



ISE

Industrial and
Systems Engineering

A Sequential Algorithm for Solving Nonlinear Optimization Problems with Chance Constraints

FRANK E. CURTIS

Department of Industrial and Systems Engineering
Lehigh University, Bethlehem, PA, USA

ANDREAS WÄCHTER

Department of Industrial Engineering and Management Science
Northwestern University, Evanston, IL, USA

VICTOR M. ZAVALA

Department of Chemical and Biological Engineering
University of Wisconsin, Madison, WI, USA

COR@L Technical Report 16T-12



LEHIGH
UNIVERSITY.

COR@L
COMPUTATIONAL OPTIMIZATION
RESEARCH AT LEHIGH 

1 **A SEQUENTIAL ALGORITHM FOR SOLVING NONLINEAR**
2 **OPTIMIZATION PROBLEMS WITH CHANCE CONSTRAINTS**

3 FRANK E. CURTIS*, ANDREAS WÄCHTER†, AND VICTOR M. ZAVALA‡

4 *Dedicated to Roger Fletcher, a great pioneer for the field of nonlinear optimization.*

5 **Abstract.** An algorithm is presented for solving nonlinear optimization problems with chance
6 constraints, i.e., those in which a constraint involving an uncertain parameter must be satisfied with at
7 least a minimum probability. In particular, the algorithm is designed to solve cardinality-constrained
8 nonlinear optimization problems that arise in sample average approximations of chance-constrained
9 problems, as well as in other applications in which it is only desired to enforce a minimum number
10 of constraints. The algorithm employs a novel penalty function, which is minimized sequentially
11 by solving quadratic optimization subproblems with linear cardinality constraints. Properties of
12 minimizers of the penalty function in relation to minimizers of the corresponding nonlinear opti-
13 mization problem are presented, and convergence of the proposed algorithm to a stationary point of
14 the penalty function is proved. The effectiveness of the algorithm is demonstrated through numerical
15 experiments with a nonlinear cash flow problem.

16 **Key words.** nonlinear optimization, chance constraints, cardinality constraints, sample average
17 approximation, exact penalization, sequential quadratic optimization, trust region methods

18 **AMS subject classifications.** 90C15, 90C30, 90C55

19 **1. Introduction.** The focus of this paper is the proposal of an algorithm for
20 solving nonlinear optimization problems with *cardinality constraints*. Such problems
21 are those in which only a subset of constraints is desired to be enforced. Similar to
22 penalty-function-based sequential quadratic optimization (penalty-SQP) methods for
23 solving standard nonlinear optimization problems, the algorithm that we propose is
24 based on the sequential minimization of a penalty function. Each search direction
25 is computed by solving a subproblem involving Taylor approximations of the prob-
26 lem objective and constraint functions. An important application area in which such
27 cardinality-constrained problems arise—and the area in which we believe that our
28 proposed algorithm can be particularly effective—is that of *chance-constrained opti-*
29 *mization*. Hence, to put the contribution and ideas presented in this paper in context,
30 we briefly review the notion of a chance-constrained optimization problem and the
31 numerical methods that have been proposed for solving such problems. In short, our
32 proposed algorithm can be seen as a technique for extending, to nonlinear settings,
33 the ideas originally proposed by Luedtke et al. [34, 35, 36] and others (e.g., see [29])
34 for solving linear chance-constrained problems.

35 Broadly, the subject of interest in this paper is that of optimization problems
36 involving uncertain parameters, which are pervasive throughout science and engineer-
37 ing. Modern approaches to handling uncertainty in optimization involve formulating
38 and solving *stochastic optimization* problems. For example, if the uncertain param-
39 eters appear in an objective function related to a system performance measure, then

*Department of Industrial and Systems Engineering, Lehigh University. This author was partially supported by U.S. Department of Energy grant DE-SC0010615 and U.S. National Science Foundation grant DMS-1319356. E-mail: frank.e.curtis@gmail.com

†Department of Industrial Engineering and Management Sciences, Northwestern University. This author was partially supported by U.S. National Science Foundation grant DMS-1522747. E-mail: waechter@iems.northwestern.edu

‡Department of Chemical and Biological Engineering, University of Wisconsin-Madison. This author was partially supported by U.S. Department of Energy grant DE-SC0014114. E-mail: victor.zavala@wisc.edu

40 a standard technique has been to optimize the expected system performance. In
 41 practice, this can be done either by assuming certain probability distributions for
 42 the uncertain parameters, or approximately by optimizing over a discrete distribu-
 43 tion defined by a known set of scenarios (or realizations). In any case, an issue with
 44 such a strategy is that it does not safeguard against potentially high variability of the
 45 system performance; moreover, it does not protect against poor worst-case perfor-
 46 mance. Hence, popular alternative methodologies have focused on optimization over
 47 all or worst-case scenarios. In *robust optimization* [4, 6], one formulates and solves
 48 a problem over the worst-case when the uncertain parameters are presumed to lie in
 49 a tractable uncertainty set. Alternatively, one can formulate a problem in which the
 50 constraints are to be satisfied *almost surely*, i.e., they must hold with probability one.
 51 Again, in practice, this can be done with presumed knowledge of distributions of the
 52 uncertain parameters, or by satisfying all constraints defined by a sufficiently large set
 53 of scenarios of the uncertain parameters [8]. The downside of all of these techniques
 54 is that they can lead to quite conservative decisions in many practical situations.

55 A modeling paradigm that offers more flexibility is that of optimization with
 56 *chance constraints* (or *probabilistic constraints*). Chance-constrained optimization
 57 problems (CCPs) are notable in that they offer the decision maker the ability to dictate
 58 the probability of dissatisfaction that they are willing to accept. Formally, given an
 59 objective function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, a vector constraint function $c : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m$, and
 60 a probability of dissatisfaction $\alpha \in [0, 1]$, a basic CCP has the form

$$61 \quad (\text{CCP}) \quad \min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t.} \quad \mathbb{P}[c(x, \Xi) \leq 0] \geq 1 - \alpha.$$

62 Here, $\Xi : \Omega \rightarrow \mathbb{R}^p$ is a random variable with associated probability space (Ω, \mathcal{F}, P) .
 63 Contained within this paradigm are optimization models in which the constraints
 64 must be satisfied almost surely, i.e., when the probability of dissatisfaction is $\alpha = 0$.

65 Starting with the first publications over 60 years ago [13, 14], a large body of
 66 research has evolved that addresses the properties and numerical solution of CCPs.
 67 Since then, chance-constrained optimization problem formulations have been studied
 68 in a variety of application areas including supply chain management [31], power sys-
 69 tems [7], production planning [37], finance [9, 19, 42], water management [1, 20, 33, 46],
 70 chemical processes [24], telecommunications [2, 32, 45], and mining [12].

71 In terms of numerical solution methods for solving CCPs, most approaches have
 72 aimed at computing a global minimizer, at least in some sense. For example, when
 73 the CCP itself is convex, one can apply standard convex optimization techniques
 74 to find a global minimizer [11, 13, 26, 43, 44]. On the other hand, if the CCP is
 75 not convex, then one may instead be satisfied by finding a global minimizer of an
 76 approximate problem constructed so that the feasible region of the chance constraint
 77 is convex [9, 15, 38, 40, 41]. Global solutions of nonconvex CCPs can also be found
 78 using combinatorial techniques, which may be effective when one can exploit problem-
 79 specific structures such as separability of the chance constraint, linearity (or at least
 80 convexity) of the almost-sure constraint, and/or explicit knowledge of an α -concave
 81 distribution of the random variable Ξ ; e.g., see [5, 18, 30].

82 Additional opportunities arise when Ξ is discrete (i.e., when $\Omega := \{\xi_1, \xi_2, \dots, \xi_N\}$
 83 is finite) and the associated realizations have equal probability of $1/N$. Such a situa-
 84 tion might arise naturally, but also arises when one approximates problem (CCP) by
 85 drawing a number of samples of the random variable (either from a presumed distri-
 86 bution or from a set of historical data) to construct a *sample average approximation*

87 (SAA) [28, 44]. This leads to a cardinality-constrained problem

88 (P)
$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t.} \quad |\{i \in \mathcal{I} : c_i(x) \leq 0\}| \geq M,$$

89 where $|\cdot|$ denotes the cardinality of a set, $c_i : \mathbb{R}^n \rightarrow \mathbb{R}^m$ for all $i \in \mathcal{I} := \{1, \dots, N\}$,
90 and $M \in \{0, 1, \dots, N\}$. We denote the elements of the vector function c_i as c_{ij} with
91 $j = 1, \dots, m$. In an SAA of problem (CCP), one obtains (P) with $c_i := c(\cdot, \xi_i)$ for all
92 $i \in \mathcal{I}$ and $M := \lceil (1 - \alpha)N \rceil$. Note that the cardinality of the set of constraints that
93 are allowed to be violated in (P) is $N - M$ (corresponding to the probability level α).

94 Optimization problems of the form (P) also arise in other decision-making set-
95 tings. For example, each constraint $c_i(x) \leq 0$ might represent the satisfaction of
96 demand for a particular set of customers (or interested parties), and one may be in-
97 terested in making an optimal decision subject to satisfying the demands of most, if
98 not all, sets of customers. This situation is found, for instance, in power grid systems
99 where the system operator might need to select a subset of demands to curtail or shed
100 to prevent a blackout [27]. Instances of (P) also arises in model predictive control
101 when the controller has flexibility to violate constraints at certain times. These so-
102 called soft constraints are often dealt with using penalization schemes. However, these
103 can be difficult to tune to achieve a desired behavior [16]. Cardinality-constrained for-
104 mulations offer the ability to achieve the desired effect directly.

105 In this paper, we propose, analyze, and provide the results of numerical exper-
106 iments for a new method for solving problems of the form (P) when the objective
107 function f and constraint functions $\{c_1, \dots, c_N\}$ are continuously differentiable. Our
108 algorithm and analysis can readily be extended for situations in which the codomain
109 of c_i varies with $i \in \mathcal{I}$, there are multiple cardinality constraints, and/or the problem
110 also involves other smooth nonlinear equality and inequality constraints. However, for
111 ease of exposition, we present our approach in the context of problem (P) and return
112 to discuss extensions in the description of our software implementation. For cases
113 when f and $\{c_1, \dots, c_N\}$ are linear, recent methods have been proposed for solving
114 such problems using mixed-integer linear optimization techniques [29, 34, 35, 36]. In
115 our method, we allow these problems functions to be nonlinear and even nonconvex.

116 In summary, problems (CCP) and (P) represent powerful classes of optimization
117 models, and the purpose of this paper is to propose an algorithm for solving instances
118 of (P), which may arise as approximations of (CCP). We stress that solving such
119 instances is extremely challenging. Even when f and each c_i are linear, the feasi-
120 ble region of (P) is nonconvex and the problem typically has many local minima.
121 Rather than resorting to nonlinear mixed-integer optimization (MINLP) techniques,
122 the method that we propose performs a local search based on first-order and, if de-
123 sired, second-order derivatives of the problem functions. In this manner, we are only
124 able to prove that our algorithm guarantees that a stationarity measure will vanish,
125 not necessarily that a global minimizer will be found. That being said, we claim that
126 our approach is of interest for a least a few reasons. First, we claim that numerical
127 methods based on convex relaxations of nonlinear chance-constrained problems can
128 lead to conservative decisions in many practical situations and that, in such cases,
129 solving nonlinear cardinality-constrained problems can offer improved results. Sec-
130 ond, for large-scale problems, applying MINLP techniques to solve instances of (P)
131 may be intractable, in which case a local search method such as ours might represent
132 the only viable alternative. Indeed, we demonstrate in our numerical experiments
133 that our approach can find high quality (i.e., nearly globally optimal) solutions.

134 It is worth emphasizing that if one were to employ our approach for solving (P)

135 with the underlying goal of solving an instance of (CCP), then one would also need to
 136 consider advanced scenario sampling schemes, such as those discussed in [25]. While
 137 this is an important consideration, our focus in this paper is on a technique for
 138 solving a given instance of (P), including its convergence properties and practical
 139 performance. This represents a critical first step toward the design of numerical
 140 methods for solving large-scale instances of (CCP). We leave the remaining issues
 141 toward solving (CCP), such as sample set size and selection, as future work.

142 The paper is organized as follows. In §2, we introduce preliminary definitions and
 143 notation. Specifically, since our algorithm is based on a technique of *exact penaliza-*
 144 *tion*, we introduce concepts and notation related to constraint violation measures and
 145 penalty functions that will be used in the design of our algorithm and its analysis.
 146 We then present our algorithm in §3 and a global convergence theory for it in §4.
 147 In §5, we present the results of numerical experiments with an implementation of our
 148 method when applied to solve a simple illustrative example as well as a nonlinear cash
 149 flow optimization problem. These results show that our method obtains high quality
 150 solutions to nonconvex test instances. We end with concluding remarks in §6.

151 **2. Exact Penalty Framework.** Our problem of interest (P) requires that a
 152 subset of the constraints $\{i \in \mathcal{I} : c_i(x) \leq 0\}$ be satisfied. In particular, the cardinality
 153 of the subset of constraints that are satisfied must be at least M . Hence, we define
 154 $\mathfrak{S}_M \subseteq \mathcal{I}$ as the set of all lexicographically ordered subsets of the constraint index set \mathcal{I}
 155 with cardinality M . With problem (CCP) in mind, we refer to each subset in \mathfrak{S}_M as
 156 a scenario selection. For any lexicographically ordered subset $\mathcal{S} \subseteq \mathcal{I}$ (not necessarily
 157 in \mathfrak{S}_M), let $c_{\mathcal{S}}(x)$ denote the $(m|\mathcal{S}|)$ -dimensional vector consisting of a concatenation
 158 of all $c_i(x)$ with $i \in \mathcal{S}$. For convenience, we also let $c(x) := c_{\mathcal{I}}(x)$.

159 With this notation and letting $[\cdot]_+ := \max\{\cdot, 0\}$ (componentwise), we measure
 160 the constraint violation for (P) through the function $\langle\langle \cdot \rangle\rangle_M : \mathbb{R}^{mN} \rightarrow \mathbb{R}$ defined by

$$161 \quad (1) \quad \langle\langle w \rangle\rangle_M := \min_{\mathcal{S} \in \mathfrak{S}_M} \|[w_{\mathcal{S}}]_+\|_1.$$

162 In particular, given $x \in \mathbb{R}^n$, its constraint violation with respect to (P) is $\langle\langle c(x) \rangle\rangle_M$,
 163 which is equal to the minimum value of the ℓ_1 -norm constraint violation $\|[c_{\mathcal{S}}(x)]_+\|_1$
 164 over all possible choices of the scenario selection \mathcal{S} from \mathfrak{S}_M . Note that $\langle\langle \cdot \rangle\rangle_M$ is Lips-
 165 chitz continuous, since it is the pointwise minimum of Lipschitz continuous functions.
 166 Also note that when $M = N$, we have that $\langle\langle c(x) \rangle\rangle_M = \|[c(x)]_+\|_1$, which corresponds
 167 to the ℓ_1 -norm constraint violation of the standard nonlinear optimization problem:

$$168 \quad (\text{RP}) \quad \min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t.} \quad c_i(x) \leq 0 \quad \text{for all } i \in \mathcal{I}.$$

169 This is a “robust” counterpart of (P) in which all constraints are enforced.

170 A useful and intuitive alternative definition of the violation measure (1) which
 171 also takes into account the amount by which the relevant constraints are satisfied is
 172 derived in the following manner. First, let $v : \mathbb{R}^{mN} \times \mathcal{I} \rightarrow \mathbb{R}$ be defined by

$$173 \quad (2) \quad v(w, i) := \begin{cases} \|[w_i]_+\|_1 & \text{if } \|[w_i]_+\|_1 > 0 \\ \max_{j \in \{1, \dots, m\}} \{w_{i,j}\} & \text{otherwise.} \end{cases}$$

174 The value $v(c(x), i)$ measures the ℓ_1 -norm violation of the constraint $c_i(x) \leq 0$ if
 175 this violation is positive; however, if this constraint is satisfied, then it provides the
 176 element of (the vector) $c_i(x)$ that is closest to the threshold of zero. The function v

177 also induces an ordering of the entire set of constraint function indices. In particular,
 178 for a given $w \in \mathbb{R}^{mN}$, let us define $\{i_{w,1}, i_{w,2}, \dots, i_{w,N}\}$ such that

$$179 \quad (3) \quad v(w, i_{w,1}) \leq v(w, i_{w,2}) \leq \dots \leq v(w, i_{w,N}).$$

180 To make this ordering well-defined, we assume that ties are broken lexicographically;
 181 i.e., if $v(w, i_{w,j_1}) = v(w, i_{w,j_2})$ and $j_1 \leq j_2$, then $i_{w,j_1} \leq i_{w,j_2}$. It can now be easily
 182 verified that our constraint violation measure is given by

$$183 \quad (4) \quad \langle\langle c(x) \rangle\rangle_M = \sum_{j=1}^M \max\{0, v(c(x), i_{c(x),j})\} = \|[c_{\mathcal{S}(x)}(x)]_+\|_1,$$

184 where $\mathcal{S}(x) := \{i_{c(x),1}, \dots, i_{c(x),M}\} \in \mathfrak{S}_M$.

