Enhancing Transient Stability of DC Microgrid by Enlarging the Region of Attraction Through Nonlinear Polynomial Droop Control

Bernardo Severino and Kai Strunz

***Abstract*— A methodology for enlarging the region of attraction (ROA) of a DC microgrid with constant power loads (CPLs) is proposed. The enlargement is achieved through the optimal design of a polynomial droop controller. The design of this nonlinear controller is done by solving a sum of squares (SOS) program. The proposed SOS program allows finding a Lyapunov function that serves to estimate the ROA. Therefore, the coefficients of the polynomial droop controller are optimized to maximize that estimate. Using the SOS approach, the estimate of the ROA exceeds the performance previously attained with alternative methods. It is illustrated how this nonlinear droop control approach is able to enlarge the ROA compared with a linear droop technique. Numerical simulations confirm that the proposed polynomial controller makes the system more robust against large disturbances and so enhances the transient stability.**

***Index Terms*— DC microgrids, constant power loads, region of attraction, transient stability, Lyapunov theory, nonlinear control, sum of squares, semidefinite programming.**

I. Introduction

DC MICROGRIDS receive attention thanks to their ability to integrate renewables, storage, and loads efficiently [1]. A typical architecture of a DC microgrid is shown in Figure 1. In this configuration, the sources depicted at the left feed a common DC bus to which several loads are connected. All the sources and loads are interfaced by power electronic converters. The loads that are interfaced by converters with a high control bandwidth act as Constant Power Loads (CPL), adding nonlinear dynamics to the overall system. These nonlinearities have a negative impact on the system stability, due to the negative impedance behavior of CPLs [2]–[4]. Therefore, to promote the adoption of DC microgrids, it is important to study the stability of DC networks taking into account the presence of CPLs.

Often, the small-signal modeling is used for local stability analysis and control design of DC networks with CPLs [5]–[11]. This approach ensures asymptotic stability of the equilibrium point if all the eigenvalues of the linearized system are strictly in the left-half complex plane [12]. Using this technique, linear controllers can be designed to move the eigenvalues to the left-half plane and to add more damping to the system. Even though local stability is ensured, it is not possible to quantify what "local" means. To answer this question, the Region of Attraction (ROA) of the equilibrium point must be studied [12]. Additionally, it is of interest to design controllers to enlarge the ROA, making the system more robust against large disturbances. One approach to this end is the usage of time-domain simulations to check every point in the neighborhood of the stable equilibrium point. However, it does not give any closed form for control design purposes and could present scalability issues [13]. In contrast, direct methods based on the construction of Lyapunov energy functions have shown a good balance between accuracy and computational effort, and they are also control-oriented tools.

Several Lyapunov-based methods have been used to estimate and, in some cases, to enlarge the ROA of DC networks with CPLs. The Brayton-Moser potential theory, used in [14]–[17], allows an analytic construction of a Lyapunov function based on a special nonlinear representation of the electrical elements. Despite the simple computation of the Lyapunov function, this method is too conservative because the Lyapunov function is not defined to give a good estimate of the ROA. In [18]–[20], the Takagi-Sugeno fuzzy model is used to represent the nonlinear system as the weighted sum of linear systems. Then, under some conditions for the linear systems,

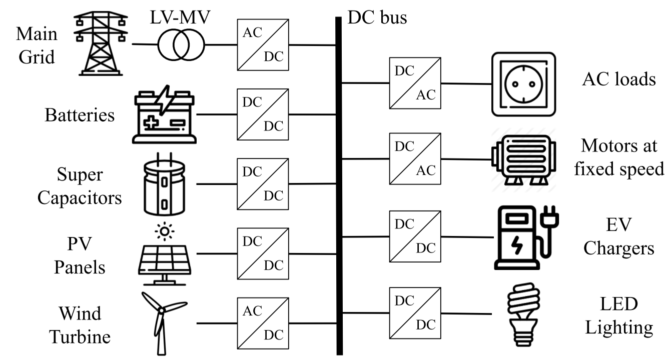

Fig. 1. Schematic of DC microgrid for wind and solar power integration.

Manuscript received January 3, 2019; revised April 15, 2019 and May 24, 2019; accepted June 4, 2019. Date of publication July 19, 2019; date of current version October 30, 2019. This work was supported by the Deutsche Akademische Austauschdienst (DAAD). This paper was recommended by Associate Editor H. R. Karimi. *(Corresponding author: Bernardo Severino.)*

The authors are with the Chair of Sustainable Electric Networks and Sources of Energy, Technische Universität Berlin, 10587 Berlin, Germany (e-mail: bernardo.severino@campus.tu-berlin.de; kai.strunz@tu-berlin.de).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2019.2924169

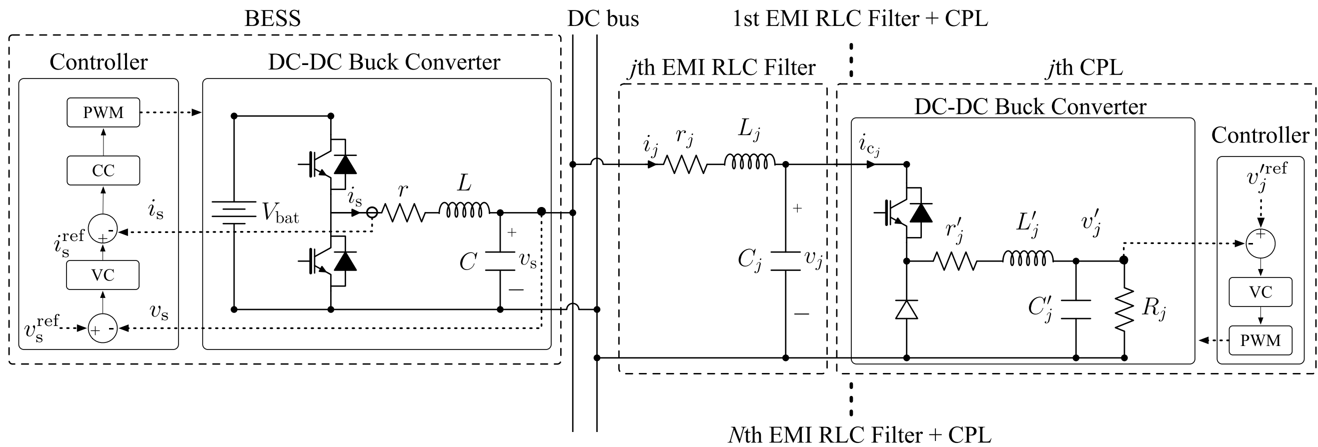

Fig. 2. Circuit details of a DC microgrid with one bidirectional source and N CPLs.

the asymptotic stability is ensured if a Linear Matrix Inequality (LMI) problem is feasible. This approach allows a less conservative estimate of the ROA compared with the Brayton-Moser potential theory. In [21], it is shown that the Takagi-Sugeno fuzzy approach becomes computational expensive for systems with a large number of CPLs. To overcome this issue, the nonlinear system is rewritten as a Lur'e problem in which stability can be studied by solving an LMI optimization problem [22]. The Takagi-Sugeno and Lur'e approaches have a good balance between accuracy and computational effort and allow to enlarge the ROA through control design. However, they cannot be used to synthesize nonlinear controllers, due to the structure of the LMI optimization problem.

Nonlinear controllers could be more suitable for enlarging the ROA as they take advantage of the nonlinear nature of the CPLs. The sum of squares (SOS) optimization technique, introduced in [23], is a promising approach for studying nonlinear systems. It allows computing polynomial Lyapunov functions, improving the accuracy of the estimated ROA [24]. In [25], the SOS technique was used to solve general control applications, and a methodology for enlarging the ROA with state feedback was proposed. SOS approach was used in [26] and [27] to study the ROA of an autonomous AC electrical power system. Also in other fields of engineering, it is observed that stability analysis using SOS shows accuracy improvements. However, one of the drawbacks is the polynomial growth of the computational effort with the number of variables [28]. Recently, [29] proposed two new polynomial decompositions which prove to reduce the computational time significantly. These are named as the diagonally-dominant sum of squares (DSOS) and the scaled-diagonally-dominant sum of squares (SDSOS).

In this paper, a methodology for estimating the ROA of DC microgrids with CPLs is proposed. The methodology relies on solving a sequence of SOS optimization problems. These problems are formulated and solved by exploiting the mathematical model of a DC microgrid. Since rational functions arise when modeling CPLs, a Taylor series approximation is used to recast the system into a polynomial one. The performance and scalability of the SOS-based modeling approach are analyzed. Results show that the proposed method improves the state of the art. Moreover, the methodology supports the design of nonlinear controllers for enlarging the ROA and enhancing stability, making the system more robust against large disturbances. A specific polynomial control structure is proposed and optimally designed using the SOS framework. The performance improvement is confirmed through simulation. The remainder of this paper is organized as follows. In Section II, the modeling of the DC microgrid with CLPs is presented and the ROA is defined, emphasizing its importance for the transient stability problem. In Section III, the Lyapunov theorem is recalled, the SOS decomposition introduced, and it is revealed how to find Lyapunov functions through polynomial certificates. In Section IV, the methodology for estimating the ROA is described, and its performance studied in Section V. In Section VI, the methodology is adapted for enlarging the ROA through a linear-cubic droop controller. Finally, in Section VII the conclusions of this work are discussed.

II. Problem Formulation

Figure 2 gives details of the general subsystem of the DC microgrid shown in Figure 1. As shown, N CPLs are connected to a DC bus trough EMI RLC filters. The DC bus voltage is regulated by a Battery Energy Storage System (BESS). The BESS encompasses an internal constant voltage source representing a battery, a bidirectional DC-DC buck converter, and its controller. The controller adopts a droop control strategy, commonly used to allow current sharing, active damping and plug and play capability [8]. Here, the current-voltage strategy is implemented [10]. The voltage v_{s} across the output capacitor C is measured and compared to a reference voltage $v_{\mathrm{s}}^{\mathrm{ref}}$, to generate an error signal. This error signal is applied to the voltage controller (VC) which computes the reference current $i_{\mathrm{s}}^{\mathrm{ref}}$ via a droop curve. The reference $i_{\mathrm{s}}^{\mathrm{ref}}$ is compared to the current i_{s} flowing through the inductance L, to generate an error signal. This error signal feeds the inner current controller (CC) that computes the duty cycle signal for the PWM block. Finally, the PWM synthesizes two square signals which drive the switches of the buck converter.

The generic jth CPL is composed of an internal load resistance R_j, a DC-DC buck converter, and its controller [2]. The controller measures the load voltage v'_j and compares it with the voltage reference $v_j'^{\mathrm{ref}}$. This error signal feeds a voltage controller (VC) which is designed to have zero steady-state error and high bandwidth. If the bandwidth of the closed-loop system is high enough, the DC-DC buck converter presents a CPL characteristic.

A. System Modeling

For the modeling of the BESS, it is assumed that the inner current controller is much faster than the outer voltage controller. Then, the closed-loop current dynamic is approximated by a first-order model with a time constant τ_{s}, i.e.,

$$G_{\mathrm{i}}(s) = \frac{i_{\mathrm{s}}}{i_{\mathrm{s}}^{\mathrm{ref}}} = \frac{1}{(s\tau_{\mathrm{s}} + 1)}, \tag{1}$$

where i_{s} is the current flowing through the upper switch, and $i_{\mathrm{s}}^{\mathrm{ref}}$ is the current reference. The outer voltage controller is considered as a general polynomial droop curve, thus

$$i_{\mathrm{s}}^{\mathrm{ref}} = i_{\mathrm{s}}^{0} + \sum_{d=1}^{d_{\max}} k_d (v_{\mathrm{s}}^{\mathrm{ref}} - v_{\mathrm{s}})^d, \tag{2}$$

where i_{s}^0 is a free parameter, k_d is the droop coefficient associated with the term of degree d, and $d_{\max}$ is the maximum degree of the polynomial droop curve. Note that when $d_{\max} = 1$ the linear droop curve is recovered. Then, from (1) and (2) the dynamic equations for the BESS when it is connected at the DC bus are:

$$\begin{aligned} C\dot{v}_{\mathrm{s}} &= i_{\mathrm{s}} - \sum_{j=1}^{N} i_j, \\ \tau_{\mathrm{s}}\dot{i}_{\mathrm{s}} &= i_{\mathrm{s}}^0 + \sum_{d=1}^{d_{\max}} k_d (v_{\mathrm{s}}^{\mathrm{ref}} - v_{\mathrm{s}})^d - i_{\mathrm{s}}, \end{aligned} \tag{3}$$

where i_j is the current flowing through the jth RLC filter.

