J Optim Theory Appl (2014) 161:783–806
DOI 10.1007/s10957-013-0459-5

Feedback Stabilization Methods for the Solution of Nonlinear Programming Problems

Iasson Karafyllis

Received: 5 November 2012 / Accepted: 21 October 2013 / Published online: 12 November 2013

Abstract In this work, we show that, given a nonlinear programming problem, it is possible to construct a family of dynamical systems, defined on the feasible set of the given problem, so that: (a) the equilibrium points are the unknown critical points of the problem, which are asymptotically stable, (b) each dynamical system admits the objective function of the problem as a Lyapunov function, and (c) explicit formulas are available without involving the unknown critical points of the problem. The construction of the family of dynamical systems is based on the Control Lyapunov Function methodology, which is used in mathematical control theory for the construction of stabilizing feedback. The knowledge of a dynamical system with the previously mentioned properties allows the construction of algorithms, which guarantee the global convergence to the set of the critical points.

Keywords Nonlinear programming · Feedback stabilization · Lyapunov functions · Nonlinear systems

1 Introduction

Differential equations have been used in the past for the solution of Nonlinear Programming (NLP) problems. The reader may consult [1–8] for various results on the topic. Some methods are interior-point methods (in the sense that they are defined only on the feasible set), while other methods are exterior-point methods (in the sense that they are defined at least in a neighborhood of the feasible set). As remarked in

Communicated by Nobuo Yamashita.

I. Karafyllis (✉)
Department of Mathematics, National Technical University of Athens, Zografou Campus,
15780 Athens, Greece
e-mail: iasonkar@central.ntua.gr

[9], each system of differential equations that solves an NLP problem, when combined with a numerical scheme for solving Ordinary Differential Equations (ODEs), provides a numerical scheme for solving the NLP problem. Differential equations have been utilized for the solution of Linear Programming and NLP problems in the literature of neural networks (see, for example, [10–12], the review paper [13], and the references therein).

In this work, we are interested in the application of feedback stabilization methods for solving NLP problems. The feedback stabilization methods can be applied in two ways:

1. for the construction of the dynamical system that solves the NLP problem and
2. for the selection of the step size of the Runge–Kutta scheme that is used for the solution of the resulting system of ODEs (see [14, 15]).

More specifically, consider a standard NLP problem with sufficient regularity properties and such that the necessary Karush–Kuhn–Tucker conditions of the NLP hold. Inspired by the methods employed in the book [16], we would like to construct a well-defined dynamical system on the feasible set of the NLP with the following properties:

Property 1: The conditions of Nagumo's theorem (given on p. 27 of the book [17]) must be satisfied. This property is required because the local existence of solutions of the dynamical system is guaranteed and the feasible set is viable, i.e., the solution exists and belongs to the feasible set.

Property 2: The vector field appearing in the right-hand side of the dynamical system is a locally Lipschitz vector field. This property is required for the uniqueness of a solution of the dynamical system. Moreover, this property is required because we would like to be able to apply first-order Runge-Kutta schemes for the simulation of the solutions of the dynamical system. Higher regularity is also desirable because high-order Runge–Kutta schemes can be used for the simulation of the solutions of the dynamical system.

Property 3: The equilibrium points of the dynamical system are exactly the points for which the necessary Karush–Kuhn–Tucker conditions of the NLP hold.

Property 4: The objective function of the NLP problem is a (strict) Lyapunov function for the dynamical system. In other words, we would like the value of the objective function to decrease along the solution of the dynamical system. This property is important because it guarantees useful stability properties. Furthermore, the fact that the Lyapunov function of the dynamical system does not involve the solutions of the NLP problem is important for numerical purposes (see [14, 15]): the time derivative of the Lyapunov function along the solutions of the system and the difference of the values of the Lyapunov function between two points can be computed without knowledge of the solutions of the NLP problem.

Property 5: The vector field appearing in the right-hand side of the dynamical system must be explicitly known. Formulas for the vector field must be provided: the formulas must not involve the solution of the NLP problem.

Property 6: The vector field appearing in the right-hand side of the dynamical system must have free parameters, which can be selected in an appropriate way so that the convergence properties of the corresponding numerical schemes to the global attractor of the dynamical system are optimal. In other words, we want to construct a parameterized family of vector fields with all the above properties.

It must be noted that properties 1–6 are rarely satisfied by other differential equation methods for solving NLPs. For example, in [1] and [2], the constructed Lyapunov function involves the solution of the NLP problem, and this does not meet our requirements. Moreover, in [2], the solution of the NLP problem is not an equilibrium point for the constructed time-varying dynamical system. Antipin [1] constructs an autonomous dynamical system for which the solution of the NLP problem is an equilibrium point and for which the locally Lipschitz vector field appearing in the right-hand side of the dynamical system does not depend on the location of the unknown point. However, the definition of the vector field appearing in the right-hand side of the dynamical system is involved (it requires the solution of an NLP since it involves a projection on the feasible set). The NLP problem without equality constraints under additional convexity hypotheses has been studied in [10]. However, even in this work, the constructed Lyapunov function involves the solution of the NLP problem; the same feature appears in almost all neural networks proposed for the solution of mathematical programming problems (see [11, 12] and the references in the review paper [13]). On the other hand, the papers [6, 7] propose systems of differential equations that satisfy properties 1–6 for systems without inequality constraints. Local results are provided in the paper [8], and differential equations based on barrier methods were considered in [4].

Clearly, the knowledge of the Lyapunov function can allow us to construct the vector field appearing in the right-hand side of the dynamical system by means of the Control Lyapunov Function methodology of feedback design (see [18–21]) for a completely controllable control system. However, there are certain obstructions for the direct application of the classical Control Lyapunov Function methodology: (i) the system is not defined on the whole space but on the closed feasible set, (ii) for every given point of the feasible set, the vector field appearing in the right-hand side of the dynamical system must belong to the contingent cone to the feasible set at the given point, and (iii) the position of the equilibrium points, i.e., the set of points of the feasible set that satisfy the Karush–Kuhn–Tucker conditions is unknown (this is what we are looking for).

The contribution of the paper is twofold:

1. The main result of the present work (Theorem 2.1) shows that all the previously mentioned obstructions can be overcome under appropriate assumptions.
2. Based on the ideas described in [14, 15], in Sect. 3 of the present work, we present an algorithm for the solution of the NLP, which is based on the application of the explicit Euler scheme for the numerical solution of the resulting system of ODEs, with appropriate step selection (Theorem 3.1). The algorithm will converge for every initial condition (global convergence). A modified and simpler version of the algorithm can work under slightly more demanding assumptions (Remark 3.3).

It should be noticed that the convergence rates of the proposed algorithms depend on the selection of certain matrices, which are the free parameters described in property 5 above. However, since the proposed algorithms are global, they can be used in combination with any other local algorithm that guarantees a fast convergence based on the following intuitive idea: "apply the newly proposed algorithms when you are away from a solution and apply a fast local algorithm when you are close to a solution."

It should be emphasized that no claim is made about the effectiveness of the proposed algorithms. The topic of the numerical solution of NLPs is a mature topic, and it is clear that other algorithms have much better characteristics than the algorithms proposed in this paper. However, the theory used for the construction of the algorithm is different from other existing algorithms. The algorithms contained in this work are derived by using concepts of dynamical system theory and mathematical control theory. Moreover, no claim is made about the generality of our results: the linear independence constraint qualification assumed in this work is a restrictive assumption: it is more restrictive than the Mangasarian–Fromovitz constraint qualification in [22] or the constant rank constraint qualification (see [23] and references therein), which are more restrictive than the Guignard constraint qualification. However, the linear independence constraint qualification has the advantage of being easily checkable and of being true in many interesting cases (the work [24] showed that this assumption holds generically), and it is a vital ingredient for many numerical methods (successive quadratic programming; see [25, 26]). Furthermore, the linear independence constraint qualification allows us to obtain easy formulas for the required vector field (see Remark 2.4 below).

The structure of the paper is as follows. Section 2 contains the statement and proof of Theorem 2.1, which provides the solution to the problem of the construction of a vector field with properties 1–6. Section 3 provides numerical algorithms for the exploitation of the constructed vector field. Section 4 provides some examples, which show the performance of the algorithms. Finally, Sect. 5 contains the concluding remarks. The Appendix provides proofs of certain auxiliary results.

2 Transforming an Nonlinear Programming Problem into a Feedback Stabilization Problem

Notation Throughout this paper we adopt the following notation:

* Let $A \subseteq \Re^n$ be a set. By $C^0(A\,;\,\Omega)$ we denote the class of continuous functions on A taking values in Ω. By $C^k(A\,;\,\Omega)$, where $k \geq 1$ is an integer, we denote the class of functions on A taking values in Ω and having continuous derivatives up to order k. By $C^\infty(A;\Omega)$ we denote the class of functions on A taking values in Ω and having continuous derivatives of all orders (smooth functions).
* For a vector $x \in \mathbb{R}^n$, we denote by $|x|$ its usual Euclidean norm and by x' its transpose. For a real matrix $A \in \mathbb{R}^{n\times m}$, we denote by $|A|$ its induced norm, i.e., $|A| := \max\{|Ax| : x \in \mathbb{R}^m, |x| = 1\}$ and by $A' \in \mathbb{R}^{m\times n}$ its transpose. $I_n \in \mathbb{R}^{n\times n}$ denotes the identity matrix. For every $x = (x_1, \ldots, x_n)' \in \mathbb{R}^n$, we define $x^+ = (\max(0, x_1), \ldots, \max(0, x_n))' \in \mathbb{R}^n$. Notice that if $R \in \mathbb{R}^{n\times n}$ is a positive definite and diagonal matrix and $x'Rx^+ = 0$, then $x^+ = 0$.

* $\mathbb{R}^n_+ := (\mathbb{R}_+)^n = \{(x_1, \dots, x_n)' \in \mathbb{R}^n : x_1 \geq 0, \dots, x_n \geq 0\}$. Let $x, y \in \mathbb{R}^n$. We say that $x \leq y$ iff $(y - x) \in \mathbb{R}^n_+$.
* For every continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$, $\nabla V(x)$ denotes the gradient of V at $x \in \mathbb{R}^n$, i.e., $\nabla V(x) = (\frac{\partial V}{\partial x_1}(x), \dots, \frac{\partial V}{\partial x_n}(x))$, and $\nabla^2 V(x)$ denotes the Hessian matrix of V at $x \in \mathbb{R}^n$.