185 With these definitions of constraint violation measures in place, we now define a
 186 penalty function for problem (P) as $\phi(\cdot; \rho) : \mathbb{R}^n \rightarrow \mathbb{R}$ such that, for any $x \in \mathbb{R}^n$,

$$187 \quad \phi(x; \rho) := \rho f(x) + \langle\langle c(x) \rangle\rangle_M.$$

188 This penalty function can be expressed in terms of the standard exact ℓ_1 -norm penalty
 189 function $\phi_{\mathcal{S}}(\cdot; \rho) : \mathbb{R}^n \rightarrow \mathbb{R}$ given by

$$190 \quad (5) \quad \phi_{\mathcal{S}}(x; \rho) := \rho f(x) + \|[c_{\mathcal{S}}(x)]_+\|_1,$$

191 which takes into account only those constraints with indices in $\mathcal{S} \subseteq \mathcal{I}$. In particular,
 192 from (1), it follows that, for any $x \in \mathbb{R}^n$,

$$193 \quad (6) \quad \phi(x; \rho) = \min_{\mathcal{S} \in \mathfrak{S}_M} \phi_{\mathcal{S}}(x; \rho).$$

194 Importantly, we can characterize local minima of the penalty function ϕ by means
 195 of local minima of $\phi_{\mathcal{S}}$. In order to do this, we define a few additional quantities.
 196 First, for a given $x \in \mathbb{R}^n$, we define the critical value $v_M(x) := v(c(x), i_{c(x),M})$. This
 197 can be viewed as the M th smallest ℓ_1 -norm constraint violation or, if at least M
 198 constraints are satisfied, it can be interpreted as the distance to the zero threshold
 199 corresponding to the M th most satisfied constraint. One can view this as the *value-at-*
 200 *risk* of $\{v(c(x), i_{c(x),j})\}_{j=1}^N$ corresponding to the confidence level α . The problem (P)
 201 can be written as $\min f(x)$ s.t. $v_M(x) \leq 0$.

202 For a given $\epsilon \geq 0$, we define a partition of the indices in \mathcal{I} through the subsets

$$203 \quad \mathcal{P}(x; \epsilon) := \{i \in \mathcal{I} : v(c(x), i) - v_M(x) > \epsilon\},$$

$$204 \quad \mathcal{N}(x; \epsilon) := \{i \in \mathcal{I} : v(c(x), i) - v_M(x) < -\epsilon\},$$

$$205 \quad \text{and } \mathcal{C}(x; \epsilon) := \{i \in \mathcal{I} : |v(c(x), i) - v_M(x)| \leq \epsilon\}.$$

207 One may think of \mathcal{P} as determining the indices corresponding to constraint violations
 208 that are sufficiently larger than the critical value $v_M(x)$, \mathcal{N} as determining indices
 209 that are sufficiently smaller than $v_M(x)$, and \mathcal{C} as determining constraint violations
 210 that are close to the critical value. We now define the set of critical scenario selections

$$211 \quad (7) \quad \mathfrak{S}_M(x; \epsilon) := \{\mathcal{S} \in \mathfrak{S}_M : \mathcal{N}(x; \epsilon) \subseteq \mathcal{S} \text{ and } \mathcal{S} \subseteq \mathcal{N}(x; \epsilon) \cup \mathcal{C}(x; \epsilon)\}.$$

212 Importantly, this set includes those scenario selections that are ϵ -critical in the sense
 213 that their violation measure is within ϵ of the critical value $v_M(x)$.

214 Using these definitions, it follows as in (6) that

$$215 \quad (8) \quad \phi(x; \rho) = \min_{\mathcal{S} \in \mathfrak{S}_M} \phi_{\mathcal{S}}(x; \rho) = \phi_{\tilde{\mathcal{S}}}(x; \rho) \text{ for all } \tilde{\mathcal{S}} \in \mathfrak{S}_M(x; 0),$$

216 where we have used that the scenario selections in $\mathfrak{S}_M(x, 0)$ each correspond to min-
217 imizers of the constraint violation measure for a given $x \in \mathbb{R}^n$. We also have that

$$218 \quad (9) \quad \mathfrak{S}_M(x; \epsilon_1) \subseteq \mathfrak{S}_M(x; \epsilon_2) \text{ if } \epsilon_1 \in [0, \epsilon_2],$$

219 from which it follows that

$$220 \quad (10) \quad \phi(x; \rho) = \min_{\mathcal{S} \in \mathfrak{S}_M(x; \epsilon)} \phi_{\mathcal{S}}(x; \rho) \text{ for all } \epsilon \geq 0,$$

221 where again we have used that the scenario selections in $\mathfrak{S}_M(x, 0) \subseteq \mathfrak{S}_M(x, \epsilon)$ (for
222 any $\epsilon \geq 0$) correspond to minimizers of the constraint violation measure.

223 We now prove the following lemma relating minimizers of the penalty function ϕ
224 with those of the exact ℓ_1 -norm penalty function $\phi_{\mathcal{S}}$ for certain \mathcal{S} .

225 **LEMMA 1.** *Suppose the functions f and c_i for all $i \in \mathcal{I}$ are continuous. For any*
226 *$\rho \geq 0$, a point $x_* \in \mathbb{R}^n$ is a local minimizer of $\phi(\cdot; \rho)$ if and only if it is a local*
227 *minimizer of $\phi_{\mathcal{S}}(\cdot; \rho)$ for all $\mathcal{S} \in \mathfrak{S}_M(x_*; 0)$.*

228 *Proof.* Suppose that x_* is not a local minimizer of $\phi_{\mathcal{S}}(\cdot; \rho)$ for all $\mathcal{S} \in \mathfrak{S}_M(x_*; 0)$.
229 That is, suppose that there exists $\tilde{\mathcal{S}} \in \mathfrak{S}_M(x_*; 0)$ such that x_* is not a local minimizer
230 of $\phi_{\tilde{\mathcal{S}}}(\cdot; \rho)$. Then, there exists a sequence $\{\tilde{x}_j\}_{j=0}^{\infty} \subset \mathbb{R}^n$ converging to x_* such that
231 $\phi_{\tilde{\mathcal{S}}}(\tilde{x}_j; \rho) < \phi_{\tilde{\mathcal{S}}}(x_*; \rho)$ for all $j \in \{0, 1, 2, \dots\}$. Along with (8), this means that

$$232 \quad \phi(\tilde{x}_j; \rho) \leq \phi_{\tilde{\mathcal{S}}}(\tilde{x}_j; \rho) < \phi_{\tilde{\mathcal{S}}}(x_*; \rho) = \phi(x_*; \rho) \text{ for all } j \in \{0, 1, 2, \dots\},$$

233 implying that x_* is not a local minimizer of $\phi(\cdot; \rho)$.

234 Now suppose that x_* is not a local minimizer of $\phi(\cdot; \rho)$, meaning that there exists
235 a sequence $\{\hat{x}_j\}_{j=0}^{\infty} \subset \mathbb{R}^n$ converging to x_* such that $\phi(\hat{x}_j; \rho) < \phi(x_*; \rho)$ for all
236 $j \in \{0, 1, 2, \dots\}$. By the definition of $\mathfrak{S}_M(x_*; 0)$, we have from (8) that

$$237 \quad \langle\langle c(x_*) \rangle\rangle_M = \min_{\mathcal{S} \in \mathfrak{S}_M} \|[c_{\mathcal{S}}(x_*)]_+\|_1 = \|[c_{\tilde{\mathcal{S}}}(x_*)]_+\|_1 \text{ for all } \tilde{\mathcal{S}} \in \mathfrak{S}_M(x_*; 0).$$

238 Let $\overline{\mathfrak{S}}_M^* := \mathfrak{S}_M \setminus \mathfrak{S}_M(x_*; 0)$ and $\bar{c} := \min_{\tilde{\mathcal{S}} \in \overline{\mathfrak{S}}_M^*} \|[c_{\tilde{\mathcal{S}}}(x_*)]_+\|_1$. We then have, for all
239 $\tilde{\mathcal{S}} \in \overline{\mathfrak{S}}_M^*$, that $\|[c_{\tilde{\mathcal{S}}}(x_*)]_+\|_1 \geq \bar{c} > \langle\langle c(x_*) \rangle\rangle_M$. From continuity of the constraint
240 functions and finiteness of the set $\overline{\mathfrak{S}}_M^*$, there exists $\hat{j} \in \{0, 1, 2, \dots\}$ such that

$$241 \quad (11) \quad \|[c_{\tilde{\mathcal{S}}}(\hat{x}_j)]_+\|_1 > \frac{\bar{c} + \langle\langle c(x_*) \rangle\rangle_M}{2} > \langle\langle c(\hat{x}_j) \rangle\rangle_M \text{ for all } j \geq \hat{j} \text{ and } \tilde{\mathcal{S}} \in \overline{\mathfrak{S}}_M^*.$$

242 For any $j \geq \hat{j}$, choose some $\hat{\mathcal{S}}_j \in \mathfrak{S}_M(\hat{x}_j; 0)$ (i.e., so that $\|[c_{\hat{\mathcal{S}}_j}(\hat{x}_j)]_+\|_1 = \langle\langle c(\hat{x}_j) \rangle\rangle_M$).
243 By (11), we must have that $\hat{\mathcal{S}}_j \notin \overline{\mathfrak{S}}_M^*$, meaning that $\hat{\mathcal{S}}_j \in \mathfrak{S}_M(x_*; 0)$. Since $\mathfrak{S}_M(x_*; 0)$
244 is finite, we may assume without loss of generality that $\hat{\mathcal{S}}_j = \hat{\mathcal{S}}$ for some $\hat{\mathcal{S}} \in \mathfrak{S}_M(x_*; 0)$.
245 Finally, we obtain from our supposition and (8) that, for all $j \geq \hat{j}$,

$$246 \quad \phi_{\hat{\mathcal{S}}}(\hat{x}_j; \rho) \stackrel{\hat{\mathcal{S}} \in \mathfrak{S}_M(\hat{x}_j, 0)}{=} \phi(\hat{x}_j; \rho) < \phi(x_*; \rho) \stackrel{\hat{\mathcal{S}} \in \mathfrak{S}_M(x_*, 0)}{=} \phi_{\hat{\mathcal{S}}}(x_*; \rho).$$

247 Therefore, x_* is not a local minimizer of $\phi_{\hat{\mathcal{S}}}(\cdot; \rho)$ where $\hat{\mathcal{S}} \in \mathfrak{S}_M(x_*; 0)$. \square

248 We also have the following lemmas which justify our choice of penalty function.
 249 For similar classical results for nonlinear optimization, see, e.g. [23, Thm. 4.1].

250 **LEMMA 2.** *Suppose the functions f and c_i for all $i \in \mathcal{I}$ are continuous, and let*
 251 *$x_* \in \mathbb{R}^n$. Then, the following hold true.*

- 252 (i) *If x_* is feasible for problem (P) and is a local minimizer of $\phi(\cdot; \rho)$ for some*
 253 *$\rho > 0$, then x_* is a local minimizer of (P).*
 254 (ii) *If x_* is infeasible for problem (P) and there exists an open ball $\mathcal{B}(x_*, \theta)$*
 255 *about x_* with radius $\theta > 0$ and some $\bar{\rho} > 0$ such that x_* is a minimizer*
 256 *of $\phi(\cdot; \rho)$ in $\mathcal{B}(x_*, \theta)$ for all $\rho \in (0, \bar{\rho}]$, then x_* is a local minimizer of the*
 257 *constraint violation measure $\langle\langle c(\cdot) \rangle\rangle_M$.*

258 *Proof.* Suppose x_* is feasible for (P) and a local minimizer of $\phi(\cdot; \rho)$ for some
 259 $\rho > 0$. Then, there exists an open ball $\mathcal{B}(x_*, \theta)$ about x_* with radius $\theta > 0$ such that
 260 $\phi(x_*; \rho) \leq \phi(x; \rho)$ for all $x \in \mathcal{B}(x_*, \theta)$. Let \bar{x} be any point in $\mathcal{B}(x_*, \theta)$ that is feasible for
 261 problem (P), i.e., such that $\langle\langle c(\bar{x}) \rangle\rangle_M = 0$. Then, $f(x_*) = \phi(x_*; \rho) \leq \phi(\bar{x}; \rho) = f(\bar{x})$.
 262 Since \bar{x} was chosen as an arbitrary feasible point in $\mathcal{B}(x_*, \theta)$, this implies that x_* is a
 263 minimizer of (P) in $\mathcal{B}(x_*, \theta)$, i.e., it is a local minimizer of problem (P).

264 Now suppose that x_* is infeasible for problem (P), i.e., suppose $\langle\langle c(x_*) \rangle\rangle_M > 0$,
 265 and that there exists $\theta > 0$ and $\bar{\rho} > 0$ such that x_* is a minimizer of $\phi(\cdot; \rho)$ in an
 266 open ball $\mathcal{B}(x_*, \theta)$ about x_* with radius θ for all $\rho \in (0, \bar{\rho}]$. In order to derive a
 267 contradiction, suppose that x_* is not a minimizer of $\langle\langle c(\cdot) \rangle\rangle_M$ in $\mathcal{B}(x_*, \theta)$. Then, there
 268 exists a sequence $\{\hat{x}_j\}_{j=0}^\infty \subset \mathcal{B}(x_*, \theta)$ converging to x_* such that $\langle\langle c(\hat{x}_j) \rangle\rangle_M < \langle\langle c(x_*) \rangle\rangle_M$
 269 for all $j \in \{0, 1, 2, \dots\}$. This, along with the fact that the properties of x_* imply that
 270 $\phi(\hat{x}_j; \bar{\rho}) \geq \phi(x_*; \bar{\rho})$ for all $j \in \{0, 1, 2, \dots\}$, means that

$$271 \quad f(\hat{x}_j) - f(x_*) \geq (\langle\langle c(x_*) \rangle\rangle_M - \langle\langle c(\hat{x}_j) \rangle\rangle_M) / \bar{\rho} > 0$$

272 for all $j \in \{0, 1, 2, \dots\}$. Then, for any fixed $\rho \in (0, \bar{\rho}]$ with

$$273 \quad \rho < (\langle\langle c(x_*) \rangle\rangle_M - \langle\langle c(\hat{x}_j) \rangle\rangle_M) / (f(\hat{x}_j) - f(x_*)),$$

274 it follows that

$$275 \quad \phi(\hat{x}_j; \rho) = \rho f(\hat{x}_j) + \langle\langle c(\hat{x}_j) \rangle\rangle_M < \rho f(x_*) + \langle\langle c(x_*) \rangle\rangle_M = \phi(x_*; \rho),$$

276 contradicting that x_* is a minimizer of $\phi(\cdot; \rho)$ in $\mathcal{B}(x_*, \theta)$. We may conclude that x_*
 277 is a minimizer of $\langle\langle c(\cdot) \rangle\rangle_M$ in $\mathcal{B}(x_*, \theta)$, i.e., it is a local minimizer of $\langle\langle c(\cdot) \rangle\rangle_M$. \square

278 **LEMMA 3.** *Suppose the functions f and c_i for all $i \in \mathcal{I}$ are continuously differ-*
 279 *entiable, and let $x_* \in \mathbb{R}^n$ be a local minimizer of (P). Furthermore, suppose that,*
 280 *for each $\mathcal{S} \in \mathfrak{S}_M(x_*; 0)$, a constraint qualification holds for the optimization problem*
 281 *$\min_{x \in \mathbb{R}^n} f(x)$ such that $c_{\mathcal{S}}(x) \leq 0$ in that there exists $\bar{\rho}_{\mathcal{S}} > 0$ such that x_* is a local*
 282 *minimizer of $\phi_{\mathcal{S}}(\cdot; \rho)$ for all $\rho \in (0, \bar{\rho}_{\mathcal{S}}]$. Then, there exists $\bar{\rho} > 0$ such that x_* is a*
 283 *local minimizer of $\phi(\cdot; \rho)$ for all $\rho \in (0, \bar{\rho}]$.*

284 *Proof.* The proof follows easily for $\bar{\rho} = \min\{\bar{\rho}_{\mathcal{S}} : \mathcal{S} \in \mathfrak{S}_M(x_*, 0)\} > 0$. \square

285 An example of a constraint qualification that implies the conclusion of Lemma 3 is
 286 the linear independence constraint qualification (LICQ); however, many other less
 287 restrictive constraint qualifications are also sufficient [3, 39].

