-7 in Algorithm 2 terminates successfully, i.e., there exists a $\pi_k > 0$ such that $\delta_k^{\pi_k} \geq \eta_f \delta_k^f$ and such a π_k can be found within finite steps.*

Proof. Since d_k is the solution to (88) (and equivalently (87)), we have by (99) and (95)

$$\begin{aligned}\delta_k^{\pi_k} \geq & \pi_k \left(-g_k^T d_k^f - \frac{1}{2}\alpha_k \left\|d_k^f\right\|^2\right) + \|c(x_k)\|_1 - \left\|c(x_k) + \nabla c(x_k)^T d_k^f\right\|_1 \\ = & \pi_k \left(-g_k^T d_k^f - \frac{1}{2}\alpha_k \left\|d_k^f\right\|^2\right) + \delta_k^f \geq \pi_k \left(-\|g_k\| \left\|d_k^f\right\| - \frac{1}{2}\alpha_k \left\|d_k^f\right\|^2\right) + \delta_k^f,\end{aligned} \tag{100}$$

where the last inequality uses Cauchy-Schwarz inequality. From Lemma 3.12 and Lipschitz continuity, d_k^f, g_k, and α_k are all bounded. Assigning $D = \left\|d_u^k - d_l^k\right\| = \|x_u\|$, we derive the condition on π_k such that $\delta_k^{\pi_k} \geq \eta_f \delta_k^f$ as

$$\pi_k \leq \frac{(1-\eta_f)\delta_k^f}{\|g_k\| D + \frac{1}{2}\alpha_k D^2}. \tag{101}$$

Thus, such $\pi_k > 0$ exists as long as $\delta_k^f > 0$. And since π_k is reduced through $\eta_\pi < 1$, only a finite number of steps are needed to obtain a π_k that satisfied (101) through step 9. If $\delta_k^f = 0$ and consistency restoration Algorithm 2 is still called, the algorithm would terminate at step 5 (and had converged to a critical point to the linearized constraint). □

Lemma 3.14. *Given a nonzero penalty parameter $\pi_k > 0$ and Assumption 3.2, the line search step 10 of Algorithm 2 finds $\beta_k \in (0, 1]$ satisfying the condition (97) in a finite number of steps.*

Proof. Since $c(\cdot)$ is smooth, by Taylor expansion of its jth component,

$$c_j(x_k + d_k) - c_j(x_k) = \nabla c_j(x_k)^T d_k + \frac{1}{2} d_k^T H_k^j d_k, \tag{102}$$

where $1 \leq j \leq m$ and the Hessian H_k^j depends on both x_k and d_k. Similarly,

$$c_j(x_{k+1}) = c_j(x_k) + \beta_k \nabla c_j(x_k)^T d_k + \frac{1}{2} \beta_k^2 d_k^T H_{k\beta}^j d_k. \tag{103}$$

From Assumption 3.2,

$$\begin{aligned} |c_j(x_{k+1})| &= |c_j(x_k) + \beta_k \nabla c_j(x_k)^T d_k + \frac{1}{2} \beta_k^2 d_k^T H_{k\beta}^j d_k| \\ &\leq |c_j(x_k) + \beta_k \nabla c_j(x_k)^T d_k| + \beta_k^2 H_u^c \|d_k\|^2 . \end{aligned} \tag{104}$$

From the definition of σ_j^k in (91), we can write

$$|c_j(x_k) + \nabla c_j(x_k)^T d_k| = \sigma_j^k(d_k) \left[c_j(x_k) + \nabla c_j(x_k)^T d_k \right] . \tag{105}$$

Using the fact that the absolute function $|\cdot|$ is convex, the function $|c_j(x_k) + \nabla c_j(x_k)^T d|$ is convex in d. Taking its value at $d = 0, d = d_k$ and $\beta_k d_k$, we have by convexity

$$|c_j(x_k) + \beta_k \nabla c_j(x_k)^T d_k| \leq (1 - \beta_k)|c_j(x_k)| + \beta_k |c_j(x_k) + \nabla c_j(x_k)^T d_k|. \tag{106}$$

Applying (106) to (104),

$$\begin{aligned} |c_j(x_{k+1})| &\leq (1 - \beta_k)|c_j(x_k)| + \beta_k |c_j(x_k) + \nabla c_j(x_k)^T d_k| + \beta_k^2 H_u^c \|d_k\|^2 \\ &= (1 - \beta_k)|c_j(x_k)| + \beta_k \sigma_j^k(d_k) \left[c_j(x_k) + \nabla c_j(x_k)^T d_k \right] + \beta_k^2 H_u^c \|d_k\|^2 . \end{aligned} \tag{107}$$

Let us denote the cardinality of A_k and V_k by $|A_k|$ and $|V_k|$, respectively. By summing up (107) over $j \in V_k$ and realizing that $|c_j(x_k)| \geq \sigma_j^k(d_k) c_j(x_k)$, we have

$$\sum_{j \in V_k} |c_j(x_{k+1})| \leq \sum_{j \in V_k} |c_j(x_k)| + \beta_k \sum_{j \in V_k} \sigma_j^k(d_k) \nabla c_j(x_k)^T d_k + |V_k| \beta_k^2 H_u^c \|d_k\|^2 . \tag{108}$$

Similarly, we sum up (107) over $j \in A_k$ and apply its definition in (92) to write

$$\sum_{j \in A_k} |c_j(x_{k+1})| \leq (1 - \beta_k) \sum_{j \in A_k} |c_j(x_k)| + |A_k| \beta_k^2 H_u^c \|d_k\|^2 . \tag{109}$$

Further, by (108) and (109), we have

$$\begin{aligned}\sum_{j\in V_k}(|c_j(x_k)|-|c_j(x_{k+1})|) \geq &-\beta_k\sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k - |V_k|\beta_k^2 H_u^c\|d_k\|^2,\\ \sum_{j\in A_k}(|c_j(x_k)|-|c_j(x_{k+1})|) \geq &\beta_k\sum_{j\in A_k}|c_j(x_k)| - |A_k|\beta_k^2 H_u^c\|d_k\|^2.\end{aligned} \tag{110}$$

Summing the two equations in (110) and applying $|A_k|+|V_k|=m$ gives us

$$\|c(x_k)\|_1 - \|c(x_{k+1})\|_1 \geq -\beta_k\sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k + \beta_k\sum_{j\in A_k}|c_j(x_k)| - m\beta_k^2 H_u^c\|d_k\|^2. \tag{111}$$

