SOS Methods for Demonstrating Dissipativity for Switched Systems

Michael J. McCourt and Panos J. Antsaklis, *Fellow IEEE*

***Abstract*—Dissipativity is a valuable tool for analysis and synthesis of dynamical systems, however, the search for an appropriate storage function to show the property is often not straightforward. This paper uses sum of squares (SOS) optimization methods to determine whether a nonlinear or switched system is dissipative. This is especially useful for switched systems where notions of dissipativity involve finding multiple storage functions. Examples and relevant software code are provided to illustrate these methods.**

I. Introduction

Dissipativity is an energy based property of dynamical systems [1], [2]. A system is dissipative if it only stores and dissipates energy provided by the environment without generating its own energy. While general dissipativity allows for the most general results, the special case of QSR dissipativity [3], [4] is of interest because of its more applicable form and it has computational advantages. The special case of passivity is of significance because passive systems are Lyapunov stable and the property of passivity is preserved when systems are combined in feedback or parallel [5], [6]. These results can be used together to design stable large scale systems. In our previous work, these methods have been used in analyzing network control systems [7]–[9].

The results provided by dissipativity theory are useful when it is possible to show that a given system is dissipative. Demonstrating dissipativity requires finding an energy storage function which is analogous to finding a Lyapunov function when demonstrating stability. When dissipativity can be shown computationally, the analysis and synthesis involved in control system design is greatly simplified. Traditionally this could be done for linear systems with quadratic supply rates using Linear Matrix Inequalities (LMIs) [10]. Recently, this has been extended to a class of nonlinear systems by using Sum of Squares (SOS) methods [11].

The notions of passivity and dissipativity have been extended to switched systems in continuous [12], [13] and discrete time [14]–[16]. Additionally, passivity indices have been defined for switched systems [17]. These notions of dissipativity rely on multiple storage functions which is based on stability of switched systems using multiple Lyapunov functions [18]. For dissipativity, multiple storage functions capture the fact that energy may be stored differently for each mode of the system. While there are promising results using multiple storage functions, the burden of demonstrating dissipativity is increased significantly compared to finding a single storage function in the non-switched case.

M. J. McCourt is with the Research and Engineering Education Facility (REEF) at the University of Florida, email: mccourt@ufl.edu.

P. J. Antsaklis is with the Department of Electrical Engineering at the University of Notre Dame, email: antsaklis.1@nd.edu.

The support of the National Science Foundation under Grant No. CNS-1035655 is gratefully acknowledged.

This paper explores computational methods to aid in demonstrating dissipativity for nonlinear and switched systems. This includes utilizing SOS methods to demonstrate dissipativity or find passivity indices for nonlinear systems. These methods are also used to find both storage functions and cross supply rates for switched systems. Examples are provided to demonstrate how these methods can be used in practice.

This paper is organized as follows. Section 2 contains background material on dissipativity theory and existing SOS methods. Section 3 covers SOS methods applied to finding storage functions for non-switched systems. This includes an example as well as the special case of passivity indices. Using SOS methods for dissipativity of switched systems is given in Section 4. Section 5 provides an illustrative example for switched systems. Concluding remarks are given in Section 6.

II. Background Material

A. Dissipativity Theory

For now, the systems of interest are nonlinear systems of the form,

$$\begin{array}{rcl}\dot{x} &=& f(x)+g(x)u \\ y &=& h(x)+j(x)u.\end{array} \qquad (1)$$

where the functions f, g, h, and j are of appropriate dimension given by the system state $x \in \mathbb{R}^n$, input $u \in \mathbb{R}^m$, and output $y \in \mathbb{R}^p$. It is assumed that the vector field f is Lipschitz with respect to x. It can be assumed without a loss of generality that $f(0)=0$ and $h(0)=0$. This system is considered polynomial when the functions f, g, h, and j are polynomial functions of the state.

Nonlinear systems can be shown to be stable in the sense of Lyapunov when there exists a function $V(x) > 0, \forall x \neq 0$ such that $\dot{V}(x) \leq 0$. When the Lyapunov function is a polynomial function of the state, it is referred to as a polynomial Lyapunov function. This class of functions is more general than quadratic Lyapunov functions.

A dissipative system is one that stores and dissipates energy supplied by the environment without generating its own energy. Dissipativity theory provides sufficient conditions for stability of single systems and for feedback interconnections. As a property of dynamical state space systems, dissipativity is more general than passivity and $\mathcal{L}_2$ gain. In many cases it is possible to use stability results from dissipativity theory when the passivity theorem or small gain theorem do not apply.