288 **3. Algorithm Description.** Our algorithm is inspired by sequential quadratic
 289 optimization (commonly known as SQP) methods for solving nonlinear optimization

290 problems that use a trust region (TR) mechanism for promoting global convergence.
 291 In particular, following standard penalty-SQP techniques [21, 22], in this section we
 292 describe a sequential method in which trial steps are computed through piecewise-
 293 quadratic penalty function models. The subproblems in this approach have quadratic
 294 objective functions and linear cardinality constraints that can be solved with tailored
 295 methods such as those proposed in [34, 35, 36].

296 We focus here on a method that employs a fixed penalty parameter $\rho \in (0, \infty)$ and
 297 a fixed $\epsilon \in (0, \infty)$ to define sets of scenario selections as in (7). It is this method with
 298 fixed (ρ, ϵ) for which we provide global convergence guarantees in §4. To simplify our
 299 notation here and in §4, we drop ρ from function expressions in which they appear;
 300 e.g., we use $\phi(\cdot) \equiv \phi(\cdot; \rho)$, $\phi_{\mathcal{S}}(\cdot) \equiv \phi_{\mathcal{S}}(\cdot; \rho)$, etc.

301 Let $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be the gradient function of $f(\cdot)$ and, for a given $\mathcal{S} \subseteq \mathcal{I}$, let
 302 $\nabla c_{\mathcal{S}} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times (m|\mathcal{S}|)}$ be the transpose of the Jacobian function of $c_{\mathcal{S}}(\cdot)$. For a given
 303 iterate $x_k \in \mathbb{R}^n$, scenario selection $\mathcal{S} \subseteq \mathcal{I}$, and symmetric positive semidefinite matrix
 304 $H_k \in \mathbb{R}^{n \times n}$, we define convex piecewise-linear and piecewise-quadratic local models
 305 of $\phi_{\mathcal{S}}$ at x_k as the functions $l_{\mathcal{S},k} : \mathbb{R}^n \rightarrow \mathbb{R}$ and $q_{\mathcal{S},k} : \mathbb{R}^n \rightarrow \mathbb{R}$, respectively, where

$$306 \quad (12a) \quad l_{\mathcal{S},k}(d) := \rho(f(x_k) + \nabla f(x_k)^T d) + \|[c_{\mathcal{S}}(x_k) + \nabla c_{\mathcal{S}}(x_k)^T d]_+\|_1$$

$$307 \quad (12b) \quad \text{and } q_{\mathcal{S},k}(d) := l_{\mathcal{S},k}(d) + \frac{1}{2}d^T H_k d.$$

309 We then define the reductions in these models corresponding to a given $d \in \mathbb{R}^n$ as

$$310 \quad \Delta l_{\mathcal{S},k}(d) := l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(d) \quad \text{and} \quad \Delta q_{\mathcal{S},k}(d) := q_{\mathcal{S},k}(0) - q_{\mathcal{S},k}(d).$$

311 Within a standard TR penalty-SQP methodology, a trial step toward minimizing
 312 $\phi_{\mathcal{S}}(\cdot)$ from x_k is computed as a minimizer of $q_{\mathcal{S},k}(\cdot)$ within a bounded region. Ex-
 313 tending this for the minimization of ϕ (recall (8)), it is natural to consider a trial step
 314 as a minimizer of a local quadratic-linear model of ϕ , defined as $q_k : \mathbb{R}^n \rightarrow \mathbb{R}$ where

$$315 \quad (13) \quad q_k(d) := \min_{\mathcal{S} \in \mathfrak{S}_M} q_{\mathcal{S},k}(d_k).$$

316 A method based around this idea is stated formally as Algorithm 1 below. As is
 317 common in TR methods, the ratio r_k , which essentially measures the accuracy of the
 318 predicted progress in the penalty function, is used to decide whether a trial point
 319 should be accepted and how the trust-region radius should be updated.

320 A critically important feature of Algorithm 1 is that the scenario selections $\mathfrak{S}_{M,k}$
 321 considered in subproblem (14) do not need to contain all possible scenario selections
 322 in \mathfrak{S}_M , in contrast to what (13) suggests. Choosing a smaller subset makes it poss-
 323 ible to reduce the computational effort of the step computation, which is the most
 324 computationally-intensive subroutine of the algorithm. In particular, it requires a
 325 solution of a cardinality-constrained problem (with linear constraints and a quadratic
 326 objective). For example, if $\mathfrak{S}_{M,k} = \mathfrak{S}_M(x_k; \epsilon)$, then subproblem (14) is equivalent to

$$327 \quad (18) \quad \begin{aligned} & \min_{\substack{d \in \mathbb{R}^n \\ s_i \in \mathbb{R}^m \\ \forall i \in \mathcal{N}_k \cup \mathcal{C}_k}} \rho(f(x_k) + \nabla f(x_k)^T d) + \frac{1}{2}d^T H_k d + \sum_{i \in \mathcal{C}_k \cup \mathcal{N}_k} e_m^T s_i \\ & \text{s.t.} \quad \begin{cases} |\{i \in \mathcal{C}_k : c_i(x_k) + \nabla c_i(x_k)^T d \leq s_i\}| \geq M - |\mathcal{N}_k|, \\ c_i(x_k) + \nabla c_i(x_k)^T d \leq s_i \quad \text{for all } i \in \mathcal{N}_k, \\ -\delta_k e_n \leq d \leq \delta_k e_n, \\ s_i \geq 0 \quad \text{for all } i \in \mathcal{C}_k \cup \mathcal{N}_k, \end{cases} \end{aligned}$$

Algorithm 1 SQP Algorithm for Problem (P) (for fixed (ρ, ϵ))

Require: penalty parameter $\rho \in (0, \infty)$, criticality parameter $\epsilon \in (0, \infty)$, initial point $x_0 \in \mathbb{R}^n$, initial TR radius $\delta_0 \in (0, \infty)$, sufficient decrease constant $\mu \in (0, \infty)$, TR norm $\|\cdot\|$, and TR update constants $\beta_1 \in (1, \infty)$, $\beta_2 \in (0, 1)$, and $\delta_{\text{reset}} \in (0, \delta_0]$

- 1: **for** $k \in \mathbb{N} := \{0, 1, 2, \dots\}$ **do**
- 2: Choose $\mathfrak{S}_{M,k} \supseteq \mathfrak{S}_M(x_k; \epsilon)$.
- 3: Compute a trial step d_k by solving the TR subproblem

$$(14) \quad \min_{d \in \mathbb{R}^n} \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d) \quad \text{s.t.} \quad \|d\| \leq \delta_k.$$

- 4: Compute the actual-to-prediction reduction ratio

$$(15) \quad r_k \leftarrow \frac{\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k + d_k)}{\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k)}.$$

- 5: Update the iterate and trust region radius by

$$(16) \quad x_{k+1} \leftarrow \begin{cases} x_k + d_k & \text{if } r_k \geq \mu \\ x_k & \text{otherwise} \end{cases}$$

$$(17) \quad \text{and } \delta_{k+1} \leftarrow \begin{cases} \max\{\beta_1 \delta_k, \delta_{\text{reset}}\} & \text{if } r_k \geq \mu \\ \beta_2 \|d_k\| & \text{otherwise.} \end{cases}$$

- 6: **end for**
-

328 where $(e_m, e_n) \in \mathbb{R}^m \times \mathbb{R}^n$ are vectors of ones, $\mathcal{N}_k := \mathcal{N}(x_k, \epsilon)$, and $\mathcal{C}_k := \mathcal{C}(x_k, \epsilon)$.
329 Clearly, the complexity of solving this problem depends on $|\mathcal{C}(x_k; \epsilon)|$.

330 This being said, to ensure convergence, it is necessary to include in $\mathfrak{S}_{M,k}$ at least
331 all ϵ -critical scenario selections $\mathfrak{S}_M(x_k; \epsilon)$ for some fixed $\epsilon > 0$, since otherwise the
332 algorithm might have a myopic view of the feasible region around x_k , causing it to
333 ignore constraints that are asymptotically relevant to characterize the local behavior
334 of ϕ at a limit point x_{lim} . More precisely, consider a situation in which there is a
335 constraint $i \in \mathcal{I}$ that has $v(c(x_k), i) > v_M(x_k)$ for all $k \in \mathbb{N}$, but $v(c(x_{\text{lim}}), i) =$
336 $v_M(x_{\text{lim}})$. In such a situation, one finds $i \notin \mathcal{C}(x_k, 0)$ for all $k \in \mathbb{N}$. Therefore, no
337 $\mathcal{S} \in \mathfrak{S}_M(x_k; 0)$ includes i , meaning that the method with $\epsilon = 0$ would compute trial
338 steps via (14) that completely ignore the constraint function c_i . This is problematic
339 since, at x_{lim} , one finds $i \in \mathcal{C}(x_{\text{lim}}, 0)$. Recalling Lemma 1, we see that the constraint
340 function c_i needs to be considered to draw conclusions about the relevance of x_* as a
341 minimizer—or even a stationary point (see (20) below)—of the penalty function ϕ .

342 Overall, the parameter $\epsilon > 0$ plays a critical role. If ϵ is large, then the method
343 considers a good approximation of q_k (and, hence, of ϕ) when computing each trial
344 step, but the computational cost of computing each step might be large. On the other
345 hand, if ϵ is small (near zero), then the objective function in (14) might be a poor
346 approximation of q_k (and ϕ). In such cases, since only a small subset of scenario
347 selections is considered in (14), the method might be unable to “see” local minima
348 of ϕ that correspond to scenario selections that are ignored, meaning that it might
349 converge to an inferior local minimum, even if better ones are “close by”. In the
350 remainder of this section, we shall see that our analysis holds for any $\epsilon > 0$, but we
351 will explore the crucial practical balance in the choice of ϵ in §5.2.

352 **4. Analysis.** The main purpose of this section is to prove a global convergence
 353 result for Algorithm 1 in terms of driving a stationarity measure for the (nonconvex
 354 and nonsmooth) penalty function ϕ to zero. We also provide some commentary on
 355 the usefulness of our convergence result, particularly as it relates to the potential
 356 convergence of the method to a “poor” local minimizer; see §4.2.

357 Our analysis generally follows that for the penalty-SQP method in [10], which we
 358 have extended to account for our unique constraint violation measure. We stress that
 359 the analysis is not straightforward, especially since—in contrast to standard penalty-
 360 SQP—even our linear and quadratic models of the penalty function are nonconvex.

361 **4.1. Global convergence.** We present our analysis under the following assump-
 362 tion. In this assumption and throughout the remainder of the section, $\|\cdot\|$ refers to
 363 the trust region norm used in Algorithm 1.

364 **ASSUMPTION 4.** *The problem functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $c_i : \mathbb{R}^n \rightarrow \mathbb{R}^m$ for all*
 365 *$i \in \mathcal{I}$ are Lipschitz continuous with Lipschitz continuous first-order derivatives over*
 366 *a bounded convex set whose interior contains the closure of the iterate sequence $\{x_k\}$.*
 367 *In addition, the sequence $\{H_k\}$ is bounded in norm in the sense that there exists a*
 368 *scalar $H_{\max} > 0$ such that $d^T H_k d \leq H_{\max} \|d\|^2$ for all k and any $d \in \mathbb{R}^n$.*

369 To state the global convergence theorem that we prove, we first need to define a
 370 valid stationarity measure for ϕ . In order to do this, let us first draw from standard
 371 exact penalty function theory to define a valid measure for $\phi_{\mathcal{S}}$ for a given $\mathcal{S} \subseteq \mathcal{I}$.
 372 Similar to our piecewise-linear model $l_{\mathcal{S},k}$ for $\phi_{\mathcal{S}}$ at x_k (recall (12a)), let us define a
 373 local piecewise-linear model of $\phi_{\mathcal{S}}$ at x as $l_{\mathcal{S}}(\cdot; x) : \mathbb{R}^n \rightarrow \mathbb{R}$, as defined by

$$374 \quad l_{\mathcal{S}}(d; x) = \rho(f(x) + \nabla f(x)^T d) + \|[c_{\mathcal{S}}(x) + \nabla c_{\mathcal{S}}(x)^T d]_+\|_1.$$

375 We denote the reduction in this model corresponding to $d \in \mathbb{R}^n$ as $\Delta l_{\mathcal{S}}(d; x) :=$
 376 $l_{\mathcal{S}}(0; x) - l_{\mathcal{S}}(d; x)$. Letting $d_{\mathcal{S}}^L(x)$ denote a minimizer of $l_{\mathcal{S}}(\cdot; x)$ within the $\|\cdot\|$ unit
 377 ball, or equivalently a maximizer of the reduction within the ball, i.e.,

$$378 \quad (19) \quad d_{\mathcal{S}}^L(x) \in \arg \max_{d \in \mathbb{R}^n} \{\Delta l_{\mathcal{S}}(d; x) : \|d\| \leq 1\},$$

379 we define the measure $\chi_{\mathcal{S}}(x) := \Delta l_{\mathcal{S}}(d_{\mathcal{S}}^L(x); x)$. The following lemma confirms that
 380 $\chi_{\mathcal{S}}(x)$ is a valid criticality measure for $\phi_{\mathcal{S}}$.

381 **LEMMA 5.** *Let $\mathcal{S} \subseteq \mathcal{I}$. Then, $\chi_{\mathcal{S}} : \mathbb{R}^n \rightarrow \mathbb{R}$ is continuous. In addition, one has*
 382 *$\chi_{\mathcal{S}}(x_*) = 0$ if and only if x_* is stationary for $\phi_{\mathcal{S}}$ in the sense that $0 \in \partial \phi_{\mathcal{S}}(x_*)$.*

383 *Proof.* See, e.g., Lemma 2.1 in [48]. □

384 In light of Lemmas 1 and 5, it is now natural to define a criticality measure for ϕ
 385 in terms of the criticality measures for $\phi_{\mathcal{S}}$ for each scenario selection $\mathcal{S} \in \mathfrak{S}_M(\cdot; 0)$.
 386 Specifically, for the penalty function ϕ , we define the measure $\chi : \mathbb{R}^n \rightarrow \mathbb{R}$ by

$$387 \quad (20) \quad \chi(x) := \max_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \chi_{\mathcal{S}}(x).$$

388 We now state our global convergence result.

389 **THEOREM 6.** *Under Assumption 4, one of the following outcomes will occur from*
 390 *any run of Algorithm 1 for given scalars $\rho > 0$ and $\epsilon > 0$:*

- 391 (i) $\chi(x_k) = 0$ for some (finite) $k \in \mathbb{N}$ or
- 392 (ii) the set $\mathcal{K} := \{k \in \mathbb{N} : r_k \geq \mu\}$ of successful iterations has infinite cardinality
 393 and any limit point x_* of $\{x_k\}$ has $\chi(x_*) = 0$.

