

A smoothing SQP method for nonlinear programs with stability constraints arising from power systems

Xiaojiao Tong · Liqun Qi · Soon-Yi Wu · Felix F. Wu

Received: 10 February 2009 / Published online: 1 September 2010
© The Author(s) 2010. This article is published with open access at Springerlink.com

Abstract This paper investigates a new class of optimization problems arising from power systems, known as nonlinear programs with stability constraints (NPSC), which is an extension of ordinary nonlinear programs. Since the stability constraint is described generally by eigenvalues or norm of Jacobian matrices of systems, this results in the semismooth property of NPSC problems. The optimal conditions of both NPSC and its smoothing problem are studied. A smoothing SQP algorithm is proposed for solving such optimization problem. The global convergence of algorithm is established. A numerical example from optimal power flow (OPF) is done. The computational results show efficiency of the new model and algorithm.

This work is supported by Natural Science Foundation of China (NSF10871031,10926189), the Hong Kong Research Grant Council, the Research Grants Council under Grant (HKU7179/07E, HKU7180/08E), and the Union Natural Science Foundation of Hunan-Hengyang (10JJ8008).

X. Tong (✉)

Department of Mathematics, Hengyang Normal University, Hengyang, Hunan 421008, China
e-mail: xjtong-csust@hotmail.com

L. Qi

Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
e-mail: maqilq@polyu.edu.hk

S.-Y. Wu

Department of Mathematics, National Cheng Kung University/National Center for Theoretical Science, Tainan, Taiwan
e-mail: soonyi@mail.ncku.edu.tw

F.F. Wu

Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong, Hong Kong
e-mail: ffwu@eee.hku.hk

Keywords Power systems · Stability constraint · Nonlinear programs · Smoothing SQP method · Convergence

1 Introduction

In this paper, we consider a class of optimization problems

$$\begin{aligned} \min & f(x, u) \\ \text{s.t.} & h(x, u) = 0, \\ & g(x, u) \leq 0, \\ & \phi(x, u) \geq \delta > 0, \end{aligned} \tag{1.1}$$

where $\delta > 0$ is a small constant vector; $x \in R^n$ stands for the state variable; $u \in R^m$ represents a control variable; $f : R^{n+m} \rightarrow R$, $h(x, u) = (h_1(x, u), h_2(x, u), \dots, h_n(x, u))^T : R^{n+m} \rightarrow R^n$, $g(x, u) = (g_1(x, u), g_2(x, u), \dots, g_l(x, u))^T : R^{n+m} \rightarrow R^l$ are twice continuously differentiable; $\phi(x, u) : R^{n+m} \rightarrow R^{n_s}$ is locally Lipschitz, possibly semismooth (see [17] for the semismooth concept). Here the last inequality constraint describes the stability property of dynamic systems. We call (1.1) *Nonlinear Programs with Stability Constraints (NPSC)* throughout this paper.

Problem (1.1) has a strong background and arises from power systems. The OPF is a fundamental tool of power systems for planning and operation. The classical OPF is an ordinary nonlinear programming problem, which has been studied widely, and many solution methods have been proposed and implemented over the years [6, 7, 19–21, 23]. On the other hand, it is well-known that power systems are described by differential and algebraic equations (DAEs), and they belong to dynamic systems. One of the concerned issues for such systems is the static stability at the equilibrium point (see [13]), which is the ability to recover the normal operation under some small disturbances. We note that classical OPF models consider little about system stability. This indicates that the solution of classical OPF cannot ensure the stable operation, especially for the current market mechanism where it requires the system operation under economic, stable and secure properties. Hence, it is necessary to extend the OPF models and algorithms with stability consideration. This is our motivation for this paper.

Both static and transient stability of power systems without considering optimal operation, have been studied extensively in the literature (see [2, 3, 10, 13, 14, 24]). In this paper, we are interested in static stability (also called small signal stability), which is not related to the dynamic operation of the system. Whereas it is associated with eigenvalues or singular values of Jacobian of function h in (1.1) (see [13, 18]). Recently, the study of OPF with static stability receives many attention in power system analysis. For example, Canizares and his research group have made a series of investigations (see [1, 12] and references therein). Their main research topic focuses on calculating technology with respect to bifurcation analysis of power systems, such as the saddle-node bifurcation, the Hopf bifurcation, and the limit-induced bifurcation. We also note that in the existing study of OPF with stability constraints, the stability requirement implements are mainly from physical properties of systems, or

via multi-layer optimization models (see [25] and its comments), which may not address the stability exactly or may result in some calculation difficulty. Moreover, the research of general static stability, i.e., considering negative real part of eigenvalues of Jacobian matrices, has little progress since the eigenvalue function is complicated. According to the system requirement and the research progress, we will set up a new mathematical model to describe the steady stability of power systems, as well as some effective algorithms for solving such model.

Some stability models and solution methods of dynamic systems with respect to nonlinear equations are studied in our previous work [11]. In this paper, the semi-smooth methodology is used for solving stability equilibrium solutions in power systems. In this paper, along with the line of stability constraints, we develop the stability solution of nonlinear equations to optimization problem, which has the form of (1.1). Due to the nonsmooth property of stability constraints, we then study (1.1) on correlative contents, such as optimal conditions, nonsmooth property and smoothing approach. Furthermore, we consider its solution method by using the smoothing technology in [22] and a smoothing SQP algorithm is presented for solving (1.1). The global convergence of the algorithm is investigated. An example arising from the OPF with stability constraints is applied to test the model and the algorithm. The numerical results show the efficiency of this study. The main contribution of this paper is twofold. First, it presents a uniform mathematical model to describe a class of practical optimization problems. The model enjoys nice semismooth characteristic. Second, related issues such as optimal conditions, computational method, and convergence of the method, are investigated, which are necessary and usable from the viewpoint of practical implementations.

The paper is organized as follow: In Sect. 2, we present a stability constraint model associated with some typical stability problems in dynamic systems. Section 3 discusses the optimal conditions of the NPSC problem, as well as its smoothing one. In Sect. 4, a smoothing SQP method is addressed. Section 5 investigates the convergence of the algorithm. Numerical example is done in Sect. 6. Section 7 gives some final conclusions.

2 Typical stability constraints in dynamic systems

This section presents some typical models of stability constraints for dynamic systems. Here we just describe the models. The detailed contents are introduced in our previous work [11].

2.1 Stability constraints

A nonlinear dynamic system can be described by

$$\dot{x} = F(x, u), \quad (2.1)$$

where $F(x, u) = (f_1(x, u), \dots, f_n(x, u))^T : R^n \times R^m \rightarrow R^n$ is a twice continuously differentiable function, and x and u are called state and control variables, respectively. Such a system arises from various engineering problems. In this paper, we are not

interested in the dynamic operation of (2.1), but concerned with steady-state stability analysis. In other words, the system (2.1) provides just a background of stability analysis.

We call x an *equilibrium point* of the system (2.1) if it satisfies

$$F(x, u) = 0$$

with given control variable u . In addition, combining with the stability requirement of dynamic systems, we define a so-called *stability equilibrium operation* of (2.1) in our research [11] as

$$\begin{cases} F(x, u) = 0, \\ \phi(x, u) \geq \delta > 0, \end{cases} \tag{2.2}$$

where $\delta > 0$ is a given constant vector of R^{m1} . Here $\phi(x, u) \in R^{m1}$ is called a *stability function*. Such function has a property that if there exists j such that $\phi_j(x) = 0$, then the correspondent state of system is unstable.

Let u be a variable. We then set up an optimal operation with some optimal objective. The optimal operation is called optimal power flow (OPF) in power systems. Furthermore, if we combine it with the stability requirement, this will formulate the optimal problem of stability constraints (1.1).

Problem (1.1) is an extension of classical OPF in power systems. The key issue is the way to set the stability constraints. Generally, there are three typical stability problems associated with dynamic system (2.1), which will result in voltage collapse and system oscillation in power systems (see [13]).

Definition 2.1 Consider the dynamic system (2.1) with given u . Let $\lambda(A)$ be the eigenvalue of matrix A , and $\nabla F(x, u)$ is the derivative of F with respect to the variable x .

(i) If x satisfies

$$F(x, u) = 0, \quad \text{Re}(\lambda(\nabla F(x, u))) < 0, \tag{2.3}$$

then we call x a *Static Stable Equilibrium Point* of (2.1).