The notion of internally stored energy in a system is captured by an energy storage function $V(x)$ that is analogous to a Lyapunov function. The rate energy is supplied to the system is captured by an energy supply rate $\omega(u, y)$. This rate can be varied to capture a range of dynamical behaviors.

Definition 1. *[1] A system is dissipative if there exists a non-negative energy storage function $V(x)$ such that the energy stored in the system is always bounded above by the energy supplied to the system, i.e. for $u(t) \in U$ and $\forall t_1, t_2$ s.t. $t_1 \leq t_2$*

$$\int_{t_1}^{t_2} \omega(u, y) dt \geq V(x(t_2)) - V(x(t_1)). \tag{2}$$

For this paper, the input space will be assumed to be, $U = \mathbb{R}^m$. When V is continuously differentiable, which will be assumed in this paper, dissipativity can be written, $\dot{V}(x) \leq \omega(u, y)$. When using computational methods, it is not always possible to use the general definition of dissipativity due to the unconstrained form of ω. It is possible to use SOS methods when ω is restricted to be polynomial as it is in the following quadratic form.

Definition 2. *[3] A system is QSR-dissipative if it is dissipative with respect to the supply rate,*

$$\omega(u, y) = \begin{bmatrix} y \\ u \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} y \\ u \end{bmatrix}, \tag{3}$$

where $Q = Q^T$ and $R = R^T$.

The parameters Q, S, and R can be chosen to assess a variety of dissipative behaviors including passivity and finite-gain $\mathcal{L}_2$ stability. QSR dissipative systems are $\mathcal{L}_2$ stable when $Q < 0$. It is also possible to assess stability for the feedback interconnection of two dissipative systems by analyzing the QSR parameters of the two systems.

The special case of passivity is of interest as many practical systems are passive including many electrical circuits, mechanical systems, and chemical processes. Passive systems are Lyapunov stable and the property of passivity is preserved when systems are combined in parallel or feedback. As a result, stability of large scale systems may be assessed quickly when the components are passive. For more details on passivity theory, refer to [5], [6].

Definition 3. *[2] A system is passive if it is dissipative with respect to the supply rate,*

$$\omega(u, y) = u^T y. \tag{4}$$

Passive linear time-invariant (LTI) systems are also known as positive real systems [19]. A necessary and sufficient test for an LTI system to be passive is the KYP Lemma [20]. This result can be written as an LMI so that systems can be shown to be passive using computational methods [10], [19]. The use of SOS methods for analysis and control of nonlinear systems arose from the success of LMI methods for LTI systems.

B. Sum of Squares Methods

Several problems in control systems can be formulated as searches for a positive definite or positive semi-definite function. It is clear that this is the case for Lyapunov stability, by showing that $V > 0$ and $-\dot{V} \geq 0$ the system is stable. Traditionally, these problems are computationally efficient to solve for linear systems using LMIs while they are np-hard for the general nonlinear case [21]. Other problems in control systems have been studied using SOS methods including robustness [22], region of attraction [22], hybrid system verification [23], and stability with delays [24], among others.

These problems have recently been approached for nonlinear systems using semi-definite programming. The key step in formulating the optimization problem is in replacing the positive semi-definite condition with an alternative sufficient condition. This condition is that the function is instead a sum of squares (SOS) of lower order polynomials [21].

$$F(x) = \sum_i f_i^2(x) \geq 0 \tag{5}$$

Clearly, a function being SOS implies that the function is positive although it is not a necessary condition in general.

These methods can be directly applied to a given system when the dynamics are polynomial. A detailed list of common polynomial nonlinearities can be found in [11]. We consider polynomials that are functions of an n^{th} dimensional variable. In order for a polynomial to be a sum of squares, the degree m must be even. We take the function of interest $F(x)$ and write it in the following form,

$$F(x) = z^T(x) \Theta z(x), \tag{6}$$

where z is the stacked vector of monomials up to degree $m/2$. It has been shown that $F(x)$ has a sum of squares decomposition if and only if it can be written as in (6) with a positive semidefinite Θ [25]. This result enabled the use of semi-definite programming for polynomial nonlinear systems. When an appropriate matrix factorization of Θ can be found, it is possible to determine the SOS representation of the function F [26].