394 REMARK 7. We recover a standard Sl_1 QP method [21] if all constraints are con-
 395 sidered, i.e., if $M = N$. In such a case, one finds that $\mathfrak{S}_M = \mathcal{I}$, (P) reduces to (RP),
 396 and Theorem 6 implies known convergence properties of a standard Sl_1 QP method.

397 We prove this theorem after proving a sequence of lemmas. A first observation is
 398 that the actual reduction and predicted reduction as defined in the ratio in Step 4 of
 399 Algorithm 1 are close for sufficiently small trial steps.

400 LEMMA 8. There exist $L_f > 0$ and $L_c > 0$ independent of $k \in \mathbb{N}$ such that

$$401 \quad \left| \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k + d_k) \right| \leq (\rho L_f + L_c + H_{\max}) \|d_k\|^2 \text{ for any } k.$$

402 *Proof.* Under Assumption 4, there exist Lipschitz constants $L_f > 0$ and $\tilde{L}_c > 0$
 403 independent of $k \in \mathbb{N}$ such that, for any $k \in \mathbb{N}$,

$$404 \quad \left| \rho(f(x_k) + \nabla f(x_k)^T d_k) + \frac{1}{2} d_k^T H_k d_k - \rho f(x_k + d_k) \right| \leq (\rho L_f + H_{\max}) \|d_k\|^2$$

$$405 \quad \text{and } \|c(x_k) + \nabla c(x_k)^T d_k - c(x_k + d_k)\| \leq \tilde{L}_c \|d_k\|^2.$$

407 Because the mapping $w_{\mathcal{S}} \mapsto \|w_{\mathcal{S}}\|_+$ is Lipschitz continuous for any $\mathcal{S} \subseteq \mathcal{I}$ and the
 408 pointwise minimum of a set of Lipschitz continuous functions is Lipschitz continuous,
 409 the mapping $w \mapsto \min_{\mathcal{S} \in \mathfrak{S}_M} \|w_{\mathcal{S}}\|_+$ is Lipschitz continuous. Thus, there exists a
 410 constant $L_c > 0$ independent of $k \in \mathbb{N}$ such that, for any $k \in \mathbb{N}$,

$$411 \quad \left| \left(\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k) \right) - \left(\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k + d_k) \right) \right|$$

$$= \left| \left(\rho(f(x_k) + \nabla f(x_k)^T d_k) + \frac{1}{2} d_k^T H_k d_k + \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \|c_{\mathcal{S}}(x_k) + \nabla c_{\mathcal{S}}(x_k)^T d_k\|_+ \right) \right.$$

$$\left. - \left(\rho f(x_k + d_k) + \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \|c_{\mathcal{S}}(x_k + d_k)\|_+ \right) \right| \leq (\rho L_f + L_c + H_{\max}) \|d_k\|^2,$$

412 which is the desired conclusion. \square

413 We now establish lower bounds for the reduction in the piecewise-quadratic model
 414 of the penalty functions corresponding to each trial step. Following standard trust
 415 region terminology, we quantify this reduction in terms of a *Cauchy point* from x_k
 416 corresponding to a model of $\phi_{\mathcal{S}}$ for each $\mathcal{S} \in \mathfrak{S}_M$. Specifically, we define the Cauchy
 417 point in iteration $k \in \mathbb{N}$ for a given scenario selection $\mathcal{S} \in \mathfrak{S}_M$ as $d_{\mathcal{S},k}^C := \alpha_{\mathcal{S},k}^C d_{\mathcal{S},k}^L$,
 418 where $d_{\mathcal{S},k}^L := d_{\mathcal{S}}^L(x_k)$ (recall (19)) and, for some $\alpha \in (0, 1)$ and $\eta \in (0, 1)$ that are
 419 fixed for the remainder of this section, the value $\alpha_{\mathcal{S},k}^C > 0$ is the largest element in
 420 the sequence $\{\alpha^j \min\{1, \delta_k / \|d_{\mathcal{S},k}^L\|\}\}_{j \in \mathbb{N}}$ such that

$$421 \quad (21) \quad \Delta q_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^C d_{\mathcal{S},k}^L) \geq \eta \Delta l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^C d_{\mathcal{S},k}^L).$$

422 Our next lemma reveals that the reduction the Cauchy point yields in the piecewise-
 423 quadratic model of $\phi_{\mathcal{S}}$ is proportional to the criticality measure $\chi_{\mathcal{S},k} := \chi_{\mathcal{S}}(x_k)$.

424 LEMMA 9. For any $k \in \mathbb{N}$ and $\mathcal{S} \in \mathfrak{S}_M$, it follows that

$$425 \quad (22) \quad \Delta q_{\mathcal{S},k}(d_{\mathcal{S},k}^C) \geq \eta \alpha_{\mathcal{S},k}^C \chi_{\mathcal{S},k}.$$

426 *Proof.* For any $k \in \mathbb{N}$ and $\mathcal{S} \in \mathfrak{S}_M$, convexity of $l_{\mathcal{S},k}$ ensures that, corresponding
 427 to any step $d \in \mathbb{R}^n$ and stepsize $\tau \in [0, 1]$, one has

$$428 \quad (23) \quad \Delta l_{\mathcal{S},k}(\tau d) = l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(\tau d) \geq \tau(l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(d)) = \tau \Delta l_{\mathcal{S},k}(d).$$

430 Hence, by the definitions of $d_{\mathcal{S},k}^{\mathcal{C}}$ and $\alpha_{\mathcal{S},k}^{\mathcal{C}}$ (specifically, (21)), it follows that

$$431 \quad \Delta q_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{C}}) \geq \eta \Delta l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}) \geq \eta \alpha_{\mathcal{S},k}^{\mathcal{C}} \Delta l_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{L}}) = \eta \alpha_{\mathcal{S},k}^{\mathcal{C}} \chi_{\mathcal{S},k},$$

432 as desired. \square

433 To accompany Lemma 9, we now establish a lower bound on the stepsize defining
 434 any Cauchy step. To do this, note that Assumption 4 ensures that each model $l_{\mathcal{S},k}$ is
 435 Lipschitz continuous with Lipschitz constant independent of k and \mathcal{S} ; in particular,
 436 under Assumption 4, there exists $L_l > 0$ such that, for any $k \in \mathbb{N}$ and $\mathcal{S} \in \mathfrak{S}_M$,

$$437 \quad (24) \quad \chi_{\mathcal{S},k} = \Delta l_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{L}}) = l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{L}}) \leq L_l \|d_{\mathcal{S},k}^{\mathcal{L}}\|.$$

438 We now prove the following result.

439 LEMMA 10. *For any $k \in \mathbb{N}$ and $\mathcal{S} \in \mathfrak{S}_M$, the Cauchy stepsize satisfies*

$$440 \quad (25) \quad \alpha_{\mathcal{S},k}^{\mathcal{C}} \geq \|d_{\mathcal{S},k}^{\mathcal{C}}\| \geq \min \left\{ \frac{\chi_{\mathcal{S},k}}{L_l}, \delta_k, \frac{2(1-\eta)\alpha\chi_{\mathcal{S},k}}{H_{\max}} \right\}.$$

441 *Proof.* The first inequality follows since (19) involves $\|d_{\mathcal{S},k}^{\mathcal{L}}\| \leq 1$, from which it
 442 follows that $\|d_{\mathcal{S},k}^{\mathcal{C}}\| = \alpha_{\mathcal{S},k}^{\mathcal{C}} \|d_{\mathcal{S},k}^{\mathcal{L}}\| \leq \alpha_{\mathcal{S},k}^{\mathcal{C}}$. To establish the second inequality, consider
 443 two cases. First, if $\alpha_{\mathcal{S},k}^{\mathcal{C}} = \min\{1, \delta_k / \|d_{\mathcal{S},k}^{\mathcal{L}}\|\}$ (i.e., if the Cauchy condition (21) is
 444 satisfied by the stepsize corresponding to $j = 0$), then (24) yields

$$445 \quad \|d_{\mathcal{S},k}^{\mathcal{C}}\| = \alpha_{\mathcal{S},k}^{\mathcal{C}} \|d_{\mathcal{S},k}^{\mathcal{L}}\| = \min \left\{ 1, \frac{\delta_k}{\|d_{\mathcal{S},k}^{\mathcal{L}}\|} \right\} \|d_{\mathcal{S},k}^{\mathcal{L}}\| \geq \min \left\{ \frac{\chi_{\mathcal{S},k}}{L_l}, \delta_k \right\},$$

446 as desired. Second, if $\alpha_{\mathcal{S},k}^{\mathcal{C}} < \min\{1, \delta_k / \|d_{\mathcal{S},k}^{\mathcal{L}}\|\}$, then the Cauchy condition (21) must
 447 be violated when $\alpha_{\mathcal{S},k}^{\mathcal{C}}$ is replaced by $\alpha_{\mathcal{S},k}^{\mathcal{C}}/\alpha$, from which it follows that

$$448 \quad \begin{aligned} & l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}/\alpha) - \frac{1}{2}(\alpha_{\mathcal{S},k}^{\mathcal{C}}/\alpha)^2 (d_{\mathcal{S},k}^{\mathcal{L}})^T H_k d_{\mathcal{S},k}^{\mathcal{L}} \\ &= \Delta q_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}/\alpha) < \eta \Delta l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}/\alpha) = \eta(l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}/\alpha)). \end{aligned}$$

449 This inequality and (23) then reveal that, under Assumption 4,

$$450 \quad \begin{aligned} & \frac{1}{2}(\alpha_{\mathcal{S},k}^{\mathcal{C}}/\alpha)^2 H_{\max} \|d_{\mathcal{S},k}^{\mathcal{L}}\|^2 \geq \frac{1}{2}(\alpha_{\mathcal{S},k}^{\mathcal{C}}/\alpha)^2 (d_{\mathcal{S},k}^{\mathcal{L}})^T H_k d_{\mathcal{S},k}^{\mathcal{L}} \\ & > (1-\eta)(l_{\mathcal{S},k}(0) - l_{\mathcal{S},k}(\alpha_{\mathcal{S},k}^{\mathcal{C}} d_{\mathcal{S},k}^{\mathcal{L}}/\alpha)) \geq (1-\eta)(\alpha_{\mathcal{S},k}^{\mathcal{C}}/\alpha) \Delta l_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{L}}). \end{aligned}$$

451 Since (19) involves $\|d_{\mathcal{S},k}^{\mathcal{L}}\| \leq 1$, it now follows by (24) that

$$452 \quad \|d_{\mathcal{S},k}^{\mathcal{C}}\| = \alpha_{\mathcal{S},k}^{\mathcal{C}} \|d_{\mathcal{S},k}^{\mathcal{L}}\| \geq \frac{2(1-\eta)\alpha\Delta l_{\mathcal{S},k}(d_{\mathcal{S},k}^{\mathcal{L}})}{H_{\max}} = \frac{2(1-\eta)\alpha\chi_{\mathcal{S},k}}{H_{\max}},$$

453 as desired. \square

454 The next lemma reveals that, in a sufficiently small neighborhood of any point
 455 $x \in \mathbb{R}^n$ representing either an element or a limit point of the iterate sequence $\{x_k\}$,
 456 the reduction obtained by the Cauchy point defined with respect to $\tilde{\mathcal{S}} \in \mathfrak{S}_M(x; 0)$
 457 in the piecewise-quadratic model of ϕ is sufficiently large whenever the trust region
 458 radius and criticality measure for $\phi_{\tilde{\mathcal{S}}}$ are sufficiently large.

459 LEMMA 11. Let $x \in \mathbb{R}^n$ be an element or a limit point of the sequence $\{x_k\}$, let
 460 $\tilde{\mathcal{S}} \in \mathfrak{S}_M(x; 0)$, and consider $\delta_{\min} > 0$ and $\chi_{\min} > 0$ such that

$$461 \quad (26) \quad \delta_{\min} \leq \min \left\{ \frac{\chi_{\min}}{L_l}, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\}.$$

462 Then, there exists a neighborhood \mathcal{X} of x such that

$$463 \quad \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S}, k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \geq \frac{1}{2}\eta\chi_{\min} \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_k, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\}$$

464 whenever $\delta_k \geq \delta_{\min}$, $x_k \in \mathcal{X}$, and $\chi_{\tilde{\mathcal{S}}, k} \geq \chi_{\min}$.

465 *Proof.* As in (8), it follows that $\min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x) = \phi_{\tilde{\mathcal{S}}}(x)$ since $\tilde{\mathcal{S}} \in \mathfrak{S}_M(x; 0)$.
 466 Hence, from continuity of both $\min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(\cdot)$ and $\phi_{\tilde{\mathcal{S}}}(\cdot)$, there exists a neighbor-
 467 hood \mathcal{X} of x such that, whenever $x_k \in \mathcal{X}$,

$$468 \quad (27) \quad \phi_{\tilde{\mathcal{S}}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x_k) \leq \frac{1}{2}\eta\delta_{\min}\chi_{\min}.$$

469 Now, under the conditions of the lemma, with $\delta_k \geq \delta_{\min}$, $x_k \in \mathcal{X}$, and $\chi_{\tilde{\mathcal{S}}, k} \geq \chi_{\min}$, it
 470 follows from Lemma 10 and (26) that $\alpha_{\tilde{\mathcal{S}}, k}^{\mathcal{C}} \geq \delta_{\min}$. One then obtains from (27) that

$$471 \quad \frac{1}{2}\eta\alpha_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}\chi_{\min} \geq \phi_{\tilde{\mathcal{S}}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x_k).$$

472 Thus, along with Lemmas 9 and 10,

$$\begin{aligned} 473 \quad & \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S}, k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \\ 474 \quad & \geq \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S}, k}(0) - q_{\tilde{\mathcal{S}}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \\ 475 \quad & = \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x_k) - q_{\tilde{\mathcal{S}}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \\ 476 \quad & = \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x_k) - \phi_{\tilde{\mathcal{S}}}(x_k) + \phi_{\tilde{\mathcal{S}}}(x_k) - q_{\tilde{\mathcal{S}}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \\ 477 \quad & \geq -\frac{1}{2}\eta\alpha_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}\chi_{\min} + \Delta q_{\tilde{\mathcal{S}}, k}(d_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}) \\ 478 \quad & \geq \frac{1}{2}\eta\alpha_{\tilde{\mathcal{S}}, k}^{\mathcal{C}}\chi_{\min} \\ 479 \quad & \geq \frac{1}{2}\eta\chi_{\min} \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_k, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\}, \\ 480 \end{aligned}$$

481 as desired. \square

482 We now prove that around any non-stationary point representing an iterate or a
 483 limit point of the iterate sequence, there exists a neighborhood such that the trust
 484 region radius must be set sufficiently large. For obtaining this result, a key role is
 485 played by the trust region reset value $\delta_{\text{reset}} > 0$.

486 LEMMA 12. Let $x \in \mathbb{R}^n$ be an element or a limit point of the sequence $\{x_k\}$ and
 487 suppose that $\chi(x) > 0$. In addition, let $\chi_{\min} := \frac{1}{2}\chi(x)$ and

$$488 \quad (28) \quad \delta_{\min} := \beta_2 \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_{\text{reset}}, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}}, \frac{(1-\mu)\eta\chi_{\min}}{2L} \right\}.$$

489 Then, there exists a neighborhood \mathcal{X} of x such that $x_k \in \mathcal{X}$ implies $\delta_k \geq \delta_{\min}$.