(ii) If $x \in C_S$, where

$$C_S = \{x \in R^n \mid F(x, u) = 0, \det(\nabla F(x, u)) = 0\}, \tag{2.4}$$

then we call x a *Saddle-Node Bifurcation Point* of (2.1).

(iii) If $x \in C_H$ with

$$C_H = \{x \in R^n \mid F(x, u) = 0, \text{Re}(\lambda(\nabla F(x, u))) = 0\}, \tag{2.5}$$

then we call x a *Hopf Bifurcation Point* of (2.1), i.e., $\nabla F(x, u)$ has a pair of conjugate eigenvalues passing the imaginary axis.

The stable equilibrium models of power systems are required to satisfy (2.3), or to avoid the point of (2.4) and (2.5). On the other hand, from the viewpoint of numerical calculations, the proposed models should be solved by some typical numerical

methods, such as Newton-type methods. To this end, we construct stability functions satisfying some semismooth property. For example, the Saddle-Node Bifurcation can be avoided by setting $\det(\nabla F(x)) \neq 0$. Then we set the stability function as

$$\phi(x, u) \equiv \min_{1 \leq i \leq n} \lambda_i(\nabla F(x, u)^T \nabla F(x, u)) > 0.$$

Based on the stable and semismooth requirements, by using mathematical analysis, three semismooth stability constraints with respect to Definition 2.1 and $m_1 = 1$ can be set up (see [11] for the detailed analysis). Here $m_1 = 1$ means that for each stable model, we just consider one stability requirement.

In the following analysis of this section, the control variable u is assumed to be constant.

- (i) *Static Stability Constraint:*

$$\begin{aligned} \phi_1^I(x, u) &= - \max_{1 \leq i \leq n} \lambda_i(\nabla F(x, u) + \nabla F(x, u)^T) \geq \delta, \\ \phi_1^{II}(x, u) &= 1 - \|C(\nabla F(x, u))\|_1 \geq \delta. \end{aligned} \tag{2.6}$$

- (ii) *Avoiding Saddle-node Bifurcation:*

$$\begin{aligned} \phi_2(x, u) &= \min_{1 \leq i \leq n} \lambda_i(\nabla F(x, u)^T \nabla F(x, u)) \\ &= - \max_{1 \leq i \leq n} \lambda_i(-\nabla F(x, u)^T \nabla F(x, u)) \geq \delta. \end{aligned} \tag{2.7}$$

- (iii) *Avoiding Hope Bifurcation:*

$$\phi_3(x, u) = 1 + \max\{-\|C(\nabla F(x, u))\|_1, -\|C^{-1}(\nabla F(x, u))\|_1\} \geq \delta. \tag{2.8}$$

Here $C(A) = (A - \sigma I)^{-1}(A + \sigma I)$ is the Cayley transformation of matrix A .

We use $\phi(x, u)$ to express the functions $\phi_i(x, u)$ ($i = 1, 2, 3$). In addition, we introduce a slack variable $y \in R$ to transfer the inequality constraint to equality one. Then the stability inequality constraint $\phi(x, u) \geq \delta$ can be transformed equivalently to an equality constraint:

$$\Phi(x, u, y) = -\phi(x, u) + |y| + \delta = 0, \tag{2.9}$$

where $|y|$ expresses the absolute value of y , and is semismooth [5].

2.2 Smoothing function of maximal eigenvalue function

We note that the functions defined in (2.9) are related to the so-called maximum function as follow:

$$\hat{\phi}(x) \equiv \max\{x_1, \dots, x_n\}.$$

Since $\hat{\phi}(x)$ is semismooth, we will consider its smoothing version as

$$\hat{\phi}_s(\epsilon, x) = \begin{cases} \epsilon \ln(\sum_{i=1}^n e^{x_i/\epsilon}), & \text{if } \epsilon \neq 0, \\ \max_{1 \leq i \leq n} \{x_i\}, & \text{if } \epsilon = 0. \end{cases}$$

Here $\epsilon > 0$ is called a smoothing parameter. Such smoothing function is also called an exponential penalty function and has some interesting characteristics (see Lemma 2.1 in [4] and Lemma 7 in [16]).

Lemma 2.1 $\hat{\phi}_s(\epsilon, x)$ has the following properties: for any $\epsilon > 0$.

- (i) $\hat{\phi}_s(\epsilon, x)$ is increasing with respect to ϵ , i.e., for any $\epsilon_1 > \epsilon_2 > 0$, it holds $\hat{\phi}_s(\epsilon_1, x) \geq \hat{\phi}_s(\epsilon_2, x)$. Further, we have

$$0 \leq \hat{\phi}_s(\epsilon, x) - \hat{\phi}(x) \leq \epsilon \ln n. \tag{2.10}$$

- (ii) $\hat{\phi}_s(\epsilon, x)$ is continuously differentiable and

$$\nabla \hat{\phi}_s(\epsilon, x) = \sum_{i=1}^n a_i(\epsilon, x) e^i, \tag{2.11}$$

$$\begin{cases} \frac{d\hat{\phi}_s(\epsilon, x)}{d\epsilon} = \ln \sum_{i=1}^n e^{x_i/\epsilon} - \frac{1}{\epsilon} \sum_{i=1}^m a_i(\epsilon, x) x_i, \\ \lim_{\epsilon \rightarrow 0^+} \frac{d\hat{\phi}_s(\epsilon, x)}{d\epsilon} = \ln |I(x)|, \quad \ln |I(x)| \leq \frac{df_s(\epsilon, x)}{d\epsilon} \leq \ln n \end{cases} \tag{2.12}$$

with fixed x , where

$$a_i(\epsilon, x) = \frac{e^{x_i/\epsilon}}{\sum_{i=1}^m e^{x_i/\epsilon}}, \quad I(x) = \{i \mid \hat{\phi}(x) = x_i\}, \tag{2.13}$$

and e^i is the unit vector with its i -th component to be 1.

- (iii) For any fixed $x \in \mathbb{R}^n$,

$$\text{dist}(\nabla_x \hat{\phi}_s(\epsilon, x), \partial \hat{\phi}(x)) = o(\epsilon). \tag{2.14}$$

On the other hand, the formulas (2.6)–(2.8) indicate that stability functions are related to the maximal eigenvalue of matrix. We then introduce the so-called spectral function (denoted by $(f \circ \lambda)$), which is a composition of symmetric function $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$ and eigenvalue function $\lambda(\cdot)$ of symmetric matrices, see [4] for detailed discussion. Here the symmetric function f means $f(x) = f(Px)$ with any permutation matrix P . The spectral function has the following properties [17].

Lemma 2.2 A spectral function $(f \circ \lambda)$ satisfies the following properties:

- (i) it is semismooth if and only if f is semismooth. If f is ρ -order semismooth ($0 < \rho \leq 1$), the spectral function $(f \circ \lambda)$ is $\min\{1, \rho\}$ -order semismooth.;
- (ii) it is LC^1 if and only if f is LC^1 ;
- (iii) it is SC^1 if and only if f is SC^1 .
- (iv) $(f \circ \lambda)$ is differentiable at a matrix $X \in \mathcal{S}_n$ if and only if f is differentiable at point $\lambda(X)$. The gradient of $(f \circ \lambda)$ at X is given by

$$\nabla(f \circ \lambda)(X) = U^T (\text{diag}(\nabla f(\lambda(X))))U, \quad \forall U \in O(X). \tag{2.15}$$

More generally, the gradient of $(f \circ \lambda)$ has the following formula

$$\nabla(f \circ \lambda)(X) = V^T (\text{diag}(\nabla f(\mu)))V, \tag{2.16}$$

for any orthogonal matrix $V \in \mathcal{O}_n$ and $\mu \in \mathfrak{R}^n$ satisfying $X = V^T (\text{diag } \mu)V$. Here \mathcal{S}_n and \mathcal{O}_n denote the real $n \times n$ symmetric matrix set and orthogonal matrix set respectively, and $O(X) = \{P \in \mathcal{O}_n \mid PXP^T = \text{diag}(\lambda(X))\}$.

As an example, we choose the following special spectral function:

$$(\hat{\phi} \circ \lambda)(X) = \max_{1 \leq i \leq n} \lambda_i(X), \quad \mathbb{R}^{n \times n} \rightarrow \mathbb{R}. \tag{2.17}$$

Here X belongs to the set of symmetric matrices. Then from Lemma 2.2 we have that $(\hat{\phi} \circ \lambda)(X)$ is semismooth, so does the correspondent functions of stability constraints proposed in (2.6)–(2.8).