The jth CPL is modeled as a controlled current source with internal dynamics, which is approximated by a first-order model [20], [30]. With the voltage being tightly regulated, the input of the controlled current source is P_j/v_j, leading to the nonlinear behavior of the CPL. Then, the dynamic equations for the jth CPL connected to the DC bus through the jth filter are:

$$\begin{aligned} L_j \dot{i}_j &= -r_j i_j - v_j + v_{\mathrm{s}}, \\ C_j \dot{v}_j &= i_j - i_{\mathrm{c}_j}, \\ \tau_j \dot{i}_{\mathrm{c}_j} &= P_j/v_j - i_{\mathrm{c}_j}, \end{aligned} \tag{4}$$

where v_j is the voltage across the filter capacitor C_j, i_j is the current through the filter inductor L_j, r_j is the resistance of the filter, P_j is the power consumption, i_{c_j} is the current through the switch, and τ_j is the time constant of the closed-loop current dynamic.

For convenience, a change of variables is performed as follows to express the dynamic behavior around the origin: $y = \bar{y} + \tilde{y}$, where y denotes the voltage or current variable, $\bar{y}$ its equilibrium point and $\tilde{y}$ its variations around the equilibrium point. Additionally, in (2) the setting is $v_{\mathrm{s}}^{\mathrm{ref}} = \bar{v}_{\mathrm{s}}$ and $i_{\mathrm{s}}^0 = \bar{i}_{\mathrm{s}}$. The resulting dynamic equations for the BESS are:

$$\begin{aligned} C\dot{\tilde{v}}_{\mathrm{s}} &= \tilde{i}_{\mathrm{s}} - \sum_{j=1}^{N} \tilde{i}_j, \\ \tau_{\mathrm{s}}\dot{\tilde{i}}_{\mathrm{s}} &= K(\tilde{v}_{\mathrm{s}}) - \tilde{i}_{\mathrm{s}}, \end{aligned} \tag{5}$$

where

$$K(\tilde{v}_{\mathrm{s}}) = \sum_{d=1}^{d_{\max}} k_d (-\tilde{v}_{\mathrm{s}})^d \tag{6}$$

is the polynomial control droop, and the resulting dynamic equations for the jth Filter-CPL are:

$$\begin{aligned} L_j \dot{\tilde{i}}_j &= -r_j \tilde{i}_j - \tilde{v}_j + \tilde{v}_{\mathrm{s}}, \\ C_j \dot{\tilde{v}}_j &= \tilde{i}_j - \tilde{i}_{\mathrm{c}_j}, \\ \tau_j \dot{\tilde{i}}_{\mathrm{c}_j} &= -\bar{i}_j h(\tilde{v}_j) - \tilde{i}_{\mathrm{c}_j}, \end{aligned} \tag{7}$$

where $\bar{i}_j = P_j/\bar{v}_j$, and

$$h(\tilde{v}_j) = \frac{\tilde{v}_j}{\tilde{v}_j + \bar{v}_j} \tag{8}$$

is the nonlinear behavior of the CPL.

From (5) to (8) the dynamical system around the origin can be written as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\Gamma}(\boldsymbol{x}), \tag{9}$$

where the vector $\boldsymbol{x}$, the matrix $\boldsymbol{A}$, and the vector of nonlinear functions $\boldsymbol{\Gamma}(\boldsymbol{x})$ are given in Appendix A. Finally, note from (5) to (8) that the ith state variable of (9) can be written as a rational function, i.e.,

$$\dot{x}_i = \frac{n_i(\boldsymbol{x})}{d_i(\boldsymbol{x})} \tag{10}$$

where $n_i(\boldsymbol{x})$ and $d_i(\boldsymbol{x})$ are polynomial functions. Therefore, the system under study is a rational system.

Other types of loads can be included in this modeling framework. In general terms, the proposed method supports any load that can be described as a polynomial function. In the case of loads that are not polynomials, such as rational systems, they may be approximated using the Taylor series as it is proposed in Section IV-A. For example, ZIP loads [31] could be modeled with the proposed approach.

B. Region of Attraction

Consider the autonomous system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \in \mathcal{D}, \tag{11}$$

where $\boldsymbol{f} : \mathcal{D} \to \mathbb{R}^n$ is a vector field of nonlinear functions, $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state vector, $\boldsymbol{x}_0$ is the initial condition, and $\mathcal{D} \subseteq \mathbb{R}^n$ is a neighborhood of the origin.

Let the origin $\boldsymbol{x} = \boldsymbol{0}$ be an asymptotically stable equilibrium point for (11), and $\phi(t, \boldsymbol{x}_0)$ be the solution of (11) that starts at initial state $\boldsymbol{x}_0$ at time $t = 0$. The ROA of the origin, denoted by R_A, is defined by

$$R_A = \{\boldsymbol{x}_0 \in \mathcal{D} \mid \phi(t, \boldsymbol{x}_0) \to \boldsymbol{0} \text{ as } t \to \infty\}. \tag{12}$$

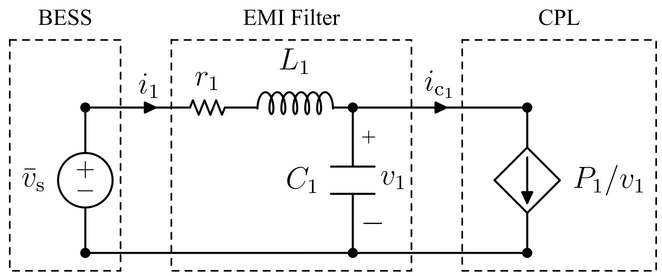

Fig. 3. Equivalent reduced circuit of the DC microgrid.

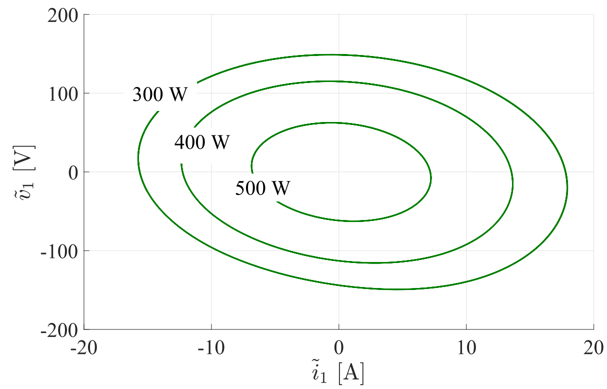

Fig. 4. Boundary of the ROA for different values of power in the phase plane.

Thus, R_A defines the neighborhood of all initial conditions for which the system (11) will return to the origin. Often, to have a better understanding of the ROA, it is plotted in the phase plane, i.e., the plane having the states of the system as coordinates.

To appreciate the importance of determining the ROA, let recall the transient stability problem [12], [13], [32]. Suppose that at time t_0 the system (11) is subjected to a severe transient disturbance, commonly named as a fault. The disturbance could be for example a short circuit or a sudden large change of load. Henceforth, the system variables are governed by the so-called fault-on system, different from the pre-fault system (11). The differences between them are due to a change of the network or to a change of the parameters of the system (11). During the fault-on process, the system response to such disturbances involves large excursions of system variables. Suppose that the disturbance ends or is cleared at time t_1. At this time, the fault-on system reaches the state $\boldsymbol{x}_{\text{pos}}(t_1)$, and it switches to the post-fault system. The post-fault system could be equal to the pre-fault system. In such a case, the equilibrium point of the post-fault system is $\boldsymbol{x} = \boldsymbol{0}$, and the state variables are governed by (11). The transient stability problem considers whether the trajectory $\boldsymbol{x}(t)$ for (11) with initial conditions $\boldsymbol{x}(0) = \boldsymbol{x}_{\text{pos}}(t_1)$ will converge to $\boldsymbol{x} = \boldsymbol{0}$, as time t goes to infinity. Whether or not the system will return to steady-state operation depends on whether $\boldsymbol{x}_{\text{pos}}(t_1)$ belongs to the ROA R_A defined in (12).

To illustrate the transient stability problem, consider the system in Figure 2 when $N = 1$, $C \gg C_1$, $\tau_1 = 0$. In such a situation, a constant voltage source approximates the behavior of the BESS, and the CPL acts as a controlled current source without internal dynamics. Therefore, from (7) and (8) the dynamic equations around the origin are given by

$$\begin{aligned} L_1 \dot{\tilde{i}}_1 &= -r_1 \tilde{i}_1 - \tilde{v}_1, \\ C_1 \dot{\tilde{v}}_1 &= \tilde{i}_1 + \frac{P_1}{\bar{v}_1}\left(\frac{\tilde{v}_1}{\tilde{v}_1 + \bar{v}_1}\right). \end{aligned} \tag{13}$$

Figure 3 shows the equivalent circuit of (13) which corresponds to a single CPL connected to a constant voltage source through an EMI RLC filter.

Using the time-domain simulation approach, the exact ROA can be obtained. Figure 4 shows the boundary of R_A, for different values of P_1 and the parameters given in Table I. As expected, the ROA decreases as P_1 increases. For all values of P_1, the equilibrium point is shown to remain asymptotically stable when using small-signal analysis. Although the time-domain simulation approach is applied to this example, its use becomes prohibitive when the number of state variables of the system increases, and it does not provide a closed-form solution of the transient stability problem, making it impractical for a control design process [13]. The use of eigenvalue analysis is not an alternative either. While the size of the ROA relates to the distance between the absolute real part of the eigenvalues and the imaginary axes, there is no methodology to translate that distance into the phase plane where the ROA is defined. These disadvantages encourage the use of direct methods, also called Lyapunov-based methods. In particular, the SOS approach has the potential for a less conservative estimate of the ROA compared to other techniques, while preserving computation times for small and medium-size problems [28].

TABLE I
CIRCUIT PARAMETERS FOR THE CIRCUIT IN FIG. 3

Variable	Value	Unit	Variable	Value	Unit
L_1	39.5	mH	C_1	500	μF
r_1	1.1	Ω	$\bar{v}_s$	200	V

III. BACKGROUND

The methodology for estimating and enlarging the ROA of a DC microgrid with CPLs is based on the Lyapunov stability theory, and the SOS of polynomials decomposition approach. In the following, a brief discussion about these topics is presented.

A. Lyapunov Stability

Direct methods are based on the Lyapunov stability theorem which allows assessing the stability of nonlinear systems by solving the problem of the existence of a *Lyapunov function* [12].

Theorem 1. *Let* $\boldsymbol{x} = \boldsymbol{0}$ *be an equilibrium point for* (11) *and* $\mathcal{D} \subset \mathbb{R}^n$ *be a domain containing* $\boldsymbol{x} = \boldsymbol{0}$. *Let* $V : \mathcal{D} \to \mathbb{R}$ *be a continuously differentiable function such that* $V(\boldsymbol{0}) = 0$ *and*

$$\begin{aligned} (i) &\quad V(\boldsymbol{x}) > 0 \quad in \quad \mathcal{D} - \{0\}, \\ (ii) &\quad -\dot{V}(\boldsymbol{x}) > 0 \quad in \quad \mathcal{D} - \{0\}, \end{aligned}$$

then $\boldsymbol{x} = \boldsymbol{0}$ is asymptotically stable. Moreover, any region defined by $\Omega_c := \{\boldsymbol{x} \in \mathbb{R}^n \mid V(\boldsymbol{x}) \le c\}$ such that $\Omega_c \subset \mathcal{D}$ is an estimate of the ROA defined in (12).

Note that $\dot{V}(\boldsymbol{x})$ is the time derivative of $V(\boldsymbol{x})$ along the trajectories of (11), i.e., $\dot{V}(\boldsymbol{x}) = \nabla V \boldsymbol{f}(\boldsymbol{x})$. A function $V(\boldsymbol{x})$ that might satisfy the conditions of Theorem 1 is called a *Lyapunov function candidate*. While a function $V(\boldsymbol{x})$ satisfying the conditions is called a *Lyapunov function*.