From Lemma (3.11) and the definition of A_k in (92), we can write

$$\begin{aligned}\sum_{j=1}^m \lambda_j^{k+1}\nabla c_j(x_k)^T d_k &= \sum_{j\in V_k}\lambda_j^{k+1}\nabla c_j(x_k)^T d_k + \sum_{j\in A_k}\lambda_j^{k+1}\nabla c_j(x_k)^T d_k\\ &= \sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k - \sum_{j\in A_k}\lambda_j^{k+1}c_j(x_k).\\ &\geq \sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k - \sum_{j\in A_k}|c_j(x_k)|.\end{aligned} \tag{112}$$

The inequality in (112) comes from the second part of Lemma 3.11. Through simple algebraic calculations and applying (111) and (112)

$$\begin{aligned}&\frac{1}{\pi_k}\|c(x_k)\|_1 - \frac{1}{\pi_k}\|c(x_{k+1})\|_1 + \frac{\beta_k}{\pi_k}\sum_{j=1}^m\lambda_j^{k+1}\nabla c_j(x_k)^T d_k \geq\\ &\quad\frac{1}{\pi_k}\sum_{j\in A_k}\beta_k|c_j(x_k)| - \frac{\beta_k}{\pi_k}\sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k - \frac{1}{\pi_k}m\beta_k^2 H_u^c\|d_k\|^2\\ &\quad+\frac{\beta_k}{\pi_k}\sum_{j\in V_k}\sigma_j^k(d_k)\nabla c_j(x_k)^T d_k - \frac{\beta_k}{\pi_k}\sum_{j\in A_k}|c_j(x_k)| \geq -\frac{1}{\pi_k}m\beta_k^2 H_u^c\|d_k\|^2.\end{aligned} \tag{113}$$

Thus, if β_k satisfies

$$0<\beta_k\leq\frac{\eta_\beta\alpha_k\pi_k}{2mH_u^c}, \tag{114}$$

where both the denominator and numerator are positive and independent of the line search, we have (97) satisfied. Using ceiling function $\lceil\cdot\rceil$, we can write

$$\beta_k \geq \frac{1}{2}^{\lceil\log_{\frac{1}{2}}\frac{\eta_\beta\alpha_k\pi_k}{2mH_u^c}\rceil}. \tag{115}$$

If $\pi_k>0$, the line search then successfully terminates after a finite number of steps based on the backtracking rule. □

The decrease in merit function follows, similar to Lemma 3.7.

Lemma 3.15. *The serious step $x_{k+1} = x_k + \beta_k d_k$ is a descent step for the merit function (96) if β_k is obtained through line search in Algorithm 2. Further, if π_k stabilizes at a finite value, i.e., there exists k such that $\pi_t = \pi_k := \bar{\pi} > 0$ for all $t \geq k$, then the speed of descent satisfies $\phi_{1\pi_k}(x_k) - \phi_{1\pi_k}(x_{k+1}) > c_\phi^\pi \|d_k\|^2$ for some $c_\phi^\pi > 0$.*

Proof. Since $\rho_k > 0, \rho_k^\beta > 0$ at any serious step, and $\eta_l^+ = \eta_l^- = \eta_\gamma^+ = \eta_\gamma^- = 1$, we can compactly write based on definitions (44) and (46)

$$\begin{aligned} r_k - r(x_k + d_k) >& \Phi_k(0) - \Phi_k(d_k), \\ r_k - r(x_{k+1}) >& \Phi_k(0) - \Phi_k(\beta_k d_k). \end{aligned} \tag{116}$$

Using the upper-C^2 property of $r(\cdot)$, as in (57) from Lemma 3.7, we have

$$r(x_k) - r(x_{k+1}) > -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2 . \tag{117}$$

Let us rearrange the first equation in the KKT conditions (89) and obtain

$$\pi_k \left[g_k + \alpha_k d_k\right] = -\sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k) + \zeta_l^{k+1} - \zeta_u^{k+1}. \tag{118}$$

Taking the dot product with $-d_k$ on both sides of (118) leads to

$$-\pi_k \left[g_k^T d_k + \alpha_k \|d_k\|^2\right] = \sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k)^T d_k - d_k^T \zeta_l^{k+1} + d_k^T \zeta_u^{k+1}. \tag{119}$$

Recall that $d_l^k = -x_k$ and $d_u^k = x_u - x_k$. Applying the complementarity conditions, which are the fourth, fifth equation in (89), (119) is simplified to

$$\begin{aligned} -\pi_k \left[g_k^T d_k + \alpha_k \|d_k\|^2\right] &= \sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k)^T d_k - (d_k - d_l^k + d_l^k)^T \zeta_l^{k+1} + d_k^T \zeta_u^{k+1} \\ &= \sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k)^T d_k + x_k^T \zeta_l^{k+1} + (d_k - d_u^k + d_u^k)^T \zeta_u^{k+1} \\ &= \sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k)^T d_k + x_k^T \zeta_l^{k+1} + (x_u - x_k)^T \zeta_u^{k+1} \\ &\geq \sum_{j=1}^m \lambda_j^{k+1} \nabla c_j(x_k)^T d_k. \end{aligned} \tag{120}$$

The last inequality utilizes the bound constraints from the previous iteration $0 \leq x_k \leq x_u$, and $\zeta_u^{k+1}, \zeta_l^{k+1} \geq 0$. Multiplying by β_k and subtracting $\pi_k \frac{1}{2}\alpha_k \beta_k^2 \|d_k\|^2$ from both sides

of (120) and using $\beta_k \in (0,1]$ results in

$$\begin{aligned}\pi_k\left[-\beta_k g_k^T d_k - \frac{1}{2}\alpha_k\beta_k^2\left\|d_k\right\|^2\right] \geq & \pi_k\beta_k\alpha_k\left\|d_k\right\|^2 - \pi_k\frac{1}{2}\alpha_k\beta_k^2\left\|d_k\right\|^2 + \beta_k\sum_{j=1}^{m}\lambda_j^{k+1}\nabla c_j(x_k)^T d_k \\ \geq & \pi_k\frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2 + \beta_k\sum_{j=1}^{m}\lambda_j^{k+1}\nabla c_j(x_k)^T d_k \\ = & \pi_k\frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2 + \beta_k(\lambda^{k+1})^T\nabla c(x_k)^T d_k.\end{aligned} \tag{121}$$