490 *Proof.* From (20) and the definition of $\chi_{\min} > 0$, there exists $\mathcal{S} \in \mathfrak{S}_M(x; 0)$
491 with $\chi_{\mathcal{S}}(x) = 2\chi_{\min}$. Thus, by continuity of $\chi_{\mathcal{S}}$ (recall Lemma 5), it follows that
492 $\chi_{\mathcal{S}}(x_k) \geq \chi_{\min} > 0$ for all x_k in a sufficiently small neighborhood \mathcal{X}_1 of x . One also
493 has by continuity of the constraint functions that there exists $\tilde{\epsilon} \in (0, \epsilon]$ such that
494 $\mathfrak{S}_M(x_k; \tilde{\epsilon}) = \mathfrak{S}_M(x; 0)$ for all x_k in a sufficiently small neighborhood $\mathcal{X}_2 \subseteq \mathcal{X}_1$ of x .
495 Since $\tilde{\epsilon} \in (0, \epsilon]$, it follows from (9) that $\mathfrak{S}_{M,k} \supseteq \mathfrak{S}_M(x_k; \epsilon) \supseteq \mathfrak{S}_M(x_k; \tilde{\epsilon}) = \mathfrak{S}_M(x; 0)$.
496 Since (28) implies (26), the result of Lemma 11 holds for some neighborhood
497 $\mathcal{X} \subseteq \mathcal{X}_2$. In particular, noting that d_k is the global minimizer of (14), $\|d_{\tilde{\mathcal{S}},k}^C\| \leq \delta_k$,
498 and $q_{\mathcal{S},k}(0) = \phi_{\mathcal{S}}(x_k)$, it follows that, for any $\tilde{\mathcal{S}} \in \mathfrak{S}_M(x; 0) \subseteq \mathfrak{S}_{M,k}$ with $\delta_k \geq \delta_{\min}$
499 and $x_k \in \mathcal{X}$,

$$\begin{aligned}
500 & \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k) \\
501 & \stackrel{(8)}{\geq} \phi_{\tilde{\mathcal{S}}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_{\tilde{\mathcal{S}},k}^C) \\
502 & \stackrel{(10)}{=} \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} \phi_{\mathcal{S}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_{\tilde{\mathcal{S}},k}^C) \\
503 & \geq \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S},k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_M(x; 0)} q_{\mathcal{S},k}(d_{\tilde{\mathcal{S}},k}^C) \\
504 \quad (29) & \geq \frac{1}{2}\eta\chi_{\min} \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_k, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\}. \\
505 &
\end{aligned}$$

506 The desired result can now be proved by contradiction. For this purpose, suppose
507 that for some $\tilde{k} \in \mathbb{N}$ with $x_{\tilde{k}} \in \mathcal{X}$ the algorithm has $\delta_{\tilde{k}} < \delta_{\min}$. Since $\delta_0 \geq \delta_{\text{reset}}$ and
508 the trust region radius is reset to at least δ_{reset} after each accepted trial step, it follows
509 from the fact that $\delta_{\min} < \delta_{\text{reset}}$ that $\tilde{k} > 0$ and there must be some $k \in \{0, \dots, \tilde{k} - 1\}$
510 with $x_k = x_{\tilde{k}}$ and

$$511 \quad (30) \quad \delta_{\min}/\beta_2 \geq \delta_k \geq \delta_{\min}$$

512 where d_k was rejected. However, since (28) and (30) imply that

$$513 \quad \delta_k \leq \min \left\{ \frac{\chi_{\min}}{L_l}, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}}, \frac{(1-\mu)\eta\chi_{\min}}{2(\rho L_f + L_c + H_{\max})} \right\},$$

514 it follows with (29) and Lemma 8 that

$$\begin{aligned}
515 \quad 1 - r_k &= 1 - \frac{\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k + d_k)}{\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k)} \\
516 &\leq \frac{|\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k + d_k) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} \phi_{\mathcal{S}}(x_k)|}{\min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(0) - \min_{\mathcal{S} \in \mathfrak{S}_{M,k}} q_{\mathcal{S},k}(d_k)} \\
517 &\leq \frac{2(\rho L_f + L_c + H_{\max})\|d_k\|^2}{\eta\chi_{\min} \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_k, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\}} \\
518 &\leq \frac{2(\rho L_f + L_c + H_{\max})\delta_k^2}{\eta\chi_{\min}\delta_k} = \frac{2(\rho L_f + L_c + H_{\max})\delta_k}{\eta\chi_{\min}} \leq 1 - \mu, \\
519 &
\end{aligned}$$

520 contradicting the assertion that d_k was rejected. Hence, no such $\tilde{k} \in \mathbb{N}$ with $x_{\tilde{k}} \in \mathcal{X}$
521 and $\delta_{\tilde{k}} < \delta_{\min}$ may exist, meaning that $x_k \in \mathcal{X}$ implies $\delta_k \geq \delta_{\min}$, as desired. \square

522 We now prove that if the number of accepted steps is finite, then the algorithm
523 must have arrived at a point with the stationarity measure for ϕ equal to zero.

LEMMA 13. If $\mathcal{K} := \{k \in \mathbb{N} : r_k \geq \mu\}$ has $|\mathcal{K}| < \infty$, then $x_k = x_*$ for some $x_* \in \mathbb{R}^n$ for all sufficiently large $k \in \mathbb{N}$ where x_* is stationary for ϕ in that $\chi(x_*) = 0$.

Proof. If the iteration index set \mathcal{K} is finite, then the iterate update (16) ensures that $x_k = x_*$ for all $k \geq k_*$ for some $x_* \in \mathbb{R}^n$ and $k_* \in \mathbb{N}$. Hence, $\chi(x_k) = \chi(x_*)$ for all $k \geq k_*$. If $\chi(x_*) = 0$, then the desired result holds. Otherwise, if $\chi(x_*) > 0$, then Lemma 12 implies the existence of $\delta_{\min} > 0$ such that $\delta_k \geq \delta_{\min}$ for all $k \geq k_*$. However, this contradicts the fact that $|\mathcal{K}| < \infty$ and (17) ensure $\{\delta_k\} \searrow 0$. \square

We are now ready to prove our global convergence theorem.

Proof (of Theorem 6). If $\mathcal{K} := \{k \in \mathbb{N} : r_k \geq \mu\}$ has $|\mathcal{K}| < \infty$, then Lemma 13 implies that $\chi(x_k) = 0$ for some finite $k \in \mathbb{N}$, which is represented by outcome (i). Thus, for the remainder of the proof, suppose that $|\mathcal{K}| = \infty$. For the purpose of deriving a contradiction, suppose that there exists a limit point x_* of the iterate sequence $\{x_k\}$ that has $\chi(x_*) > 0$. Let $\{x_{k_i}\}_{i \in \mathbb{N}}$ be an infinite subsequence of $\{x_k\}$ that converges to x_* . Without loss of generality, it can be assumed that $k_i \in \mathcal{K}$ for all $i \in \mathbb{N}$, i.e., that all steps from the elements of $\{x_{k_i}\}$ are successful in that $x_{k_i+1} = x_{k_i} + d_{k_i}$ for all $i \in \mathbb{N}$. In addition, along the lines of Lemma 12, let $\chi_{\min} := \frac{1}{2}\chi(x_*)$ and δ_{\min} be defined as in (28). Since $\{x_{k_i}\}$ converges to x_* , it can be assumed without loss of generality that $x_{k_i} \in \mathcal{X}$ for all $i \in \mathbb{N}$ with \mathcal{X} defined as in Lemma 12, from which it follows that $\delta_{k_i} \geq \delta_{\min}$.

As in the proof of Lemma 12 (recall (29)), it follows by (28) that

$$\begin{aligned} \min_{S \in \mathfrak{S}_{M,k_i}} q_{S,k_i}(0) - \min_{S \in \mathfrak{S}_{M,k_i}} q_{S,k_i}(d_{k_i}) &\geq \frac{1}{2}\eta\chi_{\min} \min \left\{ \frac{\chi_{\min}}{L_l}, \delta_{k_i}, \frac{2(1-\eta)\alpha\chi_{\min}}{H_{\max}} \right\} \\ &\geq \frac{1}{2}\eta\chi_{\min}\delta_{\min}. \end{aligned} \tag{31}$$

On the other hand, using (8) and (10), one has for all $k \in \mathbb{N}$ that

$$\begin{aligned} \phi(x_k) &= \min_{S \in \mathfrak{S}_M} \phi_S(x_k) = \min_{S \in \mathfrak{S}_{M,k}} \phi_S(x_k) \\ \text{and } \phi(x_k + d_k) &= \min_{S \in \mathfrak{S}_M} \phi_S(x_k + d_k) \leq \min_{S \in \mathfrak{S}_{M,k}} \phi_S(x_k + d_k). \end{aligned}$$

Consequently, for any $k \in \mathcal{K}$ it follows that $r_k \geq \mu$ and, hence,

$$\begin{aligned} \phi(x_k) - \phi(x_{k+1}) &\geq \min_{S \in \mathfrak{S}_{M,k}} \phi_S(x_k) - \min_{S \in \mathfrak{S}_{M,k}} \phi_S(x_k + d_k) \\ &\geq \mu \left(\min_{S \in \mathfrak{S}_{M,k}} q_{k,S}(0) - \min_{S \in \mathfrak{S}_{M,k}} q_{k,S}(d_k) \right) \geq 0. \end{aligned}$$

For the subsequence $\{x_{k_i}\}$ we can strengthen this, using (31), to

$$\phi(x_{k_i}) - \phi(x_{k_i+1}) \geq \frac{1}{2}\eta\mu\chi_{\min}\delta_{\min}.$$

Summing this inequality over all $k \in \mathbb{N}$ and noting that $\phi(x_k) - \phi(x_{k+1}) = 0$ for each unsuccessful $k \in \mathbb{N} \setminus \mathcal{K}$ implies that $\{\phi(x_k)\} \searrow -\infty$. However, this contradicts Assumption 4. Consequently, a limit point x_* with $\chi(x_*) > 0$ cannot exist. \square

4.2. Avoiding “poor” local minimizers. The analysis presented in the previous subsection concerns convergence to a local minimizer (or at least a stationary point) of the penalty function ϕ , which under nice conditions (recall Lemmas 2 and 3) corresponds to a local minimizer (or stationary point) of the cardinality-constrained problem (P). It is important to note, however, that such convergence is of little

561 meaning if the algorithm is likely to get attracted to a “poor” local minimizer. In our
 562 setting of cardinality-constrained optimization, this is of particular concern since the
 563 feasible region of such a problem is often very jagged; see, e.g., Figure 1 for the two-
 564 dimensional example studied in §5.2. As a consequence, these problems have many
 565 local minimizers (in fact, increasingly more if N grows, such as in SAA approxima-
 566 tions of chance-constrained problems), many of which can be considered “poor” in
 567 the sense that there are better local minima in a small neighborhood.

568 Fortunately, there is good reason to believe that our method will not get trapped
 569 at a local minimizer that is particularly poor. The important feature that guarantees
 570 this desirable behavior is that our method employs subproblem (18), which is itself a
 571 cardinality-constrained problem involving local Taylor models of the original problem
 572 functions, meaning that it inherits the jagged structure of problem. This allows our
 573 method to, at least locally, be aware of better local minimizers that are nearby.

574 To make this claim more precise, suppose that the algorithm has reached an iterate
 575 $x_k \in \mathbb{R}^n$ such that subproblem (14) with $\mathfrak{S}_{M,k} = \mathfrak{S}_M$ yields $d_k = 0$ for any sufficiently
 576 small $\delta_k > 0$. In addition, suppose that there exists another local minimizer $x \in \mathbb{R}^n$ of
 577 the penalty function such that $\phi(x_k) > \phi(x)$. Let the distance between the minimizers
 578 be $\delta := \|x - x_k\|$. By the same reasoning as in the proof of Lemma 8, it follows that

$$579 \quad \begin{aligned} & \left| \min_{\mathcal{S} \in \mathfrak{S}_M} q_{\mathcal{S},k}(x - x_k) - \min_{\mathcal{S} \in \mathfrak{S}_M} \phi_{\mathcal{S}}(x) \right| \leq (\rho L_f + L_c + H_{\max}) \delta^2 \\ \implies & \quad \phi(x) \geq \min_{\mathcal{S} \in \mathfrak{S}_M} q_{\mathcal{S},k}(x - x_k) - (\rho L_f + L_c + H_{\max}) \delta^2. \end{aligned}$$

580 In other words, the penalty function value at x cannot be less than that predicted
 581 by the quadratic model at x_k minus a term involving only the penalty parameter,
 582 Lipschitz constants of f and $\{c_i\}$, the upper bound on the Hessian, and the distance
 583 between x_k and x . If this term is small in that $\rho L_f + L_c + H_{\max}$ and/or δ is small,
 584 then a step from x_k to x yields a predicted reduction that is sufficiently close to the
 585 actual reduction $\phi(x_k) - \phi(x)$, meaning that our algorithm would accept such a step.

586 There are few observations to make as a result of this discussion. First, we have
 587 argued that our algorithm will not get “stuck” due to a wrong scenario selection.
 588 This can be seen in the fact that any discrepancy between the actual and prediction
 589 reduction in our penalty function is independent of the particular $\mathcal{S} \in \mathfrak{S}_M$ that
 590 minimizes the constraint violation. Second, we have shown that if a better local
 591 minimizer is nearby, then our algorithm would compute and accept a step toward
 592 such a point unless the problem functions are highly nonlinear (i.e., if L_f and/or L_c is
 593 large) and/or if H_{\max} is large. Finally, if the problem functions are linear, $H_{\max} = 0$,
 594 and our algorithm considers a sufficiently large trust region radius, then our algorithm
 595 will compute a global minimizer of the penalty function.

596 **5. Numerical Experiments.** We now describe a prototype implementation of
 597 Algorithm 1 that we have written in `Matlab B2015b`, pose two interesting test prob-
 598 lems, and present the results of numerical experiments when the implementation is
 599 tasked to solve various instances of the problems. The primary purpose of this inves-
 600 tigation is to demonstrate that, while Algorithm 1 has only the convergence guaran-
 601 tees of Theorem 6, in practice it has the potential to generate high-quality solutions
 602 to realistic problems. We also illustrate that the algorithm is flexible and allows for
 603 trade-offs between speed and quality of the solution through the tuning of algorithmic
 604 parameters. All computations were executed on a Ubuntu 12.04 Linux workstation
 605 with 256GB RAM and Intel Xeon CPU with 20 cores, running at 3.10GHz.

606 **5.1. Implementation.** Our implementation solves general instances of the form

607 (Q)
$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t.} \quad |\{i \in \mathcal{I} : c_i(x) \leq 0\}| \geq M, \quad \bar{c}(x) \leq 0,$$

608 where the objective and cardinality constraint quantities are defined as in problem (P)
 609 while $\bar{c} : \mathbb{R}^n \rightarrow \mathbb{R}^{\bar{m}}$ represents an additional inequality constraint function. These
 610 additional constraints are handled through the addition of a standard ℓ_1 -norm penalty
 611 term of the form $\|[\bar{c}(x)]_+\|_1$ in the definition of $\phi_{\mathcal{S}}$ in (5). The local models (12) and all
 612 related quantities are modified accordingly. (For ease of exposition, we continue to use
 613 the notation and labels of all of these quantities with these modifications presumed.)
 614 One can verify that an analysis as in §4 still applies.