The largest estimate of the ROA using $V(\boldsymbol{x})$ can be obtained by solving the following optimization problem [12]:

$$\max c \quad \text{s.t.} \quad \Omega_c = \{\boldsymbol{x} \in \mathbb{R}^n \mid V(\boldsymbol{x}) \le c\} \subset \mathcal{D}. \tag{14}$$

Note that Theorem 1 leaves complete freedom in the selection of both the Lyapunov function V and the domain $\mathcal{D}$. Then, the problem (14) suggests that a search over V and $\mathcal{D}$ could be performed. Thus, a choice to improve the estimate of the ROA is to enlarge the domain $\mathcal{D}$ for which a Lyapunov function V exists.

Suppose now that $V(\boldsymbol{x})$ is a Lyapunov function candidate which is positive definite in $\mathcal{D}$, but it might not satisfy the condition (ii) of Theorem 1. In such a case, the largest estimate of the ROA using $V(\boldsymbol{x})$ can be obtained by solving the following optimization problem [12]:

$$\begin{aligned} \max_{c} \quad & c \\ \text{s.t.} \quad & \Omega_c = \{\boldsymbol{x} \in \mathbb{R}^n \mid V(\boldsymbol{x}) \le c\} \subset \mathcal{D}, \\ & -\dot{V}(\boldsymbol{x}) > 0 \quad \text{in} \quad \Omega_c - \{0\}, \end{aligned} \tag{15}$$

where the second constraint enforces the nonnegativity of $-\dot{V}$ over Ω_c.

As seen from Theorem 1, the problem of estimating the ROA can be reduced to finding a Lyapunov function V and solve (14); or at least to find a positive definite function V and solve (15). In the specific case of a linear system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$, a quadratic Lyapunov function $V(\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}$ can be easily found by solving the so-called Lyapunov matrix equation [12]

$$\boldsymbol{A}^{\mathsf{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} = -\boldsymbol{Q}, \tag{16}$$

where $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ is an arbitrary positive definite symmetric matrix, and $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ is the matrix to be solved. In contrast, for nonlinear systems, the search for Lyapunov functions is a challenging task [12]. In the case when $\boldsymbol{f}(\boldsymbol{x})$ is a vector of polynomial functions, this search is, in general, an NP-hard problem [28]. Nevertheless, the conditions (i) and (ii) of Theorem 1 could be relaxed to be SOS of polynomials, instead of to be positive definite functions. Thus, checking the conditions of Theorem 1 is equivalent to solving a semidefinite programming problem [28].

B. Sum of Squares Decomposition

Let $\boldsymbol{x} \in \mathbb{R}^n$ be a vector of n variables, $p(\boldsymbol{x}) \in \mathbb{R}[\boldsymbol{x}]_{n,2d}$ be a multivariate polynomial in n variables and degree $2d$, and $[\boldsymbol{x}]_d$ be a vector of all $\binom{n+d}{d}$ monomials in $\boldsymbol{x}$ of degree up to d. For example, the vector of all monomials in $\boldsymbol{x} = [x_1, x_2]^{\mathsf{T}}$ of degree up to $d = 2$ is $[\boldsymbol{x}]_2 = [1, x_1, x_2, x_1^2, x_1x_2, x_2^2]^{\mathsf{T}}$.

Definition 1. The polynomial $p(\boldsymbol{x})$ is SOS if there exist $q_1, \ldots, q_m \in \mathbb{R}[\boldsymbol{x}]_{n,d}$ such that

$$p(\boldsymbol{x}) = \sum_{k=1}^{m} q_k^2(\boldsymbol{x}).$$

Obviously, $p(\boldsymbol{x})$ being an SOS implies that $p(\boldsymbol{x})$ is nonnegative over the whole space $\mathbb{R}^n$. Note that the SOS condition can be written as $p(\boldsymbol{x}) = (\boldsymbol{C}[\boldsymbol{x}]_d)^{\mathsf{T}}(\boldsymbol{C}[\boldsymbol{x}]_d)$, where $\boldsymbol{C} \in \mathbb{R}^{m\times\binom{n+d}{d}}$, and its kth row contains the coefficients of the polynomial $q_k(\boldsymbol{x})$. The following theorem describes how to obtain an SOS decomposition [23].

Theorem 2. *A polynomial $p(\boldsymbol{x}) \in \mathbb{R}[\boldsymbol{x}]_{n,2d}$ is SOS if and only if there exists a symmetric matrix $\boldsymbol{Q} \in \mathbb{R}^{\binom{n+d}{d}\times\binom{n+d}{d}}$ such that $p(\boldsymbol{x}) = [\boldsymbol{x}]_d^{\mathsf{T}}\boldsymbol{Q}[\boldsymbol{x}]_d$ and $\boldsymbol{Q} \succeq 0$. Then, by factorizing $\boldsymbol{Q} = \boldsymbol{C}^{\mathsf{T}}\boldsymbol{C}$, an SOS decomposition of $p(\boldsymbol{x})$ can be obtained.*

Thus, an SOS decomposition problem is recast into an LMI problem. LMI problems are convex optimization problems and can be solved via semidefinite programming (SDP) [23]. Then, if the SDP problem is feasible, $p(\boldsymbol{x})$ is SOS and therefore it is nonnegative for all $\boldsymbol{x} \in \mathbb{R}^n$.

In many cases it is required to check if $p(\boldsymbol{x})$ is positive definite in a given subset $\mathcal{S} \subseteq \mathbb{R}^n$, as in Theorem 1 where the conditions need to be checked over the domain $\mathcal{D}$. Assume the set $\mathcal{S}$ has a description:

$$\mathcal{S} = \{\boldsymbol{x} \in \mathbb{R}^n \mid g_1(\boldsymbol{x}) \le 0, \ldots, g_m(\boldsymbol{x}) \le 0\}.$$

A sufficient condition to certify that $p(\boldsymbol{x})$ is positive definite in the set $\mathcal{S}$ is the existence of SOS polynomials $s_i(\boldsymbol{x})$ for $i = 0, \ldots, m$ and a positive definite polynomial function $\varphi(\boldsymbol{x})$ such that

$$s_0(\boldsymbol{x})p(\boldsymbol{x}) + \sum_{i=1}^{m} s_i(\boldsymbol{x})g_i(\boldsymbol{x}) - \varphi(\boldsymbol{x}) \quad \text{is SOS.} \tag{17}$$

Indeed, take an arbitrary point $\boldsymbol{x} \in \mathcal{S}$. Since $s_i(\boldsymbol{x})$ is SOS and $g_i(\boldsymbol{x}) \le 0$ for all $i = 1, \ldots, m$ then $s_i(\boldsymbol{x})g_i(\boldsymbol{x}) \le 0$ for all $i = 1, \ldots, m$. Therefore, $s_0(\boldsymbol{x})p(\boldsymbol{x}) - \varphi(\boldsymbol{x}) \ge 0 \Rightarrow p(\boldsymbol{x}) > 0$ because $\varphi(\boldsymbol{x}) > 0$ and $s_0(\boldsymbol{x})$ is SOS.

With the sufficient condition (17), the SOS decomposition can be applied on Theorem 1, the optimization problem (14), and the optimization problem (15). In Appendix B, three Lemmas are presented to certify the conditions of Theorem 1, problem (14), and problem (15). These Lemmas lead to optimization problems where the constraints are SOS of polynomials. These optimization problems are called SOS programs. The software SOSTOOLS [33], YALMIP [34] or SPOT [35] can be used along with an SDP solver such as SeDuMi or MOSEK to solve such SOS programs.

The software SPOT additionally handles DSOS and SDSOS decompositions, which have been proposed to reduce the computational time [29]. SOS decomposition certifies nonnegativity of $p(\boldsymbol{x})$ by finding a positive semidefinite symmetric matrix Q, as shown in Theorem 2. In contrast, DSOS and SDOS certify nonnegativity by finding a diagonally dominant symmetric matrix and a scaled diagonally dominant symmetric matrix, respectively. As a result, DSOS and SDSOS rely on

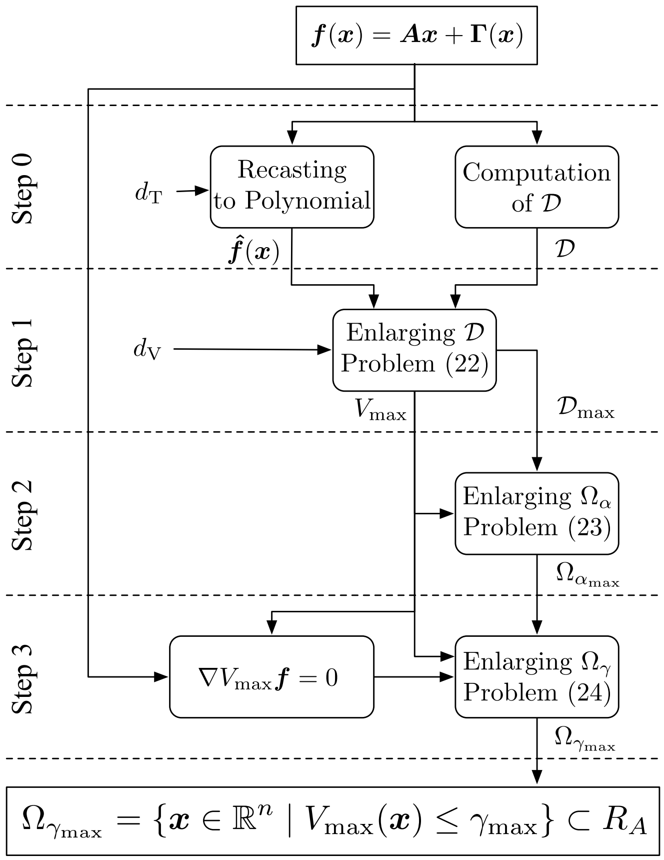

Fig. 5. Flow chart of the SOS-based modeling.

solving a linear programming problem and a second-order cone programming problem, respectively. In Appendix C, more details about the DSOS and SDSOS decompositions are given.

IV. Methodology for Estimating the ROA

The flow chart of the SOS-based modeling approach is shown in Figure 5. The input is the nonlinear system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$, and the output is the estimate of the ROA, $\boldsymbol{\Omega}_{\gamma_{\max}}$.

The methodology is composed by four steps, and it mainly relies on the sufficient condition (17). Condition (17) is used to recast Theorem 1, the optimization problem (14), and the optimization problem (15) into SOS programs by using the lemmas in Appendix B. The methodology is implemented in Matlab using the SPOT software with the capability of using SeDuMi or MOSEK SDP solvers. The SOS-based modeling is flexible in the selection of both the specific polynomial decomposition and the SDP solver. This feature allows exploring the natural compromise between computational effort and conservatism in the ROA estimation. In the following, the steps of the methodology are explained.

A. Step 0

Before the formulation of the SOS programs, the set $\mathcal{D}$ in Theorem 1 has to be defined. One option is to use the information provided by the linear approximation of (9) to shape the domain $\mathcal{D}$ wherein a Lyapunov function can be computed.

Therefore, defining $\mathcal{D}$ as

$$\mathcal{D} := \{\boldsymbol{x} \in \mathbb{R}^n \mid \beta(\boldsymbol{x}) - b \leq 0\}, \tag{18}$$

where $\beta(\boldsymbol{x})$ is a multivariate polynomial and $b \in \mathbb{R}$, this can be accomplished by taking

$$\beta(\boldsymbol{x}) = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}, \tag{19}$$

where $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ is the solution of the Lyapunov equation (20) for the linearized model:

$$\left(\boldsymbol{A} + \frac{\partial \boldsymbol{\Gamma}}{\partial \boldsymbol{x}}\right)^{\mathsf{T}} \boldsymbol{P} + \boldsymbol{P}\left(\boldsymbol{A} + \frac{\partial \boldsymbol{\Gamma}}{\partial \boldsymbol{x}}\right) = -\boldsymbol{Q}, \quad \boldsymbol{P} > 0. \tag{20}$$

In (20), $\boldsymbol{A}$ is the matrix in (9), $\frac{\partial \boldsymbol{\Gamma}}{\partial \boldsymbol{x}}$ is the Jacobian matrix of $\boldsymbol{\Gamma}(\boldsymbol{x})$ evaluated at $\boldsymbol{x} = \boldsymbol{0}$, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ is a given arbitrary positive definite symmetric matrix, and $\boldsymbol{P}$ is the matrix to be solved. The solution can be found using LMI solvers, such as the one provided in [34]. To this end, $\boldsymbol{Q}$ is set equal to the identity matrix. This selection is usually performed in [12], [22], [28]. In this manner, the boundary of $\mathcal{D}$ is an ellipsoid centered at the origin described by $\beta(\boldsymbol{x}) = b$.