Consider the NLP problem

$$\min\bigl\{\theta(x) : x \in S\bigr\}, \tag{1}$$

where $x \in \mathbb{R}^n$, and the closed set $S \subseteq \mathbb{R}^n$ is defined by

$$S := \Bigl\{x \in \mathbb{R}^n : h_1(x) = \cdots = h_m(x) = 0, \max_{j=1,\dots,k}\bigl(g_j(x)\bigr) \leq 0\Bigr\}, \tag{2}$$

where $m < n$, and all functions $\theta : \mathbb{R}^n \to \mathbb{R}$, $h_i : \mathbb{R}^n \to \mathbb{R}$ $(i = 1, \dots, m)$, $g_j : \mathbb{R}^n \to \mathbb{R}$ $(j = 1, \dots, k)$ are twice continuously differentiable, under the following assumptions:

(H1) The feasible set $S \subseteq \mathbb{R}^n$ defined by (2) is nonempty, and the sublevel sets of $\theta : \mathbb{R}^n \to \mathbb{R}$ are compact sets, i.e., for every $x_0 \in S$, the following sublevel set is compact:

$$\Xi_\theta(x_0) := \bigl\{x \in S : \theta(x) \leq \theta(x_0)\bigr\}. \tag{3}$$

(H2) For every $x \in S$, the row vectors $\nabla h_i(x)$ $(i = 1, \dots, m)$ and $\nabla g_j(x)$ for all $j = 1, \dots, k$ for which $g_j(x) = 0$ (active constraints) are linearly independent.

Assumption (H1) is a standard assumption, which guarantees that the NLP problem described by (1) and (2) is well posed and admits at least one global solution (see [27]). Assumption (H2) is a global version of the linear independence constraint qualification. As remarked in the Introduction, assumption (H2) is a restrictive assumption, which guarantees that for every solution of the NLP problem described by (1) and (2), the Karush–Kuhn–Tucker conditions hold.

We define:

$$\begin{aligned} h(x) &:= \begin{bmatrix} h_1(x) \\ \vdots \\ h_m(x) \end{bmatrix} \in \mathbb{R}^m, \qquad A(x) := \begin{bmatrix} \nabla h_1(x) \\ \vdots \\ \nabla h_m(x) \end{bmatrix} \in \mathbb{R}^{m\times n}, \\ g(x) &:= \begin{bmatrix} g_1(x) \\ \vdots \\ g_k(x) \end{bmatrix} \in \mathbb{R}^k, \qquad B(x) := \begin{bmatrix} \nabla g_1(x) \\ \vdots \\ \nabla g_k(x) \end{bmatrix} \in \mathbb{R}^{k\times n} \quad \text{for all } x \in \mathbb{R}^n. \end{aligned} \tag{4}$$

Assumption (H2) allows us to define the symmetric matrix

$$H(x) = I_n - A'(x)\bigl(A(x)A'(x)\bigr)^{-1}A(x) \quad \text{for all } x \in \mathbb{R}^n \text{ in a neighborhood of } S. \tag{5}$$

The following facts are direct consequences of definitions (4) and (5):

Fact 1: $H^2(x) = H(x)$, $A(x)H(x) = 0$, and $H(x)A'(x) = 0$.
Fact 2: $\xi' H(x)\xi = |H(x)\xi|^2$ for all $\xi \in \mathbb{R}^n$.
Fact 3: For every $\xi \in \mathbb{R}^n$, there exists $\lambda \in \mathbb{R}^m$ such that $\xi = H(x)\xi + A'(x)\lambda$.

Next, we define the set of critical points for the NLP problem defined by (1), (2).

Definition 2.1 Let $\Phi \subseteq S$ be the set of all points $x \in S$ for which there exist vectors $\lambda = (\lambda_1, \ldots, \lambda_m)' \in \mathbb{R}^m$ and $\mu = (\mu_1, \ldots, \mu_k)' \in \mathbb{R}^k_+$ such that

$$\begin{aligned} &\nabla\theta(x) + \sum_{i=1}^{m} \lambda_i \nabla h_i(x) + \sum_{j=1}^{k} \mu_j \nabla g_j(x) = 0, \\ &\sum_{j=1}^{k} \mu_j g_j(x) = 0. \end{aligned} \tag{6}$$

In other words, $\Phi \subseteq S$ is the set of critical points or Karush–Kuhn–Tucker points for the problem defined by (1) and (2).

Clearly, assumptions (H1) and (H2) guarantee that the set $\Phi \subseteq S$ is nonempty.

The following lemma provides a useful consequence of assumption (H2). Its proof is provided in the Appendix.

Lemma 2.1 *If assumption* (H2) *holds*, *then the matrix*

$$Q(x) := B(x)H(x)B'(x) - \operatorname{diag}\big(g(x)\big) \tag{7}$$

is positive definite for all $x \in S$.

We are now ready to state the main result of this section.

Theorem 2.1 *Suppose that assumptions* (H1) *and* (H2) *hold for the NLP problem described by* (1), (2). *Let $Q(x) \in \mathbb{R}^{k\times k}$ be the symmetric positive definite matrix defined by* (7). *Let $R_1(x) \in \mathbb{R}^{n\times n}$ be an arbitrary C^1, symmetric, and positive definite matrix, $R_2(x) \in \mathbb{R}^{k\times k}$ be an arbitrary C^1, symmetric, and positive semidefinite matrix, $a_i(x)$, $b_i(x)$, and $c_i(x)$ $(i = 1, \ldots, k)$ be arbitrary C^1 nonnegative functions with $b_i(x) + c_i(x) > 0$ for all $i = 1, \ldots, k$ and $x \in S$. Let at least one of the matrices $R_2(x) \in \mathbb{R}^{k\times k}$ and* $\operatorname{diag}(a(x)) \in \mathbb{R}^{k\times k}$ *be positive definite, where $a(x) := (a_1(x), \ldots, a_k(x))' \in \mathbb{R}^k$. Define the following locally Lipschitz vector field*:

$$\begin{aligned} F(x) = &-\big[H(x) - P'(x)Q(x)P(x)\big]R_1(x)\big[H(x) - P'(x)Q(x)P(x)\big]\big(\nabla\theta(x)\big)' \\ &- P'(x)\operatorname{diag}\big(g(x)\big)\big(R_2(x)\operatorname{diag}\big(g(x)\big) - \operatorname{diag}\big(a(x)\big)\big)v(x) \\ &- P'(x)R_3(x)\big(v(x)\big)^+, \end{aligned} \tag{8}$$

where

$$\begin{aligned}
P(x) &:= Q^{-1}(x)B(x)H(x) \in \mathbb{R}^{k\times n},\\
v(x) &:= P(x)\big(\nabla\theta(x)\big)' \in \mathbb{R}^k,\\
R_3(x) &:= \operatorname{diag}\big(b_1(x)+c_1(x)\big(\max\big(0,v_1(x)\big)\big)^{2p_1},\dots,\\
&\qquad b_k(x)+c_k(x)\big(\max\big(0,v_k(x)\big)\big)^{2p_k}\big),
\end{aligned}\tag{9}$$

and $p_i \ge 1$ $(i=1,\dots,k)$ are integers. Then the following properties hold:

(a) $A(x)F(x)=0$ *for all* $x\in S$,
(b) $\nabla\theta(x)F(x)<0$ *for all* $x\in S\backslash\Phi$,
(c) $F(x)=0 \Leftrightarrow x\in\Phi$,
(d) $B(x)F(x)=\operatorname{diag}(g(x))Q^{-1}(x)w(x)-R_3(x)(v(x))^+$ *for all* $x\in S$, *where*

$$\begin{aligned}
w(x) := {}& B(x)H(x)R_1(x)\big[H(x)-P'(x)Q(x)P(x)\big]\big(\nabla\theta(x)\big)'\\
&-\big[Q(x)+\operatorname{diag}\big(g(x)\big)\big]\big(R_2(x)\operatorname{diag}\big(g(x)\big)-\operatorname{diag}\big(a(x)\big)\big)v(x)\\
&-R_3(x)\big(v(x)\big)^+.
\end{aligned}\tag{10}$$

Consider the dynamical system

$$\dot{x}=F(x)\tag{11}$$

on the closed set $S\subseteq\mathbb{R}^n$. Then the following properties hold:

1. *For every $x_0\in S$, there exists a unique solution $x(t)$ of the initial-value problem (11) with $x(0)=x_0$, which is defined for all $t\ge 0$ and satisfies $x(t)\in S$ for all $t\ge 0$.*
2. *Every point $x\in\Phi$ is an equilibrium point for (11). Every strict local solution $x^*\in S$ of the NLP problem described by (1) and (2) is locally asymptotically stable for system (11).*

If we denote by $\omega(x_0)$ the set of accumulation points of the set $\{x(t):t\ge 0\}$, where $x_0\in S$, then $\omega(x_0)$ is a compact, positively invariant set for which there exists $l\le\theta(x_0)$ such that $\omega(x_0)\subseteq\Phi\cap\{x\in S:\theta(x)=l\}$.

Remark 2.1 Clearly, the matrices $R_1(x)\in\mathbb{R}^{n\times n}$, $R_2(x)\in\mathbb{R}^{k\times k}$ and the functions $a_i(x)$, $b_i(x)$, $c_i(x)$ $(i=1,\dots,k)$ can be selected in an appropriate way so that the convergence properties of the corresponding numerical schemes to the global attractor of the dynamical system are optimal. The stability properties of system (11) are analogous to the stability properties of gradient systems (see [28]).