From (117), the merit function satisfies

$$\begin{aligned}\phi_{1\pi_k}(x_k) - \phi_{1\pi_k}(x_{k+1}) = & r(x_k) - r(x_{k+1}) + \frac{1}{\pi_k}\left\|c(x_k)\right\|_1 - \frac{1}{\pi_k}\left\|c(x_{k+1})\right\|_1 \\ \geq & -\beta_k g_k^T d_k - \frac{1}{2}\alpha_k\beta_k^2\left\|d_k\right\|^2 + \frac{1}{\pi_k}\left\|c(x_k)\right\|_1 - \frac{1}{\pi_k}\left\|c(x_{k+1})\right\|_1.\end{aligned} \tag{122}$$

Applying (121) and the line search condition (97), we have

$$\begin{aligned}\phi_{1\pi_k}(x_k) - \phi_{1\pi_k}(x_{k+1}) \geq & \frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2 + \frac{\beta_k}{\pi_k}(\lambda^{k+1})^T\nabla c(x_k)^T d_k \\ & + \frac{1}{\pi_k}\left(\left\|c(x_k)\right\|_1 - \left\|c(x_{k+1})\right\|_1\right) \\ \geq & \frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2 - \eta_\beta\frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2 \\ = & (1-\eta_\beta)\frac{1}{2}\alpha_k\beta_k\left\|d_k\right\|^2.\end{aligned} \tag{123}$$

Given $\pi_k > 0$, the conclusion follows. In addition, if π_k stabilizes, based on (115)

$$\beta_k \geq \frac{1}{2}^{\left\lceil \log_{\frac{1}{2}} \frac{\eta_\beta\alpha_0\bar{\pi}}{2mH_u^c}\right\rceil} := \bar{\beta}^\pi. \tag{124}$$

Thus, from (123), $\phi_{1\pi_k}(x_k) - \phi_{1\pi_k}(x_{k+1}) > (1-\eta_\beta)\frac{1}{2}\alpha_0\bar{\beta}^\pi\left\|d_k\right\|^2$. Or equivalently, there exists $c_\phi^\pi > 0$ such that $\phi_{1\pi_k}(x_k) - \phi_{1\pi_k}(x_{k+1}) > c_\phi^\pi\left\|d_k\right\|^2$. □

In general, the global convergence analysis from Section 3.2 stands when π_k is bounded away from 0 and θ_k is bounded from above. This is reflected in the following two theorems similar to Theorem 3.9.

Theorem 3.16. *Under the Assumptions 3.1, 3.2 and LICQ conditions of Lemma 3.8, if Algorithm 2 is called finite many times, then every accumulation points of the serious step sequence $\{x_k\}$ generated from Algorithm 1 and 2 is a KKT point of problem (37).*

The proof is similar to that of Theorem 3.9 and straightforward. Since there are only finite number of consistency restoration steps, the linearized constraints become consistent

for k large enough. Thus, only Algorithm 1 is called for k large enough. We can directly apply Theorem 3.9 to obtain 3.16.

Before stating the next theorem, we note that π_k and θ_k are designed to impact each other through step 7 of Algorithm 1 and step 1 in Algorithm 2. Therefore, if $\frac{1}{\pi_k}$ does not stay bounded, θ_k will not either. On the other hand, if π_k is bounded below from a nonzero value, together with the conditions in Lemma 3.8, both $\frac{1}{\pi_k}$ and θ_k are finite for all k. In addition, for k large enough, both stop increasing. A stabilized π_k at nonzero values essentially requires step 7 in Algorithm 2 to be encountered only finitely many times.

Theorem 3.17. *Under the Assumptions 3.1, 3.2 and LICQ conditions of Lemma 3.8, if Algorithm 2 is called infinitely many times with nonzero stabilized penalty parameter, i.e., $\pi_t = \pi_k > 0$ for all $t \geq k$, then any accumulation points of the sequence $\{x_k\}$ generated by Algorithm 1 and 2 is either a KKT point of (37) or a critical point of the linearized constraint violation of $c(\cdot)$.*

The proof is again similar to that of Theorem 3.9 and a brief framework is presented here. First, by Lemma 3.12, let k be large enough such that $\alpha_t = \alpha_k$ for all $t \geq k$ and all steps produced by both algorithms are serious steps. We note that by design both penalty parameters θ_k and $\frac{1}{\pi_k}$ in merit function (42) and (96) increase monotonically across iterations and algorithms. By Lemma 3.8, λ^k is bounded for k large enough. Given a nonzero and stabilized π_k for k large enough, θ_k is also bounded and remains constant based on step 7 in Algorithm 1. Together with step 7 in Algorithm 2, the merit functions $\phi_{1\theta_k}(\cdot)$ and $\phi_{1\pi_k}(\cdot)$ for both algorithms (42) and (96) have the same parameter $\theta_k = \frac{1}{\pi_k} = \bar{\theta}$, where $\bar{\theta}$ denotes the stabilized value. Hence, by Lemma 3.7 and 3.15, $\{\phi_{1\pi_k}(x_k)\}$ and $\{\phi_{1\theta_k}(x_k)\}$ at serious steps x_k decreases monotonically in the order of $\|d_k\|^2$ and is bounded below. Therefore, d_k generated by both algorithms, regardless of the order they are called, satisfies $d_k \to 0$.

Compared to the case in Theorem 3.16, it is possible that a finite number of calls of Algorithm 1 are followed by all consistency restoration calls of Algorithm 2, in which case an accumulation point of $\{x_k\}$ might be infeasible. This can be seen from the constraints of the penalty problem in (88) where v, w could be nonzero at an accumulation point $\bar{x}$ of $\{x_k\}$ even as $d_k \to 0$.

If an accumulation point $\bar{x}$ is feasible, *i.e.*, $c(\bar{x}) = 0$, then $\{x_k\}$ converges subsequently to a KKT point of problem (37), the proof of which is the same as in Theorem 3.9 at this point. Otherwise, if $\bar{x}$ is infeasible, *i.e.*, $c(\bar{x}) \neq 0$, it is not a KKT point. Meanwhile,

from (95), given still $d_k \to 0$, we have $\delta_k^{\pi_k} \to 0$. From step 7 in Algorithm 2, the update rule of π_k enforces $\eta_f \delta_k^f \leq \delta_k^{\pi_k}$. Therefore, with a constant parameter η_f, we obtain $\delta_k^f \to 0$. Hence, for an infeasible accumulation point $\bar{x}$, the update rule for the penalty parameter π_k results in $\bar{x}$ being a critical point of linearized constraint violation of $c(\cdot)$. This convergence result is similar to the exact penalty method for smooth objectives [35].

Finally, the following theorem covers the case when the penalty parameter $\pi_k \to 0$.