To apply the SOS approach, $\boldsymbol{f}(\boldsymbol{x})$ must be a polynomial vector, as assumed in Section III-B. But, as it was described in Section II-A, the system under study is a rational system. Nevertheless, the original nonlinear system $\boldsymbol{f}(\boldsymbol{x})$ can be recast to a polynomial system by using Taylor series expansion. The truncated Taylor series of $\boldsymbol{f}(\boldsymbol{x})$ until degree d_{T} is

$$\hat{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \hat{\boldsymbol{\Gamma}}(\boldsymbol{x}), \tag{21}$$

where $\hat{\boldsymbol{\Gamma}}(\boldsymbol{x})$ is the truncated Taylor series of $\boldsymbol{\Gamma}$ until degree d_{T}. In this fashion, $\hat{\boldsymbol{f}}(\boldsymbol{x})$ approximates the nonlinear function $\boldsymbol{f}(\boldsymbol{x})$ and meets the assumption of Lemma 1.

B. Step 1

Once $\beta(\boldsymbol{x})$, $\hat{\boldsymbol{f}}(\boldsymbol{x})$ are set, the Lemma 1 in Appendix B is used to formulate the following SOS program that enlarges the domain $\mathcal{D}$ in (18) wherein a Lyapunov function V can be found.

$$\begin{aligned} \max_{b,V,s_1,s_2} \quad & b \\ \text{s.t.} \quad & V + s_1(\beta - b) - \varphi_1 \text{ is SOS} \\ & -\nabla V \hat{\boldsymbol{f}} + s_2(\beta - b) - \varphi_2 \text{ is SOS} \\ & s_1 \text{ and } s_2 \text{ are SOS.} \end{aligned} \tag{22}$$

Therefore, by solving this SOS program the conditions of the Lemma 1 are fulfilled, and then the origin is asymptotically stable.

To solve the problem (22) the following is considered. Since $V(\boldsymbol{0}) = 0$ and $V(\boldsymbol{x}) > 0$ in a domain containing the origin, $V(\boldsymbol{x})$ must not have any constant terms and cannot be of degree one. Consequently, the degree d_{V} of $V(\boldsymbol{x})$ must be equal or higher than two. The degrees of the SOS polynomials s_1 and s_2 are defined as d_{s_1} and d_{s_2}, respectively. They are set such that the polynomial expressions involved in the constraints have even degree. The positive definite polynomial functions $\varphi_1(\boldsymbol{x})$ and $\varphi_2(\boldsymbol{x})$ are defined in advance to reduce computational time. It is chosen $\varphi_i(\boldsymbol{x}) = \lambda_i \|[\boldsymbol{x}]_{d_i}\|^2$, with $\lambda_i > 0$ a fixed parameter, and $[\boldsymbol{x}]_{d_i}$ a vector of all monomials up to degree $d_i = \frac{d_{\varphi_i}}{2}$, where d_{φ_i} is the desired degree of $\varphi_i(\boldsymbol{x})$. The selection of λ_i could influence the results. It is recommended to test the problem before for some values of λ_i and check the error between the resulting SOS decomposition and the polynomial constraints.

The SOS program (22) is not affine in the decision variables s_1, s_2, and b. Nevertheless, a bisection search over b can be conducted, as described in Algorithm 1 below. The selection of the interval is $[b_l, b_u]$, where $b_l = 0$ and $b_u = \beta(-\bar{\boldsymbol{x}})$. Note from Appendix A that the vector $-\bar{\boldsymbol{x}}$ contains the value $-\bar{v}_j$. Therefore, from (8) it follows that

$$h(-\bar{v}_j) = \frac{-\bar{v}_j}{-\bar{v}_j + \bar{v}_j} \longrightarrow -\infty.$$

Therefore, $\beta(-\bar{\boldsymbol{x}})$ is a good selection for the upper bound since the point $-\bar{\boldsymbol{x}}$ does not belong to the ROA. In each step of the bisection method, a feasibility problem over the polynomials V, s_1, and s_2 needs to be solved. The process continues until the problem is not feasible. The maximum value of b corresponds to the last iteration in which the problem has found a feasible solution. The termination condition is defined by ϵ.

Algorithm 1 will require at most $\lfloor \log_2(\frac{b_u}{\epsilon}) \rfloor + 1$ iterations. In each iteration a feasibility SOS program must be solved. The complexity of solving an SOS program can be measured in terms of the number of LMI scalar variables required for establishing whether the constraints are SOS [36]. In Appendix IV the complexity of the SOS program (22) is analyzed in terms of the degree d_{V} of the Lyapunov function candidate $V(\boldsymbol{x})$ and the degree d_{T} of the truncated Taylor series $\hat{\boldsymbol{f}}(\boldsymbol{x})$. By using the ellipsoid algorithm or the interior-point method, the time needed to solve the resulting LMI will be a polynomial function of the LMI size [28].

Summarizing, at the end of Step 1 the value $b_{\max}$ is obtained. This value defines the domain $\mathcal{D}_{\max}$ in (18). Also, the polynomial $V_{\max}(\boldsymbol{x})$ is obtained.

It must be recalled that $\mathcal{D}_{\max}$ is not an estimate of the ROA [12]. Although $\phi(t, \boldsymbol{x}_0)$, defined as the solution of (9), will move from one Lyapunov level to an inner Lyapunov level in $\mathcal{D}_{\max}$, there is no guarantee that $\phi(t, \boldsymbol{x}_0)$ will remain in $\mathcal{D}_{\max}$ as t goes to infinity. Once $\phi(t, \boldsymbol{x}_0)$ leaves $\mathcal{D}_{\max}$, there is no guarantee that $\dot{V}_{\max}(\boldsymbol{x})$ will be negative. Consequently, the condition (ii) of Theorem 1 may not be met. This issue does not arise when R_A in (12) is estimated by the largest compact set $\boldsymbol{\Omega}_c \subset \mathcal{D}$ as it was defined in (14).

Algorithm 1 Bisection Search Over b.

Data: $\beta(\boldsymbol{x})$, $\hat{\boldsymbol{f}}(\boldsymbol{x})$, $\varphi_1(\boldsymbol{x})$, $\varphi_2(\boldsymbol{x})$, d_{V}, d_{s_1}, d_{s_2} and ϵ
Result: $b_{\max}$ and $V_{\max}$
initialize $b_{\max} = 0$, $b_l = 0$ and $b_u = \beta(-\bar{\boldsymbol{x}})$;
while $|b_u - b_l|/2 > \epsilon$ **do**
  set $b = (b_u + b_l)/2$;
  solve (22) for V, s_1 and s_2;
  if *feasible* **then**
    set $b_{\max} = b$, $V_{\max} = V$ and $b_l = b$;
  else
    set $b_u = b$;
  end
end

C. Step 2

The Lyapunov function $V_{\max}$ could be used to compute the largest invariant region contained in $\mathcal{D}_{\max}$, as it was described in (14). To this end, the Lemma 2 in Appendix B is used to formulate the following SOS program that expands the domain $\boldsymbol{\Omega}_\alpha$ defined in Lemma 2.

$$\begin{aligned} \max_{\alpha, s_0} \quad & \alpha \\ \text{s.t.} \quad & -s_0(\beta - b_{\max}) + (V_{\max} - \alpha) - \varphi_1 \text{ is SOS} \\ & s_0 \text{ is SOS.} \end{aligned} \tag{23}$$

Note that this problem is affine in the SOS variables. Then, it can be solved directly without the usage of a bisection method. The optimal solution of this problem is $\alpha_{\max}$, and $\boldsymbol{\Omega}_{\alpha_{\max}}$ is the largest estimate of the ROA of $\hat{\boldsymbol{f}}(\boldsymbol{x})$. It must be recalled that $\boldsymbol{\Omega}_{\alpha_{\max}}$ is the ROA of the approximated system $\hat{\boldsymbol{f}}(\boldsymbol{x})$ and would likely not meet the conditions of Theorem 1 for the original system $\boldsymbol{f}(\boldsymbol{x})$. This issue is solved in the next step.

D. Step 3

The obtained Lyapunov function $V_{\max}$ for $\hat{\boldsymbol{f}}(\boldsymbol{x})$ could be used as a Lyapunov function candidate to study the stability of the original nonlinear system $\boldsymbol{f}(\boldsymbol{x})$ in a new domain as it was described in (15). Based on that, the Lemma 3 in Appendix B is used to formulate the following SOS program that enlarges the domain $\boldsymbol{\Omega}_\gamma$ defined in Lemma 3.

$$\begin{aligned} \max_{\gamma, s_0, s_1} \quad & \gamma \\ \text{s.t.} \quad & \boldsymbol{\Omega}_\gamma \subseteq \mathcal{D}_{\max} \\ & -s_0 \dot{V}_{\max} + s_1(V_{\max} - \gamma) - \varphi_1 \text{ is SOS} \\ & s_0 \text{ and } s_1 \text{ are SOS.} \end{aligned} \tag{24}$$

To solve this problem, some aspects must be addressed. First, we have the multiplication between the decision variables s_1 and γ. Therefore, a bisection algorithm that searches over γ is used in the same manner as it was used to solve the problem (22). In this case, $V_{\max}$ is an input, and there is only one output $\gamma_{\max}$. The interval to be considered is $[\gamma_l, \gamma_u]$ where $\gamma_l = 0$, and γ_u is equal to the maximum level set of $V_{\max}$ contained in $\mathcal{D}$. Therefore, $\gamma_u = \alpha_{\max}$ where $\alpha_{\max}$ is the argument of the optimization problem (23). By doing so, it is ensured that $\boldsymbol{\Omega}_\gamma \subseteq \mathcal{D}$ for all $\gamma \in [\gamma_l, \gamma_u]$. Then, the first constraint of (24) is not needed as it is always fulfilled.

Secondly, by using (10), $\dot{V}_{\max}(\boldsymbol{x}) = \nabla V_{\max} \dot{\boldsymbol{x}}$ can be written as:

$$\dot{V}_{\max}(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{\partial V_{\max}}{\partial x_i} \frac{n_i(\boldsymbol{x})}{d_i(\boldsymbol{x})}.$$

Expanding the sum and taking the minimum common denominator, it follows that

$$\dot{V}_{\max}(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} \frac{\partial V_{\max}}{\partial x_i} n_i(\boldsymbol{x}) \prod_{\forall j=1, j\neq i}^{n} d_j(\boldsymbol{x})}{\prod_{i=1}^{n} d_i(\boldsymbol{x})}.$$

Since the interval used in the bisection method of problem (22) is $[0, \beta(-\bar{\boldsymbol{x}})]$, we obtain that $\prod_{i=1}^{n} d_i(\boldsymbol{x}) > 0$ in the domain $\mathcal{D}$. Therefore, $\dot{V}_{\max}$ is replaced by

$$\sum_{i=1}^{n} \frac{\partial V_{\max}}{\partial x_i} n_i(\boldsymbol{x}) \prod_{\forall j=1, j\neq i}^{n} d_j(\boldsymbol{x}).$$

It can be shown that the degree of this polynomial is $d_{\mathrm{V}} + N$, where N is the number of CPLs. Since the domain $\boldsymbol{\Omega}_{\gamma}$ was defined using the Lyapunov function $V_{\max}$, the biggest invariant region contained in $\boldsymbol{\Omega}_{\gamma}$ is the domain itself at $\gamma_{\max}$. Therefore, the set

$$\boldsymbol{\Omega}_{\gamma_{\max}} := \{\boldsymbol{x} \in \mathbb{R}^n \mid V_{\max}(\boldsymbol{x}) \leq \gamma_{\max}\}$$

is the estimate of the ROA at $\boldsymbol{x} = \boldsymbol{0}$.