In order to design solution algorithm for NPSC problem, we use the smoothing way of maximum function and denote the smoothing function of $(\hat{\phi} \circ \lambda)(X)$ as $(\hat{\phi}_s \circ \lambda)(\epsilon, X)$, which has the following version for $\epsilon > 0$

$$(\hat{\phi}_s \circ \lambda)(\epsilon, X) = \epsilon \ln \left(\sum_{i=1}^n e^{\lambda_i(X)/\epsilon} \right). \tag{2.18}$$

Combining with Lemma 2.1, we have the following conclusions with respect to the spectral function and its smoothing one (see Proposition 2.2 in [4]).

Lemma 2.3 *Let X be symmetric matrix. For any $\epsilon > 0$, it holds*

$$0 \leq (\hat{\phi}_s \circ \lambda)(\epsilon, X) - (\hat{\phi} \circ \lambda)(X) \leq \epsilon \ln n, \tag{2.19}$$

$$\lim_{\epsilon \rightarrow 0} \nabla(\hat{\phi}_s \circ \lambda)(\epsilon, X) \in \partial(\hat{\phi} \circ \lambda)(X). \tag{2.20}$$

2.3 Smoothing technology for stability constraints

In the reminder part of this section, we focus on the smoothing version of three stability functions $\phi_1^I(x), \phi_1^{II}(x), \phi_2(x), \phi_3(x)$.

Denote the smoothing functions of stability functions in a uniform form as $\phi_s(\epsilon, x)$. Then we obtain a smoothing system of (2.9) as

$$\Phi_s(\epsilon, x, u, y) = -\phi_s(\epsilon, x, u) + \chi(\epsilon, y) + \delta = 0, \tag{2.21}$$

where $\chi(\epsilon, y)$ is the smoothing function of $|y|$ via the smoothing version of maximum function:

$$\chi(\epsilon, y) = \epsilon \ln(e^{y/\epsilon} + e^{-y/\epsilon}).$$

From above Lemmas 2.1–2.3, we can obtain the uniform approximate property of $\Phi_s(\epsilon, x, u, y)$ and $\Phi(x, u, y)$ as follows.

Proposition 2.1 *Let $\Phi_s(\epsilon, x, u, y)$ be the smoothing function of $\Phi(x, u, y)$ with respect to three stability functions. Then the following conclusions hold for the smoothing parameter $\epsilon > 0$.*

(i) *Approximation property:*

$$|\Phi_s(\epsilon, x, u, y) - \Phi(x, u, y)| = O(\epsilon). \tag{2.22}$$

(ii) *The derivative of $\Phi_s(\epsilon, x, u, y)$ with respect to (x, y) :*

$$\nabla \Phi_s(\epsilon, x, u, y) = - \begin{pmatrix} \nabla_x \phi_s(\epsilon, x, u) \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{e^{y/\epsilon} - e^{-y/\epsilon}}{e^{y/\epsilon} + e^{-y/\epsilon}} \end{pmatrix}. \tag{2.23}$$

(iii) *Property of Jacobian:*

$$\lim_{\epsilon \rightarrow 0} \text{dist}(\nabla \Phi_s(\epsilon, x, u, y), \partial \Phi(x, u, y)) = 0. \tag{2.24}$$

Here $\partial \Phi(x, u, y)$ expresses the generalized Jacobian of $\Phi(x, u, y)$ in the sense of Clarke derivative [5]. The symbol *dist* indicates the distance of a point to the set.

Proof As an example, we prove the proposition by the static stability constraint function $\phi_1^I(x, u)$. For such case, we have the stability function and their smoothing one as:

$$\begin{cases} \phi_1^I(x, u) = -\max_{1 \leq i \leq n} \lambda_i (\nabla F(x, u) + \nabla F(x, u)^T), \\ \phi_s(\epsilon, x, u) = -\epsilon \ln \left(\sum_{i=1}^n e^{\lambda_i (\nabla F(x, u) + \nabla F(x, u)^T) / \epsilon} \right). \end{cases}$$

(i) From Lemma 2.1 and Lemma 2.3 we have

$$\begin{cases} -\epsilon \ln n \leq \phi_s(\epsilon, x, u) - \phi_1^I(x, u) \leq 0, \\ 0 \leq \chi(\epsilon, y) - |y| \leq \epsilon \ln 2. \end{cases}$$

Then it follows

$$0 \leq \Phi_s(\epsilon, x, u, y) - \Phi(x, u, y) \leq \epsilon(\ln n + \ln 2),$$

which shows the result of (2.22).

(ii) This is a direct derivative calculation of function.

(iii) We obtain the result from (2.20) in Lemma 2.3 and (2.14) in Lemma 2.1 for functions $\phi_1^I(x, u)$ and $|y|$, respectively.

For other stability constraint functions, the proposition can be proved by the same method. The difference with 1-norm $\|\cdot\|_1$ in the stability constraint is where two-times smoothing process of maximum function is needed.

We complete the proof. □

3 Optimal conditions of NPSC problems

3.1 Optimal conditions for semismooth NPSC problems

We analyze the optimal conditions of (1.1) with ϕ defined in (2.6)–(2.8) and $n_s = 1$ in this section. It is obvious that (1.1) is equivalent to the following problem by adding a slack variable $y \in R$:

$$\begin{aligned} & \min_{(x,u,y) \in R^{n+1}} f(x, u) \\ & \text{s.t. } h(x, u) = 0, \\ & \quad g(x, u) \leq 0, \\ & \quad \Phi(x, u, y) \equiv -\phi(x, u) + |y| + \delta = 0. \end{aligned} \tag{3.1}$$

Here the typical stability problems are discussed in Sect. 2, i.e., the stability functions $\phi(x, u)$ are defined in (2.6)–(2.8) as $\phi_1^I(x, u)$, $\phi_1^{II}(x, u)$, $\phi_2(x, u)$, $\phi_3(x, u)$.

According to the optimal condition of nonsmooth optimization [5], we introduce the concept of generalized stationary points of (3.1).

Definition 3.1 A point $(x^*, u^*) \in R^{n+1}$ is said to be a generalized stationary point of (3.1) if there exists a KKT multiplier vector $(\lambda_h, \lambda_g, \lambda_\phi) \in R^{n+l+1}$ such that the following generalized Karush-Kuhn-Tucker (GKKT) conditions hold:

$$\begin{aligned} 0 \in & \begin{pmatrix} \nabla f(x^*, u^*) \\ 0 \end{pmatrix} + \begin{pmatrix} h'(x^*, u^*)^T \\ 0 \end{pmatrix} \lambda_h + \begin{pmatrix} g'(x^*, u^*)^T \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} \partial\phi(x^*, u^*)^T \\ \alpha \end{pmatrix} \lambda_\phi, \\ h(x^*, u^*) &= 0, \end{aligned} \tag{3.2}$$

$$0 \leq -g(x^*, u^*) \perp \lambda_g \geq 0,$$

$$\Phi(x^*, u^*, y^*) \equiv -\phi(x^*, u^*) + |y^*| + \delta = 0,$$

where $\alpha \in [-1, 1]$.

For convenience, we may assume that in the inequality constraints of (3.1), the first l_1 ($l_1 \leq l$) inequality constraints are active and the rest are inactive at (x^*, u^*) . Denote

$$\begin{aligned} G(x, u) &\equiv (h_1(x, u), \dots, h_n(x, u), g_1(x, u), \dots, g_{l_1}(x, u))^T, \\ \bar{G}(x, u, y) &\equiv \begin{pmatrix} G(x, u) \\ \Phi(x, u, y) \end{pmatrix}. \end{aligned}$$

If each element of the generalized Jacobian of $\bar{G}(x, u, y)$ has full row rank, then we say that the *Generalized Linear Independent Constraint Qualification* (GLICQ) condition of (3.1) is satisfied. For the addressed problem, the following conclusion gives a sufficient condition of constrained qualification.