Theorem 3.18. *Under the Assumptions 3.1, 3.2 and LICQ conditions of Lemma 3.8, if Algorithm 2 is called infinite many times and $\pi_k \to 0$, then any accumulation points of the serious steps $\{x_k\}$ generated from Algorithm 1 and 2 is a critical point of the linearized constraint violation $c(\cdot)$.*

Proof. From Lemma 3.8 , we know that the Lagrange multipliers from Algorithm 1 are bounded. Therefore, a $\pi_k \to 0$ is caused by step 7 in Algorithm 2 being called infinitely many times which in turn increases θ_k as well. Because we are considering an infinite number of iterations where $\delta_k^f > 0$ (otherwise the algorithm would have terminated at step 4), by Lemma 3.13, the number of loops between step 6 and 7 is finite for each k. Thus, to have infinite many step 7, we must have an infinite number of iterations that would enter step 7 at least once. Let k be one of the iterations where π_k is reduced through step 7. To simplify the analysis, we denote by $\pi_k^0 = \min\left(\pi_{k-1}, \frac{1}{\theta_k}\right)$ the penalty parameter at iteration k after step 2. Then the change in objective of the penalized problem (88) with π_k^0, before the update to π_k at step 7, is

$$\delta_k^0 = \pi_k^0 \left(-g_k^T d_k^0 - \frac{1}{2}\alpha_k \left\| d_k^0 \right\|^2 \right) + \|c(x_k)\|_1 - \left\| c(x_k) + \nabla c(x_k)^T d_k^0 \right\|_1, \tag{125}$$

where d_k^0 is the solution of (88) with π_k^0. Since π_k^0 enters step 7 in Algorithm 2, $\delta_k^0 < \eta_f \delta_k^f$. Given d_k^0 as the solution to (88) with π_k^0 and using the definition of δ_k^f in (99),

$$\begin{aligned}
\eta_f \delta_k^f > \delta_k^0 =& \pi_k^0 \left(-g_k^T d_k^0 - \frac{1}{2}\alpha_k \left\| d_k^0 \right\|^2 \right) + \|c(x_k)\|_1 - \left\| c(x_k) + \nabla c(x_k)^T d_k^0 \right\|_1 \\
\geq& \pi_k^0 \left(-g_k^T d_k^f - \frac{1}{2}\alpha_k \left\| d_k^f \right\|^2 \right) + \|c(x_k)\|_1 - \left\| c(x_k) + \nabla c(x_k)^T d_k^f \right\|_1 \\
=& \pi_k^0 \left(-g_k^T d_k^f - \frac{1}{2}\alpha_k \left\| d_k^f \right\|^2 \right) + \delta_k^f \\
\geq& \pi_k^0 \left(-\|g_k\| \left\| d_k^f \right\| - \frac{1}{2}\alpha_k \left\| d_k^f \right\|^2 \right) + \delta_k^f.
\end{aligned} \tag{126}$$

Since x_k, g_k and α_k are all bounded, assigning $D = \|d_u - d_l\| = \|x_u\|$, we have

$$\begin{aligned}(1-\eta_f)\delta_k^f \leq & \pi_k^0 \left(\|g_k\| \left\|d_k^f\right\| + \frac{1}{2}\alpha_k \left\|d_k^f\right\|^2 \right) \\ = & \pi_k^0 \left(\|g_k\| D + \frac{1}{2}\alpha_k D^2 \right).\end{aligned} \tag{127}$$

Therefore, as π_k and π_k^0 approach 0, so does δ_k^f. This proves that as $x_k \to \bar{x}$, $\delta_k^f \to 0$. Thus, $\bar{x}$ is a critical point of the linearized constraint violation of $c(\cdot)$. □

Theorem 3.18 is a relatively weak result in the sense that it does not distinguish between an accumulation point $\bar{x}$ that is feasible, *i.e.*, $c(\bar{x}) = 0$ and infeasible. Stronger results are possible for smooth optimization even under a less restrictive constraint qualification, Mangasarian–Fromovitz constraint qualification. For example, Byrd et. al. [6] show that if $\pi_k \to 0$ and $c(\bar{x}) = 0$, then their SLQP algorithm converges to a KKT point. Such result for (nonsmooth) upper-C^2 objective function is not evident to us and will be the subject of our future research.

4 Numerical Applications

We present three numerical examples to demonstrate the theoretical and numerical capabilities the proposed algorithm offers. The examples are chosen within the general formulation of two-stage optimization problems. For nonsmooth nonconvex optimization problems, a wildly popular assumption of the objective function is lower-C^2 or prox-regular and the constraints are often assumed to be convex. The first two problems are synthetic problems designed to showcase the extra theoretical convergence analysis our algorithm brings to problems that do not satisfy these lower-type of properties. They are simple and computationally inexpensive. We also present a comparison of results with the redistributed bundle method in [45], which we drew inspiration from.

Example 1. (Differentiable but not continuously differentiable objective) Example 1 has the following mathematical formulation:

$$\begin{aligned}\min_{x\in\mathbb{R}^3} \quad & f(x_1) + \mu[(x_2 - \frac{1}{2})^2 + x_3^2] + r(x) \\ \text{s.t.} \quad & -5 \leq x_1 \leq 5,\ 0 \leq x_2 \leq 50,\ -1 \leq x_3 \leq 10,\end{aligned} \tag{128}$$

where $\mu = 10^5$ and $f(x_1) : \mathbb{R} \to \mathbb{R}$ is a continuously differentiable function. The function

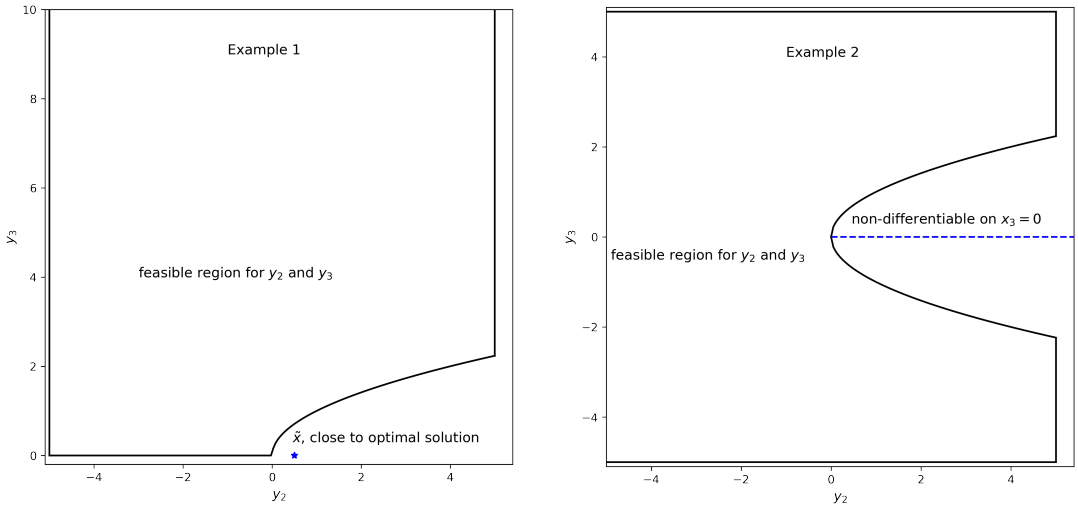

Figure 2: Feasible set of y_2, y_3 plane of example 1 (left) and example 2 (right)