Remark 2.2 It should be noted that all properties 1–6 mentioned in the Introduction are satisfied for the dynamical system (11). Indeed, property 1 is a direct consequence of (a) and (d). More specifically, a direct consequence of definition (9) and the fact that $\frac{d}{dt}g(x)=B(x)\dot{x}=B(x)F(x)=\operatorname{diag}(g(x))Q^{-1}(x)w(x)-R_3(x)(v(x))^+$ is that the following implication holds: "if $g_j(x)=0$ for some $j\in\{1,\dots,k\}$, then

$\frac{d}{dt}g_j(x) = -(b_j(x) + c_j(x)(\max(0, v_j(x)))^{2p_j})\max(0, v_j(x)) \leq 0$." The previous implication and property (a) guarantee that for every $x \in S$, $F(x)$ belongs to the contingent cone to S at x.

– Property 2 is a direct consequence of definitions (5), (7), (8), (9) and the fact that all functions $\theta : \mathbb{R}^n \to \mathbb{R}$, $h_i : \mathbb{R}^n \to \mathbb{R}$ $(i = 1, \dots, m)$, $g_j : \mathbb{R}^n \to \mathbb{R}$ $(j = 1, \dots, k)$ are twice continuously differentiable. It should be noticed that if at least one of the functions $b_i(x)$ $(i = 1, \dots, k)$ takes positive values, then the vector field $F(x)$ defined by (8) is simply locally Lipschitz and not C^1. When $b_i(x) \equiv 0$ for $i = 1, \dots, k$, then the vector field $F(x)$ defined by (8) is C^1. Higher regularity is possible by assuming higher regularity for all functions and matrices involved in (5), (7), (8), (9), sufficiently large values for the integers $p_i \geq 1$ $(i = 1, \dots, k)$, and $b_i(x) \equiv 0$ for $i = 1, \dots, k$.
– Properties 3 and 4 are direct consequences of (c) and (b), respectively. Indeed, notice that the function $V(x) = \theta(x) - \theta(x^*)$, where $x^* \in S$ is one of the global solutions of the NLP described by (1), (2), satisfies the equation $\nabla V(x)F(x) = \nabla\theta(x)F(x)$.
– Finally, properties 5 and 6 are evident.

Remark 2.3 The inspiration for Theorem 2.1 is the transformation of the NLP problem into a feedback stabilization problem. First, we notice that the Control Lyapunov Function (see [18–21]) is selected to be the function defined by $V(x) = \theta(x) - \theta(x^*)$, where $x^* \in S$ is one of the global solutions of the NLP problem described by (1), (2). The only problem is that we must define in an appropriate way the control system so that S is a positively invariant set for all possible inputs. In other words, we must have

$$\frac{d}{dt}h(x) = A(x)\dot{x} = 0 \quad \text{and} \quad \frac{d}{dt}g(x) = B(x)\dot{x} = \operatorname{diag}\big(g(x)\big)v - u \tag{12}$$

for all possible inputs $v \in \mathbb{R}^n$ and $u \in \mathbb{R}^n_+$. Notice that the second equation in (12) guarantees the implication "if $g_j(x) = 0$ for some $j \in \{1, \dots, k\}$, then $\frac{d}{dt}g_j(x) = -u_j \leq 0$." The first equation in (12) implies that $\dot{x} = H(x)w$ for arbitrary $w \in \mathbb{R}^n$. Combining, we get $B(x)H(x)w = \operatorname{diag}(g(x))v - u$. By redefining the input variables $w = B'(x)p + q$ and $v = p + z$ we get $p = Q^{-1}(x)(\operatorname{diag}(g(x))z - u - B(x)H(x)q)$. Consequently, the required control system is

$$\begin{aligned}\dot{x} = {} & H(x)\big(I_n - B'(x)Q^{-1}(x)B(x)H(x)\big)q \\ & + H(x)B'(x)Q^{-1}(x)\operatorname{diag}\big(g(x)\big)z - H(x)B'(x)Q^{-1}(x)u\end{aligned} \tag{13}$$

with inputs $q, z \in \mathbb{R}^n$ and $u \in \mathbb{R}^n_+$. The computation of the feedback law for the control system (13) with Control Lyapunov Function $V(x) = \theta(x) - \theta(x^*)$, gives the dynamical system (11), where F is defined by (8), (9). More specifically, we get:

$$\begin{aligned}\nabla V(x)\dot{x} = {} & \nabla\theta(x)\big(H(x) - H(x)B'(x)Q^{-1}(x)B(x)H(x)\big)q \\ & + \nabla\theta(x)H(x)B'(x)Q^{-1}(x)\operatorname{diag}\big(g(x)\big)z - \nabla\theta(x)H(x)B'(x)Q^{-1}(x)u.\end{aligned} \tag{14}$$

The Control Lyapunov Function approach requires that each input must be selected so that each term appearing in the right-hand side of (14) takes negative values. The feedback laws

$$
\begin{aligned}
q &= -R_1(x)\big(H(x) - H(x)B'(x)Q^{-1}(x)B(x)H(x)\big)\big(\nabla\theta(x)\big)', \\
z &= \big(\operatorname{diag}\big(a(x)\big) - R_2(x)\operatorname{diag}\big(g(x)\big)\big)Q^{-1}(x)B(x)H(x)\big(\nabla\theta(x)\big)', \\
u &= R_3(x)\big(Q^{-1}(x)B(x)H(x)\big(\nabla\theta(x)\big)'\big)^+,
\end{aligned}
\tag{15}
$$

where $R_1(x) \in \mathbb{R}^{n\times n}$ is an arbitrary C^1, symmetric, and positive definite matrix, $R_2(x) \in \mathbb{R}^{k\times k}$ is an arbitrary C^1, symmetric, and positive semidefinite matrix, $a_i(x)$, $b_i(x)$, and $c_i(x)$ $(i = 1,\dots,k)$ are arbitrary C^1 nonnegative functions with $b_i(x)+c_i(x) > 0$ for all $i = 1,\dots,k$ and $x \in S$, and at least one of the matrices $R_2(x)$, $\operatorname{diag}(a(x)) \in \mathbb{R}^{k\times k}$ is positive definite, where $a(x) := (a_1(x),\dots,a_k(x))' \in \mathbb{R}^k$, and $R_3(x) \in \mathbb{R}^{k\times k}$ is defined by (9), give us the vector field $F(x)$ defined by (8), (9).

Proof of Theorem 2.1 We first notice that statements (a) and (d) are direct consequences of definitions (7), (8), (9) and Fact 1. We next prove statements (b) and (c).

We first notice that definitions (9) and the fact that $g(x) \le 0$ imply that the following equality holds for all $x \in S$:

$$
\begin{aligned}
\nabla\theta(x)F(x) = {} & -\xi(x)R_1(x)\big(\xi(x)\big)' - \big(\operatorname{diag}\big(g(x)\big)v(x)\big)'R_2(x)\operatorname{diag}\big(g(x)\big)v(x) \\
& - \sum_{j=1}^{k} a_j(x)\big|g_j(x)\big|v_j^2(x) - \sum_{j=1}^{k} b_j(x)\big(\max\big(0, v_j(x)\big)\big)^2 \\
& - \sum_{j=1}^{k} c_j(x)\big(\max\big(0, v_j(x)\big)\big)^{2p_j+2},
\end{aligned}
\tag{16}
$$

where $\xi(x) = \nabla\theta(x)[H(x) - H(x)B'(x)Q^{-1}(x)B(x)H(x)]$. Clearly, Eq. (16) shows that $\nabla\theta(x)F(x) \le 0$ for all $x \in S$. We next investigate the nature of points $x \in S$ for which $\nabla\theta(x)F(x) = 0$. Equation (16) and the facts that $R_1(x) \in \mathbb{R}^{n\times n}$ is a positive definite matrix, $R_2(x) \in \mathbb{R}^{k\times k}$ is a positive semidefinite matrix, $a_i(x)$, $b_i(x)$, and $c_i(x)$ $(i = 1,\dots,k)$ are nonnegative functions with $b_i(x) + c_i(x) > 0$ for all $i = 1,\dots,k$ and $x \in S$, and at least one of the matrices $R_2(x) \in \mathbb{R}^{k\times k}$, $\operatorname{diag}(a(x)) \in \mathbb{R}^{k\times k}$ is positive definite, where $a(x) := (a_1(x),\dots,a_k(x))' \in \mathbb{R}^k$, and $R_3(x) \in \mathbb{R}^{k\times k}$ is defined by (9), show that $\nabla\theta(x)F(x) = 0$ is equivalent to the following equations:

$$\big(v(x)\big)^+ = 0, \tag{17}$$

$$g_j(x)v_j(x) = 0, \quad j = 1,\dots,k, \tag{18}$$

$$\nabla\theta(x)H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x) = 0. \tag{19}$$

We define

$$\mu = -v(x). \tag{20}$$

Definition (20) in conjunction with (17) and (18) implies that

$$\mu \geq 0 \quad \text{and} \quad \mu' g(x) = 0. \tag{21}$$

Using (19) and the identity $H^2(x) = H(x)$, we obtain:

$$H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)H(x)\big]\big(\nabla\theta(x)\big)' = 0. \tag{22}$$

Definitions (9) and (20), in conjunction with (22), imply that

$$H(x)\big(\nabla\theta(x) + \mu' B(x)\big)' = 0. \tag{23}$$

Equation (23), in conjunction with Fact 3 and (21), implies that conditions (6) hold. Therefore, $x \in \Phi$.

Thus, we have proved the implication $\nabla\theta(x)F(x) = 0 \Rightarrow x \in \Phi$.

Consequently, we have proved statement (b) and one of the implications of statement (c) (namely, the implication $F(x) = 0 \Rightarrow x \in \Phi$).