Proposition 3.1 *Suppose that $G'(x, u)$ has full row rank at (x^*, u^*) and $y^* \neq 0$. Then the GLICQ holds for NPSC problem (3.1).*

Proof The active constraints of (3.1) can be written as

$$\bar{G}(x, u, y) = \begin{pmatrix} G(x, u) \\ -\phi(x, u) + |y| + \delta \end{pmatrix}.$$

It is obvious that for each element of generalized Jacobian $V \in \partial \bar{G}(x, u, y)$ with $V_\phi \in \partial \phi(x, u)$ it has the following form

$$V = \begin{pmatrix} G'(x, u) & 0 \\ -V_\phi & \alpha \end{pmatrix}. \tag{3.3}$$

On the other hand, the generalized derivative of $|y|$ is

$$\partial |y| = \begin{cases} 1, & \text{if } y > 0, \\ [-1, 1], & \text{if } y = 0, \\ -1, & \text{if } y < 0. \end{cases}$$

By the condition of proposition $y^* \neq 0$, it holds

$$\alpha = \begin{cases} 1, & \text{if } y^* > 0, \\ -1, & \text{if } y^* < 0, \end{cases}$$

which combines with the expression of V to obtain that V is full row rank. We complete the proof. □

Above proposition relies on the assumption $y^* \neq 0$. We note that on the one hand, the GLICQ condition is a sufficient condition of constraint qualification of nonlinear programming. This means that for case $y^* = 0$, the constrained qualification condition of NPSC may hold. On the other hand, for given $\delta > 0$, if it happens $y^* = 0$ at an optimal solution (i.e., the stability constraint is active), we can choose $\delta_1 < \delta$ in (3.1). Then we may obtain an optimal solution with $y^* \neq 0$, which satisfies the assumption of the proposition. The constrained qualification condition of NPSC is a valuable problem.

3.2 Optimal conditions for smoothing problems

According to the smoothing functions of the stability functions, we have nonlinear program with the smoothed stability equilibrium operation in the constraint set of (3.1) as:

$$\begin{aligned} & \min_{(x,u,y) \in \mathbb{R}^{n+m+1}} f(x, u) \\ & \text{s.t. } h(x, u) = 0, \\ & \quad g(x, u) \leq 0, \\ & \quad \Phi_s(\epsilon, x, u, y) \equiv -\phi_s(\epsilon, x, u) + \chi(\epsilon, y) + \delta = 0. \end{aligned} \tag{3.4}$$

Obviously, if (x, u, y) is a feasible point (local solution, global solution) of (3.1), it is a feasible point (local solution, global solution) of (3.4) with $\epsilon = 0$.

We now write the KKT condition of the smoothing nonlinear program (3.4). For $\epsilon \neq 0$, it holds

$$\begin{aligned}
 0 &= \begin{pmatrix} \nabla f(x, u) \\ 0 \end{pmatrix} + \begin{pmatrix} h'(x, u)^T \\ 0 \end{pmatrix} \lambda_h + \begin{pmatrix} g'(x, u)^T \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} \phi'_s(\epsilon, x, u)^T \\ \chi'(\epsilon, y) \end{pmatrix} \lambda_{\Phi_s}, \\
 h(x, u) &= 0, \\
 0 &\leq -g(x, u) \perp \lambda_g \geq 0, \\
 \Phi_s(\epsilon, x, u, y) &= 0.
 \end{aligned}
 \tag{3.5}$$

Here Jacobian of $\phi'_s(\epsilon, x, u)$ with respect to (x, u) can be obtained from the smoothing function of ϕ_s . The derivative of $\chi(\epsilon, y)$ with respect to y is

$$\chi'(\epsilon, y) = \frac{e^{y/\epsilon} - e^{-y/\epsilon}}{e^{y/\epsilon} + e^{-y/\epsilon}}.
 \tag{3.6}$$

4 A smoothing SQP algorithm

We now present a *smoothing SQP method* for solving problem (3.4). For more discussion on smoothing SQP methods, see [9, 15].

Let $d = (dx, du, dy)$. For given $(x, u, y) \in R^{n+m+1}$ and $r > 0, \epsilon > 0$. Similar to reference [8], we define a modified quadratic program $QP(x, u, y, \epsilon, r)$ subproblem of (3.4) by

$$\begin{aligned}
 \min_{d \in R^{n+m+1}, \xi \in R^l} & \nabla f(x, u)^T \begin{pmatrix} dx \\ du \end{pmatrix} + \frac{1}{2} d^T W d + r \sum_{i=1}^l \xi_i \\
 \text{s.t. } & h'(x, u) \begin{pmatrix} dx \\ du \end{pmatrix} + h(x, u) = 0, \\
 & g'(x, u) \begin{pmatrix} dx \\ du \end{pmatrix} + g(x, u) \leq \xi, \\
 & \Phi'_s(\epsilon, x, u, y) d + \Phi_s(\epsilon, x, u, y) = 0, \\
 & \xi \geq 0,
 \end{aligned}
 \tag{4.1}$$

where $W \in R^{(n+m+1) \times (n+m+1)}$ is a symmetric positive definite matrix, and $r > 0$ is a penalty parameter. Here we add one variable ξ to ensure the feasibility of the inequality constraint. In addition, we can see that the feasibility of equality constraints in (4.1) is satisfied if the matrix

$$V_1 = \begin{pmatrix} h'(x, u) & 0 \\ -\phi'_s(x, u) & \chi'(\epsilon, y) \end{pmatrix}$$

has full row rank.

It is not difficult to write the KKT system of (4.1) as

$$\begin{aligned} &\begin{pmatrix} \nabla f(x, u) \\ 0 \end{pmatrix} + Wd + \begin{pmatrix} h'(x, u)^T \\ 0 \end{pmatrix} \lambda_h + \begin{pmatrix} g'(x, u)^T \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} \phi'_s(\epsilon, x, u)^T \\ \chi'(\epsilon, y) \end{pmatrix} \lambda_{\Phi_s} = 0, \\ &r = \lambda_g + \lambda_\xi, \\ &h'(x, u) \begin{pmatrix} dx \\ du \end{pmatrix} + h(x, u) = 0, \\ &0 \leq - \left[g'(x, u) \begin{pmatrix} dx \\ du \end{pmatrix} + g(x, u) - \xi \right] \perp \lambda_g \geq 0, \\ &-\phi'_s(\epsilon, x, u) \begin{pmatrix} dx \\ du \end{pmatrix} + \chi'(\epsilon, y) dy + \Phi_s(\epsilon, x, u, y) = 0, \\ &0 \leq \xi \perp \lambda_\xi \geq 0. \end{aligned} \tag{4.2}$$

Define a merit function by a penalty function as:

$$\begin{aligned} \Theta_{(r^g, r^h, r^\Phi, \epsilon)}(x, u, y) = & f(x, u) + r^g \sum_{i=1}^l \max\{g_i(x, u), 0\} + r^h \sum_{i=1}^n |h_i(x, u)| \\ & + r^\Phi |\Phi_s(\epsilon, x, u, y)|, \end{aligned} \tag{4.3}$$

where r^g, r^h and r^Φ are positive numbers.

Now we state the SQP method for solving (1.1) by a smoothing method. We handle ϵ as the smoothing parameter and adjust it in every iteration.

Algorithm 4.1 Smoothing SQP Algorithm.

Step 0. (Initialization)

Let $r_{-1} > 0, \delta_1 > 0, \delta_2 > 0, \beta_\epsilon \in (0, 1), \sigma \in (0, 1), \tau \in (0, 1)$; Choose $w^0 = (x^0, u^0, y^0) \in R^{n+m+1}$, and $\epsilon_0 > 0$; and a symmetric positive definite matrix $W_0 \in R^{(n+m+1) \times (n+m+1)}$. Set $k := 0$.

Step 1. (Search direction)

Solve the QP subproblem (4.1) with $w^k = (x^k, u^k, y^k), \epsilon = \epsilon_k, W = W_k, r = r_{k-1}$. Let (d^k, ξ^k) be a solution of the QP, and $\lambda^k = (\lambda_h, \lambda_g, \lambda_\Phi, \lambda_\xi)$ be its corresponding KKT multiplier.

Step 2. (Termination check)

If a stopping rule is satisfied, terminate. Otherwise, go to Step 3.