$r(\cdot)$ is the solution to the second-stage problem

$$\begin{aligned} \min_{y \in \mathbb{R}^3} \quad & \|x - y\|^2 \\ \text{s.t.} \quad & y_2 \leq y_3^2, \ -5 \leq y_1 \leq 5, \\ & -5 \leq y_2 \leq 5, \ 0 \leq y_3 \leq 10. \end{aligned} \tag{129}$$

It is obvious that $r(\cdot)$ is a squared-distance function and thus upper-C^2. In addition, $r(\cdot)$ turns out to also be lower-C^1, but not lower-C^2 at $\tilde{x} = [x_1, \frac{1}{2}, 0]$, where x_1 can be any value within its bounds. This translates to $r(\cdot)$ being differentiable but not continuously differentiable at $\tilde{x}$, as shown in the feasible region plot on the left of Figure 2. In this case, our proposed algorithm offers global convergence support compared to algorithms that require a lower-C^2 objective. It needs to be pointed out that $r(\cdot)$ is continuously differentiable at remaining points in the domain and thus other algorithms with carefully chosen parameters can succeed in solving Example 1 regardless.

The true solution is obtained by treating the two-stage problem as one problem with variables in $\mathbb{R}^6$ and solved with Ipopt. The proposed Algorithm 1 starts with $\alpha_0 = 1.0, \epsilon = 10^{-8}$ and the redistributed bundle method in [45] is implemented with $\Gamma = 2, \mu_0 = 1, \eta_0 = 1$. The initial point is set to $x_0 = [1, 50, 5]^T$. The simplified bundle method exits in 4 iterations. While both algorithms quickly moved close to the solution, due to the lack of lower-C^2 property at $\tilde{x} = [x_1, \frac{1}{2}, 0]$ ($\forall x_1 \in [-5, 5]$), the convexification parameter η_n registers a large value for the redistributed bundle method. Given the error tolerance at 10^{-8}, the redistributed bundle method will require more iterations and could potentially

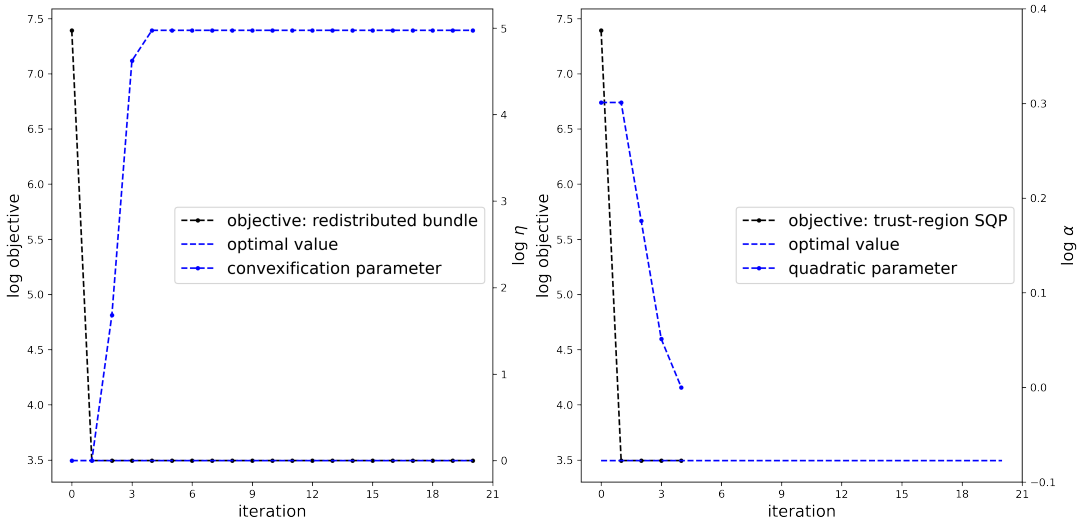

Figure 3: Convergence and quadratic coefficient plots for example 1

be destabilized due to numerical error from the large value of η_n. This problem disappears if error tolerance is set to a larger number. Figure 3 shows the numerical result of error measure against the number of iterations for both redistributed and simplified bundle method. The quadratic coefficient, which decreases in the simplified bundle method as explained in Remark 3.4, is also plotted for both algorithms.

Example 2. (Non-differentiable) Example 2 has the following form:

$$\begin{aligned} \min_{x\in\mathbb{R}^3} \quad & f(x_1)+\mu[(x_2-\frac{1}{2})^2+x_3^2]+r(x) \\ \text{s.t.} \quad & -5 \le x_1 \le 5,\ 0 \le x_2 \le 50, \\ & -5 \le x_3 \le 5. \end{aligned} \tag{130}$$

Again, $\mu = 10^5$ and $f(x_1)$ is a continuously differentiable function. The function $r(\cdot)$ is the solution to the second-stage problem

$$\begin{aligned} \min_{y\in\mathbb{R}^3} \quad & \|x-y\|^2 \\ \text{s.t.} \quad & y_2 \le y_3^2,\ -5 \le y_1, y_2, y_3 \le 5. \end{aligned} \tag{131}$$

Example 2 is designed to only vary slightly from Example 1 to illustrate the large group of problems the proposed algorithm can tackle. With a slight change in the constraint to allow $y_3 < 0$, the solution function $r(\cdot)$ is no longer differentiable on $x_3 = 0$ as multiple solutions y exist. This is illustrated on the right plot in Figure 2.