The optimization problem (24) is challenging from the point of view of the numerical resolution [28]. Changes in the structure of $V_{\max}$ and in the parameters of the system (9) could lead to numerical problems. It has been noted, from the application of the method, that the estimated ROA $\boldsymbol{\Omega}_{\gamma_{\max}}$ increases with the expansion of the domain $\mathcal{D}$. Therefore, for control purposes, it is sufficient to solve the problem (22) and use the value $b_{\max}$ as a measure of robustness.

V. Analysis of Performance

The performance of the SOS-based modeling is analyzed in terms of its accuracy and its scalability. Regarding the accuracy, it is evaluated how conservative the solution $\boldsymbol{\Omega}_{\gamma_{\max}}$ is compared with the exact ROA. The scalability refers to the number of CPLs that the methodology can handle. To this end, the methodology is applied to two different DC networks. One is used to study the accuracy, and the other one is to show the scalability. These systems can be obtained from the general system presented in Section II-A, whose dynamics are governed by (9).

A. Accuracy

The proposed approach has three sources of conservativeness. The first source is given by the set $\mathcal{D}$ defined in (18). As it was described, the selection of $\mathcal{D}$ is based on the local behavior of the system under study. For this specific application, selecting $\mathcal{D}$ in this manner has shown good results. However, it could be the case that a different set $\mathcal{D}$ improves the estimation of the ROA. The second source of conservativeness is given by the approximation of $\boldsymbol{f}(\boldsymbol{x})$ by its truncated Taylor series in (21). By increasing the degree d_{T} of the truncated Taylor series $\hat{\boldsymbol{f}}(\boldsymbol{x})$, a better approximation of the original nonlinear system is obtained. Nevertheless, estimating the ROA would require a higher computational time. The third source of conservativeness depends on the degrees of the polynomials defined for the SOS problem (22) in Step 1 of Section IV. These degrees could be made as large as possible with the aim of finding a Lyapunov function valid for the set $\mathcal{D}$. Nevertheless, that would require a higher computational effort.

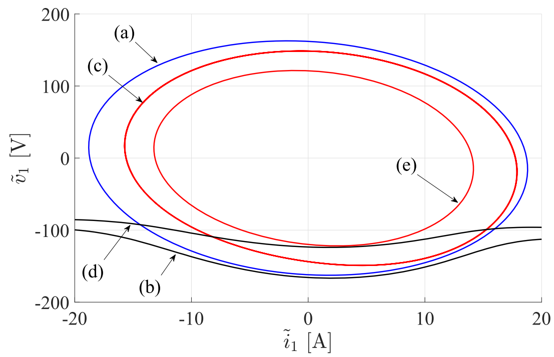

Fig. 6. Estimated ROA; a) the boundary of $\mathcal{D}_{\max}$, b) the equation $\nabla V_{\max}\hat{\boldsymbol{f}} = 0$, c) the boundary of $\boldsymbol{\Omega}_{\alpha_{\max}}$, d) the equation $\nabla V_{\max}\boldsymbol{f} = 0$, and e) the boundary of $\boldsymbol{\Omega}_{\gamma_{\max}}$.

The method is now applied to the system depicted in Figure 3. Since only two state variables are involved in this problem, the outcomes of the methodology can be illustrated graphically in the phase plane as shown in Figure 4. Thus, its accuracy can be directly quantified and compared with other methods. The power of the CPL is set to 300 W, the degree of V is set to $d_{\mathrm{V}} = 4$, the degree of the Taylor series is set to $d_{\mathrm{T}} = 3$, the SOS decomposition is selected, and the SeDuMi SDP solver is used.

Figure 6 shows five curves that illustrate how the methodology proceeds. From Step 0 it is obtained that

$$\hat{\boldsymbol{f}} = \begin{bmatrix} -27.848x_1 - 25.316x_2 \\ 2000x_1 + 15.253x_2 - 0.076903x_2^2 + 0.00038774x_2^3 \end{bmatrix},$$

$$\boldsymbol{P} = \begin{bmatrix} 75.5512 & 0.8231 \\ 0.8231 & 1.0091 \end{bmatrix}.$$

Curve (a) is the boundary of the domain $\mathcal{D}_{\max}$ obtained from solving problem (22) in Step 1. Therefore, it is the equation $\boldsymbol{x}^{\top}\boldsymbol{P}\boldsymbol{x} = b_{\max} = 30271$. At this step the resulting Lyapunov function is

$$\begin{aligned} V_{\max}(\boldsymbol{x}) = {} & 306270\, x_1^2 - 8156.9x_1^3 + 3529.4x_1^4 + 3887.4\, x_2^2 \\ & -1.68x_2^3 + 0.57x_2^4 + 8041.1\, x_2x_1 - 336.19\, x_2x_1^2 \\ & +188.1\, x_2x_1^3 - 156.1\, x_2^3x_1 + 92.7\, x_2^2x_1^2 + 2.7\, x_2^3x_1. \end{aligned}$$

Curve (b) is the equation $\nabla V_{\max}\hat{\boldsymbol{f}} = 0$, as approximation for $\nabla V_{\max}\boldsymbol{f} = 0$. Since the curve (b) does not go into the ellipse (a), $\nabla V_{\max}\hat{\boldsymbol{f}} < 0$ in the domain $\mathcal{D}_{\max}$. Curve (c) is the boundary of $\boldsymbol{\Omega}_{\alpha_{\max}}$ obtained from solving problem (23) in Step 2. Therefore, (c) is the equation $V_{\max}(\boldsymbol{x}) = \alpha_{\max} = 2.65 \cdot 10^4$. It can be seen that $\boldsymbol{\Omega}_{\alpha_{\max}}$ is the biggest set contained in $\mathcal{D}_{\max}$. Curve (d) is the equation $\nabla V_{\max}\boldsymbol{f} = 0$, i.e., the time derivative of $V_{\max}$ along the trajectories of the original system (13). It can be seen that $\nabla V_{\max}\boldsymbol{f}$ does not meet the last condition of Theorem 1 in the set $\mathcal{D}_{\max}$, since the curve (d) goes into the ellipse (a). The curve (e) is the boundary of the domain $\boldsymbol{\Omega}_{\gamma_{\max}}$ obtained by solving problem (24) in Step 3. Therefore, (e) is the equation $V_{\max}(\boldsymbol{x}) = \gamma_{\max} = 1.78 \cdot 10^4$. It can be seen that now the curve (d) does not go into the

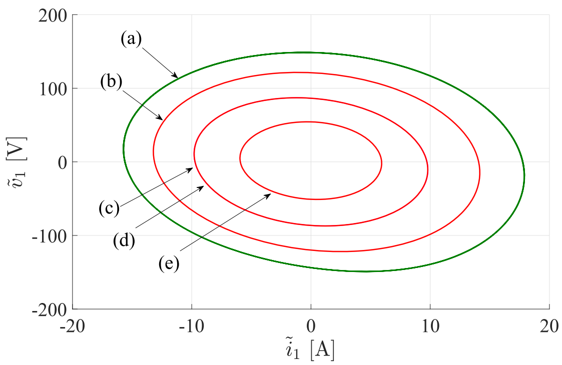

Fig. 7. Estimated ROA obtained by different methods; a) numerical simulation, b) proposed SOS approach, c) Lur'e approach, d) Takagi-Sugeno approach, and e) Brayton-Moser approach.

domain defined by the boundary (e). Thus, the last condition of Theorem 1 is achieved. Therefore, the curve (e) is the boundary of the estimated ROA of the system (13) obtained by applying the methodology.

The proposed approach is compared with the Lur'e problem approach [21], the Takagi-Sugeno fuzzy approach [20], and the Brayton-Moser method [17]. To observe the accuracy of the methodology, five curves are compared in Figure 7. Curve (a) is the boundary of the exact ROA computed by time-domain simulation as given in Section II-B. Curve (b) is the boundary of $\Omega_{\gamma_{\max}}$, i.e. the curve (e) in Figure 6. Curves (c), (d) and (e) are the boundaries of the ROA obtained by the methods used in [20], [21], and [17], respectively. It can be seen that the proposed method achieves a better ROA estimation compared with alternative state-of-the-art methods. Moreover, it can be shown that the results obtained by [20], [21] are equivalent to the results obtained by the SOS-based modeling when the search space is limited to quadratic Lyapunov functions, i.e. when setting $d_{\mathrm{V}} = 2$. Therefore, the proposed method is more general and can achieve less conservative solutions.

B. Scalability

The analysis of the scalability is performed by comparing the computational times of using different polynomial decompositions. Then, the difference between using sum of squares (SOS) and scaled-diagonally-dominant sum of squares (SDSOS) decompositions is studied. The SOS-based modeling is applied to the system (9) when the CPLs are controlled current source without internal dynamics, i.e. $\tau_j = 0$ for all $j = 1, \ldots, N$. Note that, although $\tau_j = 0$, the nonlinear behavior of the constant power load remains. This adaptation was done to scale the number of constant power loads.

Several experiments were run on a 4.2GHz Intel i7 processor, for different numbers of CPLs and different optimization options. The optimization options of the problem (22) are: the polynomial decomposition method, the degree d_{V} of the Lyapunov function candidate $V(\boldsymbol{x})$, and the degree d_{T} of the truncated Taylor expansion $\hat{\boldsymbol{f}}(\boldsymbol{x})$. The polynomial decomposition method can be set to SOS or SDSOS. To make the results comparable, the quantity $d_{\mathrm{V}} + d_{\mathrm{T}}$ is conserved. Otherwise, the complexity of problem (22) changes significantly as it is explained in Appendix IV. After each experiment, two quantities are recorded: the solution time and the maximum level of $\mathcal{D}$, $b_{\max}$. Note that $b_{\max}$ is not the parameter that defines the ROA, but it helps to assess how conservative the ROA will be. If $b_{\max}$ increases, then a better estimation of the ROA will be obtained.

TABLE II

COMPUTATIONAL EFFORT AND ACCURACY FOR MULTIPLE CPLS USING SOS DECOMPOSITION

N	$d_{\mathrm{V}} = 3$ $d_{\mathrm{T}} = 4$		$d_{\mathrm{V}} = 4$ $d_{\mathrm{T}} = 3$	
	Time (s)	$b_{\max}$	Time (s)	$b_{\max}$
1	2.2	5985	3.1	13022
2	12	3768	13.8	8551
3	129	3507	141	7830
4	1096	2997	1280	6247
5	9171	2387	9574	4824

TABLE III

DIFFERENCES BETWEEN SOS AND SDSOS DECOMPOSITIONS WHEN THE NUMBER OF CPLS IS $N = 5$

Decomp.	SOS	SDSOS	SOS	SDSOS
d_{V}	3	3	4	4
d_{T}	4	4	3	3
Time [s]	9171	674	9574	818
$b_{\max}$	2387	1619	4824	3172

Table II shows the performance of the proposed method when a different number of CPLs are considered. For this analysis, two optimization options are considered. The first one with $d_{\mathrm{V}} = 3$ and $d_{\mathrm{T}} = 4$, and the second with $d_{\mathrm{V}} = 4$ and $d_{\mathrm{T}} = 3$. In both cases, the SOS decomposition method is selected. Results show that the optimization option $d_{\mathrm{V}} = 4$ and $d_{\mathrm{T}} = 3$ obtains a larger domain $\mathcal{D}$ without a significant increase in the solution time. However, in both cases the computational time growths fast with the number of CPLs.

To overcome this issue, the SDSOS decomposition is applied. To illustrate the difference between using SOS or SDSOS decompositions, the results are shown in Table III. The number N of CPLs was set to five. The results show that SDSOS decomposition is able to solve the problem approximately 10 times faster than using SOS decomposition. Using SDSOS the solution $b_{\max}$ decreases approximately by 35% compared with using SOS. It was to be expected that the cost of such speed-up would be a reduction of the accuracy. Nevertheless, considering the significant speed-up, the observed reduction in accuracy can offer an acceptable trade-off.