Step 3. (Penalty update)

Let

$$\tilde{r}_k = \begin{cases} r_{k-1}, & \text{if } r_{k-1} \geq \max_{1 \leq i \leq m+n+1} \{|\lambda_i^k|\}, \\ \delta_1 + \max_{1 \leq i \leq n+l+2} \{|\lambda_i^k|\}, & \text{otherwise.} \end{cases} \tag{4.4}$$

Define $r_k^g = r_{k-1}$, and $r_k^h = r_k^\Phi = \tilde{r}_k$ and

$$r_k = \begin{cases} \tilde{r}_k, & \text{if } \sum_{i=1}^l \xi_i^k = 0, \\ \tilde{r}_k + \delta_2, & \text{otherwise.} \end{cases} \tag{4.5}$$

Step 4. (Line Search)

Let $t_k = (\tau)^{i_k}$, where i_k is the smallest nonnegative integer such that $i = i_k$ satisfies

$$\Theta_{(r_k^g, r_k^h, r_k^\Phi, \epsilon_k)}(w^k + (\tau)^i d^k) - \Theta_{(r_k^g, r_k^h, r_k^\Phi, \epsilon_k)}(w^k) \leq -\sigma(\tau)^i (d^k)^T W_k d^k. \tag{4.6}$$

Step 5. (Update Smoothing Parameter)

Let

$$w^{k+1} = w^k + t_k d^k, \tag{4.7}$$

$$\epsilon_{k+1} = \begin{cases} \beta_\epsilon \epsilon_k, & \text{if } \|d^k\| \leq \epsilon_k, \\ \epsilon_k, & \text{otherwise.} \end{cases} \tag{4.8}$$

Choose a symmetric positive definite matrix $W_{k+1} \in \mathbb{R}^{(m+n+1) \times (m+n+1)}$. Set $k := k + 1$ and go to Step 1.

Remark 4.1 At Step 2 in Algorithm 4.1, we do not specify a stopping rule. Generally, we stop the iteration when d^k and ϵ_k are sufficiently small. For example, the stopping rule is chosen

$$\|d^k\| + \epsilon_k \leq \hat{\tau},$$

where $\hat{\tau}$ is a given small constant.

The search direction obtained from the subproblem (4.1) in Step 1 of Algorithm 4.1 has the following descent property.

Proposition 4.1 *For any $\epsilon > 0$, we have*

- (i) $\Theta_{(r^g, r^h, r^\Phi, \epsilon)}(x, u, y)$ is directionally differentiable at $w = (x, u, y)$. Furthermore, if (d, ξ) is a solution of QP subproblem (4.1), $r^g = r$ and $r^h = r^\Phi \geq \max_{1 \leq i \leq n+l+2} |\lambda_i|$ with $\lambda = (\lambda_h, \lambda_g, \lambda_\Phi, \lambda_\xi)$ to be the corresponding multiplier of (4.1), then

$$\begin{aligned} \Theta'_{(r^g, r^h, r^\Phi, \epsilon)}(x, u, y; d) &\leq \nabla f(x, u)^T \bar{d} + \lambda_g^T g'(x, u) \bar{d} + \lambda_h^T h'(x, u) \bar{d} \\ &\quad + \lambda_\Phi^T \Phi'_s(x, u, y) d, \end{aligned} \tag{4.9}$$

where $\bar{d} = \begin{pmatrix} dx \\ du \end{pmatrix}$. Moreover, we have

$$\Theta'_{(r^g, r^h, r^\Phi, \epsilon)}(x, u, y; d) \leq -d^T W d. \tag{4.10}$$

- (ii) Suppose that $W \in R^{(n+m+1) \times (n+m+1)}$ is symmetric positive definite. If (d, ξ) is a solution of (4.1) with $d \neq 0$, then d is a descent direction of the penalty function $\Theta_{(r^g, r^h, r^\Phi, \epsilon)}(w)$ for $r^g = r$, and $r^h = r^\Phi$ satisfying the condition in (i).

Proof (i) The directionally differentiable conclusion is obtained from the property of the maximal function and the absolute function. In addition, we calculate the directional derivative of Θ as

$$\begin{aligned} &\Theta'_{(r^g, r^h, r^\Phi, \epsilon)}(x, u, y; d) \\ &= \nabla f(x, u)^T \bar{d} \\ &\quad + r^g \left[\sum_{i: g_i < 0} 0 + \sum_{i: g_i = 0} \max\{g'_i(x, u)\bar{d}, 0\} + \sum_{i: g_i > 0} g'_i(x, u)\bar{d} \right] \\ &\quad + r^h \left[\sum_{j: h_j > 0} h'_j(x, u)\bar{d} + \sum_{j: h_j = 0} |h'_j(x, u)\bar{d}| + \sum_{j: h_j < 0} -h'_j(x, u)\bar{d} \right] \\ &\quad + r^\Phi \kappa_\Phi, \end{aligned} \tag{4.11}$$

where

$$\kappa_\Phi = \begin{cases} \Phi'_s(\epsilon, w)d, & \text{if } \Phi_s(\epsilon, w) > 0, \\ |\Phi'_s(\epsilon, w)d|, & \text{if } \Phi_s(\epsilon, w) = 0, \\ -\Phi'_s(\epsilon, w)d, & \text{if } \Phi_s(\epsilon, w) < 0. \end{cases}$$

In order to prove the inequality (4.9), the KKT system (4.2) of the subproblem is used repeatedly.

First, we consider the equality parts $h(x, u)$ and $\Phi_s(\epsilon, w)$. By direct calculation and the KKT condition (4.2), it holds from (4.2) that

$$\begin{aligned} &r^h \left[\sum_{j: h_j > 0} h'_j(x, u)\bar{d} + \sum_{j: h_j = 0} |h'_j(x, u)\bar{d}| + \sum_{j: h_j < 0} -h'_j(x, u)\bar{d} \right] \\ &= r^h \left[\sum_{j: h_j > 0} (-h_j(x, u)) + \sum_{j: h_j = 0} |-h_j(x, u)| + \sum_{j: h_j < 0} h_j(x, u) \right] \\ &= -r^h \|h(x, u)\|_1. \end{aligned} \tag{4.12}$$

On the other hand, from the KKT system (4.2) again and the condition of r^h we have

$$(\lambda_h)^T h'(x, u)\bar{d} = -(\lambda_h)^T h(x, u) \geq - \sum_{1 \leq j \leq n} |(\lambda_h)_j| |h_j(x, u)| \geq -r^h \|h(x, u)\|_1,$$

which combines with (4.12) to yield

$$r^h \left[\sum_{j: h_j > 0} h'_j(x, u)\bar{d} + \sum_{j: h_j = 0} |h'_j(x, u)\bar{d}| + \sum_{j: h_j < 0} -h'_j(x, u)\bar{d} \right] \leq \lambda_h^T h'(x, u)\bar{d}. \tag{4.13}$$

By using a similar way we can prove that

$$r^\Phi \kappa_\Phi \leq \lambda_\Phi \Phi'_S(\omega, w)d. \tag{4.14}$$

Second, we consider the inequality part. From (4.11), to prove the conclusion (4.9) suffices to prove the follows:

$$r^g (\max\{g_i(x, u), 0\})'(x, u; \bar{d}) \leq (\lambda_g)_i g'_i(x, u)\bar{d} \quad (i = 1, 2, \dots, l). \tag{4.15}$$

We consider two cases for proving (4.15). Note that from the second relationship in the KKT condition (4.2) it implies

$$r^g \geq (\lambda_g)_i \quad (i = 1, \dots, l). \tag{4.16}$$

Case I: $\xi_i = 0$.

If $g_i(x, u) > 0$, then from the KKT condition (4.2) we derive that

$$(\max\{g_i(x, u), 0\})'(x, u; \bar{d}) = g'_i(x, u)\bar{d} \leq -g_i(x, u) + \xi_i < 0.$$

This implies (4.15) from (4.16).

If $g_i(x, u) = 0$, then it holds

$$(\max\{g_i(x, u), 0\})'(x, u; \bar{d}) = \max\{g'_i(x, u)\bar{d}, 0\} = 0.$$

From the complementarity condition in (4.2) we have

$$(\lambda_g)_i g'_i(x, u)\bar{d} = (\lambda_g)_i (g_i(x, u) + \xi_i) = 0.$$

So (4.15) holds.

If $g_i(x, u) < 0$, on the one hand, the complementarity condition shows that

$$(\lambda_g)_i g'_i(x, u)\bar{d} = -(\lambda_g)_i (g_i(x, u) - \xi_i) \geq 0.$$

On the other hand, it has

$$r^g (\max\{g_i(x, u), 0\})'(x, u; \bar{d}) = 0.$$

This also implies (4.15).