615 Each iteration requires the solution of (14), which, using an ℓ_∞ -norm trust region,
 616 can equivalently be rewritten in terms of a cardinality-constrained problem with linear
 617 functions in the constraints and a quadratic objective function. Specifically, with
 618 $e_m \in \mathbb{R}^m$, $e_{\bar{m}} \in \mathbb{R}^{\bar{m}}$, and $e_n \in \mathbb{R}^n$ representing vectors of ones, \mathcal{N}_k representing
 619 indices of constraints to be enforced, and \mathcal{C}_k representing indices of constraints *that*
 620 *might be enforced* (such that $\mathfrak{S}_{M,k} = \mathcal{N}_k \cup \mathcal{C}_k$), our implementation considers

$$\begin{aligned}
 & \min_{\substack{d \in \mathbb{R}^n \\ t \in \mathbb{R}^{\bar{m}} \\ s_i \in \mathbb{R}^m \\ \forall i \in \mathcal{N}_k \cup \mathcal{C}_k}} \rho \nabla f(x)^T d + \frac{1}{2} d^T H_k d + \sum_{i \in \mathcal{C}_k \cup \mathcal{N}_k} e_m^T s_i + e_{\bar{m}}^T t \\
 621 \quad (32) \quad & \text{s.t.} \quad \left\{ \begin{array}{l} |\{i \in \mathcal{C}_k : c_i(x_k) + \nabla c_i(x_k)^T d \leq s_i\}| \geq M - |\mathcal{N}_k|, \\ c_i(x_k) + \nabla c_i(x_k)^T d \leq s_i \quad \text{for all } i \in \mathcal{N}_k, \\ \bar{c}(x_k) + \nabla \bar{c}(x_k)^T d \leq t, \\ -\delta_k e_n \leq d \leq \delta_k e_n, \\ s_i \geq 0 \quad \text{for all } i \in \mathcal{N}_k \cup \mathcal{C}_k, \\ t \geq 0. \end{array} \right.
 \end{aligned}$$

622 Further details about our method for choosing \mathcal{N}_k and \mathcal{C}_k are given at the end of
 623 this subsection. To solve subproblem (32), the cardinality constraint is replaced using
 624 a “big-M” approach, leading to the mixed-integer constraint set

$$\begin{aligned}
 & c_i(x_k) + \nabla c_i(x_k)^T d \leq s_i + \bar{M}(1 - z_i) \quad \text{for all } i \in \mathcal{C}_k, \\
 625 \quad (33) \quad & \sum_{i \in \mathcal{C}_k} z_i = M - |\mathcal{N}_k|, \\
 & z_i \in \{0, 1\} \quad \text{for all } i \in \mathcal{C}_k.
 \end{aligned}$$

626 Here, for each $i \in \mathcal{C}_k$, the binary variable z_i indicates whether the constraint corre-
 627 sponding to the index i is enforced and the parameter \bar{M} is chosen large enough so that
 628 the constraint is inactive at the optimal solution of (32) whenever $z_i = 0$. We stress
 629 that this “big-M” approach, while valid, likely leads to vastly inferior performance
 630 compared to a specialized branch-and-cut decomposition algorithm for solving these
 631 types of problems, such as the one proposed by Luedtke [34]. However, since adopting
 632 such a specialized approach would require a significant programming effort beyond the
 633 scope of this paper, our implementation simply employs the “big-M” approach above,
 634 invoking `Cplex` (version 12.6.2) to solve the resulting subproblem.

635 For the quadratic objective term in (32), our implementation uses

$$636 \quad H_k = \rho \nabla^2 f(x_k) + \sum_{i=1}^N \sum_{j=1}^m [\lambda_k^c]_{ij} \nabla^2 c_{ij}(x_k) + \sum_{j=1}^{\bar{m}} [\lambda_k^{\bar{c}}]_j \nabla^2 \bar{c}_j(x_k) + (\zeta + 10^{-8})I.$$

637 The quantities involved in this expression require some explanation. Corresponding
 638 to the constraint functions c_{ij} (i.e., the j th element of the vector function c_i) and \bar{c}_j
 639 (i.e., the j th element of the vector function \bar{c}), the quantities $[\lambda_k^c]_{ij}$ and $[\lambda_k^{\bar{c}}]_j$ can be
 640 interpreted as Lagrange multipliers. Thus, the first three terms defining H_k can be
 641 viewed as the Hessian of the Lagrangian corresponding to problem (Q) involving all
 642 elements of the cardinality constraints. In all iterations after the first one—the choice
 643 of the initial values is described below—the values for these Lagrange multipliers are
 644 set to the optimal multipliers for the constraints in (32)–(33) when this subproblem
 645 is solved at the previous iterate with all binary variables fixed at their optimal values.
 646 For any $i \notin \mathfrak{S}_{M,k}$ and for any $i \in \mathcal{C}_k$ such that the optimal binary variable is $z_i^* = 0$,
 647 the corresponding multipliers are set to zero. As for the scalar $\zeta > 0$, it is chosen
 648 to ensure that H_k is positive definite so that the objective of (32)–(33) is convex.
 649 Specifically, its value is chosen using “Algorithm IC (Inertia Correction)” in [47] with
 650 the parameters $\bar{\delta}_w^0 = 10^{-4}$, $\bar{\delta}_w^{\min} = 10^{-12}$, $\bar{\delta}_w^{\max} = 10^{10}$, $\bar{\kappa}_w^+ = \kappa_w^+ = 8$, and $\bar{\kappa}_w^- = 1/3$.
 651 This method first checks if H_k is positive definite with $\zeta = 0$. If this is the case,
 652 then the choice $\zeta = 0$ is used; otherwise, increasing positive values of ζ are tried
 653 until H_k is positive definite. In our implementation, to determine whether a trial
 654 matrix is positive definite, the smallest eigenvalue of H_k is computed using Matlab’s
 655 built-in `eigs` function. The small shift of the eigenvalues by 10^{-8} (regardless of the
 656 computed value for ζ) is included as a regularization term since it made the numerical
 657 performance of the `Cplex` solver more stable in our experiments.

658 The values for the parameters in Algorithm 1 that we employ in our implemen-
 659 tation are $\rho = 0.01$, $\epsilon = 10^{-3}$, $\delta_0 = 1000$, $\mu = 10^{-8}$, and $\beta_1 = \beta_2 = 10^{-8}$. The
 660 remaining parameters are given in subsequent subsections for each experiment. The
 661 algorithm terminates when $\|d_k\|_\infty \leq 10^{-6}$. We also note that, to account for numer-
 662 ical inaccuracies in the subproblem solutions, our implementation relaxes the ratio
 663 test in (15) by adding 10^{-8} to both the numerator and denominator.

664 Significant tuning of the `Cplex` solver was required to achieve reliably perfor-
 665 mance, both when solving instances of the mixed-integer QP (32)–(33) and the related
 666 QP for computing the Lagrange multipliers in the definition of H_k . Feasibility and
 667 optimality tolerances were decreased to 10^{-8} and the `emphasis.numerical` option
 668 was set to 1, asking `Cplex` to emphasize “extreme numerical caution”. When solving
 669 the QP for computing Lagrange multipliers, we set the `solutiontarget` option to 1,
 670 thereby telling `Cplex` that we are seeking an optimal solution to a convex QP.

671 When solving (32)–(33), we used the dual simplex solver with integrality gap
 672 tolerance set to 10^{-7} and absolute optimality gap tolerance set to 10^{-8} . Furthermore,
 673 we specified that the interior point QP solver is used for solving the root node in
 674 the branch-and-bound methods for the MIQP. This was necessary because otherwise
 675 `Cplex` claimed in some instances that the root node was infeasible (which, in theory,
 676 is not possible due to the presence of the slack variables). We used `Cplex`’s `MipStart`
 677 feature to provide the zero-step (setting $d = 0$, and choosing the remaining variables
 678 to minimize the objective in (32)–(33)) as a feasible incumbent. We imposed a time
 679 limit of 300 seconds for each subproblem. If the time limit was exceeded when solving
 680 an instance of (32)–(33), then we took the final incumbent solution as the step d_k .
 681 In some cases, this incumbent did not lead to a reduction in the model, and the run

682 was reported as a failure (even though this says nothing about the performance of
 683 Algorithm 1, except that the subproblem was numerically difficult to solve).

684 Finally, we recall the critical role in Algorithm 1 played by the set $\mathfrak{S}_{M,k}$, which
 685 must satisfy $\mathfrak{S}_{M,k} \supseteq \mathfrak{S}_M(x_k; \epsilon)$. As expected, in our implementation and experiments,
 686 the manner in which this set is chosen offers a trade-off between speed and quality
 687 of the ultimate solution found. If one were to set $\mathfrak{S}_{M,k} = \mathfrak{S}_M(x_k; \epsilon)$ for a relatively
 688 small ϵ (as would typically describe our default value of $\epsilon = 10^{-3}$), then one would
 689 often find the subproblem solved relatively quickly due to the smaller numbers of ele-
 690 ments in the cardinality constraints (and, hence, smaller number of binary variables)
 691 in the generated instances of (32)–(33). On the other hand, if one were to choose
 692 $\mathfrak{S}_{M,k} = \mathcal{I}$ (corresponding to $\epsilon = \infty$), then one might find that the algorithm attains
 693 a better quality solution at the expense of a *much* higher computing time. Since one
 694 of the goals of our experiments is to illustrate this trade-off, our implementation em-
 695 ploys a fractional scenario selection parameter $\gamma \in [0, 1]$ in the choice of \mathcal{C}_k which, in
 696 contrast to ϵ , is not dependent on the scale of the constraint function c . Specifically,
 697 with $M^- := \max\{1, M - \lceil \gamma N \rceil\}$ and $M^+ := \min\{N, M + \lceil \gamma N \rceil\}$ and recalling the
 698 ordering scheme defined by the function v in (2), our implementation chooses

$$699 \quad \mathcal{C}_k \leftarrow \{i_{c(x_k), M^-}, \dots, i_{c(x_k), M^+}\} \cup \mathcal{C}(x_k, \epsilon) \text{ and } \mathcal{N}_k \leftarrow \mathcal{N}(x_k, \epsilon) \setminus \mathcal{C}_k.$$

701 In this manner, to follow our convergence analysis in §4, the set \mathcal{C}_k always includes the
 702 set $\mathcal{C}(x_k, \epsilon)$, but it might also include indices corresponding to additional constraints
 703 whose violations are above and/or below the critical value $v_M(x_k)$. We explore the
 704 effects of various values of γ in our numerical experiments.

705 **5.2. A Simple Nonconvex Example.** This experiment explores how likely
 706 the algorithm gets stuck at a “poor” local minimizer (recall §4.2) using an example
 707 with $n = 2$, $f(x) = x_2$, $c_i(x) = 0.25x_1^4 - 1/3x_1^3 - x_1^2 + 0.2x_1 - 19.5 + \xi_{i,1}x_1 + \xi_{i,1}\xi_{i,0}$,
 708 $N = 5,000$, and $\alpha = 0.05$. The random values $\{\xi_{i,1}\}_{i \in \mathcal{I}}$ were drawn uniformly from
 709 $[-3, 3]$ while $\{\xi_{i,0}\}_{i \in \mathcal{I}}$ were drawn uniformly from $[-12, 12]$. Figure 1a depicts the
 710 feasible region of (P). By inspection, we see that this function has a “true” local
 711 minimizer around $x_1 = -1$, in the sense that this is the best local minimizer in a
 712 large neighborhood. The global minimizer lies around $x_1 = 2$.

713 This problem was solved with $\bar{M} = 20$ and $\delta_{\text{reset}} = 1$, using different starting
 714 points, different values of the penalty parameter ρ , and different values of the fractional
 715 parameter γ . The initial multipliers for the computation of H_0 were set to zero. The
 716 starting point was computed by choosing $[x_0]_1$ (the first component of x_0), then
 717 determining $[x_0]_2$ so that x_0 lies on the boundary of the feasible region.

718 Table 1 gives the initial points along with their corresponding function values.
 719 The starting point with $[x^{\text{bad}}]_1 = 0.08524989$ is included as one from which some
 720 poor performance was observed. This point is a “poor” local minimizer with a large
 721 objective value, away from the two “true” local minimizers (see Figure 1d). We include
 722 this starting point to explore whether the method can escape such a local solution.

$[x_0]_1$	-1.500	-1.000	-0.500	0.000	0.500	1.000	1.500	2.000	0.085
$f(x_0)$	2.299	1.823	2.127	2.357	2.562	1.657	0.880	0.548	2.310

Table 1: Values of the initial points and the corresponding objective functions.

723 Table 2 shows the objective function values for the points returned by the algo-
 724 rithm. Function values that are within 1% of one of the “true” minima are highlighted

725 in bold face, and those within 5% are given in italics. The markers in Figures 1a–1d
 726 depict the locations of all the points returned by Algorithm 1. We see that each such
 727 point appears to be a local minimizer.

$[x_0]_1$	γ							
	0.001	0.002	0.005	0.010	0.050	0.100	0.200	1.000
-1.5	<i>1.776</i>	<i>1.776</i>	1.754	1.754	1.754	1.754	1.754	1.754
-1.0	1.754	1.760	1.754	1.754	1.754	1.754	1.754	1.754
-0.5	2.049	1.951	1.814	1.754	1.754	1.754	1.754	1.754
0.0	2.289	2.289	1.814	1.754	1.754	1.754	1.754	1.754
0.5	2.382	2.382	0.460	0.460	0.460	0.460	0.460	0.460
1.0	0.912	0.460	0.460	0.460	0.463	0.463	0.463	0.463
1.5	0.579	0.460						
2.0	0.463	0.463	0.460	0.460	0.463	0.463	0.463	0.463
x^{bad}	2.310	2.289	1.814	1.754	1.754	1.754	1.754	1.754

Results for $\rho = 1$

$[x_0]_1$	γ							
	0.001	0.002	0.005	0.010	0.050	0.100	0.200	1.000
-1.5	<i>1.784</i>	<i>1.784</i>	<i>1.784</i>	1.754	<i>1.776</i>	<i>1.776</i>	<i>1.784</i>	<i>1.784</i>
-1.0	1.813	1.813	1.813	1.813	1.813	1.813	1.813	1.813
-0.5	2.049	1.951	1.951	1.951	1.814	1.814	1.814	1.814
0.0	2.289	2.289	1.951	1.951	<i>1.776</i>	<i>1.784</i>	<i>1.784</i>	<i>1.784</i>
0.5	2.382	2.382	0.579	0.579	0.624	0.460	0.460	0.460
1.0	0.912	0.579	0.624	0.624	0.579	0.579	0.579	0.579
1.5	0.624	0.579	0.460	<i>0.473</i>	0.460	0.460	0.460	0.460
2.0	0.463							
x^{bad}	2.310	2.289	1.951	1.951	<i>1.776</i>	<i>1.784</i>	<i>1.784</i>	<i>1.784</i>

Results for $\rho = 0.1$

$[x_0]_1$	γ							
	0.001	0.002	0.005	0.010	0.050	0.100	0.200	1.000
-1.5	2.001	<i>1.784</i>						
-1.0	1.813	1.813	1.813	1.813	1.813	1.813	1.813	1.813
-0.5	2.049	1.951	1.951	1.951	1.951	1.951	1.951	1.951
0.0	2.289	2.289	1.951	1.951	<i>1.784</i>	<i>1.794</i>	<i>1.794</i>	<i>1.794</i>
0.5	2.382	2.382	0.912	0.624	0.624	0.624	0.624	0.624
1.0	0.912	0.912	0.579	0.711	0.624	0.624	0.624	0.624
1.5	0.629	0.624	0.629	0.629	0.579	0.460	0.460	0.460
2.0	<i>0.475</i>							
x^{bad}	2.310	2.289	2.289	1.951	<i>1.784</i>	<i>1.784</i>	<i>1.784</i>	<i>1.784</i>

Results for $\rho = 0.01$

Table 2: Objective function values of the final iterates returned by Algorithm 1 for a simple two-dimensional example problem, using different values for the penalty parameter ρ and scenario selection parameter γ . The most-left column shows the starting point component $[x_0]_2$ together with the corresponding objective values $f(x_0)$.

728 The penalty parameter ρ balances the influence of the objective function and
 729 the constraint violation. The subproblem model q_k is defined by linearizations of

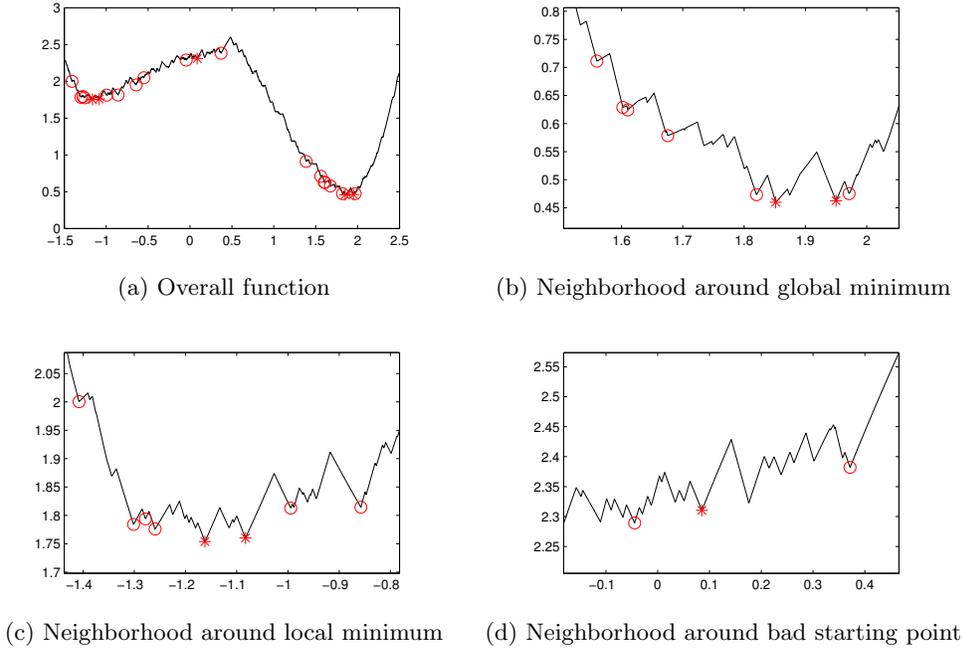


Fig. 1: Final points returned for the noisy nonconvex test function. The markers indicate the different points returned by the algorithm in the different runs from Table 2. The stars are the locations of the points with objective values within 1% of one of the “true” minima in 1a–1c. In 1d, the star denotes the point x^{bad} .