We will prove now the implication $x \in \Phi \Rightarrow F(x) = 0$. Suppose that $x \in \Phi$. Then there exist $\lambda_i \in \mathbb{R}$ $(i = 1, \dots, m)$ and $\mu_j \geq 0$ $(j = 1, \dots, k)$ such that conditions (6) hold, or in vector form,

$$\begin{aligned} &\big(\nabla\theta(x)\big)' + A'(x)\lambda + B'(x)\mu = 0, \\ &\mu' g(x) = 0. \end{aligned} \tag{24}$$

It follows from (24), Fact 1, and definitions (9) that

$$\begin{aligned} v(x) &= Q^{-1}(x)B(x)H(x)\big(\nabla\theta(x)\big)' = -Q^{-1}(x)B(x)H(x)\big(A'(x)\lambda + B'(x)\mu\big) \\ &= -Q^{-1}(x)B(x)H(x)B'(x)\mu. \end{aligned}$$

Using definition (7) and the above equality, we obtain

$$v(x) = -\mu - Q^{-1}(x)\,\mathrm{diag}\big(g(x)\big)\mu.$$

However, the facts that $g(x) \leq 0$, $\mu \geq 0$, and $\mu' g(x) = 0$ imply that $\mathrm{diag}(g(x))\mu = 0$. Consequently, it follows that $v(x) = -\mu$ and that (17) holds. Using (24), definition (8), and the facts that $v(x) = -\mu$, $\mathrm{diag}(g(x))\mu = 0$, and $(v(x))^+ = 0$, we obtain:

$$F(x) = -\big[H(x) - P'(x)Q(x)P(x)\big]R_1(x)\big[H(x) - P'(x)Q(x)P(x)\big]\big(\nabla\theta(x)\big)'.$$

Using definitions (7), (9), Fact 1, (24), the above equality, and the fact that $\mathrm{diag}(g(x))\mu = 0$, we get:

$$\begin{aligned} F(x) &= H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)R_1(x)H(x) \\ &\quad \times \big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)\big(A'(x)\lambda + B'(x)\mu\big) \\ &= H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)R_1(x)H(x) \\ &\quad \times \big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)B'(x)\mu \end{aligned}$$

$$\begin{aligned}
&= H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)R_1(x)H(x)B'(x) \\
&\quad \times \big[I_k - Q^{-1}(x)B(x)H(x)B'(x)\big]\mu \\
&= H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big]H(x)R_1(x)H(x)B'(x) \\
&\quad \times \big[I_k - Q^{-1}(x)\big(Q(x) + \mathrm{diag}\big(g(x)\big)\big)\big]\mu \\
&= -H(x)\big[I_n - B'(x)Q^{-1}(x)B(x)\big] \\
&\quad \times H(x)R_1(x)H(x)B'(x)Q^{-1}(x)\,\mathrm{diag}\big(g(x)\big)\mu = 0. \qquad (25)
\end{aligned}$$

We next turn to the proof of properties 1 and 2.

The local existence of the solution of the initial-value problem (11) with $x(0) = x_0$ is a direct consequence of properties (a), (d) and the Nagumo theorem (p. 27 in [17]). The global existence of the solution of the initial-value problem (11) with $x(0) = x_0$ follows from Theorem 1.2.3 (p. 27) in [17], assumption (H1), and the fact that $\theta(x(t))$ is nonincreasing (a direct consequence of property (b)). In fact, assumption (H1), in conjunction with the fact that $\theta(x(t))$ is nonincreasing, shows that $\{x(t) : t \geq 0\}$ is bounded.

As in the case of dynamical systems on $\mathbb{R}^n$, it follows that $\omega(x_0)$ is a compact, positively invariant set for system (11) (see [28]). The fact that $\theta(x(t))$ is nonincreasing implies that $\lim_{t\to+\infty}\theta(x(t)) = l = \inf\{\theta(x(t)) : t \geq 0\}$, which shows that $\omega(x_0) \subseteq \{x \in S : \theta(x) = l\}$. We next show that $\omega(x_0) \subseteq \Phi$. The inequality

$$\int_0^t \gamma\big(x(s)\big)ds \leq \theta(x_0) - l \quad \text{for all } t \geq 0, \qquad (26)$$

where $\gamma(x) := \nabla\theta(x)F(x)$, is a direct consequence of (16) and the definition $l := \inf\{\theta(x(t)) : t \geq 0\}$. Notice that the mapping $\mathbb{R}_+ \ni s \to \gamma(x(s))$ is uniformly continuous since $\{\dot{x}(t) = F(x(t)) : t \geq 0\}$ is bounded (a consequence of the fact that $\{x(t) : t \geq 0\}$ is bounded) and since the mapping $\mathbb{R}^n \ni x \to \gamma(x)$ is locally Lipschitz. Using (26) and applying Barbalat's lemma (see [29]), we conclude that $\lim_{s\to+\infty}\gamma(x(s)) = 0$. The validity of the implication $\nabla\theta(x)F(x) = \gamma(x) = 0 \Rightarrow x \in \Phi$ implies that $\omega(x_0) \subseteq \Phi$.

Finally, the fact that every strict local solution $x^* \in S$ of the NLP problem described by (1) and (2) follows from property (b) and the consideration of the Lyapunov function $V(x) = \theta(x) - \theta(x^*)$. The proof is complete. □

Remark 2.4 If there are no equality constraints (i.e., $h(x) \equiv 0$), then the proof of Theorem 2.1 shows that exactly the same results hold with $H(x) \equiv I_n$. Easy formulas can be obtained for nonlinear programming problems with no equality constraints. By selecting $R_1(x) = \sigma(x)I_n$, $R_2(x) \equiv 0 \in \mathbb{R}^{k\times k}$, $a_i(x) \equiv \sigma(x)$, $b_i(x) \equiv \gamma(x)$, $c_i(x) \equiv 0$ ($i = 1, \ldots, k$), where $\sigma, \gamma : \mathbb{R}^n \to]0, +\infty[$ are arbitrary C^1 functions, formulas (7), (8), and (9) give:

$$F(x) = \sigma(x)\big(\Psi(x)B(x) - I_n\big)\big(\nabla\theta(x)\big)' - \gamma(x)\Psi(x)\big(\big(\nabla\theta(x)\Psi(x)\big)'\big)^+, \qquad (27)$$

where $\Psi(x) := B'(x)(B(x)B'(x) - \mathrm{diag}(g(x)))^{-1}$.

3 Numerical Solutions of Nonlinear Programming Problems

As remarked in the Introduction and in [9], each system of differential equations that solves an NLP problem, when combined with a numerical scheme for solving Ordinary Differential Equations (ODEs), provides a numerical scheme for solving the NLP problem. However, when we try to apply a numerical scheme for the solution of (11), then we face the problem that the dynamical system (11) is not defined on $\mathbb{R}^n$ but on the closed set $S \subseteq \mathbb{R}^n$.

In the literature, projection schemes have been proposed (see [30, 31]). The projection schemes preserve the order of the applied numerical scheme (see [30, 31]) even if the projection on the closed set $S \subseteq \mathbb{R}^n$ is not exact. However, the application of a Runge–Kutta numerical scheme for (11) and its (approximate) projection on the closed set $S \subseteq \mathbb{R}^n$ means that the solution of an NLP problem is required at each time step. The corresponding NLP problem may be as difficult as the initial one, which means that this approach is not easily applicable (with the exception of the cases where the projection is easy, see [14]).

The key idea presented in this work is that the selection of the applied time step can be used for solving the above problems. First, we focus on the case without equality constraints.

The following theorem is the main result of this section, which guarantees the global convergence of the above algorithm.

Theorem 3.1 *Suppose that assumptions* (H1) *and* (H2) *hold for the NLP problem described by* (1) *and* (2) *with* $h(x) \equiv 0$. *Let* $R_1(x) \in \mathbb{R}^{n\times n}$ *be an arbitrary* C^1, *symmetric, and positive definite matrix,* $R_2(x) \in \mathbb{R}^{k\times k}$ *be an arbitrary* C^1, *symmetric, and positive semidefinite matrix,* $a_i(x)$, $b_i(x)$, *and* $c_i(x)$ $(i = 1, \ldots, k)$ *be arbitrary* C^1*nonnegative functions with* $b_i(x) + c_i(x) > 0$ *for all* $i = 1, \ldots, k$ *and* $x \in S$, *and let at least one of the matrices* $R_2(x) \in \mathbb{R}^{k\times k}$, $\mathrm{diag}(a(x)) \in \mathbb{R}^{k\times k}$ *be positive definite, where* $a(x) := (a_1(x), \ldots, a_k(x))' \in \mathbb{R}^k$. *Let* $F(x) \in \mathbb{R}^n$ *be the vector field defined by* (8), (9) *with* $H(x) \equiv I_n$. *Consider the following algorithm.*

Algorithm Given constants $r > 0$, $\varepsilon \in]0, r[$, and $\lambda \in]0, 1[$ and an initial point $x_0 \in S$, we follow the following steps for $i = 0, 1, \ldots$:

→ Step i: Calculate $F(x_i)$ using (8), (9). If $|F(x_i)| = 0$, then $x_{i+1} = x_i$. If $|F(x_i)| > 0$, then set $s^{(0)} = r$ and $p = 0$. Moreover, let $I(x_i) \subseteq \{1, \ldots, k\}$ be the set of all indices $j \in \{1, \ldots, k\}$ with $\max_{0\le s\le\varepsilon}(g_j(x_i + sF(x_i))) > -\varepsilon$.

→ Step p: Calculate $x_i^{(p)} = x_i + s^{(p)}F(x_i)$.

  → Solve $\min\{|y - x_i^{(p)}|^2 : \max_{j\in I(x_i)}(g_j(y)) \le 0\}$ for the case $I(x_i) \neq \emptyset$ or set $y = x_i^{(p)}$ for the case $I(x_i) = \emptyset$.

  → If $y \in S$ and $\theta(y) \le \theta(x_i) + \lambda s^{(p)}\nabla\theta(x_i)F(x_i)$, then set $x_{i+1} = y$, $i = i + 1$, and go to Step i.

  → If $y \notin S$ or $\theta(y) > \theta(x_i) + \lambda s^{(p)}\nabla\theta(x_i)F(x_i)$, then set $s^{(p+1)} = \frac{1}{2}s^{(p)}$, $p = p + 1$, and go to Step p.

Then every accumulation point x^* of the sequence x_i produced by the above algorithm satisfies $x^* \in \Phi$.