Case II: $\xi_i > 0$.

For this case, we have $r^g = (\lambda_g)_i$ since $\lambda_\xi = 0$ and the second expression in (4.2).

If $g_i(x, u) > 0$, the conclusion (4.15) is straightforward.

If $g_i(x, u) = 0$, the complementarity condition in (4.2) implies the follows

$$\begin{aligned} r^g (\max\{g_i(x, u), 0\})'(x, u; \bar{d}) &= r^g \max\{g'_i(x, u)\bar{d}, 0\} \\ &= \max\{(\lambda_g)_i g'_i(x, u)\bar{d}, 0\} = \max\{(\lambda_g)_i \xi_i, 0\} \\ &= (\lambda_g)_i \xi_i = (\lambda_g)_i g'_i(x, u)\bar{d}. \end{aligned}$$

This also proves (4.15).

If $g_i(x, u) < 0$, the complementarity condition implies that

$$r^g (\max\{g_i(x, u), 0\})'(x, u; \bar{d}) = 0 \leq (\lambda_g)_i (-g_i(x, u) + \xi_i) = (\lambda_g)_i g'(x, u) \bar{d}.$$

This proves (4.15).

Hence, for all cases, the result (4.15) holds. Combining with (4.11), (4.13)–(4.14) and (4.15) the inequality (4.9) is proved.

The conclusion (4.10) follows from (4.9) and the first equality of (4.2).

(ii) The conclusion is obtained from the conclusion (i), the positive definite matrix W and $d \neq 0$.

We complete the proof of the proposition. □

Remark 4.2 (i) Proposition 4.1 shows that line search in Step 4 of Algorithm 4.1 is well-defined for $d \neq 0$, so does the smoothing SQP method when $\epsilon_k > 0$ and W_k to be a symmetric positive definite matrix at each iteration. On the other hand, from Remark 4.1 the algorithm will stop when $\|d\|$ is sufficiently small.

(ii) Assume that $r_k = r_*$ for large k with constant r_* . Then Algorithm 4.1 implies $\sum_{i=1}^l \xi_i = 0$. From the KKT system (4.2) of subproblem (4.1), we can derive that $d = 0$ to be a KKT point of (3.4).

5 Global convergence

We make the following standard assumptions in order to prove the global convergence of Algorithm 4.1.

(A1) There exist two positive scalars m and M with $m \leq M$ such that the symmetric matrix W_k in Algorithm 4.1 satisfies the following condition

$$m \|w\|^2 \leq w^T W_k w \leq M \|w\|^2 \tag{5.1}$$

with any vector $w = (x, u, y) \in R^{n+m+1}$.

(A2) For all large k , it holds $r_k = r_* > 0, r_k^g = r_k = r_*, r_k^h = r_k^\Phi = r_*^h$.

We have the following global convergence theorem.

Theorem 5.1 *Assume that (A1) and (A2) hold. Let $\{w^k\}$ and $\{\epsilon_k\}$ be the sequences generated by the smoothing SQP algorithm.*

(i) *Assume that $\{w_k\}$ has a limiting point. Then*

$$\lim_{k \rightarrow \infty} \epsilon_k = 0. \tag{5.2}$$

(ii) *Let*

$$K = \{k \mid \|d^k\| \leq \epsilon_k\}. \tag{5.3}$$

If $\{w^k\}_{k \in K}$ has an accumulation point $w^ = (x^*, u^*, y^*)$, then w^* is a generalized stationary point of (3.1).*

Proof (i) It is obvious that $\{\epsilon_k\}$ is bounded. Suppose that ϵ_* is an accumulation point of $\{\epsilon_k\}$. If $\epsilon_* > 0$, then $\|d^k\| \leq \epsilon_k$ occurs only finitely many times. This means that after a finite number of iterations, ϵ_k remains unchanged, i.e., for some k_0 and all $k \geq k_0$, $\epsilon_k = \epsilon_{k_0}$. In this case, our smoothing method reduces to the smooth modified SQP method proposed in [9] for a smooth nonlinear optimization, see Appendix in [9]. Then Theorem A.1 in [9] indicates that $d^k \rightarrow 0$, which implies that $\|d^k\| \leq \epsilon_{k_0}$ for sufficiently large k . This contradicts with the above conclusion. Therefore, $\lim_{k \rightarrow \infty} \epsilon_k = 0$.

(ii) By the assumption (A2), it implies that the KKT multiplier sequence $\{\lambda^k\}_{k \in K}$ is bounded and $\xi^k = 0$ for all large enough k . We also obtain that $\lim_{k \rightarrow \infty, k \in K} d^k = 0$

Without loss of generality, we assume that

$$\lim_{k \rightarrow \infty, k \in K} w^k = w^*.$$

It follows from the penalty update rule in Step 3 of the algorithm and the assumption (A2) that $\xi^k = 0$ and $r_k = r_*$ for all sufficiently large k . Moreover, from the update rule in Step 3 of Algorithm 4.1, it implies that the multiplier $\{\lambda_k\} = \{(\lambda_g, \lambda_h, \lambda_\phi, \lambda_\xi)\}$ and $\{d^k\}_{k \in K}$ to be bounded. Passing to the subsequence, we may assume that

$$\lim_{k \rightarrow \infty, k \in K} d^k = d^*, \quad \lim_{k \rightarrow \infty, k \in K} W_k = W^*, \quad \lim_{k \rightarrow \infty, k \in K} \lambda_k = \lambda^* = (\lambda_g^*, \lambda_h^*, \lambda_\phi^*).$$

Then from $\epsilon^* = 0$ in (i) and the algorithm, it holds $d^* = 0$, $\bar{d}^* = 0$ and $\xi^* = 0$. Finally, we have, from Proposition 2.1 and (4.2), that

$$\begin{aligned} 0 &\in \begin{pmatrix} \nabla f(x^*, u^*) \\ 0 \end{pmatrix} + \begin{pmatrix} h'(x^*, u^*)^T \\ 0 \end{pmatrix} \lambda_h^* + \begin{pmatrix} g'(x^*, u^*)^T \\ 0 \end{pmatrix} \lambda_g^* + \begin{pmatrix} \partial \Phi(x^*, u^*)^T \\ \partial |y^*| \end{pmatrix} \lambda_\phi^*, \\ h(x^*, u^*) &= 0, \\ 0 &\leq -g(x^*, u^*) \perp \lambda_g^* \geq 0, \\ \Phi(x^*, u^*, y^*) &\equiv -\phi(x^*, u^*) + |y^*| + \delta = 0. \end{aligned} \tag{5.4}$$

Above expression means that $w^* = (x^*, u^*, y^*)$ is the generalized stationary point of (3.1). The theorem is proved. \square

In the remainder of this section, we consider that which conditions can satisfy the assumption (A2). Denote

$$H(\epsilon, w) = \begin{pmatrix} h(x, u) \\ \Phi_s(\epsilon, x, u, y) \end{pmatrix}.$$

By using a similar proof we can obtain the following conclusion.

Proposition 5.1 *Let $\{(\epsilon_k, w^k)\}$ be generated by Algorithm 4.1 and (ϵ_*, w^*) be an any accumulation point of $\{(\epsilon_k, w^k)\}$. Suppose that*

- (i) (A1) holds;
- (ii) $\{w^k\}$ is bounded;

- (iii) The generalized Jacobian $\partial_w H(\epsilon_*, w^*)$ has full row rank with $\epsilon_* \geq 0$;
- (iv) For any $V \in \partial_w H(\epsilon_*, w^*)$, there exists $d = \begin{pmatrix} \bar{d} \\ d_y \end{pmatrix}$ with $\bar{d} = \begin{pmatrix} dx \\ du \end{pmatrix}$ such that

$$g'(x, u)\bar{d} + g(x^*, u^*) < 0, \quad Vd + H(\epsilon_*, w^*) = 0.$$

Then the condition (A2) holds.

Proof The proof process is similar to the one of Theorem A.2 in [9], just to replace the Jacobian matrix of smooth function by the generalized Jacobian V . Here we omit the detailed process of the proof. □

6 Numerical example

In this section, we present an example for testing the NPSC model and Algorithm 4.1 proposed in Sect. 4. The WSCC 3-generators with 9-buses is chosen from PST (Power System Toolbox), see Fig. 1 for the system configuration.