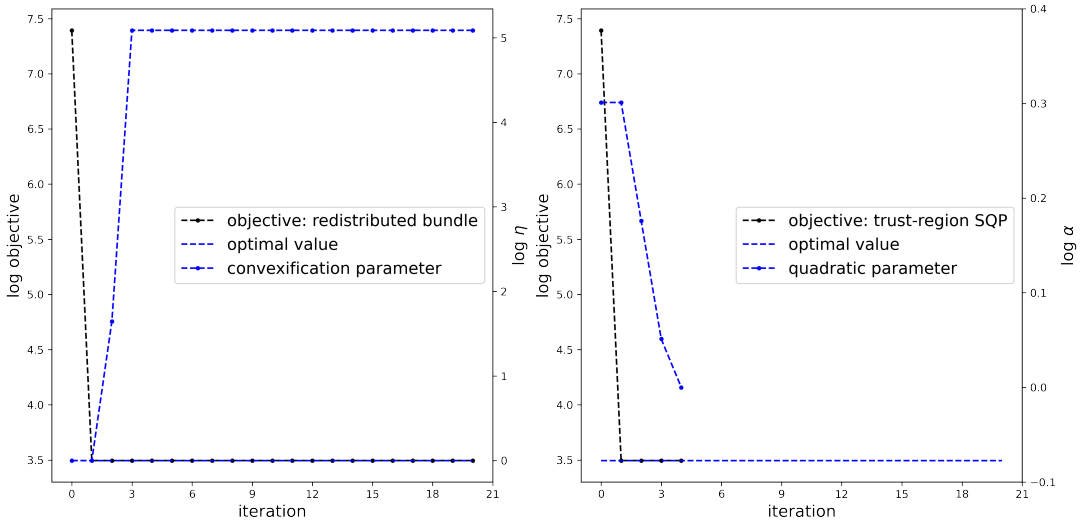

Figure 4: Convergence and quadratic coefficient plots for example 2

However, $r(\cdot)$ remains upper-C^2 and the convergence analysis for the proposed algorithm applies. Figure 4 shows the objective and quadratic coefficient for both redistributed and simplified bundle method from the same starting point as in Example 1. Similar conclusions as in Example 1 can be drawn.

Example 3. (smoothed SCACOPF) Example 3 is a SCACOPF problem with affine active power constraint for contingency (second-stage) problems. The network data used in this example is from the ARPA-E Grid Optimization competition [39]. The complete mathematical formulation is complex but the master (first-stage) problem fits in the form of (1), where r is the recourse function of the contingency problems. Details of the problem setup can be seen in [39]. The number of contingency problems that are solved to evaluate r is 100.

The coupling constraint between master and contingency variables can be viewed as linear in the former (x) but it is nonsmooth. This means recourse function r might not be upper-C^2. However, using a quadratic penalty of the coupling constraints in the contingency problems, r in (1) becomes upper-C^2 and the problem is referred to as the smoothed SCACOPF, in contrast to the original non-smoothed one. The proposed algorithm is applied to the smoothed SCACOPF, where the quadratic penalty parameter μ is set to 10^9. While this means the convergence analysis applies, we only solve an approximated problem. To verify the accuracy of the solution, the true solution is obtained by solving the extensive form of the SCACOPF with Ipopt. It is plotted with the optimization results

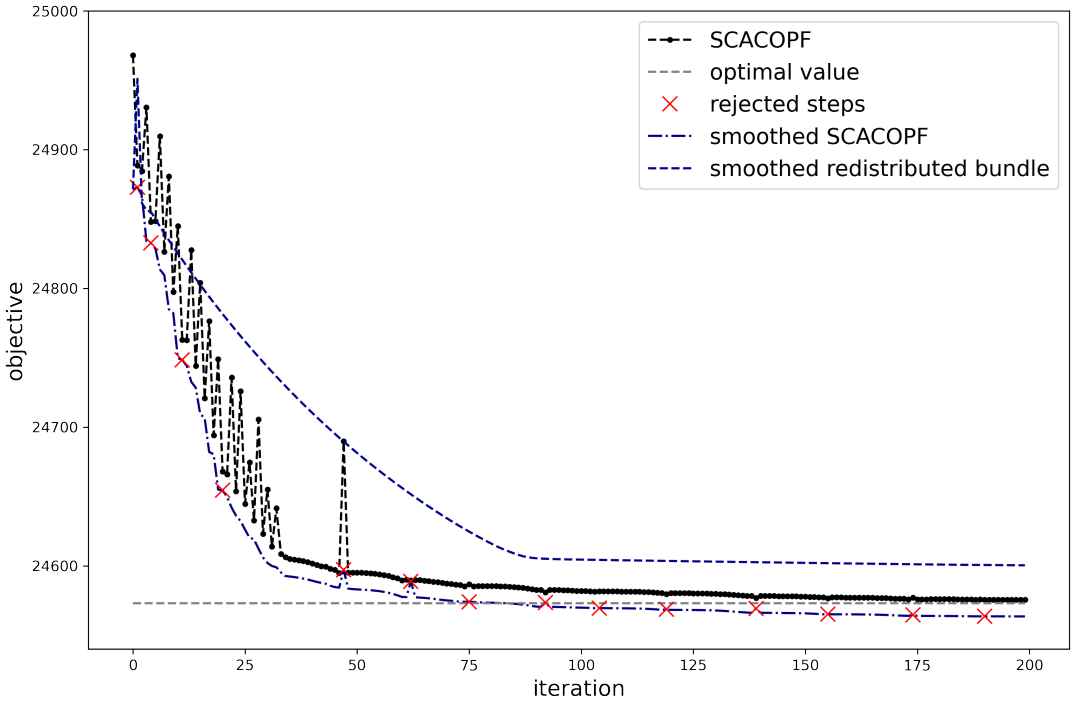

Figure 5: Convergence plots for example 3

in Figure 5. We also plot the non-smoothed objective evaluated at the optimal solution x gained from the smoothed problem at each iteration. The rejected steps are marked as well. Within 200 iterations, the non-smoothed objective reach within 0.010% error of the true solution, which is acceptable and useful in practice. To speed up convergence of this first-order method, the quadratic coefficient α_k is reduced whenever possible. For large-scale problems with 10^4 coupled optimization variables and 10^5 contingencies, the extensive form of the SCACOPF would be impractical, while the simplified bundle algorithm has been successfully deployed to supercomputers [52].

5 Conclusions

In this report, we have motivated, proposed and analyzed algorithms for a group of non-smooth, nonconvex optimization problems. We show that many two-stage (stochastic) optimization problems, including our target application SCACOPF problems exhibit interesting properties which are not thoroughly investigated previously. This has lead to our design and analysis of the simplified bundle algorithm whose global convergence can be achieved under upper-C^2 objectives. The algorithm is scalable and has been implemented

on parallel computing platforms. Numerical experiments show promising convergence and scaling results.

Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Release number LLNL-TR-833325.