Remark 3.1 Clearly, the algorithm presented in Theorem 3.1 exploits the time step used for the estimate provided by the explicit Euler scheme $x_i^{(p)} = x_i + s^{(p)} F(x_i)$. The constant $r > 0$ is the maximum allowable time step. In most iterations, the algorithm will not require the solution of an NLP problem, provided that $\varepsilon > 0$ is sufficiently small. The fact that in most cases the Euler scheme is sufficient is explained by statement (d) of Theorem 2.1: for all $j = 1, \dots, k$, it holds that

$$\begin{aligned}\nabla g_j(x)F(x) &= g_j(x)\omega_j(x) \\ &\quad - \big(b_j(x) + c_j(x)\big(\max\big(0, v_j(x)\big)\big)^{2p_j}\big)\max\big(0, v_j(x)\big),\end{aligned} \tag{28}$$

where $\omega(x) := (\omega_1(x), \dots, \omega_k(x))'$ is given by

$$\begin{aligned}\omega(x) :=\ & Q^{-1}(x)B(x)H(x)R_1(x)\big[H(x) - P'(x)Q(x)P(x)\big]\big(\nabla\theta(x)\big)' \\ & - \big[I_k + Q^{-1}(x)\operatorname{diag}\big(g(x)\big)\big]\big(R_2(x)\operatorname{diag}\big(g(x)\big) - \operatorname{diag}\big(a(x)\big)\big)v(x) \\ & - Q^{-1}(x)R_3(x)\big(v(x)\big)^+.\end{aligned} \tag{29}$$

Using the estimate

$$g_j\big(x + sF(x)\big) \le g_j(x) + s\nabla g_j(x)F(x) + \frac{1}{2}s^2 K_j(x) \tag{30}$$

for $s \in [0, r]$, where $K_j(x) := \max\{F'(x)\nabla^2 g_j(x + \mu F(x))F(x) : \mu \in [0, r]\}$, it follows that $g_j(x + sF(x)) \le -\varepsilon$, provided that

$$\begin{aligned}&\varepsilon + g_j(x) + sg_j(x)\omega_j(x) - s\big(b_j(x) + c_j(x)\big(\max\big(0, v_j(x)\big)\big)^{2p_j}\big)\max\big(0, v_j(x)\big) \\ &\quad + \frac{1}{2}s^2 K_j(x) \le 0.\end{aligned}$$

The above inequality is satisfied for $s \in [0, \varepsilon]$ in many cases, provided that $\varepsilon > 0$ is sufficiently small. This explains the additional fact that the NLP problem

$$\min\Big\{\big|y - x_i^{(p)}\big|^2 : \max_{j \in I(x_i)}\big(g_j(y)\big) \le 0\Big\} \tag{31}$$

is much simpler than the initial NLP problem: the index set $I(x_i) \subseteq \{1, \dots, k\}$ is expected to be a set with small cardinal number.

Finally, as remarked in [14], the solution of the NLP problem (31) need not be exact: it suffices to find any $y \in \mathbb{R}^n$ with $\max_{j \in I(x_i)}(g_j(y)) \le 0$ and $|y - x_i^{(p)}| \le C|y^* - x_i^{(p)}|$, where $C \ge 1$ is a constant, and $y^* \in \mathbb{R}^n$ is a global solution of problem (31).

Proof of Theorem 3.1 Define the sets

$$\tilde{\Phi} := \big\{z \in \Phi, \theta(z) \le \theta(x_0)\big\}, \qquad \tilde{S} := \big\{z \in S, \theta(z) \le \theta(x_0)\big\}. \tag{32}$$

Notice that the set $\tilde{\Phi}$ is nonempty and compact (by virtue of property (b) of Theorem 2.1, it follows that the set $\tilde{\Phi}$ coincides with the closed nonempty set $\{x \in$

$S : |\nabla\theta(x)F(x)| = 0, \theta(x) \leq \theta(x_0)\}$ for which assumption (H1) implies that it is bounded; notice that the set $\tilde{\Phi}$ contains the global solution of the NLP problem described by (1) and (2)). Moreover, assumption (H1) guarantees that $\tilde{S}$ is nonempty and compact.

Let $d(x)$ be the distance of any point $x \in \mathbb{R}^n$ from the set $\tilde{\Phi}$, i.e.,

$$d(x) := \inf\bigl\{|x - y| : y \in \tilde{\Phi}\bigr\}. \tag{33}$$

Since the set $\tilde{\Phi}$ is nonempty and compact, it follows that the function $d(x)$ is well defined, is globally Lipschitz (with unit Lipschitz constant), and satisfies $d(x) > 0$ for all $x \notin \tilde{\Phi}$.

Claim For every $\eta > 0$, there exists a constant $\delta_\eta > 0$ such that the following implication holds:

$$\begin{aligned}&\text{“If } x \in \tilde{S}, d(x) \geq \eta, \text{ and } s \leq \delta_\eta, \text{ then } y \in S \text{ and } \theta(y) \leq \theta(x) + \lambda s \nabla\theta(x)F(x),\\ &\text{where } y = x + sF(x) \text{ for the case } I(x) = \emptyset. \text{ and } y \in \mathbb{R}^n \text{is}\\ &\text{any global solution of} \min\{|y - x - sF(x)|^2 : \max_{j \in I(x)}(g_j(y)) \leq 0\}\\ &\text{for the case } I(x) \neq \emptyset.\text{”}\end{aligned} \tag{34}$$

The proof of the claim can be found in the Appendix.

By virtue of implication (34) and the fact that $\theta(x_i)$ is nonincreasing, the algorithm is well defined (i.e., for each iteration i, the variable p takes finite values). Let $s_i \in]0, r]$ $(i = 0, 1, \ldots)$ be the applied $s^{(p)}$ for which $y \in S$ and $\theta(y) \leq \theta(x_i) + \lambda s^{(p)}\nabla\theta(x_i)F(x_i)$ for the case $x_i \notin \Phi$ and $s_i = 0$ for the case $x_i \in \Phi$. For every $i = 0, 1, \ldots,$ it holds that

$$\theta(x_{i+1}) \leq \theta(x_i) + \lambda s_i \nabla\theta(x_i)F(x_i). \tag{35}$$

Assumption (H1) in conjunction with (35) guarantees that the sequence x_i is bounded with $\theta(x_i) \leq \theta(x_0)$ for all $i = 0, 1, \ldots$. Moreover, implication (34) implies the following inequality for every $i = 0, 1, \ldots$ with $x_i \notin \Phi$:

$$s_i \geq \delta_\eta/2 \quad \text{for all } \eta \in]0, d(x_i)], \tag{36}$$

where $\delta_\eta > 0$ is the constant involved in implication (34). If x^* is a global solution of the NLP problem described by (1) and (2), then by applying (35) inductively we conclude that the following inequality holds for $i = 1, 2, \ldots$:

$$\lambda \sum_{l=0}^{i-1} s_l \bigl|\nabla\theta(x_l)F(x_l)\bigr| \leq \theta(x_0) - \theta\bigl(x^*\bigr). \tag{37}$$

The above inequality implies that $\lim_{i\to+\infty}(s_i|\nabla\theta(x_i)F(x_i)|) = 0$.

In order to prove that every accumulation point x^* of the sequence x_i produced by the above algorithm satisfies $x^* \in \Phi$, we will use a contradiction argument. Consider a converging subsequence of the sequence x_i, which we again denote by x_i. Let x^*

be the unique accumulation point of the subsequence x_i. We assume that $x^* \notin \Phi$. By continuity and using property (b) of Theorem 2.1, we have

$$\lim_{i\to+\infty}\left(\left|\nabla\theta(x_i)F(x_i)\right|\right)=\left|\nabla\theta\left(x^*\right)F\left(x^*\right)\right|>0.$$

Since

$$\lim_{i\to+\infty}\left(s_i\left|\nabla\theta(x_i)F(x_i)\right|\right)=0,$$

we conclude that

$$\lim_{i\to+\infty}(s_i)=0. \tag{38}$$

Since $x^* \notin \tilde{\Phi}$, it follows that $\lim_{i\to+\infty}(d(x_i))=d(x^*)>0$. Therefore, there exists $\eta>0$ such that $d(x_i)\geq\eta$ for sufficiently large i. Inequality (36) gives $s_i\geq\delta_\eta/2$, where $\delta_\eta>0$ is the constant involved in implication (34). This contradicts (38). The proof is complete. □

Remark 3.2 Notice that when equality constraints are present, then Theorem 3.1 is still useful under the following assumption:

(H3) There exist positive integers n_1, n_2 with $n_1+n_2=n$ and a smooth function $\phi:\mathbb{R}^{n_1}\to\mathbb{R}^{n_2}$ such that for every $\xi\in\mathbb{R}^{n_1}$, it holds that $h(x)=0$, where $x=(\xi,\phi(\xi))$.

Indeed, under assumption (H3), for all $\xi\in\mathbb{R}^{n_1}$, we may define:

$$\tilde{\theta}(\xi) = \theta(x) \quad \text{with } x=\bigl(\xi,\phi(\xi)\bigr), \tag{39}$$

$$\tilde{g}_j(\xi) := g_j(x) \quad \text{with } x=\bigl(\xi,\phi(\xi)\bigr) \text{ for } j=1,\dots,k. \tag{40}$$

We can also define $\tilde{F}(\xi)$ to be the vector field that is made up from the first n_1 components of the vector field $F(x)$ defined by (8), (9) and evaluated at $x=(\xi,\phi(\xi))$. Then, we can apply Theorem 3.1 with $\tilde{\theta},\tilde{g}_j$ $(j=1,\dots,k)$, and $\tilde{F}$ in place of θ, g_j $(j=1,\dots,k)$, and F.

Remark 3.3 The algorithm may be modified in a straightforward way for other higher-order explicit Runge–Kutta numerical schemes. This is meaningful only when the vector field $F(x)$ has sufficient regularity. More specifically, the term $x_i^{(p)}=x_i+s^{(p)}F(x_i)$ may be replaced by $x_i^{(p)}=N(s^{(p)},x_i)$ with an appropriate mapping $N(s^{(p)},x_i)$, which is a characteristic of the Runge–Kutta scheme, and the definition of the set $I(x_i)\subseteq\{1,\dots,k\}$ is modified to be the set of all indices $j\in\{1,\dots,k\}$ with $\max_{0\leq s\leq\varepsilon}(g_j(N(s,x_i)))>-\varepsilon$. However, it should be noticed that for higher-order explicit Runge–Kutta schemes, the vector field $F(x)$ must be computed for various points. Since $F(x)$ is defined only in a neighborhood of the set S, it may be needed to restrict the time step $s^{(p)}$.