The tested system has three generators, three loads and nine transmission electricity lines. Three loads (Load A-Load C) adopt dynamic model where the active power and reactive power are set

$$-P_i^d = -P_i^{d0} + D_i \dot{\delta}_i, \quad -Q_i^d(V_i) = -Q_i^{d0} + \alpha_2 V_i^2 + \alpha_1 V_i$$

with $D_i = 2.5$ and Table 1, where $\dot{\delta}_i$ means the differential of δ_i to time t .

Fig. 1 WSCC 3-generators with 9-buses system

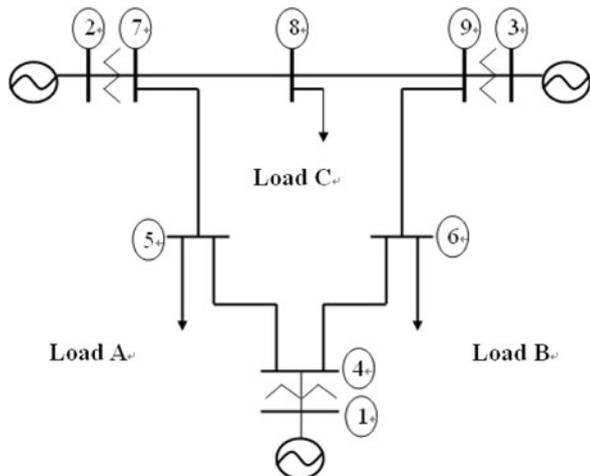


Table 1 Load constants

	Load A	Load B	Load C
P_i^{d0}	1.18	0.84	1.05
Q_i^{d0}	0.1	0.1	0.1

Table 2 Wastage coefficients of generators

	<i>a</i>	<i>b</i>	<i>c</i>
P_{G1}	8.20	12.712	30.00
P_{G2}	8.76	12.001	25.00
P_{G3}	6.46	12.290	45.00

In the tested system, the objective is to minimize the total energy wastage of generating fuel:

$$W(P) = \sum_{i=1}^{n_g} W_i(P_{Gi}) = \sum_{i=1}^{n_g} (a_i P_{Gi}^2 + b_i P_{Gi} + c_i) \tag{6.1}$$

with wastage coefficients (a_i, b_i, c_i) ($i = 1, \dots, n_g$) given by Table 2.

Let the system consist of n_g generators and $(n_l - n_g)$ load buses. Denote

$$\mathcal{Q} = \{1, \dots, n_l - n_g\}, \quad \mathcal{C} = \{n_l - n_g + 1, \dots, n_l\}, \quad \mathcal{N} = \{n_l + 1, \dots, n\}$$

to be the index sets of load buses, generator terminal buses, and generator-inner buses, respectively. For the tested system of the 3-generators with 9-buses, we have $n_l = 9, n_g = 3, n - n_l = 3$.

The dynamic description of power systems is DAEs as

$$\begin{cases} \dot{\delta} = T_2 \omega_g - T_1 D_l^{-1} T_1^T [f(\delta, V) - P], \\ \dot{\omega} = -M_g^{-1} D_g \omega_g - M_g^{-1} T_2^T [f(\delta, V) - P], \\ g(\delta, V) = 0, \end{cases} \tag{6.2}$$

where the n -th bus is chosen as the reference bus; $\delta = (\delta_1, \dots, \delta_{n-1})^T$ and $V = (V_1, \dots, V_n)^T$ are bus angle and voltage magnitude; $\omega_g = (\omega_{n_l+1}, \dots, \omega_n)^T$ is the angle speed of generators; D_l, D_g and M_g are matrices of load-frequency, generator damp coefficient, and generator inertial coefficient, respectively. Denote

$$T_1 = \begin{pmatrix} I_{n_l} \\ 0 \end{pmatrix} \in R^{(n-1) \times n_l}, \quad T_2 = \begin{pmatrix} 0 & -1_{n-n_g} \\ I_{n_g-1} & -1_{n_g-1} \end{pmatrix} \in R^{(n-1) \times n_g}$$

with I_k to be $k \times k$ unit matrix and $1_k = (1, \dots, 1)^T \in R^k$; $P_i = P_{Gi} - P_{li}$ is the injective active power of buses. The functions in (6.2) are defined by

$$\begin{cases} f(\delta, V) = (f_1(\delta, V), \dots, f_{n-1}(\delta, V))^T, \\ f_i(\delta, V) = V_i \sum_{j=1}^n V_j (G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij}). \end{cases} \tag{6.3}$$

$$\begin{cases} g(\delta, V) = (g_1(\delta, V), \dots, g_{n_l-n_g}(\delta, V), g_{n_l+1}(\delta, V), \dots, g_n(\delta, V))^T, \\ g_i(\delta, V) = V_i \sum_{j=1}^n V_j (G_{ij} \sin \delta_{ij} - B_{ij} \cos \delta_{ij}) - Q_i^d(V_i). \end{cases} \tag{6.4}$$

The corresponding function of dynamic system (6.2) is denoted by

$$F(\delta, \omega_g, V, P) = \begin{pmatrix} T_2\omega_g - T_1D_l^{-1}T_1^T[f(\delta, V) - P] \\ -M_g^{-1}D_g\omega_g - M_g^{-1}T_2^T[f(\delta, V) - P] \end{pmatrix}. \tag{6.5}$$

Then from $g(\delta, V) = 0$, the derivative of $F(\cdot)$ with respect to (δ, ω_g) is

$$\begin{pmatrix} -T_1D_l^{-1}J\tilde{F}(\delta, V) & T_2 \\ -M_g^{-1}T_2^TJ\tilde{F}(\delta, V) & -M_g^{-1}D_g \end{pmatrix} \in R^{(2n-n_l-1) \times (2n-n_l-1)}, \tag{6.6}$$

which is the function of (δ, V, P_G) , and we denote it as $\nabla F(\delta, V, P_G)$ simply. Here $J\tilde{F}(\delta, V)$ represents the reduce Jacobian of $f(\cdot)$ to variable δ

$$J\tilde{F}(\delta, V) \equiv \frac{\partial f}{\partial \delta} - \frac{\partial f}{\partial V} \left(\frac{\partial g}{\partial V} \right)^{-1} \frac{\partial g}{\partial \delta}.$$

Consider the steady stability, i.e., all the eigenvalues of ∇F are on the left half plane. By the Cayley transformation, i.e., $C(A) = (A - \sigma I)^{-1}(A + \sigma I)$, the stable constraint is formulated to the follows (see [11])

$$\phi(\delta, V) = 1 - \|C(\nabla F(\delta, V, P_G))\|_1 \geq \gamma \tag{6.7}$$

with given constant $\gamma > 0$.

Finally, the NPSC model with respect to OPF problems is reformulated as

$$\begin{aligned} & \min_{(\delta, V, P_G)} W(\delta, V, P_G) \\ & \text{s.t. } G(\delta, V, P_G) = \begin{pmatrix} f(\delta, V) - (P_G - P_l) \\ g(\delta, V) \end{pmatrix} = 0, \\ & H_{min} \leq H(\delta, V, P_G) \leq H_{max}, \\ & \phi(\delta, V) = 1 - \|C(\nabla F(\delta, V, P_G))\|_1 \geq \gamma. \end{aligned} \tag{6.8}$$

Here $G(\delta, V, P_G)$ is called power flow equations with respect to active power and reactive power, and $\delta_i = 0$ for steady state.

We choose $\sigma = 9$ in the Cayley transformation and $\gamma = 10^{-4}$ in the stability constraint. The smoothing technology proposed in [11] is used to obtain the smoothing function of $\|C(\nabla F(\delta, V))\|_1$ and to form the smoothing optimization problem.