VI. APPLICATION: ENLARGING THE ROA THROUGH CONTROL FEEDBACK

Consider a system of the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})u(t) \qquad (25)$$

with $\boldsymbol{x}(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$, and $\boldsymbol{f}$, $\boldsymbol{g}$ n-vectors of polynomials such that $\boldsymbol{f}(\mathbf{0}) = \mathbf{0}$. The objective is to synthesize a state feedback controller $u = K(\boldsymbol{x})$ with $K(\boldsymbol{x})$ being a polynomial that enlarges the ROA.

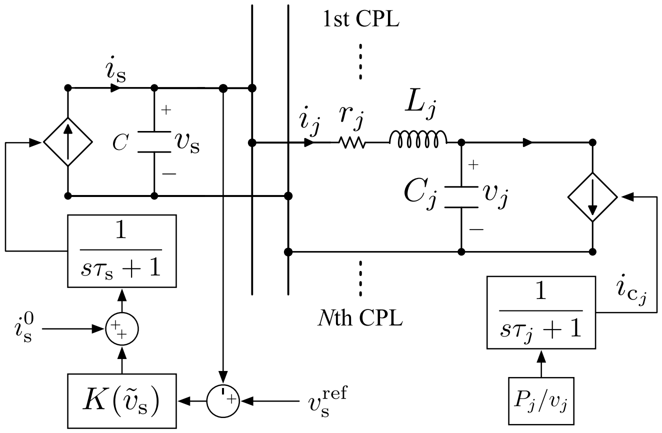

Fig. 8. Equivalent circuit of the DC microgrid with N CPLs and the polynomial droop controller.

Using Lemma 1 in Appendix B, the problem of enlarging the ROA can be recast into the following SOS program

$$\begin{aligned} \max_{b,V,K,s_1,s_2} \quad & b \\ \text{s.t.} \quad & V + s_1(\beta - b) - \varphi_1 \text{ is SOS} \\ & -\nabla V(\hat{f} + gK) + s_2(\beta - b) - \varphi_2 \text{ is SOS} \\ & s_1 \text{ and } s_2 \text{ are SOS} \end{aligned} \quad (26)$$

where b is the level, $\hat{f}$ is the polynomial approximation, V and K are polynomials, $\varphi_1(\boldsymbol{x})$ and $\varphi_2(\boldsymbol{x})$ are non-negative polynomials, and $s_1(\boldsymbol{x})$ and $s_2(\boldsymbol{x})$ are SOS polynomials.

Now, it is not possible to use the bisection method on b as it is used in problem (22) because of the bilinear term $\nabla V g K$ that appears in the second constraint. However, if K is a polynomial with few coefficients, a bisection method over the coefficients of K and b can be employed. This idea is exploited to design a nonlinear droop controller capable of enlarging the ROA of a DC system with CPLs. For this purpose, the following polynomial control structure is proposed:

$$K(\tilde{v}_{\mathrm{s}}) = -k_1\tilde{v}_{\mathrm{s}} - k_3\tilde{v}_{\mathrm{s}}^3 \quad (27)$$

where k_1 and k_3 are the linear and cubic droop coefficients, respectively.

The control structure in (27) allows higher current from the source compared with the linear droop control. This higher current allows to suppress oscillations caused by the nonlinear behavior of the CPLs in the presence of large disturbances. The current provided by the source is to have the same sign as the voltage error $\tilde{v}_{\mathrm{s}}$. Therefore, only coefficients of odd degree are taken into account. The maximum degree was set equal to three to maintain the control design problem practical. If more odd coefficients are added, a bisection search on these coefficients would be needed to solve problem (26). Coefficient k_1 is designed to achieve current sharing or voltage regulation using small-signal analysis [8]. Coefficient k_3 is designed to enlarge the ROA of the system. Therefore, k_1 is a given parameter of the problem (26), and k_3 is a decision variable.

To illustrate the problem of enlarging the estimated ROA, the system (9) is solved for three CPLs. The equivalent circuit of (9) is depicted in Figure 8. The parameters of system (9) are shown in Table IV. The linear droop coefficient k_1 is also given in Table IV. To solve the problem (26), a bisection search over k_3 in the interval from 0 to 5 is performed. On average, it takes 5.5 minutes to solve each problem. The maximum value of b is obtained for $k_3 = 1.6$. Systems with more CPLs can be handled with this methodology at an increase in solution time. If the solution time is a critical constraint, model reduction methods can be used. For example, in [9] it is shown that N CPLs connected to a common DC bus can be reduced to an equivalent CPL unit if the ratios L_j/r_j for $j = 1, \ldots, N$ are approximately equal.

TABLE IV
PARAMETERS OF SYSTEM CONTROL PROBLEM

Parameter	Value	Unit	Parameter	Value	Unit
$r_{j=\{1,2,3\}}$	0.035	Ω	τ_{s}	16	μs
$L_{j=\{1,2,3\}}$	110	μH	k_1	2	1/Ω
$C_{j=\{1,2,3\}}$	115	μF	C	2.46	μF
$\tau_{j=\{1,2,3\}}$	16	μs	$\bar{v}_{\mathrm{s}}$	48	V
P_1	80	W	$\bar{i}_{\mathrm{s}}$	2.08	A
$P_{j=\{2,3\}}$	10	W			

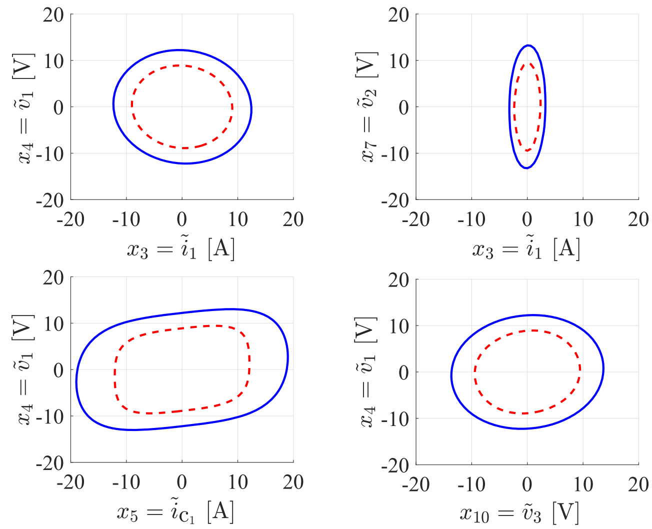

Fig. 9. Comparison of the estimated ROA: Linear-cubic droop controller (blue) and linear droop controller (red-dashed).

Figure 9 shows the estimated ROA with and without the nonlinear coefficient k_3. The ROA is given in a 2-dimensional plane for four combinations of state variables, while the rest of the variables are set equal to zero. As it can be seen, a considerable enlargement of the ROA is achieved in all the sub-spaces.

To study the performance of the nonlinear controller, the system (9) is simulated for two different initial conditions. In the first initial condition $\tilde{v}_1(0) = -0.74\bar{v}_1$ with $\bar{v}_1 = 47.94$ [V], and the remaining state variables are set to zero at $t = 0$. In the second initial condition $\tilde{v}_1(0) = -0.5\bar{v}_1$ with $\bar{v}_1 = 47.94$ [V], and the remaining state variables are set to zero at $t = 0$. These two initial conditions represent a large and a medium disturbance, respectively. Figure 10 shows the response of the system for both cases. In the case of the

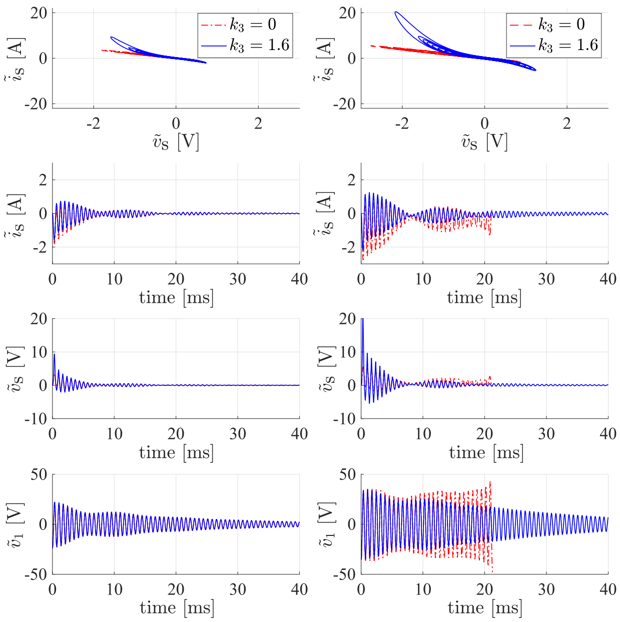

Fig. 10. Transient response and control behavior for a medium and a large disturbance. 1st column: medium disturbance. 2nd column: large disturbance. 1st row: control dynamics. 2nd row: current source $\tilde{i}_s$. 3rd row: bus voltage $\tilde{v}_s$. 4th row: voltage across CPL 1 $\tilde{v}_1$.

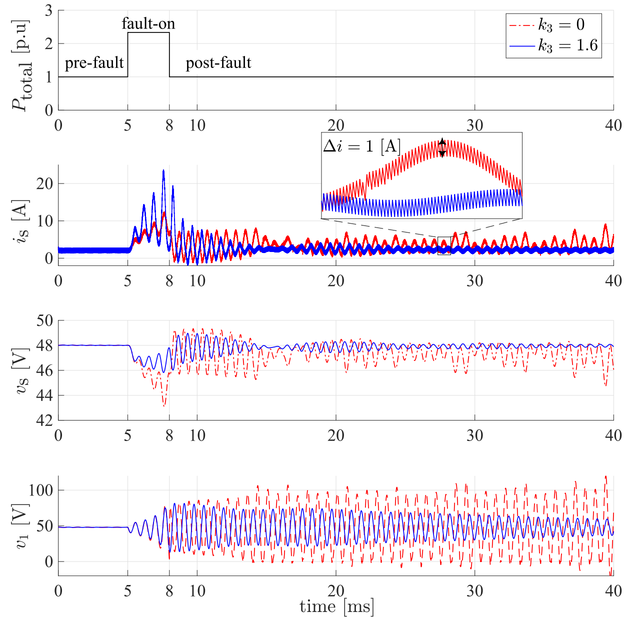

Fig. 11. Numerical results of the voltage and current dynamics when the DC microgrid with 3 CPLs is subjected to a load power disturbance.

TABLE V

PARAMETERS OF BIDIRECTIONAL DC-DC CONVERTER IN SYSTEM CONTROL PROBLEM

Parameter	Value	Unit	Parameter	Value	Unit
V_{in}	150	V	C	2.46	μF
V_{out}	48	V	L	0.26	mH
f_{sw}	100	kHz	k_a	0.3418	
r	0.1	mΩ	k_b	63000	

medium disturbance, both systems remain stable. However, the one with the linear-cubic droop controller shows better damping. In the case of the large disturbance, the system with the linear droop controller becomes unstable while the system with the nonlinear polynomial droop controller. As it was expected and designed, when the polynomial controller is employed, more current is available to support the nonlinear behavior of the CPL.

To have a more accurate validation of the polynomial controller performance, the detailed model of the buck converter connected to the voltage source is considered, referring to Figure 2. The system is modeled in Simulink using the SimPowerSystems library. For the inner current loop controller of the buck converter, a PI controller $k_a(1 + k_b/s)$ is implemented. The coefficients of the PI controller are tuned so that the inner current closed-loop cross-over frequency becomes equal to $\omega_s = 1/\tau_s$. The parameters of the converter source are listed in Table V.

Figure 11 shows the numerical results under a sudden large change of load to study transient stability. Two cases are distinguished: when the linear-cubic controller is turned off and when it is turned on. For each case, the quantities P_{total}, i_s, v_s, and v_1 are plotted. The initial conditions are set equal to the steady-state values of the system (9). During the time interval from 0 to 5 ms, the quantities converge to the operation point given by the detailed model. In both cases, during the pre-fault interval, the response is stable. During the time interval from 5 ms to 8 ms, the CPL 1 increases the power up to 4 times the initial power. This disturbance is reflected in an increment of the total system load P_{total} up to 2.33 times the initial load. In both cases, the system response to such disturbance involves a large excursion of system variables. At 8 ms the fault is cleared. During the post-fault interval, the system with the linear controller cannot converge to the original operating point. The voltage v_1 across the CPL 1 oscillates taking negative values. In contrast, the system with the polynomial controller returns to the operation point, revealing improved transient stability.