Remark 3.4 Using (28), (29), and (30) and assuming that for all $j = 1, \dots, k$, there exist positive continuous functions $q_j : S \to]0, +\infty[$ and $Q_j : S \to]0, +\infty[$ ($j = 1, \dots, k$) such that the following inequalities hold for all $j = 1, \dots, k$ and $x \in S$:

$$\begin{aligned} K_j(x) &\le -g_j(x)q_j(x) \\ &\quad + Q_j(x)\big(b_j(x) + c_j(x)\big(\max\big(0, v_j(x)\big)\big)^{2p_j}\big) \max\big(0, v_j(x)\big), \end{aligned} \tag{41}$$

where $K_j(x) := \max\{F'(x)\nabla^2 g_j(x + \mu F(x))F(x) : \mu \in [0, r]\}$ ($j = 1, \dots, k$), we can conclude that

$$g_j\big(x + sF(x)\big) \le 0 \quad \text{for all } j = 1, \dots, k \text{ and } x \in S, \tag{42}$$

provided that

$$s \le \min_{j=1,\dots,k}\left(r, \frac{\omega_j(x) + \sqrt{\omega_j^2(x) + 2q_j(x)}}{q_j(x)}, \frac{2}{Q_j(x)}\right). \tag{43}$$

Define

$$\begin{aligned} &s_g(x) := \sup\Big\{s \in [0, r] : \max_{\substack{0 \le l \le s \\ j = 1, \dots, k}} \big(g_j\big(x + lF(x)\big)\big) \le 0\Big\} \\ &\text{for all } x \in S \end{aligned} \tag{44}$$

and notice that (43) implies

$$s_g(x) \ge \min_{j=1,\dots,k}\left(r, \frac{\omega_j(x) + \sqrt{\omega_j^2(x) + 2q_j(x)}}{q_j(x)}, \frac{2}{Q_j(x)}\right)$$

for all $x \in S$. Using the analogue of inequality (30) for $\theta(x)$, i.e., the inequality

$$\theta\big(x + sF(x)\big) \le \theta(x) + s\nabla\theta(x)F(x) + s^2 K_\theta(x)/2 \tag{45}$$

for $s \in [0, r]$, where $K_\theta(x) := \max\{F'(x)\nabla^2\theta(x + \mu F(x))F(x) : \mu \in [0, r]\}$, we can conclude that the best possible choice for the time step $s \in [0, r]$ is given by

$$\begin{aligned} &s = \min\left(s_g(x), \frac{|\nabla\theta(x)F(x)|}{K_\theta(x)}\right) \quad \text{for the case } K_\theta(x) > 0 \text{ and} \\ &s = s_g(x) \text{ for the case } K_\theta(x) \le 0. \end{aligned} \tag{46}$$

Inequalities (41) hold automatically for arbitrary positive continuous functions q_j, $Q_j : S \to]0, +\infty[$ ($j = 1, \dots, k$) when all functions $g_j(x)$ ($j = 1, \dots, k$) are linear.

Therefore, if (41) holds, then we can compute the sequence $x_{i+1} = M(x_i)$, where $M(x) = x + sF(x)$, and $s \in [0, r]$ is given by (46). We notice that the implementation of an approximation of the numerical scheme $x_{i+1} = M(x_i)$ does not necessarily require knowledge of the functions $K_j(x)$, $q_j : S \to]0, +\infty[$, $Q_j : S \to]0, +\infty[$

$(j=1,\dots,k)$, and $K_\theta(x)$: evaluating $g(x+sF(x))$ and $\theta(x+sF(x))$ for certain $s\in[0,r]$ can give us estimates of $K_j(x)$ and $K_\theta(x)$ satisfying (30) and (45). Using these estimates, we can estimate $s_g(x)$. Consequently, the algorithm is implemented as follows.

Algorithm Given constants $r>0$, $\varepsilon>0$, and $\lambda\in]0,1/2]$ and an initial point $x_0\in S$, we follow the following steps for $i=0,1,\dots$:

$\rightarrow$ Step i: Calculate $F(x_i)$ using (8) and (9). If $|F(x_i)|=0$, then $x_{i+1}=x_i$.
If $|F(x_i)|>0$, then calculate
$K_j^{(0)}=\max(\varepsilon, 2r^{-2}(g_j(x_i+rF(x_i))-g_j(x_i)-r\nabla g_j(x_i)F(x_i)))$
for $j=1,\dots,k$,
$K_\theta^{(0)}=\max(\varepsilon, 2r^{-2}(\theta(x+rF(x))-\theta(x)-r\nabla\theta(x)F(x)))$,
and set $p=0$.

$\rightarrow$ Step p: Compute $s_j^{(p)}=\min(r, \frac{-\nabla g_j(x_i)F(x_i)+\sqrt{|\nabla g_j(x_i)F(x_i)|^2-2K_j^{(p)}g_j(x_i)}}{K_j^{(p)}})$
for $j=1,\dots,k$ and $s^{(p)}=\min_{j=1,\dots,k}\min(s_j^{(p)}, \frac{|\nabla\theta(x_i)F(x_i)|}{K_\theta^{(p)}})$. Calculate the vector
$x_i^{(p)}=x_i+s^{(p)}F(x_i)$.

$\rightarrow$ If $x_i^{(p)}\in S$ and $\theta(x_i^{(p)})\le\theta(x_i)+\lambda s^{(p)}\nabla\theta(x_i)F(x_i)$, then set $x_{i+1}=x_i^{(p)}$, $i=i+1$, and go to Step i.

$\rightarrow$ If $x_i^{(p)}\notin S$ or $\theta(x_i^{(p)})>\theta(x_i)+\lambda s^{(p)}\nabla\theta(x_i)F(x_i)$, then set $K_j^{(p+1)}=K_j^{(p)}+\varepsilon$ $(j=1,\dots,k)$, $K_\theta^{(p+1)}=K_\theta^{(p)}+\varepsilon$, $p=p+1$, and go to Step p.

Using exactly the same procedure as in the proof of Theorem 3.1, we can conclude that every accumulation point x^* of the sequence x_i produced by the above algorithm satisfies $x^*\in\Phi$, provided that assumptions (H1), (H2), and (41) hold. However, numerical experiments show that the algorithm can converge even when (41) does not hold.

4 Examples

In order to demonstrate the performance of the proposed algorithms, we have used two examples from [8] and one linear programming problem.

The first example is dealing with the solution of the problem

$$\begin{aligned} &\min\theta(x)=x_1^2+2x_2^2+x_1x_2-6x_1-2x_2-12x_3\\ &\text{s.t.}\quad h(x)=x_1+x_2+x_3-2=0,\\ &\qquad g(x)=[-x_1+2x_2-3,-x_1,-x_2,-x_3]'\le 0. \end{aligned}\tag{47}$$

It can be shown that assumptions (H1), (H2) hold for this problem. Moreover, assumption (H3) holds with $\xi=(x_1,x_2)'\in\mathbb{R}^2$ and $\phi(\xi)=2-x_1-x_2$. Since the inequality constraints are linear, we are in a position to use the algorithm of Remark 3.3. We have used the algorithm of Remark 3.3 with

$$\begin{aligned} &R_1(x)=\sigma I_3,\quad R_2(x)=0\in\mathbb{R}^{4\times4},\quad a_i(x)\equiv1,\quad b_i(x)\equiv1,\quad c_i(x)\equiv0\\ &(i=1,\dots,4),\quad r=1,\quad \lambda=0.1,\quad \varepsilon=10^{-6}, \end{aligned}\tag{48}$$

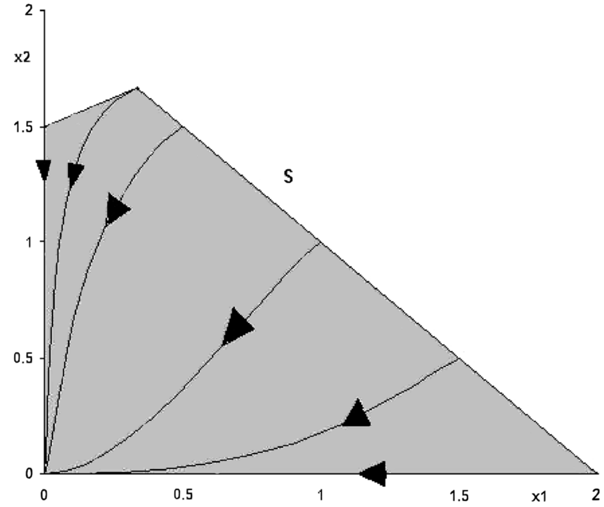

Fig. 1 The projection of the phase diagram on the x_1–x_2 plane for the dynamical system (11), where F is defined by (8), (9), for problem (47) with $R_1(x) = 2I_3$, $R_2(x) = 0 \in \mathbb{R}^{4\times 4}$, $a_i(x) \equiv 1$, $b_i(x) \equiv 1$, $c_i(x) \equiv 0$ $(i = 1, \dots, 4)$

where $\sigma > 0$. It was found that for all initial points in the feasible set and for every $\sigma \in [0.01, 200]$, the algorithm converges at the point $(x_1, x_2, x_3) = (0, 0, 2)$ in no more than three iterations. In this case, the convergence of the algorithm of Remark 3.3 is very fast.

The convergence in no more than three iterations was observed to the solution of the linear programming problem

$$\min \theta(x) = -2x_1 + x_2 - x_3,$$
$$\text{s.t.} \quad g(x) = \begin{bmatrix} 3x_1 + x_2 + x_3 - 180 \\ x_1 - x_2 + 2x_3 - 30 \\ x_1 + x_2 - x_3 - 60 \\ -x_1 \\ -x_2 \\ -x_3 \end{bmatrix} \le 0. \tag{49}$$

Figure 1 shows the projection of the phase diagram on the x_1–x_2 plane for the dynamical system (11), where F is defined by (8), (9), for problem (47) with $R_1(x) = \sigma I_3$, $\sigma = 2$, $R_2(x) = 0 \in \mathbb{R}^{4\times 4}$, $a_i(x) \equiv 1$, $b_i(x) \equiv 1$, $c_i(x) \equiv 0$ $(i = 1, \dots, 4)$. Figure 1 was created by solving numerically system (11) with the explicit Euler method and time step 0.01.