Appendix A: Continuity of the second-stage solution function

This appendix details the continuity of $r_\mu(\cdot)$ in (23). We consider continuity of $r_\mu(\cdot)$ using Proposition 4.4 from [5]. The related notations are denoted as follows. The set $S(x) \subset Y$ is the optimal solutions at x. We denote by $\Phi(x) \subset \mathbb{R}^m$ the feasible set of y in the recourse subproblem. From the constraints in (23), an important observation is that $\Phi(x) = \Phi$, for all x, *i.e.*, the feasible set for y is independent of x due to smoothing. To simplify the notations, instead of applying compact set theories on the extended real vector space, it is reasonable to assume the following.

Assumption A.1. *The optimization variables x and y are bounded. The feasibility sets for x and y, denoted as $X \in \mathbb{R}^n$ and $Y \in \mathbb{R}^m$, respectively, are bounded. Moreover, Y is compact.*

We establish in Lemma A.2 that under the given assumption, the optimization problem in (23) meets the conditions in Proposition 4.4 in [5], which directly establishes the continuity of $r_\mu(\cdot)$.

Lemma A.2. *The optimization problem (23) satisfies the conditions in Proposition 4.4 in [5] at a given x, which are (1) the function $f(x, y)$ is continuous on $X \times Y$, (2) the multifunction $\Phi(\cdot)$ is closed, (3) there exists $\alpha \in \mathbb{R}$ and a compact set $C \subset Y$ such that for every x' in a neighborhood of x, the level set*

$$lev_\alpha f(x', \cdot) := \{y \in \Phi(x) : f(x', y) \leq \alpha\} \tag{132}$$

is nonempty and contained in C, (4) for any neighborhood V_y of the set $S(\bar{x})$,$\bar{x} \in X$, there exists a neighborhood V_x of $\bar{x}$ such that $V_y \cap \Phi(x) \neq \emptyset$ for all $x \in V_x$.

Proof. The objectives and constraints in the recourse subproblem are twice continuously differentiable, as mentioned when introducing (2), which guarantees (1) for the entire feasible set $\Phi(x)$. Assumption A.1, consistent with the constraints and bounds on y, ensures a closed feasible set and thus (2) is met. Since $f(x', y)$ is continuously differentiable on $X \times Y$, it is also bounded. Denoting a neighborhood of x as V_x, the obvious choice of α to make the level set $\text{lev}_\alpha f(x', \cdot), \forall x' \in V_x$ nonempty is to let it be the maximum value of $f(x', \cdot)$ on C. Therefore, an α exists such that $f(x', y) \leq \alpha, \forall x' \in V_x, y \in \Phi(x')$. Given the compact set $Y \subset \mathbb{R}^m$, a compact subset $C \subset Y$ can be found such that the level set $\text{lev}_\alpha f(x', \cdot)$ is contained in C. Thus, (3) is satisfied. To see (4), let $\bar{y} \in S(\bar{x})$

and V_y be a neighborhood of $\bar{y}$. Since $\Phi(x)$ is independent of x and compact, it is clear $V_y \cap \Phi(x) \neq \emptyset$. □

We can then prove Lemma 2.4.

Proof. Applying Proposition 4.4 in [5] and Lemma A.2 directly, $r_\mu(\cdot)$ is continuous for any $x \in X$ and the multifunction $x \to S(x)$ is upper semicontinuous at x. □

References

[1] 2017 IRC Markets Committee. Market design executive summary, 2017. Available at: https://isorto.org/wp-content/uploads/2017/09/20170905_2017IRCMarketsCommitteeExecutiveSummaryFinal.pdf.

[2] P. Apkarian, D. Noll, and O. Prot. A trust region spectral bundle method for nonconvex eigenvalue optimization. *SIAM J. on Optimization*, 19(1):281–306, Mar 2008.

[3] J. Barzilai and J. M. Borwein. Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, 8(1):141–148, Jan 1988.

[4] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer-Verlag, New York,, 1997.

[5] J. F. Bonnans and A. Shapiro. *Perturbation Analysis of Optimization Problems*. Springer-Verlag, New York, 2000.

[6] R. H. Byrd, N. I. Gould, J. Nocedal, and R. A. Waltz. On the convergence of successive linear-quadratic programming algorithms. *SIAM Journal on Optimization*, 16(2):471–89, 2005.

[7] F. Capitanescu, J.L. Martinez Ramos, P. Panciatici, D. Kirschen, A. Marano Marcolini, L. Platbrood, and L. Wehenkel. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electric Power Systems Research*, 81(8):1731–1741, 2011.

[8] Florin Capitanescu. Critical review of recent advances and further developments needed in ac optimal power flow. *Electric Power Systems Research*, 136:57 – 68, 2016.

[9] N. Chiang, C. G. Petra, and V. M. Zavala. Structured nonconvex optimization of large-scale energy systems using pips-nlp. In *2014 Power Systems Computation Conference*, pages 1–7, 2014.

[10] F. H. Clarke, R. J. Stern, and P. R. Wolenski. Proximal smoothness and the lower-c^2 property. *Journal of Convex Analysis*, 2:117–144, 1995.

[11] F.H. Clarke. *Optimization and Nonsmooth Analysis*. John Wiley & Sons New York, 1983.

[12] Y. Cui and J. S. Pang. *Modern Nonconvex Nondifferentiable Optimization*. Society for Industrial and Applied Mathematics, 2021.

[13] F. E. Curtis, T. Mitchell, and M. L. Overton. A bfgs-sqp method for nonsmooth, nonconvex, constrained optimization and its evaluation using relative minimization profiles. *Optimization Methods Software*, 32(1):148–181, January 2017.

[14] Frank E. Curtis and M. Overton. A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization. *SIAM J. Optim.*, 22:474–500, 2012.

[15] A. Daniilidis and P. Georgiev. Approximate convexity and submonotonicity. *Journal of Mathematical Analysis and Applications*, 291(1):292–301, 2004.

[16] M. Dao. Bundle method for nonconvex nonsmooth constrained optimization. *Journal of Convex Analysis*, 22:1061–1090, 12 2015.

[17] M. Dao, J. Gwinner, D. Noll, and N. Ovcharova. Nonconvex bundle method with application to a delamination problem. *Computational Optimization and Applications*, 65, 09 2016.

[18] S. Ekisheva and H. Gugel. North American AC circuit outage rates and durations in assessment of transmission system reliability and availability. In *2015 IEEE Power Energy Society General Meeting*, pages 1–5, 2015.

[19] W. Hare, C. Sagastizabal, and M. Solodov. A proximal bundle method for nonsmooth nonconvex functions with inexact information. *Computational Optimization and Applications*, 63, 05 2015.

[20] M. Hong. A distributed, asynchronous, and incremental algorithm for nonconvex optimization: An ADMM approach. *IEEE Transactions on Control of Network Systems*, 5(3):935–945, 2018.