The simulations show that the system becomes more robust when adopting the nonlinear controller. It can tolerate larger perturbations without loss of stability. Note that this gain of robustness is not reflected in the eigenvalues of the linearized system.

VII. CONCLUSIONS

A new methodology to analyze and enhance the transient stability of a DC microgrid with constant power loads based on the sum of squares decomposition is proposed. An SOS program was formulated to compute a polynomial Lyapunov function. Results show that the proposed formulation achieves a better characterization of the ROA compared with recent references. The SOS-based modeling also handles the SDSOS decomposition, enhancing the solution time compared with the SOS decomposition without a significant compromise in the accuracy. With a similar SOS program formulation, the problem for enlarging the ROA by nonlinear control feedback is solved. A linear-cubic polynomial droop controller was computed. Results show that the ROA is enlarged, and also the damping of the system is increased. The transient stability was improved.

APPENDIX A
MODELING EQUATIONS

For the system (9)

$$\boldsymbol{x}=\begin{bmatrix}\boldsymbol{x}_0\\ \boldsymbol{x}_1\\ \vdots\\ \boldsymbol{x}_N\end{bmatrix},\quad \boldsymbol{A}\boldsymbol{x}=\begin{bmatrix}\boldsymbol{\Lambda}_0(\boldsymbol{x})\\ \boldsymbol{\Lambda}_1(\boldsymbol{x})\\ \vdots\\ \boldsymbol{\Lambda}_N(\boldsymbol{x})\end{bmatrix},\ \boldsymbol{\Gamma}(\boldsymbol{x})=\begin{bmatrix}\boldsymbol{\Gamma}_0(\boldsymbol{x})\\ \boldsymbol{\Gamma}_1(\boldsymbol{x})\\ \vdots\\ \boldsymbol{\Gamma}_N(\boldsymbol{x})\end{bmatrix},$$

where

$$\boldsymbol{x}_0=\begin{bmatrix}x_{0,1}\\ x_{0,2}\end{bmatrix}=\begin{bmatrix}\tilde{v}_{\mathrm{s}}\\ \tilde{i}_{\mathrm{s}}\end{bmatrix},\quad \boldsymbol{x}_j=\begin{bmatrix}x_{j,1}\\ x_{j,2}\\ x_{j,3}\end{bmatrix}=\begin{bmatrix}\tilde{i}_j\\ \tilde{v}_j\\ \tilde{i}_{\mathrm{c}_j}\end{bmatrix},$$

$$\boldsymbol{\Gamma}_0(\boldsymbol{x})=\begin{bmatrix}0\\ \Gamma_{0,2}(x_{0,1})\end{bmatrix},\quad \boldsymbol{\Lambda}_0(\boldsymbol{x})=\begin{bmatrix}\Lambda_{0,1}(\boldsymbol{x})\\ \Lambda_{0,2}(\boldsymbol{x})\end{bmatrix},$$

$$\boldsymbol{\Gamma}_j(\boldsymbol{x})=\begin{bmatrix}0\\ 0\\ \Gamma_{j,3}(x_{j,2})\end{bmatrix},\ \boldsymbol{\Lambda}_j(\boldsymbol{x})=\begin{bmatrix}\Lambda_{j,1}(\boldsymbol{x})\\ \Lambda_{j,2}(\boldsymbol{x})\\ \Lambda_{j,3}(\boldsymbol{x})\end{bmatrix},$$

$\forall j=1,\ldots,N$, with

$$\begin{aligned}\Lambda_{0,1}&=\frac{1}{C}\left(x_{02}-\sum_{j=1}^{N}x_{j1}\right),\\ \Lambda_{0,2}&=\frac{1}{\tau_{\mathrm{s}}}(-x_{02}),\\ \Gamma_{0,2}&=\frac{1}{\tau_{\mathrm{s}}}K(x_{01}),\\ \Lambda_{j,1}&=\frac{1}{L_j}(-r_jx_{j1}-x_{j2}+x_{01}),\\ \Lambda_{j,2}&=\frac{1}{C_j}(x_{j1}-x_{j3}),\\ \Lambda_{j,3}&=\frac{1}{\tau_j}(-x_{j3}),\\ \Gamma_{j,3}&=\frac{1}{\tau_j}-\bar{i}_jh(\tilde{v}_j).\end{aligned}$$

APPENDIX B
SOS-BASED LEMMAS

The following three lemmas are based on real algebraic geometry and the Positivstellensatz theorem. An explanation about how to derive these lemmas can be found in [28]. For all the lemmas consider the autonomous system (11) and assume that $\boldsymbol{f}(\boldsymbol{x})$ is a vector field of polynomial functions.

Lemma 1. Assume a given domain $\mathcal{D}$ defined in (18). If there exist a constant $b>0$, a polynomial function V, SOS polynomials $s_1(\boldsymbol{x})$, $s_2(\boldsymbol{x})$, and positive definite polynomial functions $\varphi_1(\boldsymbol{x})$, $\varphi_2(\boldsymbol{x})$, all of bounded degree, such that $V(0)=0$ and

$$\begin{aligned}&(i)\qquad V+s_1(\beta-b)-\varphi_1 \text{ is SOS},\\ &(ii)\ \ -\dot{V}+s_2(\beta-b)-\varphi_2 \text{ is SOS},\end{aligned}$$

then the origin is asymptotically stable.

Proof: The conditions (i) and (ii) ensure that $V(\boldsymbol{x})$ and $-\dot{V}(\boldsymbol{x})$ are positive definite on $\mathcal{D}$. Indeed, defining $g(\boldsymbol{x}):=\beta(\boldsymbol{x})-b$, the conditions of Lemma 1 are equivalent to (17) when $s_0=1$. Therefore, Lemma 1 serves to prove the conditions of Theorem 1.

Lemma 2. Assume a given Lyapunov function $V(\boldsymbol{x})$ over the domain $\mathcal{D}$ defined by (18) where $b>0$ is a given fixed value. Define the set $\Omega_\alpha:=\{\boldsymbol{x}\in\mathbb{R}^n\mid V(\boldsymbol{x})-\alpha\le 0\}$. If there exists a constant $\alpha>0$, an SOS polynomial $s_0(\boldsymbol{x})$, and a positive definite polynomial function $\varphi_1(\boldsymbol{x})$, all of bounded degree, such that

$$(i)\qquad -s_0(\beta-b)+(V-\alpha)-\varphi_1 \text{ is SOS},$$

then $\Omega_\alpha\subset\mathcal{D}$ is an estimate of the ROA.

Proof: Indeed, defining $g(\boldsymbol{x}):=V(\boldsymbol{x})-\alpha$, the condition of Lemma 2 is equivalent to (17) when $s_1=1$. Since s_0 is SOS, then $(\beta-b)<0$ in Ω_α. This proves that $\Omega_\alpha\subset\mathcal{D}$ is an estimate of the ROA, as the region defined in Theorem 1.

Lemma 3. Assume a given positive definite function $V(\boldsymbol{x})$ over the domain $\mathcal{D}$, i.e., $V(0)=0$ and $V(\boldsymbol{x})>0$ for all $\boldsymbol{x}\in\mathcal{D}-\{0\}$. Define the set $\Omega_\gamma:=\{\boldsymbol{x}\in\mathbb{R}^n\mid V(\boldsymbol{x})-\gamma\le 0\}$. If there exist a constant $\gamma>0$, SOS polynomials $s_0(\boldsymbol{x})$, $s_1(\boldsymbol{x})$, and a positive polynomial function $\varphi_1(\boldsymbol{x})$, all of bounded degree, such that

$$\begin{aligned}&(i)\qquad \Omega_\gamma\subset\mathcal{D}\\ &(ii)\ \ -s_0\dot{V}+s_1(V-\gamma)-\varphi_1 \text{ is SOS},\end{aligned}$$

then the origin is asymptotically stable, and Ω_γ is an estimate of the ROA.

Proof: The first condition ensures that $V(\boldsymbol{x})>0$ over the domain Ω_γ. Indeed, if $\Omega_\gamma\subset\mathcal{D}$ then $V(\boldsymbol{x})$ is positive definite in Ω_γ. The second condition forces $-\dot{V}>0$ over the domain Ω_γ. Indeed, defining $g(\boldsymbol{x}):=V(\boldsymbol{x})-\gamma$, the condition (ii) of Lemma 3 is equivalent to (17). Therefore, $V(\boldsymbol{x})$ fulfills the conditions of Theorem 1 over the domain Ω_γ.

APPENDIX C
DSOS AND SDSOS DECOMPOSITIONS

In this section, the DSOS and SDSOS decompositions proposed in [29] are considered. These decompositions certify nonnegativity by finding a diagonally dominant symmetric matrix and a scaled diagonally dominant symmetric matrix, respectively.

Definition 2. A polynomial $p(\boldsymbol{x})$ is a diagonally-dominant sum of squares (DSOS) if it can be written as

$$p(\boldsymbol{x})=\sum_i\alpha_im_i^2(\boldsymbol{x})+\sum_{i,j}\beta_{ij}^+(m_i(\boldsymbol{x})+m_j(\boldsymbol{x}))^2+\sum_{i,j}\beta_{ij}^-(m_i(\boldsymbol{x})-m_j(\boldsymbol{x}))^2$$

for some monomials $m_i(\boldsymbol{x})$, $m_j(\boldsymbol{x})$, and some nonnegative scalars α_i, β_{ij}^+, and β_{ij}^-.

Definition 3. A polynomial $p(\boldsymbol{x})$ is a scaled-diagonally-dominant sum of squares (SDSOS) if it can be written as

$$p(\boldsymbol{x})=\sum_i\alpha_im_i^2(\boldsymbol{x})+\sum_{i,j}(\hat{\beta}_{ij}^+m_i(\boldsymbol{x})+\tilde{\beta}_{ij}^+m_j(\boldsymbol{x}))^2+\sum_{i,j}(\hat{\beta}_{ij}^-m_i(\boldsymbol{x})-\tilde{\beta}_{ij}^-m_j(\boldsymbol{x}))^2$$

for some monomials $m_i(\boldsymbol{x})$, $m_j(\boldsymbol{x})$, and some scalars α_i, $\hat{\beta}_{ij}^{+}$, $\tilde{\beta}_{ij}^{+}$, $\hat{\beta}_{ij}^{-}$, and $\tilde{\beta}_{ij}^{-}$, with $\alpha_i \geq 0$.

Definition 4. A symmetric matrix $A = (a_{ij})$ is diagonally dominant (DD) if $a_{ij} \geq \sum_{j \neq i} |a_{ij}|$ for all i. A symmetric matrix A is scaled diagonally dominant (SDD) if there exists a positive definite diagonal matrix D such that DAD is DD.

The following theorem provided in [29] establishes the relation between the polynomial decomposition, the class of matrix, and the resulting optimization problem.

Theorem 3. *A polynomial $p(\boldsymbol{x}) \in \mathbb{R}[\boldsymbol{x}]_{n,2d}$ is DSOS (resp., SDSOS) if and only if there exists a symmetric matrix $\boldsymbol{Q} \in \mathbb{R}^{\binom{n+d}{d} \times \binom{n+d}{d}}$ such that $p(\boldsymbol{x}) = [\boldsymbol{x}]_d^{\top} \boldsymbol{Q} [\boldsymbol{x}]_d$ and $\boldsymbol{Q}$ is DD (resp., SDD). Moreover, for any fixed d, the optimization over DSOS (resp., SDSOS) can be done with a linear program (resp., second-order cone program) of size polynomial in n.*

IV. Complexity of SOS Programs

The complexity of an SOS program can be measured by the number of LMI scalar variables N_{SV} [36]. The number of LMI scalar variables is given by:

$$N_{\mathrm{SV}} = \sum_{i=1}^{N_{\mathrm{v}}} \eta_i + \sum_{i=1}^{N_{\mathrm{c}}} \theta_i, \tag{28}$$

where N_{v} is the number of polynomial variables, η_i is the number of free coefficients in the ith polynomial variable, N_{c} is the number of SOS constraints, and θ_i is the number of scalar variables for the ith SOS constraint. The number of free coefficients is given by

$$\eta_i = c(n, 2p_i), \tag{29}$$

where $c(k, q) = (k+q)!/(k!q!)$, n is the number of variables, and $2p_i$ is the degree of the ith polynomial variable. The number of scalar variables is given by

$$\theta_i = \frac{c(n, h_i)(c(n, h_i) + 1)}{2} - c(n, 2h_i) \tag{30}$$

where $2h_i$ is the degree of the ith SOS constraint.