Again, since the inequality constraints are linear, we are in a position to use the algorithm of Remark 3.3. We have used the algorithm of Remark 3.3 with

$$\begin{aligned} &R_1(x) = \sigma I_3, \quad R_2(x) = 0 \in \mathbb{R}^{6\times 6}, \quad a_i(x) \equiv 1, \quad b_i(x) \equiv 1, \quad c_i(x) \equiv 0 \\ &(i = 1, \dots, 6), \quad r = 10^3, \quad \lambda = 0.1, \quad \varepsilon = 10^{-6}, \end{aligned} \tag{50}$$

where $\sigma > 0$. It was found that for all initial points in the feasible set and for every $\sigma \in [0.1, 20]$, the algorithm converges at the point $(x_1, x_2, x_3) = (45, 15, 0)$ in no more than three iterations.

The third example is dealing with the Rosen–Suzuki problem

$$\begin{aligned} &\min \theta(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4 \\ &\text{s.t.} \quad h(x) = 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5 = 0, \\ &\qquad g(x) = \begin{bmatrix} x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8 \\ x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10 \end{bmatrix} \le 0. \end{aligned} \tag{51}$$

It can be shown that assumptions (H1) and (H2) hold for this problem. Moreover, assumption (H3) holds with $\xi = (x_1, x_2, x_3)' \in \mathbb{R}^3$ and $\phi : \mathbb{R}^3 \to \mathbb{R}$ defined by $\phi(\xi) = 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - 5$. The vector field $F(x)$ defined by (8), (9) is constructed with

$$\begin{aligned} &R_1(x) = \sigma I_4, \quad R_2(x) = 0 \in \mathbb{R}^{2\times 2}, \quad a_i(x) \equiv 1, \quad b_i(x) \equiv 1, \quad c_i(x) \equiv 0 \\ &(i = 1, 2), \end{aligned} \tag{52}$$

where $\sigma > 0$. This is a problem with nonlinear inequality constraints. Therefore, we cannot assume the validity of (41). Indeed, there are points in the feasible set with $g_1(x) = 0$, $\max(0, v_1(x)) = 0$, and for which $\tilde{g}_1(\xi + s\tilde{F}(\xi)) > 0$ for $s > 0$, where $\tilde{g}_1$ is defined by (40), and $\tilde{F}(\xi)$ is the vector field that is made up from the first three components of the vector field $F(x)$ evaluated at $x = (\xi, \phi(\xi))$. Such a point is $x = (-1, -1, 2, 1)' \in \mathbb{R}^4$, and it is clear that we cannot apply the algorithm of Remark 3.3 at any one of these points. However, the algorithm of Remark 3.3 may be applied with other initial points: for example, if the numerical algorithm of Remark 3.3 is applied with $\sigma = 0.2$, $r = 1$, $\lambda = 0.1$, and $\varepsilon = 10^{-6}$ to the initial point $x_0 = (-0.9, -1, 2, 0.82)' \in \mathbb{R}^4$ (which is close to the "problematic point" $x = (-1, -1, 2, 1)' \in \mathbb{R}^4$), then the produced sequence reaches the neighborhood $N = \{|x - x^*| \le 10^{-5}\}$ with $x^* = (0, 1, 2, -1)' \in \mathbb{R}^4$ in 33 iterations. It was also found that different values of $\sigma > 0$ and $r > 0$ affect the convergence properties of the algorithm. For example, the values lower than 1 for $r > 0$ and higher than 0.5 for $\sigma > 0$ require more iterations for convergence. The algorithm of Remark 3.3 performs well from almost all points of the feasible set: for example, if the algorithm of Remark 3.3 is applied with $\sigma = 0.2$, $r = 1$, $\lambda = 0.1$, and $\varepsilon = 10^{-6}$ to the initial point $x_0 = (-1, -1, -2, 1)' \in \mathbb{R}^4$, then the produced sequence reaches the neighborhood $N = \{|x - x^*| \le 10^{-5}\}$ with $x^* = (0, 1, 2, -1)' \in \mathbb{R}^4$ in 47 iterations. For the initial point $x_0 = (-1, -1, 2, 1)' \in \mathbb{R}^4$, we can apply the algorithm of Theorem 3.1. If the algorithm of Theorem 3.1 is applied with $\sigma = 0.2$, $r = 0.5$, $\lambda = 0.1$, and $\varepsilon = 10^{-6}$ to the initial point $x_0 = (-1, -1, 2, 1)' \in \mathbb{R}^4$, then the produced sequence reaches the neighborhood $N = \{|x - x^*| \le 10^{-5}\}$ with $x^* = (0, 1, 2, -1)' \in \mathbb{R}^4$ in 39 iterations.

In general, the convergence of the proposed algorithms is linear. For superlinear convergence, we can either use different selections for $R_1(x) \in \mathbb{R}^{4\times 4}$, $R_2(x) \in \mathbb{R}^{2\times 2}$, $a_i(x)$, $b_i(x)$, $c_i(x)$ $(i = 1, 2)$ or use a different algorithm once we are close to the set Φ. The quantity $|F(x)|$ can be used in order to signal the approach of a neighborhood of Φ.

5 Conclusions

In this work, we have showed that given a nonlinear programming problem, it is possible, under mild assumptions, to construct a family of dynamical systems defined on the feasible set of the given problem so that: (a) the equilibrium points are the unknown critical points of the problem, (b) each dynamical system admits the objective function of the problem as a Lyapunov function, and (c) explicit formulae are available without involving the unknown critical points of the problem. The construction of the family of dynamical systems is based on the Control Lyapunov Function methodology, which is used in mathematical control theory for the construction of stabilizing feedback.

The knowledge of a dynamical system with the previously mentioned properties allows the construction of algorithms that guarantee the global convergence to the set of the critical points. However, we make no claim about the effectiveness of the proposed algorithms. The topic of the numerical solution of NLPs is a mature topic, and it is clear that other algorithms have much better characteristics than the algorithms proposed in this paper. However, the theory used for the construction of the algorithm is different from other existing algorithms. The algorithms contained in this work are derived by using concepts of dynamical system theory and mathematical control theory. Many more remain to be studied for the improvement of the resulting algorithms and for the relaxation of the assumption of the linear independence constraint qualification.

The obtained results have nothing to do with extremum seeking (see [32, 33]) but may open the way of using different extremum seeking control schemes in the future for constrained problems. Finally, it may be beneficial to compare the proposed algorithms with other global sequential quadratic programming algorithms (see [25, 26] and references therein); this is a future research topic.

Acknowledgements The author would like to thank Prof. Lars Grüne for his valuable comments and suggestions. The contribution of Prof. Lars Grüne to this work is major. George Athanasiou helped very much with the typesetting of the paper.

Appendix

Proof of Lemma 2.1 First notice that by virtue of Fact 2, the following equality holds for all $\xi = (\xi_1, \dots, \xi_k)' \in \mathbb{R}^k$:

$$\xi' Q(x)\xi = \left|H(x)B'(x)\xi\right|^2 + \sum_{j=1}^{k} \left|g_j(x)\right|\xi_j^2. \tag{53}$$

Equation (53) implies that $Q(x) \in \mathbb{R}^{k\times k}$ is positive semidefinite. Suppose that $Q(x)$ is not positive definite. Then there exists a nonzero vector $\xi = (\xi_1, \dots, \xi_k)' \in \mathbb{R}^k$ with $\xi' Q(x)\xi = 0$. Consequently, Eq. (53) shows that we must have $H(x)B'(x)\xi = 0$ and

$\xi_j = 0$ for all $j = 1, \ldots, k$ with $g_j(x) < 0$. Fact 3 implies that there exists $\lambda \in \mathbb{R}^m$ such that $B'(x)\xi = A'(x)\lambda$. The previous equality implies that

$$\sum_{j=1}^{k} \xi_j \nabla g_j(x) - \sum_{i=1}^{m} \lambda_i \nabla h_i(x) = 0. \tag{54}$$

Since $\xi_j = 0$ for all $j = 1, \ldots, k$ with $g_j(x) < 0$ and since $\xi = (\xi_1, \ldots, \xi_k)' \in \mathbb{R}^k$ is non-zero, we conclude from (54) that assumption (H2) is violated. The proof is complete. □

Proof of the claim made in the proof of Theorem 3.1 Let $\eta > 0$ be arbitrary. We distinguish two cases.

Case 1: The set $\{x \in \tilde{S} : d(x) \geq \eta\}$ is empty, where $\tilde{S} \subseteq S$ is defined by (32), and $d(x)$ is defined by (33). In this case, implication (34) holds trivially with arbitrary $\delta_\eta > 0$.

Case 2: The set $\{x \in \tilde{S} : d(x) \geq \eta\}$ is nonempty.