For SOS problems, the optimization problem is typically written in the following way.

$$\begin{array}{rl} \min & c_1 u_1 + ... + c_n u_n \\ \text{subject to} & P_i(x) = A_{i,0}(x) + A_{i,1}(x) u_1 + ... + A_{i,n}(x) u_n \\ \text{for} & i = 1, ..., n \end{array} \tag{7}$$

This problem can be shown to be convex and solvable using semi-definite programming. Computational solvers are available such as SOSTOOLS [27] for MATLAB. SOSTOOLS relies on semi-definite solvers such as SeDuMi.

A typical control problem using SOS methods begins with a given system with polynomial dynamics and assumes a polynomial form for the unknown functions, $A_{i,j}$ in (7). The functions are linearly parameterized by coefficients u_j such that, when appropriate coefficients are found, the system has a desired property. Semi-definite solvers may be used to find the coefficients subject to the constraints including the system dynamics. When the problem is feasible, there exists a function that is SOS, and positive semi-definite, which implies the system has the desired property. Often the quantity to minimize is not important as any feasible solution solves the problem. For these feasibility problems, the value of the cost coefficients c_j are arbitrary.

III. SOS Methods for Dissipativity

A. Finding Storage functions

Using SOS methods to demonstrate passivity or dissipativity for a dynamical system can be done similarly to demonstrating Lyapunov stability. Showing dissipativity requires finding a non-negative storage function $V(x)$ such that $\dot{V} \leq \omega(u, y)$. Both of these conditions can be relaxed to the following SOS conditions where the functions $p_i(x)$ are polynomials. The optimization problem has the form (7) subject to

$$V(x) = \sum_i a_i p_i(x)^2 \tag{8}$$

$$\omega(u, y) - \frac{\partial V}{\partial x}(f(x) + g(x)u) = \sum_j b_j p_j(x)^2. \tag{9}$$

Typically finding a single storage function is sufficient so the values of c_i are arbitrary. An important exception, finding passivity indices, will be addressed later. Regardless of the linear quantity to minimize, a solution to the problem may be found using semi-definite solvers. This problem can be solved using SOSTOOLS with the following constraints.

```
ineq1 = V-a*(x'*x);
ineq2 = u'*(h+j*u)-dVdx*(f+g*u);
prog = sosineq(prog,ineq1);
prog = sosineq(prog,ineq2);
```

The first inequality is used to ensure $V(x) > 0$. The parameter $a = .001$ is chosen to ensure $V(x) \neq 0$ for $x \neq 0$. The second inequality forces the dissipative inequality (2) to hold.

Like the other SOS problems, the existence of a SOS storage function is only sufficient for showing passivity or dissipativity. When the SOS solver fails to find a solution, the system may or may not be dissipative.

SOS optimization methods have been employed to find energy storage functions for dissipative polynomial systems for specific energy supply rates [11], [28], [29]. These papers show that dissipation inequalities involving an unknown storage function can be formulated as a SOS problem. The authors investigate dissipative inequalities for the minimum phase property, robustness, and synchronizing feedbacks.

B. Nonlinear System Example

It may help to illustrate how SOS methods can be used with an example. This example demonstrates how a storage function can be found to show passivity. Consider a polynomial nonlinear system,

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -(x_1^3 + x_1 x_3^2)(1 + x_3^2) \\ -(x_1^2 x_2 + x_2)(1 + x_3^2) \\ -(x_3 + x_1^2 x_3)(1 + x_3^2) - 3x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \\ y &= \begin{bmatrix} x_3 \end{bmatrix}. \end{aligned} \tag{10}$$

A storage function is chosen to be of the form $V(x) = x^T P x + a_1 x_1^4 + a_2 x_2^4 + a_3 x_3^4$. Using SOSTOOLS in MATLAB the system is shown to be passive with $P = diag\{1.70, 1.29, 0.5\}$, $a_1 = 0.867$, $a_2 = 0.815$, and $a_3 = 0$ which yields the storage function $V(x) = 1.70x_1^2 + 1.29x_2^2 + 0.5x_3^2 + 0.867x_1^4 + 0.815x_2^4$. A quick check can be done to verify passivity by evaluating $\dot{V}(x) = \frac{\partial V(x)}{\partial x}[f(x) + g(x)u]$ which will be bounded above by $u^T y$.

It should be noted that this is just one such storage function to show that this system is passive. By adjusting the coefficients c_i of the cost function it is possible to find other storage functions.