730 the constraints, meaning that it approximates the constraint violation well near an
731 iterate. However, in order to make progress towards a local minimizer that is not in the
732 immediate neighborhood of the current iterate, it might be necessary to take a larger
733 step. In that case, q_k might overestimate the constraint violation and not permit a
734 sufficiently large trial step, unless a decrease in the objective function outweighs the
735 increase in the constraint violation. A larger value of the penalty parameter gives
736 more weight to the progress in the objective function, and makes it more likely that a
737 larger step can be taken. We see this confirmed in Table 2, where better final objective
738 values are obtained when ρ is large. The algorithm is then able to escape from an
739 inferior local minimum more often. Of course, the penalty parameter must still be
740 small enough to ensure that the limit point is feasible; see Lemma 2.

741 Another important factor that determines the solution quality is the number of
742 scenarios that are considered in subproblem (14). We vary this by choosing different
743 values of the scenario selection parameter γ . When γ is small, only scenarios are con-
744 sidered that are relevant to describe the feasible region locally in a small neighborhood
745 around the current iterate. For instances, in our example, $\gamma = 0.001$ leads to a sce-
746 nario selection $\mathfrak{S}_{M,k}$ of size 21 chosen out of 5000. As a consequence, the subproblem
747 might again overestimate the constraint violation (recall that the model objective in
748 (14) is defined as the minimum over the scenario selection $\mathfrak{S}_{M,k}$) and not permit trial
749 steps that are sufficiently large to escape a poor local minimizer. Indeed, we note in

750 Table 2 that inferior final objective values are obtained for small values of γ . In fact,
751 the method is not able to escape the local minimizer x^{bad} when $\gamma = 0.001$. On the
752 other hand, in the extreme case when $\gamma = 1$, the method typically achieves the best
753 results for a given penalty parameter. However, the computational effort required to
754 solve the subproblem can be significantly higher for large values of γ , and in practice a
755 trade-off has to be made between solution quality and solution time. We also explore
756 this trade-off in our larger example in §5.3.

757 The trust-region reset value δ_{reset} is another parameter that determines the final
758 solution quality. Consider a situation where a new point is accepted with a small
759 trust-region radius δ_k after a sequence of rejected trial points due to disagreement in
760 (15) of the actual reduction and the predicted reduction. After this, the trust-region
761 is increased to δ_{reset} . However, if this reset value is too small to reach a better local
762 minimizer in the model objective in (14), the method might be restricted to stay close
763 to an inferior local minimizer. Again, in practice a trade-off has to be made between
764 solution quality and computation time, since a large value of δ_{reset} might require more
765 rejected trial steps until an acceptable trial point is computed. A large value of δ_{reset}
766 also makes the solution of the subproblem more difficult because of the increased
767 feasible region. While the results above were found with $\delta_{\text{reset}} = 1$, we also ran the
768 method with $\delta_{\text{reset}} \in \{0.1, 10\}$. This resulted in different final objective values for a
769 few cases compared to Table 2, but none of the choices for δ_{reset} was clearly superior.

770 Overall, for this example, Algorithm 1 finds “true” local solutions in most cases
771 when the penalty parameter is large and when γ is at least 0.05. This corresponds
772 to 501 critical scenarios, chosen out of a total of 5,000 scenarios. These choices even
773 allow the method to escape from the inferior local minimizer x^{bad} .

774 **5.3. A Cash Flow Problem.** Our second test problem is a nonlinear and non-
775 convex variation of the cash flow problem considered in [17] given by

$$\begin{aligned}
776 \quad & \max_{\substack{x \in \mathbb{R}^T \\ z \in \mathbb{R}^{T+1}}} z_{T+1} - \mathbb{E} \left[\sum_{t=1}^T L_t \right] \\
777 \quad & \text{s.t.} \quad \left| \left\{ k = 1, \dots, N : z_{t+1} \geq \sum_{\hat{t}=1}^t l_{\hat{t},k} \text{ for all } t = 1, \dots, T \right\} \right| \geq M, \\
778 \quad & z_t = z_{t-1} - \sum_{j \in J: s_j=t} x_j + \sum_{j \in J: s_j+d_j=t} I_j(x_j)^{\frac{d_j}{2}} x_j \text{ for all } t = 2, \dots, T+1, \\
779 \quad & z_1 = z^{\text{init}}, \quad z \geq 0, \quad x \geq 0.
\end{aligned}$$

781 Here, the task is to decide how much money x_j should be invested in each investment
782 option $j \in J$ (in our experiments, $|J| = 75$) over 10 years, divided into $T = 20$ time
783 periods. Option j can be bought at start time $s_j \in \{1, \dots, T\}$ and is sold after d_j
784 periods. When sold, interest has been earned with an annual interest rate of $I_j(x_j)$.
785 In contrast to [17], the interest rate depends on the amount of money invested and
786 increases as the investment becomes larger, making the chance constraints nonlinear
787 and nonconvex. We assume that the decision about how much to invest in the different
788 options must be made upfront (with no recourse). In each time period t , an unknown
789 random liability L_t is incurred and has to be paid from money that is not currently
790 invested. The realizations of L_t are denoted $l_{t,k}$. The variables z_t keep track of the
791 amount of cash available in period t , and we need to ensure that there is always
792 enough cash on hand to pay off the liabilities $l_{t,k}$. For the objective, we maximize

793 the expected amount of cash available at the end of the 10 years. We note that this
 794 problem resembles other important stochastic optimal control applications such as
 795 inventory and battery management where one seeks to balance profit with the risk
 796 of depleting stored resources and not being able to satisfy demands. The chance-
 797 constrained approach allows us to explore such trade-offs.

798 The initial budget is $z^{\text{init}} = 10$. The interest rate for investment $j \in J$ is given by

$$799 \quad I_j(x_j) = \underline{I}_j + (\bar{I}_j - \underline{I}_j) \frac{\log(1 + \psi_j x_j)}{\log(1 + \psi_j z^{\text{init}})},$$

800 where \underline{I}_j is the initial interest rate for very small investments, and \bar{I}_j is the interest
 801 rate that would be earned if all of the initial cash z^{init} is invested in that option. This
 802 function is monotonically increasing and concave, with diminishing interest increase
 803 as the investment increases. Varying $\psi_j > 0$ changes the curvature of I_j ; with this,
 804 we can explore the effects of nonconvexities on algorithm performance.

805 We report the performance of the proposed method averaged over five instances
 806 with randomly generated data. For generating each instance, the parameter d_j was
 807 drawn uniformly from $\{1, \dots, 10\}$ and s_j was drawn uniformly from $\{1, \dots, T - d_j\}$.
 808 The values \underline{I}_j and \bar{I}_j were drawn uniformly from $[0.01, 0.05]$ and $(\underline{I}_j + 0.005, \bar{I}_j + 0.015)$,
 809 respectively. The parameter ψ_j was chosen as 10^{-p_j} , where p_j was drawn uniformly
 810 from $[2, 6]$. The unknown liabilities L_j followed a normal distribution with mean
 811 z^{init}/T and variance $0.2z^{\text{init}}/T$, i.e., 20% of the mean. With this, $\mathbb{E}[\sum_{t=1}^T L_t] = z^{\text{init}}$.

812 We chose $\delta_{\text{reset}} = 0.1$, $\bar{M} = 15$, and $\rho = 0.1$. We verified that the penalty
 813 parameter ρ was small enough so that, in each successful outcome, the infeasibility of
 814 the solution \tilde{x}^* returned from our algorithm, i.e., $\langle\langle c(\tilde{x}^*) \rangle\rangle_M$, was at most 10^{-6} .

815 The initial point x_0 was set as the optimal solution of the “robust” counterpart of
 816 (P), i.e., problem (RP). This point was computed by the `Ipopt` solver [47] in negligible
 817 time. We highlight that the ability to initialize the search using the robust solution
 818 implicitly allows our algorithm to quickly identify subsets of constraints that can be
 819 relaxed to improve the objective. Furthermore, the optimal multipliers obtained by
 820 `Ipopt` are taken as the initial values of the multipliers $[\lambda_0^c]_{ij}$ and $[\lambda_0^s]_j$.

821 Table 3 details the results of our numerical experiment, where we varied the
 822 number of scenarios N between 100 and 5,000 with $\alpha = 0.05$. The relevance of the
 823 size of the critical set \mathcal{C}_k was assessed using values of the scenario selection parameter
 824 γ between 0.001 and 1. We count an instance as solved (“OK”) if the algorithm
 825 terminated without an error. Unsuccessful outcomes were sometimes observed for
 826 one of the five randomly generated instances when $N \geq 2000$, where for some values
 827 of γ the QP for computing the multipliers could not be solved. We emphasize that
 828 these outcomes do not represent a failure of the proposed algorithm, but rather are a
 829 consequence of the inefficient manner in which we are solving the subproblems (recall
 830 the discussion surrounding (33)) in this preliminary implementation of the method.

831 Table 3 reports geometric averages over the successfully solved instances (“OK”)
 832 for the number of iterations, the number the critical scenarios \mathcal{C}_k per iteration, the
 833 total number of changes—from one iteration to the next—in the set of scenarios
 834 $\mathcal{W}_k = \mathcal{C}_k \cup \mathcal{N}_k$ considered in the subproblem (summed over all iterations after the
 835 first), and the wall clock time.

836 The quality of the solution returned by the algorithm is assessed using arithmetic
 837 averages of the final objective function values $f(x^{\text{rob}})$ and the relative improvement (a
 838 percentage) of the objective function compared to the robust solution x^{rob} , computed
 839 as $100 \frac{f(x^{\text{rob}}) - f(x_*)}{f(x^{\text{rob}})}$. To ensure consistency when comparing the influence of the choice

840 of γ on the relative improvement, the averages were taken only over all instances that
 841 were solved for all values of γ . The numbers in parentheses indicate the number of
 842 instances over which the relative improvement was averaged.

843 Interestingly, the improvements achieved by the proposed algorithm over the ro-
 844 bust formulation become more apparent as the number of scenarios increases. This
 845 is because the robust solution becomes increasingly conservative as N grows, which
 846 is reflected in smaller values of $f(x^{\text{rob}})$ (recall that this is a maximization problem).
 847 For reference, the (arithmetic) averages of the robust optima are given in Table 4.

N	100	200	500	1000	2000	3000	5000
$f(x^{\text{rob}})$	2.2456	2.2270	2.1852	2.1412	2.0844	2.0678	2.0441

Table 4: Average optimal objective value for robust formulation.

848 The main purpose of these experiments is the exploration of the trade-off between
 849 solution quality and computational effort. As in the experiments of §5.2, we observe
 850 that better objective function values are obtained when more scenarios are considered
 851 in the subproblem. The column “ $|\mathcal{C}_k|$ ” shows how many discrete variables are in the
 852 MIQP formulation of (32). For this problem, we see that the best results are obtained
 853 if we choose γ as small as 0.05, so that about only 10% of all scenarios are critical in
 854 the subproblem. Even if we reduce γ further, the outcome is often very good.

855 Looking at the number of changes in the scenario set \mathcal{W}_k considered in the sub-
 856 problem from one iteration to the next, we see that this is not a trivial problem.
 857 Specifically for the large cases, the total number of changes is several hundreds. On
 858 the other hand, for $N = 100$, the subproblem in the first iteration finds the scenario
 859 selection that is considered relevant in the end. We also observed that the method
 860 seemed to converged at a superlinear rate once the final \mathcal{W}_k was found, unless a
 861 suboptimal incumbent was returned by the subproblem solver due to the time limit.

862 Regarding the computational effort, we note that the number of iterations is
 863 moderate, indicating that the initial point provided by the robust solution is useful.
 864 Clearly, the computation times increase as the size of the subproblems grows. The
 865 reason that they are almost constant with $N = 5,000$ for varying γ is that, in most
 866 iterations, the 5 minutes time limit for the `Cplex` solver was exceeded, and the method
 867 continued with the best incumbent as the trial step in that iteration. As we can see, the
 868 final solution quality might even decline as γ increases to 1, because these incumbents
 869 are inferior solutions of the subproblem (14), and the method might then stall early.

870 We stress that our implementation is a proof-of-concept that has not attempted
 871 to reduce the overall computing time. Given that the MIQP subproblem can become
 872 quite large (the MIQP formulations for the last line in the table has 5,000 discrete
 873 variables, 100,213 continuous variables, and 100,097 constraints), the current times
 874 might be considered somewhat reasonable. However, the computation times can be
 875 expected to be significantly smaller in an implementation that uses the branch-and-cut
 876 method by Luedtke [34] to solve the subproblems.

877 **6. Summary and Extensions.** We presented an algorithm for solving nonlin-
 878 ear optimization problems with cardinality-constraints that arise from sample average
 879 approximations of chance constraints. Our analysis showed that, under standard as-
 880 sumptions, the method produces stationary points for an exact penalty function that
 881 coincide with stationary points for the optimization problem. A potential drawback

882 of the proposed method is that it might converge to poor local minimizers resulting
883 from the jaggedness of the feasible region caused by the SAA discretization of the
884 chance constraint. Our numerical experiments demonstrate that this is only of a mi-
885 nor concern, even if only a fraction of all scenarios are considered in each subproblem.

886 In the paper, we assumed that the value of the penalty parameter ρ is fixed
887 throughout the optimization. In practice, a suitable value is often not known *a priori*,
888 and an update mechanism is necessary to adjust its value. As with standard nonlinear
889 optimization problems, the penalty parameter has to be sufficiently small to ensure
890 convergence to feasible points. On the other hand, as seen in §5.2, too small a value
891 might result in convergence to inferior local solutions. This is in contrast to standard
892 nonlinear optimization, where a small value usually affects only the convergence speed,
893 not the quality of the solution.

894 The numerical experiments in the paper were carried out with a proof-of-concept
895 Matlab implementation. A practical implementation requires that the subproblems
896 are solved by a much more efficient method than the generic Cplex MIQP solver
897 applied to a big-M formulation (33). Here, the decomposition algorithm by Luedtke
898 [34] is a very promising option, since it is tailored specifically to chance constraints.
899 It models the chance constraint for a given scenario i as the general implication

$$900 \quad (34) \quad z_i \implies x \in P_i,$$

901 where z_i is a binary indicator variable that is one when scenario i is enforced, and P_i
902 is a polyhedron. Second-stage (cost-free) recourse actions can be expressed naturally in
903 this manner, where the hidden recourse variables are part of the definition of P_i . The
904 method proposed in [34] has shown tremendous improvement over a big-M formula-
905 tion (like (33)) in this context. We expect that Algorithm 1 can reap these benefits
906 for the solution of nonlinear chance-constrained problems with resource actions as
907 well. Here, conceptually, the hidden second-stage variables are included explicitly
908 in the optimization variables x in (P), but subproblem (32) can be posed using the
909 implicit formulation (34) without recourse variables, thereby making it amenable to
910 the decomposition method in [34].