In the numerical example, inequality constraints in (6.8) are set limits of line transfer power $P_{line_{max}} = 2.8$. The inequality limits are with respect to the active and reactive power of generators, and the voltages and angles of buses. The upper and lower boundaries of them are set as:

$$\begin{aligned} P_{g_{min}} &= (0.25, 0.25, 0.25)^T, & P_{g_{max}} &= (2.5, 2.5, 2.5)^T, \\ Q_{g_{min}} &= (-0.8, -0.8, -0.8)^T, & Q_{g_{max}} &= (2.5, 2.5, 2.5)^T, \\ V_{min} &= 0.9 * (1, 1, 1, 1, 1, 1, 1, 1)^T, & V_{max} &= 1.1 * (1, 1, 1, 1, 1, 1, 1, 1)^T, \end{aligned}$$

Table 3 Numerical results for OPF problem

	W_{min}	CEV	P_{G1}^*	P_{G2}^*	P_{G3}^*	Iter
Without-SC	138.4477	$0.1134 \pm 4.7099i$	0.9221	1.8852	0.3335	40
$\gamma = 0.003$	138.9961	$-0.0599 \pm 4.9673i$	0.4334	0.2500	2.4837	25
$\gamma = 0.0003$	138.6684	$-0.0528 \pm 4.9052i$	0.4269	0.5194	2.1973	30
$\gamma = 0.00003$	138.6159	$-0.0452 \pm 5.0449i$	0.3941	0.5495	2.1976	28
$\gamma = 0.000003$	138.6146	$-0.0448 \pm 5.0532i$	0.3923	0.5504	2.1985	29

$$\delta_{min} = -45 * (1, 1, 1, 1, 1, 1, 1, 1)^T, \quad \delta_{max} = 45 * (1, 1, 1, 1, 1, 1, 1, 1)^T.$$

The parameters of Algorithm 4.1 are chosen as follows:

$$\begin{aligned} r_{-1} &= 10^3, & \delta_1 &= \delta_2 = 20, & \beta_\epsilon &= 0.1, & \sigma &= 0.5, \\ \tau &= 0.8, & \epsilon_0 &= 10^{-4}. \end{aligned}$$

We use the usual stopping rule in SQP algorithm, i.e., $\|d^k\| \leq 10^{-4}$, which is also the adopted way in reference [9, 15]. It needs to explain that since the initial smoothing parameter is chosen $\epsilon_0 = 10^{-4}$, from the algorithm it holds $\epsilon_k \leq 10^{-4}$. So we omit the stopping rule of smoothing parameter ϵ .

In order to compare the effect of stability constraint, we also consider the ordinary OPF problem in the example, i.e., without including stability constraint. The computing results are reported in Table 3, where W_{min} means the optimal value of the objective; CEV expresses the Critical Eigenvalue, which is the maximal value of the real part of eigenvalues; P_{Gi}^* is the optimal solution with respect to active power at the i -th generator; Iter indicates iterative number by using Algorithm 4.1.

From Table 3 we have the following conclusions:

- Without stability constraint in the OPF problem (denoted as Without-SC in Table 3), we obtain an unsteady optimal solution. On the other hand, we find a stable solution by solving NPSC with differently stable constant γ ;
- The larger the stability constant γ , the better the resulting stability of the system becomes. Here the stability performance is described via the magnitude of negative real part of the CEV(the Critical Eigenvalue of ∇F).

7 Conclusions

This paper studies a class of nonlinear programs with stability constraints (NPSC). The problem has a strong background and is an extension of ordinary nonlinear programming problems. The models of stability constraints are discussed in detail in our research for stability solution of nonlinear equations. In this paper, we combine the stability constraint into optimization problems, which results in semismooth nonlinear programming problems. Then the optimal conditions and smoothing algorithm are investigated in this paper. The global convergence is also established. A numerical example arising from power systems is tested. The computational result shows

the efficiency of the NPSC model and the algorithm. The further research is to study large-scale NPSC problems.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. Avalos, R.J., Canizares, C.A., Milano, F., Conejo, A.J.: Equivalency of continuation and optimization methods to determine saddle-node and limit-induced bifurcations in power systems. *IEEE Trans. Circuits Syst. I* **56**(1), 210–223 (2009)
2. Canizares, C.A.: Calculating optimal system parameters to maximize the distance to saddle-node bifurcation. *IEEE Trans. Circuits Syst.* **45**, 225–237 (1998)
3. Chen, L., Tada, Y., Okamoto, H., Tanabe, R., Ono, A.: Optimal operation solutions of power systems with transient stability constraints. *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.* **48**(3), 327–339 (2001)
4. Chen, X., Qi, H., Qi, L., Teo, K.L.: Smooth convex approximation to the maximum eigenvalue function. *J. Glob. Optim.* **30**, 253–270 (2004)
5. Clarke, F.H.: *Optimization and Nonsmooth Analysis*. John Wiley, New York (1983)
6. Dommel, H.W., Tinney, W.F.: Optimal power flow solutions. *IEEE Trans. Power Appar. Syst.* **87**, 1866–1876 (1968)
7. El-Hawary, M.E. (ed.): *Optimal Power Flow: Solution Techniques, Requirement and Challenges*. IEEE Tutorial Course, IEEE Service Center, Piscataway, USA, 1996
8. Jiang, H., Ralph, D.: Smooth SQP methods for mathematical programs with nonlinear complementarity constraints. *SIAM J. Optim.* **30**(3), 779–808 (2000)
9. Jiang, H., Ralph, D.: Smooth SQP methods for mathematical programs with nonlinear equilibrium constraints. See <http://www.ms.unimelb.edu.au/danny/smooth-mpec.ps>
10. Kaye, R.J., Wu, F.F.: Dynamic security regions of power systems. *IEEE Trans. Circuits Syst.* **29**(9), 612–623 (1982)
11. Kelly, C.T., Qi, L., Tong, X., Yin, H.: Finding a stable solution of a system of nonlinear equations. *J. Ind. Manag. Optim.* (to appear)
12. Kodsi, S.K.M., Canizares, C.A.: Application of a stability-constrained optimal power flow to tuning of oscillation controls in competitive electricity markets. *IEEE Trans. Power Syst.* **22**(4), 1944–1954 (2007)
13. Kundur, P., Paserba, J., Ajarapu, V., Andersson, G., Bose, A., Canizares, C., Hatziargyriou, N., Hill, D., Stankovic, A., Taylor, C., Van Cutsem, T., Vittal, V.: Definition and classification of power system stability. *IEEE Trans. Power Syst.* **19**, 1387–1401 (2004)
14. Kwatny, H.G., Fischl, R.F., Nwankpa, C.O.: Local bifurcation in power systems: theory, computation, and application. *Proc. IEEE* **83**(11), 1456–1483 (1995)
15. Ling, C., Qi, L., Zhou, G., Wu, S.Y.: Global convergence of a robust smoothing SQP method for semi-infinite programming. *J. Optim. Theory Appl.* **129**, 147–164 (2006)
16. Peng, J.M., Lin, Z.H.: A non-interior continuation method for generalized linear complementarity problems. *Math. Program., Ser. A* **86**, 533–563 (1999)
17. Qi, H., Yang, X.: Semismoothness of spectral functions. *SIAM J. Matrix Anal. Appl.* **25**(3), 766–783 (2004)
18. Qi, L., Tong, X.J., Wang, Y.: Computing power system parameters to maximize the small signal stability margin based on min-max models. *Optim. Eng.* (2008). doi:10.1007/s11081-008-9049-z
19. Sun, D.I., Ashley, B., Brewer, B., Hughes, A., Tinney, W.F.: Optimal power flow solution by Newton approach. *IEEE Trans. Power Appar. Syst.* **103**, 2864–2880 (1984)
20. Tong, X.J., Ling, C., Qi, L.: A semi-infinite programming algorithm for solving optimal power flow with transient stability constraints. *J. Comput. Appl. Math.* **217**, 432–447 (2008)
21. Tong, X.J., Wu, F.F., Qi, L.: On the convergence of decoupled optimal power flow method. *Numer. Funct. Anal. Optim.* **28**(3–4), 1–19 (2007)
22. Tong, X.J., Wu, F.F., Qi, L.: Available transfer capability using a smoothing pointwise maximum function. *IEEE Trans. Circuits Syst. I* **55**(1), 462–474 (2008)

23. Tong, X.J., Wu, F.F., Zhang, Y., Yan, Z., Ni, Y.: A semismooth Newton method for solving optimal power flow. *J. Ind. Manag. Optim.* **3**(3), 553–567 (2007)
24. Yuan, Y., Kubokawa, J., Sasaki, H.: A solution of optimal power flow with multicontingency transient stability constraints. *IEEE Trans. Power Syst.* **18**(3), 1094–1102 (2003)
25. Zhao, J., Zhang, B., Chiang, H.D.: An optimal power flow model and approach with static voltage stability constraints. 2005 IEEE/PES Transmission and Distribution Conference and Exhibition: Asia and Pacific, Dalian, China, pp. 1–6