The continuity of the distance function $d(x)$ and the compactness of $\tilde{S} \subseteq S$ implies that the set $\{x \in \tilde{S} : d(x) \geq \eta\}$ is compact. Statements (b) and (c) of Theorem 2.1 guarantee that the quantity

$$\rho := \min\left\{ \frac{|\nabla\theta(x)F(x)|}{|F(x)|(|\nabla\theta(x)| + |F(x)|)} : x \in \tilde{S}, d(x) \geq \eta \right\} \tag{55}$$

is well defined and positive. Let $x \in \tilde{S}$ with $d(x) \geq \eta$ be an arbitrary point. We denote by $z(t)$ the unique solution of the initial-value problem $\dot{z} = F(z)$ with $z(0) = x$. We also notice that the vector field F as defined by (8), (9) is locally Lipschitz on a neighborhood of S. By the compactness of $\tilde{S} \subseteq S$, there exists a constant $L \geq 0$ such that

$$\left|F(y) - F(x)\right| \leq L|y - x| \quad \text{for all } x, y \in \tilde{S}. \tag{56}$$

Inequality (56), the fact that $z(s)$ belongs to the compact set of all $z \in S$ with $\theta(z) \leq \theta(x)$, and standard arguments show that the following inequality holds for all $s \geq 0$:

$$\left|z(s) - x - sF(x)\right| \leq Le^{Ls}\frac{s^2}{2}|F(x)|. \tag{57}$$

Next, we notice that the problem

$$\min\left\{\left|y - x - sF(x)\right|^2 : \max_{j \in I(x)} \left(g_j(y)\right) \leq 0\right\} \tag{58}$$

with $I(x) \neq \emptyset$ admits at least one solution (since the mapping $y \to |y - x - sF(x)|^2$ is radially unbounded). Any solution $y \in \mathbb{R}^n$ of problem (58) with $I(x) \neq \emptyset$ satisfies for all $s \geq 0$:

$$\left|y - x - sF(x)\right| \leq \left|z(s) - x - sF(x)\right| \quad \text{and} \quad \left|y - x - sF(x)\right| \leq s|F(x)|. \tag{59}$$

The above inequalities hold trivially for the case $I(x)=\emptyset$ and $y=x+sF(x)$. Define:

$$q := \max\left\{\sum_{j=1}^{k}\left|\nabla g_j(z)\right| : z\in\mathbb{R}^n, |z|\le\beta\right\},$$
$$\text{where } \beta := \max\left\{|x| + 2r\left|F(x)\right| : x\in\tilde{S}, d(x)\ge\eta\right\}, \tag{60}$$

$$Q := \frac{Le^{Lr}}{2} + 2\max\left\{\left|\nabla^2\theta(z)\right| : z\in\mathbb{R}^n, |z|\le\beta\right\}. \tag{61}$$

We will show next that implication (34) holds with $\delta_\eta > 0$ defined by

$$\delta_\eta := \min\left(\varepsilon, \left(\frac{2\varepsilon}{1+qL\gamma e^{Lr}}\right)^{1/2}, \frac{(1-\lambda)\rho}{1+Q}\right),$$
$$\text{where } \gamma := \max\left\{\left|F(x)\right| : x\in\tilde{S}, d(x)\ge\eta\right\}. \tag{62}$$

First, we show the implication

$$\text{“If } x\in\tilde{S}, d(x)\ge\eta \text{ and } s\le\delta_\eta, \text{ then } y\in S.\text{”} \tag{63}$$

It suffices to show that $g_j(y)\le 0$ for all $j\notin I(x)$. Notice that by the definition of the set $I(x)\subseteq\{1,\dots,k\}$ it follows that $g_j(x+sF(x))\le-\varepsilon$ for all $s\in[0,\varepsilon]$ and $j\notin I(x)$. Using (59), we obtain for all $s\in[0,r]$:

$$g_j(y)\le g_j\bigl(x+sF(x)\bigr) + \left|y-x-sF(x)\right|\max\left\{\left|\nabla g_j(z)\right| : |z-x|\le 2r\left|F(x)\right|\right\}. \tag{64}$$

Since $g_j(x+sF(x))\le-\varepsilon$ for all $s\in[0,\varepsilon]$ and $j\notin I(x)$, we obtain from (57), (59), (64) and (60) for all $s\in[0,\varepsilon]$ and $j\notin I(x)$:

$$g_j(y)\le -\varepsilon + qL\left|F(x)\right|e^{Lr}\frac{s^2}{2}. \tag{65}$$

Inequality (65) in conjunction with definition (62) shows that $g_j(y)\le 0$ for all $j\notin I(x)$, provided that $s\le\delta_\eta$.

By implication (63) we are left with the task of proving the inequality $\theta(y)\le\theta(x)+\lambda s\nabla\theta(x)F(x)$ for all $s\le\delta_\eta$ and $x\in\tilde{S}$ with $d(x)\ge\eta$. Using (59), we obtain for all $s\in[0,r]$:

$$\theta(y)\le\theta(x)+s\nabla\theta(x)F(x)+\nabla\theta(x)\bigl(y-x-sF(x)\bigr)+K|y-x|^2, \tag{66}$$

where $K=\frac{1}{2}\max\{|\nabla^2\theta(z)| : |z|\le\beta\}$. The derivation of (66) follows from majorizing the second derivative of the mapping $w\to p(w)=\theta(x+w(y-x))$ and using the inequality $|y-x|\le 2s|F(x)|$ (which is a direct consequence of (59)). It follows from (59), (57), (66), and (61) that the following inequality holds for all $s\in[0,r]$:

$$\theta(y)\le\theta(x)+s\nabla\theta(x)F(x)+Qs^2\bigl(\left|\nabla\theta(x)\right|\left|F(x)\right|+\left|F(x)\right|^2\bigr). \tag{67}$$

Definitions (55), (62) and inequality (67) allow us to conclude that the inequality $\theta(y)\le\theta(x)+\lambda s\nabla\theta(x)F(x)$ holds for all $s\le\delta_\eta$. The proof is complete. $\square$

References

1. Antipin, A.S.: Minimization of convex functions on convex sets by means of differential equations. Differ. Equ. **30**(9), 1365–1375 (1994)
2. Cabot, A.: The steepest descent dynamical system with control. Applications to constrained minimization. ESAIM Control Optim. Calc. Var. **10**, 243–258 (2004)
3. Evtushenko, Y.G., Zhadan, V.: Barrier-projective methods for nonlinear programming. Comput. Math. Math. Phys. **34**(5), 579–590 (1994)
4. Goh, B.S.: Algorithms for unconstrained optimization problems via control theory. J. Optim. Theory Appl. **92**(3), 581–604 (1997)
5. Pan, P.Q.: New ODE methods for equality constrained optimization equations. J. Comput. Math. **10**(1), 77–92 (1992)
6. Tanabe, K.: An algorithm for constrained maximization in nonlinear programming. J. Oper. Res. Soc. Jpn. **17**, 184–201 (1974)
7. Yamashita, H.: A differential equation approach to nonlinear programming. Math. Program. **18**, 155–168 (1980)
8. Zhou, L., Wu, Y., Zhang, L., Zhang, G.: Convergence analysis of a differential equation approach for solving nonlinear programming problems. Appl. Math. Comput. **184**, 789–797 (2007)
9. Brown, A.A., Bartholomew-Biggs, M.C.: ODE versus SQP methods for constrained optimization. J. Optim. Theory Appl. **62**(3), 371–386 (1989)
10. Xia, Y., Wang, J.: A recurrent neural network for nonlinear convex optimization subject to nonlinear inequality constraints. IEEE Trans. Circuits Syst. **51**(7), 1385–1394 (2004)
11. Xia, Y., Wang, J.: A recurrent neural network for solving nonlinear convex programs subject to linear constraints. IEEE Trans. Neural Netw. **16**(2), 379–386 (2005)
12. Xia, Y., Feng, G., Wang, J.: A novel recurrent neural network for solving nonlinear optimization problems with inequality constraints. IEEE Trans. Neural Netw. **19**(8), 1340–1353 (2008)
13. Wena, U.P., Lan, K.M., Shih, H.S.: A review of Hopfield neural networks for solving mathematical programming problems. Eur. J. Oper. Res. **198**(3), 675–687 (2009)
14. Grüne, L., Karafyllis, I.: Lyapunov function based step size control for numerical ODE solvers with application to optimization algorithms. In: Hüper, K., Trumpf, J. (eds.) Mathematical System Theory, pp. 183–210 (2013) Festschrift in Honor of Uwe Helmke on the Occasion of his 60th Birthday. CreateSpace, Smart-Link: http://users.cecs.anu.edu.au/~trumpf/UH60Festschrift.pdf
15. Karafyllis, I., Grüne, L.: Feedback stabilization methods for the numerical solution of systems of ordinary differential equations. Discrete Contin. Dyn. Syst., Ser. B **16**(1), 283–317 (2011)
16. Helmke, U., Moore, J.B.: Optimization and Dynamical Systems, 2nd edn. Springer, Berlin (1996)
17. Aubin, J.P.: Viability Theory. Birkhäuser, Boston (1991)
18. Artstein, Z.: Stabilization with Relaxed Controls. Nonlinear Anal., Theory Methods Appl. **7**, 1163–1173 (1983)
19. Freeman, R.A., Kokotovic, P.V.: Robust nonlinear control design-state space and Lyapunov techniques. Birkhäuser, Boston (1996)
20. Karafyllis, I., Jiang, Z.-P.: Stability and Stabilization of Nonlinear Systems. Communications and Control Engineering. Springer, London (2011)
21. Sontag, E.D.: A "Universal" construction of Artstein's theorem on nonlinear stabilization. Syst. Control Lett. **13**, 117–123 (1989)
22. Mangasarian, O.L., Fromovitz, S.: The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. J. Math. Anal. Appl. **17**, 37–47 (1967)
23. Andreani, R., Echagüe, C.E., Schuverdt, M.L.: Constant-rank condition and second order constraint qualification. J. Optim. Theory Appl. **146**(2), 255–266 (2010)
24. Scholtes, S., Stohr, M.: How Stringent is the linear independence assumption for mathematical programs with complementarity constraints? Math. Oper. Res. **26**(4), 851–863 (2001)
25. Solodov, M.V.: Global convergence of an SQP method without boundedness assumptions on any of the iterative sequences. Math. Program., Ser. A **118**, 1–12 (2009)
26. Fiacco, A.V., McCormick, G.P.: Nonlinear Programming: Sequential Unconstrained Minimization Techniques. Wiley, New York (1968)
27. Avriel, M.: Nonlinear Programming: Analysis and Methods. Dover, New York (2003)
28. Stuart, A.M., Humphries, A.R.: Dynamical Systems and Numerical Analysis. Cambridge University Press, Cambridge (1998)
29. Khalil, H.K.: Nonlinear Systems, 2nd edn. Prentice-Hall, New York (1996)

30. Hairer, E., Norsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I Nonstiff Problems, 2nd edn. Springer, Berlin (1993)
31. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (2002)
32. Ghaffari, A., Krstic, M., Nesic, D.: Multivariable Newton-based extremum seeking. Automatica **48**, 1759–1767 (2012)
33. Liu, S.-J., Krstic, M.: Stochastic Averaging and Stochastic Extremum Seeking. Springer, Berlin (2012)