911

REFERENCES

- 912 [1] L. ANDRIEU, R. HENRION, AND W. RÖMISCH, *A model for dynamic chance constraints in*
913 *hydro power reservoir management*, European Journal of Operational Research, 207 (2010),
914 pp. 579–589.
- 915 [2] J. BARRERA, T. HOMEM-DE MELLO, E. MORENO, B. K. PAGNONCELLI, AND G. CANESSA,
916 *Chance-constrained problems and rare events: an importance sampling approach*, Math-
917 *ematical Programming*, 157 (2016), pp. 153–189, doi:10.1007/s10107-015-0942-x, <http://dx.doi.org/10.1007/s10107-015-0942-x>.
- 919 [3] M. S. BAZARAA, H. D. SHERALI, AND C. M. SHETTY, *Nonlinear Programming*, John Wiley &
920 Sons, Inc., 2005, doi:10.1002/0471787779, <http://dx.doi.org/10.1002/0471787779>.
- 921 [4] A. BEN-TAL AND A. NEMIROVSKI, *Robust optimization—methodology and applications*, Mathe-
922 *matical Programming*, 92 (2002), pp. 453–480.
- 923 [5] P. BERARDI AND A. RUSZCZYŃSKI, *The probabilistic set-covering problem*, Operations Research,
924 50 (2002), pp. 956–967.
- 925 [6] D. BERTSIMAS, D. B. BROWN, AND C. CARAMANIS, *Theory and applications of robust optimiza-*
926 *tion*, SIAM Review, 53 (2011), pp. 464–501.
- 927 [7] D. BIENSTOCK, M. CHERTKOV, AND S. HARNETT, *Chance-constrained optimal power flow: Risk-*
928 *aware network control under uncertainty*, SIAM Review, 56 (2014), pp. 461–495.
- 929 [8] J. R. BIRGE AND F. LOUVEAUX, *Introduction to stochastic programming*, Springer Science &
930 Business Media, 2011.
- 931 [9] P. BONAMI AND M. A. LEJEUNE, *An exact solution approach for portfolio optimization problems*
932 *under stochastic and integer constraints*, Operations Research, 57 (2009), pp. 650–670.

- 933 [10] R. H. BYRD, N. I. M. GOULD, J. NOCEDAL, AND R. A. WALTZ, *On the convergence*
934 *of successive linear-quadratic programming algorithms*, SIAM Journal on Optimiza-
935 tion, 16 (2005), pp. 471–489, doi:10.1137/S1052623403426532, [http://dx.doi.org/10.1137/](http://dx.doi.org/10.1137/S1052623403426532)
936 [S1052623403426532](http://dx.doi.org/10.1137/S1052623403426532), arXiv:<http://dx.doi.org/10.1137/S1052623403426532>.
- 937 [11] G. C. CALAFIORE AND L. EL GHAOUI, *On distributionally robust chance-constrained linear*
938 *programs*, Journal of Optimization Theory and Applications, 130 (2006), pp. 1–22.
- 939 [12] A. CARNIATO AND E. CAMPONOGARA, *Integrated coal-mining operations planning: Modeling*
940 *and case study*, International Journal of Coal Preparation and Utilization, 31 (2011),
941 pp. 299–334.
- 942 [13] A. CHARNES AND W. W. COOPER, *Deterministic equivalents for optimizing and satisficing*
943 *under chance constraints*, Operations Research, 11 (1963), pp. 18–39.
- 944 [14] A. CHARNES, W. W. COOPER, AND G. H. SYMONDS, *Cost horizons and certainty equivalents:*
945 *An approach to stochastic programming of heating oil*, Management Science, 4 (1958),
946 pp. 235–263.
- 947 [15] W. CHEN, M. SIM, J. SUN, AND C.-P. TEO, *From CVaR to uncertainty set: Implications in*
948 *joint chance-constrained optimization*, Operations Research, 58 (2010), pp. 470–485.
- 949 [16] N. DE OLIVEIRA AND L. T. BIEGLER, *Constraint handling and stability properties of model-*
950 *predictive control*, AIChE journal, 40 (1994), pp. 1138–1155.
- 951 [17] D. DENTCHEVA, B. LAI, AND A. RUSZCZYŃSKI, *Dual methods for probabilistic optimization*
952 *problems*, Mathematical Methods of Operations Research, 60 (2004), pp. 331–346.
- 953 [18] D. DENTCHEVA, A. PRÉKOPA, AND A. RUSZCZYŃSKI, *Concavity and efficient points of discrete*
954 *distributions in probabilistic programming*, Mathematical Programming, 89 (2000), pp. 55–
955 77.
- 956 [19] G. DORFLEITNER AND S. UTZ, *Safety first portfolio choice based on financial and sustainability*
957 *returns*, European Journal of Operational Research, 221 (2012), pp. 155–164.
- 958 [20] J. DUPAČOVÁ, A. GAIVORONSKI, Z. KOS, AND T. SZANTAI, *Stochastic programming in water*
959 *management: A case study and a comparison of solution techniques*, European Journal of
960 Operational Research, 52 (1991), pp. 28–44.
- 961 [21] R. FLETCHER, *An ℓ_1 penalty method for nonlinear constraints*, in Numerical Optimization 1984,
962 P. T. Boggs, R. H. Byrd, and R. B. Schnabel, eds., SIAM, Philadelphia, PA, USA, 1985,
963 pp. 26–40.
- 964 [22] R. FLETCHER, *Practical Methods of Optimization*, John Wiley and Sons, New York, USA,
965 second ed., 1987.
- 966 [23] S. P. HAN AND O. L. MANGASARIAN, *Exact penalty functions in nonlinear programming*, Math-
967 ematical Programming, 17 (1979), pp. 251–269, doi:10.1007/BF01588250, [http://dx.doi.](http://dx.doi.org/10.1007/BF01588250)
968 [org/10.1007/BF01588250](http://dx.doi.org/10.1007/BF01588250).
- 969 [24] R. HENRION, P. LI, A. MÖLLER, M. C. STEINBACH, M. WENDT, AND G. WOZNY, *Stochastic*
970 *optimization for operating chemical processes under uncertainty*, in Online Optimization
971 of Large Scale Systems, M. Grötschel, S. O. Krumke, and J. Rambau, eds., Springer Berlin
972 Heidelberg, 2001, pp. 457–478, doi:10.1007/978-3-662-04331-8_24, [http://dx.doi.org/10.](http://dx.doi.org/10.1007/978-3-662-04331-8_24)
973 [1007/978-3-662-04331-8_24](http://dx.doi.org/10.1007/978-3-662-04331-8_24).
- 974 [25] T. HOMEM-DE MELLO AND G. BAYRAKSAN, *Monte carlo sampling-based methods for stochastic*
975 *optimization*, Surveys in Operations Research and Management Science, 19 (2014), pp. 56–
976 85.
- 977 [26] L. J. HONG, Y. YANG, AND L. ZHANG, *Sequential convex approximations to joint chance con-*
978 *strained programs: A Monte Carlo approach*, Operations Research, 59 (2011), pp. 617–630.
- 979 [27] T. KIM AND S. J. WRIGHT, *An ℓ_1 -active set approach for feasibility restoration in*
980 *power systems*, Optimization and Engineering, (2014), pp. 1–35.
- 981 [28] A. J. KLEYWEGT, A. SHAPIRO, AND T. HOMEM-DE-MELLO, *The sample average approximation*
982 *method for stochastic discrete optimization*, SIAM Journal on Optimization, 12 (2001),
983 pp. 479–502.
- 984 [29] S. KÜÇÜKYAVUZ, *On mixing sets arising in chance-constrained programming*, Mathematical
985 Programming, 132 (2012), pp. 31–56.
- 986 [30] M. A. LEJEUNE, *Pattern definition of the p -efficiency concept*, Annals of Operations Research,
987 200 (2012), pp. 23–36.
- 988 [31] M. A. LEJEUNE AND A. RUSZCZYŃSKI, *An efficient trajectory method for probabilistic*
989 *production-inventory-distribution problems*, Operations Research, 55 (2007), pp. 378–394.
- 990 [32] W. LI, Y. ZHANG, A. SO, AND M. WIN, *Slow adaptive OFDMA systems through chance con-*
991 *strained programming*, IEEE Transactions on Signal Processing, 58 (2010), pp. 3858–3869.
- 992 [33] Y. LIU, H. GUO, F. ZHOU, X. QIN, K. HUANG, AND Y. YU, *Inexact chance-constrained lin-*
993 *ear programming model for optimal water pollution management at the watershed scale*,
994 Journal of Water Resources Planning and Management, 134 (2008), pp. 347–356.

- 995 [34] J. LUEDTKE, *A branch-and-cut decomposition algorithm for solving chance-constrained mathe-*
996 *matical programs with finite support*, Mathematical Programming, 146 (2014), pp. 219–244.
- 997 [35] J. LUEDTKE AND S. AHMED, *A sample approximation approach for optimization with proba-*
998 *bilistic constraints*, SIAM Journal on Optimization, 19 (2008), pp. 674–699.
- 999 [36] J. LUEDTKE, S. AHMED, AND G. L. NEMHAUSER, *An integer programming approach for linear*
1000 *programs with probabilistic constraints*, Mathematical Programming, 122 (2010), pp. 247–
1001 272.
- 1002 [37] M. R. MURR AND A. PRÉKOPA, *Solution of a product substitution problem using stochastic*
1003 *programming*, in Probabilistic Constrained Optimization, S. Uryasev, ed., Springer, 2000,
1004 pp. 252–271.
- 1005 [38] A. NEMIROVSKI AND A. SHAPIRO, *Convex approximations of chance constrained programs*,
1006 SIAM Journal on Optimization, 17 (2006), pp. 969–996.
- 1007 [39] J. NOCEDAL AND S. WRIGHT, *Numerical Optimization*, Springer, New York, NY, USA, sec-
1008 ond ed., 2006.
- 1009 [40] J. PINTÉR, *Deterministic approximations of probability inequalities*, Zeitschrift für Operations
1010 Research, 33 (1989), pp. 219–239.
- 1011 [41] R. T. ROCKAFELLAR AND S. URYASEV, *Optimization of conditional value-at-risk*, Journal of
1012 Risk, 2 (2000), pp. 21–42.
- 1013 [42] R. T. ROCKAFELLAR AND S. URYASEV, *Conditional value-at-risk for general loss distributions*,
1014 Journal of banking & finance, 26 (2002), pp. 1443–1471.
- 1015 [43] F. SHAN, L. ZHANG, AND X. XIAO, *A smoothing function approach to joint chance-constrained*
1016 *programs*, Journal of Optimization Theory and Applications, (2014), pp. 1–19.
- 1017 [44] A. SHAPIRO, D. DENTCHEVA, AND A. RUSZCZYŃSKI, *Lectures on Stochastic Programming: Mod-*
1018 *eling and Theory*, SIAM, 2009.
- 1019 [45] Q.-T. THIEU AND H.-Y. HSIEH, *Use of chance-constrained programming for solving the oppor-*
1020 *tunistic spectrum sharing problem under Rayleigh fading*, in Wireless Communications and
1021 Mobile Computing Conference (IWCMC), 2013 9th International, IEEE, 2013, pp. 1792–
1022 1797.
- 1023 [46] W. VAN ACKOOLJ, R. HENRION, A. MÖLLER, AND R. ZORGATI, *Joint chance constrained pro-*
1024 *gramming for hydro reservoir management*, Optimization and Engineering, 15 (2014),
1025 pp. 509–531.
- 1026 [47] A. WÄCHTER AND L. T. BIEGLER, *On the implementation of a primal-dual interior point filter*
1027 *line search algorithm for large-scale nonlinear programming*, Mathematical Programming,
1028 106 (2006), pp. 25–57.
- 1029 [48] Y. YUAN, *Conditions for convergence of trust region algorithms for nonsmooth optimization*,
1030 Mathematical Programming, 31 (1985), pp. 220–228, doi:10.1007/BF02591750, http://dx.
1031 doi.org/10.1007/BF02591750.

N	γ	OK	iter	$ C_k $	$\Delta(\mathcal{W}_k)$	$f(x^*)$	rel. impr.	time sec
100	0.001	5	7.00	6.31	0.00	2.3306	3.61 (5)	3.43
100	0.002	5	7.00	6.31	0.00	2.3306	3.61 (5)	2.83
100	0.005	5	7.00	6.31	0.00	2.3306	3.61 (5)	2.96
100	0.010	5	7.00	6.31	0.00	2.3306	3.61 (5)	2.87
100	0.050	5	6.75	11.61	0.00	2.3312	3.64 (5)	4.15
100	0.200	5	6.75	26.00	0.00	2.3312	3.64 (5)	5.82
100	1.000	5	6.75	100.00	0.00	2.3312	3.64 (5)	6.97
200	0.001	5	7.42	7.91	0.00	2.3245	4.16 (5)	5.52
200	0.002	5	7.42	7.91	0.00	2.3245	4.16 (5)	5.33
200	0.005	5	7.42	7.91	0.00	2.3245	4.16 (5)	5.37
200	0.010	5	7.42	9.27	0.00	2.3280	4.32 (5)	6.30
200	0.050	5	7.42	21.00	2.46	2.3288	4.39 (5)	10.57
200	0.200	5	7.42	51.00	2.95	2.3288	4.39 (5)	14.66
200	1.000	5	7.42	200.00	2.95	2.3288	4.39 (5)	22.19
500	0.001	5	9.84	9.40	6.14	2.3103	5.72 (5)	25.95
500	0.002	5	9.84	9.40	6.14	2.3103	5.72 (5)	27.59
500	0.005	5	8.19	11.84	8.68	2.3156	5.96 (5)	37.45
500	0.010	5	7.69	14.27	9.58	2.3155	5.95 (5)	39.33
500	0.050	5	7.04	51.00	13.06	2.3171	6.03 (5)	52.96
500	0.200	5	7.04	126.00	13.66	2.3171	6.03 (5)	101.41
500	1.000	5	7.04	500.00	15.52	2.3171	6.03 (5)	260.08
1000	0.001	5	15.98	10.90	15.15	2.3023	7.40 (5)	117.13
1000	0.002	5	13.71	11.96	19.17	2.3059	7.57 (5)	130.13
1000	0.005	5	12.29	15.22	23.72	2.3092	7.73 (5)	147.30
1000	0.010	5	11.04	22.26	25.96	2.3104	7.79 (5)	176.23
1000	0.050	5	8.12	101.00	34.29	2.3126	7.91 (5)	313.17
1000	0.200	5	7.43	251.00	49.23	2.3126	7.91 (5)	411.45
1000	1.000	5	7.43	1000.00	57.14	2.3126	7.91 (5)	819.24
2000	0.001	5	21.94	13.84	43.67	2.2872	9.73 (4)	652.19
2000	0.002	4	17.22	15.88	51.58	2.2887	9.81 (4)	550.45
2000	0.005	4	12.85	23.01	56.61	2.2907	9.90 (4)	513.20
2000	0.010	4	13.39	41.00	67.10	2.2920	9.97 (4)	1013.33
2000	0.050	4	9.90	201.00	88.99	2.2946	10.10 (4)	1099.41
2000	0.200	4	9.90	501.00	113.26	2.2946	10.10 (4)	2208.06
2000	1.000	5	14.33	2000.00	129.18	2.2945	10.09 (4)	3519.53
3000	0.001	5	26.00	16.38	74.36	2.2895	10.74 (4)	1207.26
3000	0.002	5	19.86	19.24	79.64	2.2906	10.80 (4)	1093.91
3000	0.005	5	16.72	31.42	92.40	2.2921	10.86 (4)	1777.77
3000	0.010	4	14.56	61.00	111.11	2.2935	10.93 (4)	1736.09
3000	0.050	4	10.75	301.00	137.77	2.2945	10.98 (4)	2389.05
3000	0.200	5	9.14	751.00	183.12	2.2942	10.97 (4)	2251.02
3000	1.000	4	11.97	3000.00	227.11	2.2927	10.89 (4)	3124.97
5000	0.001	5	28.20	20.74	137.24	2.2866	11.92 (4)	2965.06
5000	0.002	5	21.49	25.84	144.26	2.2873	11.95 (4)	3469.57
5000	0.005	4	18.58	51.00	172.91	2.2897	12.06 (4)	4570.27
5000	0.010	4	15.90	101.00	192.78	2.2917	12.17 (4)	3954.34
5000	0.050	4	14.37	501.00	230.20	2.2917	12.16 (4)	3853.37
5000	0.200	4	11.66	1251.00	344.47	2.2912	12.14 (4)	3014.36
5000	1.000	4	8.99	5000.00	410.05	2.2676	10.89 (4)	2211.95

Table 3: Numerical results for cash flow problem with $\alpha = 0.05$.