[21] P. Kall and S. W. Wallace. *Stochastic Programming*. John Wiley & Sons, Chichester, 2nd edition, 1994.

[22] K. Kiwiel. Restricted step and levenberg-marquardt techniques in proximal bundle methods for nonconvex nondifferentiable optimization. *SIAM J. Optim.*, 6:227–249, 1996.

[23] K.C. Kiwiel. A linearization algorithm for nonsmooth minimization. *Mathematics of Operations Research*, 10(2):185–94, May 1985.

[24] C. Lemaréchal. Bundle methods in nonsmooth optimization. In C. Lemaréchal and R. Mifflin, editors, *Nonsmooth optimization (Proc. IIASA Workshop, Laxenburg, 1977),*, volume 3, pages 79–102. Pergamon, Oxford, 1978.

[25] C. Lemaréchal. Lagrangian relaxation. In M. Jünger and D. Naddef, editors, *Computational Combinatorial Optimization: Optimal or Provably Near-Optimal Solutions*, page 112–156. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[26] C. Lemaréchal and C. Sagastizabal. Variable metric bundle methods: From conceptual to implementable forms. *Math. Program.*, 76:393–410, 01 1996.

[27] J. Liu, Y. Cui, J. S. Pang, and S. Sen. Two-stage stochastic programming with linearly bi-parameterized quadratic recourse. *SIAM Journal on Optimization*, 30(3):2530–2558, 2020.

[28] L. Liu, A. Khodaei, W. Yin, and Z. Han. A distribute parallel approach for big data scale optimal power flow with security constraints. In *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 774–778, 2013.

[29] J. Lv, L. Pang, and F. Meng. A proximal bundle method for constrained nonsmooth nonconvex optimization with inexact information. *Journal of Global Optimization*, 70, 03 2018.

[30] R. Mifflin. A modification and an extension of Lemarechal's algorithm for nonsmooth minimization. In D.C. Sorensen and R.J.B. Wets, editors, *Nondifferential and Variational Techniques in Optimization*, volume 17 of *Mathematical Programming Studies*, pages 77–90. Springer, Berlin, Heidelberg, 1982.

[31] D. K. Molzahn and I. A. Hiskens. *A Survey of Relaxations and Approximations of the Power Flow Equations*. 2019.

[32] B.S. Mordukhovich. Necessary conditions in nonsmooth minimization via lower and upper subgradients. *Set-Valued Analysis*, 12(1):163–193, 2004.

[33] M. M. Mäkelä and P. Neittaanmäki. *Nonsmooth Optimization*. WORLD SCIENTIFIC, 1992.

[34] S. Nobakhtian N. Hoseini Monjezi. A new infeasible proximal bundle algorithm for nonsmooth nonconvex constrained optimization. *Comput Optim Appl*, 74:443–480, 2019.

[35] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.

[36] D. Noll. Cutting plane oracles to minimize non-smooth non-convex functions. *Set-Valued and Variational Analysis*, 18:531–568, 01 2009.

[37] D. Noll. Bundle method for non-convex minimization with inexact subgradients and function values. *Springer Proceedings in Mathematics and Statistics*, 50, 01 2013.

[38] North America Electric Reliability Corporation (NERC). Reliability standards for the bulk electric systems of north america, July 2020. Available at: https://www.nerc.com/pa/Stand/Pages/default.aspx.

[39] C. G. Petra and I. Aravena. Solving realistic security-constrained optimal power flow problems. *Operations Research*, submitted, 2021.

[40] C. G. Petra, O. Schenk, and M. Anitescu. Real-time stochastic optimization of complex energy systems on high performance computers. *Computing in Science and Engineering*, 99:1–9, 2014.

[41] C. G. Petra, O. Schenk, M. Lubin, and K. Gärtner. An augmented incomplete factorization approach for computing the Schur complement in stochastic optimization. *SIAM Journal on Scientific Computing*, 36(2):C139–C162, 2014.

[42] R. Poliquin, R. T. Rockafellar, and L. Thibault. Local differentiability of distance functions. *Transactions of the American Mathematical Society*, 352:5231–5249, 2000.

[43] W. Qiu, A. J. Flueck, and F. Tu. A parallel algorithm for security constrained optimal power flow with an interior point method. In *IEEE Power Engineering Society General Meeting, 2005*, pages 447–453 Vol. 1, 2005.

[44] M. Raydan. On the barzilai and borwein choice of steplength for the gradient method. *IMA Journal of Numerical Analysis*, 13(3):321–6, Jul 1993.

[45] A redistributed proximal bundle method for nonconvex optimization. W. hare and c. sagastizábal. *SIAM Journal on Optimization*, 20(5):2442–73, 2010.

[46] R. T. Rockafellar. Favorable classes of lipschitz continuous functions in subgradient optimization. 1981.

[47] R. T. Rockafellar and R. J-B Wets. *Variational Analysis*. Springer-Verlag, Berlin Heidelberg, 1998.

[48] H. Schramm and J. Zowe. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. *SIAM J. Optim.*, 2:121–152, 1992.

[49] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014.

[50] N. Z. Shor. *Minimization methods for non-differentiable functions*. Springer-Verlag, Berlin Heidelberg, 3 edition, 1985.

[51] J. Spingarn. Submonotone subdifferentials of lipschitz functions. *Transactions of the American Mathematical Society*, 264:77–89, 1981.

[52] J. Wang, N. Y. Chiang, and C. G. Petra. An asynchronous distributed-memory optimization solver for two-stage stochastic programming problems. In *20th International Symposium on Parallel and Distributed Computing (ISPDC)*, pages 33–40. IEEE, Jul 2021.

[53] Y. Wang, W. Yin, and J. Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *J. Sci. Comput.*, 78(1):29–63, January 2019.

[54] M. Xu, J. Ye, and L.W. Zhang. Smoothing sqp methods for solving degenerate nonsmooth constrained optimization problems with applications to bilevel programs. *SIAM Journal on Optimization*, 25:1388–1410, 01 2015.

[55] Y. Yang, L. Pang, X. Ma, and J. Shen. Constrained nonconvex nonsmooth optimization via proximal bundle method. *J. Optim. Theory Appl.*, 163(3):900–925, December 2014.

[56] J. Zowe. The BT-Algorithm for Minimizing a Nonsmooth Functional Subject to Linear Constraints. In F.H. Clarke, V.F. Dem'yanov, and F. Giannessi, editors, *Nonsmooth Optimization and Related Topics*, volume 43 of *Ettore Majorana International Science Series*. Springer, Boston, MA, 1989.