In problem (22), the polynomial variables are V, s_1, and s_2. Therefore, $N_{\mathrm{V}} = 3$. The number of SOS constraints is $N_{\mathrm{C}} = 4$. The degrees of the first and second SOS constraint of problem (22) are defined as d_1 and d_2, respectively. Then,

$$\begin{aligned} d_1 &= \mathbf{max}\{\mathbf{deg}(V), \mathbf{deg}(s_1(\beta - b)), \mathbf{deg}(\varphi_1)\}, \\ d_2 &= \mathbf{max}\{\mathbf{deg}(\nabla V \hat{\boldsymbol{f}}), \mathbf{deg}(s_2(\beta - b)), \mathbf{deg}(\varphi_2)\}. \end{aligned}$$

Given d_{V} and d_{T}, the degrees of s_1 and s_2 are set as follows:

$$d_{\mathrm{s}_1} = \begin{cases} d_{\mathrm{V}} - d_\beta & \text{if } d_{\mathrm{V}} \text{ is even} \\ d_{\mathrm{V}} - d_\beta + 1 & \text{if } d_{\mathrm{V}} \text{ is odd} \end{cases}$$

$$d_{\mathrm{s}_2} = \begin{cases} d_{\mathrm{V}} + d_{\mathrm{T}} + d_\beta - 1 & \text{if } d_{\mathrm{V}} + d_{\mathrm{T}} - 1 \text{ is even} \\ d_{\mathrm{V}} + d_{\mathrm{T}} + d_\beta & \text{if } d_{\mathrm{V}} + d_{\mathrm{T}} - 1 \text{ is odd} \end{cases}$$

where $d_\beta = 2$ because $\beta(\boldsymbol{x})$ in (19) is quadratic. Assuming that $\mathbf{deg}(\varphi_1) \leq \mathbf{deg}(V)$ and $\mathbf{deg}(\varphi_2) \leq \mathbf{deg}(\nabla V \hat{\boldsymbol{f}})$, it follows that

$$d_1 = \begin{cases} d_{\mathrm{V}} & \text{if } d_{\mathrm{V}} \text{ is even} \\ d_{\mathrm{V}} + 1 & \text{if } d_{\mathrm{V}} \text{ is odd} \end{cases}$$

$$d_2 = \begin{cases} d_{\mathrm{V}} + d_{\mathrm{T}} - 1 & \text{if } d_{\mathrm{V}} + d_{\mathrm{T}} - 1 \text{ is even} \\ d_{\mathrm{V}} + d_{\mathrm{T}} & \text{if } d_{\mathrm{V}} + d_{\mathrm{T}} - 1 \text{ is odd} \end{cases}$$

The degrees of the third and fourth SOS constraint of problem (22) are equal to d_{s_1} and d_{s_2}, respectively. Therefore, given d_{V} and d_{T}, N_{SV} can be computed by (28)-(30).

For example, if $n = 2$, $d_{\mathrm{V}} = 4$, and $d_{\mathrm{T}} = 3$, then $N_{\mathrm{SV}} = 75$. The number of scalar variables is lower if it is considered that V, s_1 and s_2 contain only monomials of degree larger or equal than two [36].

References

[1] K. Strunz, E. Abbasi, and D. N. Huu, "DC microgrid for wind and solar power integration," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 2, no. 1, pp. 115–126, Mar. 2014.

[2] A. Emadi, A. Khaligh, C. H. Rivetta, and G. A. Williamson, "Constant power loads and negative impedance instability in automotive systems: Definition, modeling, stability, and control of power electronic converters and motor drives," *IEEE Trans. Veh. Technol.*, vol. 55, no. 4, pp. 1112–1125, Jul. 2006.

[3] A. Kwasinski and C. N. Onwuchekwa, "Dynamic behavior and stabilization of DC microgrids with instantaneous constant-power loads," *IEEE Trans. Power Electron.*, vol. 26, no. 3, pp. 822–834, Mar. 2011.

[4] X. Lu, K. Sun, J. M. Guerrero, J. C. Vasquez, L. Huang, and J. Wang, "Stability enhancement based on virtual impedance for DC microgrids with constant power loads," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2770–2783, Nov. 2015.

[5] N. Barabanov, R. Ortega, R. Griñó, and B. Polyak, "On existence and stability of equilibria of linear time-invariant systems with constant power loads," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 1, pp. 114–121, Jan. 2016.

[6] S. Anand and B. G. Fernandes, "Reduced-order model and stability analysis of low-voltage DC microgrid," *IEEE Trans. Ind. Electron.*, vol. 60, no. 11, pp. 5040–5049, Nov. 2013.

[7] M. Wu and D. D.-C. Lu, "A novel stabilization method of LC input filter with constant power loads without load performance compromise in DC microgrids," *IEEE Trans. Ind. Electron.*, vol. 62, no. 7, pp. 4552–4562, Jul. 2015.

[8] T. Dragičević, X. Lu, J. C. Vasquez, and J. M. Guerrero, "DC microgrids—Part I: A review of control strategies and stabilization techniques," *IEEE Trans. Power Electron.*, vol. 31, no. 7, pp. 4876–4891, Jul. 2016.

[9] L. Guo, S. Zhang, X. Li, Y. W. Li, C. Wang, and Y. Feng, "Stability analysis and damping enhancement based on frequency-dependent virtual impedance for DC microgrids," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 5, no. 1, pp. 338–350, Mar. 2017.

[10] F. Gao *et al.*, "Comparative stability analysis of droop control approaches in voltage-source-converter-based DC microgrids," *IEEE Trans. Power Electron.*, vol. 32, no. 3, pp. 2395–2415, Mar. 2017.

[11] A. Riccobono *et al.*, "Stability of shipboard DC power distribution: Online impedance-based systems methods," *IEEE Electrific. Mag.*, vol. 5, no. 3, pp. 55–67, Sep. 2017.

[12] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[13] H.-D. Chang, C.-C. Chu, and G. Cauley, "Direct stability analysis of electric power systems using energy functions: Theory, applications, and perspective," *Proc. IEEE*, vol. 83, no. 11, pp. 1497–1529, Nov. 1995.

[14] M. Belkhayat, R. Cooley, and A. Witulski, "Large signal stability criteria for distributed systems with constant power loads," in *Proc. Power Electron. Spec. Conf.*, vol. 2, Jun. 1995, pp. 1333–1338.

[15] A. Griffo and J. Wang, "Large signal stability analysis of 'More Electric' aircraft power systems with constant power loads," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 1, pp. 477–489, Jan. 2012.

[16] W. Du, J. Zhang, Y. Zhang, and Z. Qian, "Stability criterion for cascaded system with constant power load," *IEEE Trans. Power Electron.*, vol. 28, no. 4, pp. 1843–1851, Apr. 2013.

[17] K. Cavanagh, J. A. Belk, and K. Turitsyn, "Transient stability guarantees for ad hoc DC microgrids," *IEEE Contr. Syst. Lett.*, vol. 2, no. 1, pp. 139–144, Jan. 2018.

[18] P. Magne, D. Marx, B. Nahid-Mobarakeh, and S. Pierfederici, "Large-signal stabilization of a DC-link supplying a constant power load using a virtual capacitor: Impact on the domain of attraction," *IEEE Trans. Ind. Appl.*, vol. 48, no. 3, pp. 878–887, May/Jun. 2012.

[19] D. Marx, P. Magne, B. Nahid-Mobarakeh, S. Pierfederici, and B. Davat, "Large signal stability analysis tools in DC power systems with constant power loads and variable power loads—A review," *IEEE Trans. Power Electron.*, vol. 27, no. 4, pp. 1773–1787, Apr. 2012.

[20] H. Kim, S. Kang, G. Seo, P. Jang, and B. Cho, "Large-signal stability analysis of DC power system with shunt active damper," *IEEE Trans. Ind. Electron.*, vol. 63, no. 10, pp. 6270–6280, Oct. 2016.

[21] L. Herrera, W. Zhang, and J. Wang, "Stability analysis and controller design of DC microgrids with constant power loads," *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 881–888, Mar. 2017.

[22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, vol. 15. Philadelphia, PA, USA: SIAM, 1994.

[23] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Math. Program.*, vol. 96, pp. 293–320, May 2003.

[24] A. Papachristodoulou and S. Prajna, "Analysis of non-polynomial systems using the sum of squares decomposition," in *Positive Polynomials Control*, D. Henrion and A. Garulli, Eds. Berlin, Germany: Springer, 2005, pp. 23–43.

[25] Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, "Control applications of sum of squares programming," in *Positive Polynomials Control*, D. Henrion and A. Garulli, Eds. Berlin, Germany: Springer, 2005, pp. 3–22.

[26] M. Anghel, F. Milano, and A. Papachristodoulou, "Algorithmic construction of Lyapunov functions for power system stability analysis," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 9, pp. 2533–2546, Sep. 2013.

[27] S. K. Mazumder and E. P. de la Fuente, "Stability analysis of micropower network," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 4, no. 4, pp. 1299–1309, Dec. 2016.

[28] P. A. Parrilo, *Chapter 3: Polynomial Optimization, Sums of Squares, and Applications*. Philadelphia, PA, USA: SIAM, 2012, pp. 47–157.

[29] A. A. Ahmadi and A. Majumdar, "DSOS and SDSOS optimization: More tractable alternatives to sum of squares and semidefinite optimization," *SIAM J. Appl. Algebra Geometry*, vol. 3, no. 2, pp. 193–230, Apr. 2019.

[30] F. Gao, S. Bozhko, A. Costabeber, G. Asher, and P. Wheeler, "Control design and voltage stability analysis of a droop-controlled electrical power system for more electric aircraft," *IEEE Trans. Ind. Electron.*, vol. 64, no. 12, pp. 9271–9281, Dec. 2017.

[31] R. Han, M. Tucci, A. Martinelli, J. M. Guerrero, and G. Ferrari-Trecate, "Stability analysis of primary plug-and-play and secondary leader-based controllers for DC microgrid clusters," *IEEE Trans. Power Syst.*, vol. 34, no. 3, pp. 1780–1800, May 2019.

[32] P. Kundur, "Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definitions," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1387–1401, May 2004.

[33] A. Papachristodoulou, S. Prajna, P. Seiler, and P. A. Parrilo. (2013). *SOSTOOLS: Sum of Squares Optimization Toolbox for MATLAB*. [Online]. Available: http://arxiv.org/abs/1310.4716

[34] J. Lofberg, "Pre- and post-processing sum-of-squares programs in practice," *IEEE Trans. Autom. Control*, vol. 54, no. 5, pp. 1007–1011, May 2009.

[35] A. Megretski, "Systems polynomial optimization tools (SPOT)," Massachusetts Inst. Technol., Cambridge, MA, USA, 2010. [Online]. Available: https://github.com/spot-toolbox/spotless

[36] G. Chesi, "On the complexity of sos programming: Formulas for general cases and exact reductions," in *Proc. SICE Int. Symp. Control Syst. (SICE ISCS)*, Mar. 2017, pp. 1–6.

Bernardo Severino received the B.Sc. degree in electrical engineering from the Universidad de Chile, 2010. He is currently pursuing the Ph.D. degree with TU Berlin, Germany. His research fields are the stability analysis and the control design of dc microgrids.

Kai Strunz has been a Professor of sustainable electric networks and sources of energy with TU Berlin, Germany, since 2007. He is also a Guest Professor with the Chinese Academy of Sciences, Beijing. He received the IEEE PES Prize Paper Award in 2015 and the IEEE Journal of Emerging and Selected Topics in Power Electronics First Prize Paper Award in 2015. In 2012, he was the General and Technical Program Chair of the 2012 IEEE PES Innovative Smart Grid Technologies Europe, Berlin. He is also the Chair of the IEEE PES Subcommittee on Distributed Energy Resources.