C. Finding Passivity Indices

A special case of QSR dissipativity that is of importance is the passivity index framework. This framework is a generalization of passivity that allows for the level of passivity to be captured with two indices. Passive systems may be analyzed for the level of excess passivity while non-passive systems can be analyzed for how close to a passive system the dynamics are. Conceptually, a "nearly passive" but not passive system can be stabilized by the feedback of an "excessively passive" system. More details on passivity indices can be found in [6], [30].

Definition 4. *A system has passivity indices ρ and ν if it is QSR dissipative with respect to the matrices*

$$Q = -\rho I, \; S = \frac{1}{2}(1 + \rho\nu)I, \; \text{and } R = -\nu I \tag{11}$$

where I is the identity matrix in the appropriate dimension.

Since showing that a system has specific passivity indices is a special case of showing dissipativity, SOS methods may be used to find an appropriate storage function. This approach can be used to test whether a certain index pair (ρ, ν) is valid for a given system.

Alternatively, the approach can be used to actually find each index. In the stability results using passivity indices it is desirable to know the largest ρ or ν that holds for a given system. This allows for the greatest flexibility when designing stabilizing controllers. Often it is known that a particular index is zero. A SOS program can be setup to maximize the other index. In general, SOS methods cannot be used to simultaneously maximize both indices. This is due to the S matrix in (11) where the product $\rho\nu$ appears. SOS methods can only be used when the functions of interest are linear with respect to the unknowns.

An alternative approach can be used where the value of ρ is fixed in order to maximize ν and vice versa. A range of fixed values can be tested for each index and the results recorded to arrive at an appropriate pair of indices. This approach is not a blind grid search, and typically produces a number of satisfactory (ρ, ν) pairs. More on this approach can be found in the technical report [30].

IV. SOS Methods for Dissipative Switched Systems

SOS methods facilitate the process of demonstrating that a system is passive or dissipative by finding an energy storage function. This approach is particularly useful for switched systems where multiple storage functions are typically required. When a system has more than a few discrete modes, this notion of dissipativity becomes overly cumbersome to apply without relying on computational methods.

A. Dissipativity for Switched Systems

A nonlinear switched system has the following form, where $x \in R^n$, $u \in R^m$, and $y \in R^p$,

$$\begin{array}{rcl} \dot{x} & = & f_\sigma(x,u) \\ y & = & h_\sigma(x,u). \end{array} \tag{12}$$

It is assumed that for each subsystem $i \in \{1, ..., M\}$, f_i is Lipschitz with respect to x, $f_i(0,0) = 0$, and $h_i(0,0) = 0$. The function $\sigma : \mathbb{R}^+ \to \{1, ..., M\}$ is piecewise constant and indicates the index of the current active subsystem. At any given time, $\sigma(t) = i$ for $i \in \{1, ..., M\}$ and the dynamics are nonlinear and time-invariant. The switching in the system is caused by discrete behavior from an underlying hybrid process. Each subsystem of the switched system represents a discrete mode of that process so the terms *mode* and *subsystem* will be used interchangeably. The state variable $x(t)$ is continuous at all times including switching instants where it is typically not differentiable.

A single switching instant is denoted by t_{i_k}, which is the k^{th} time that the i^{th} subsystem becomes active. This system becomes inactive at time $t_{(i_k+1)}$ and becomes active again at time $t_{i_{(k+1)}}$. The values of i are a subset of $\mathbb{Z}^+$ (the positive integers) from 1 to M, and k take on values in $\mathbb{Z}^+$ that is allowed to be infinite. To avoid Zeno behavior, it is assumed that on any finite time interval, t_0 to arbitrary time T, the system switches a finite number of times K, where K may depend on the time T chosen. To avoid trivial asymptotic analysis, it is assumed that the system switches an infinite number of times on the infinite time horizon.

The notion of dissipativity for switched systems uses multiple storage functions. This is based on existing notions including [13], [15], [16]. Each storage function V_i captures the internal energy storage when mode i is active. The storage functions are positive and bounded by class-$\mathcal{K}_\infty$ functions $\underline{\alpha_i}$ and $\overline{\alpha_i}$,

$$\underline{\alpha_i}(||x||) \leq V_i(x) \leq \overline{\alpha_i}(||x||). \tag{13}$$

Recall that a function $\alpha(x)$ is class-$\mathcal{K}_\infty$ if $\alpha(0) = 0$, $\alpha(x) > 0$ for $x > 0$, α is strictly increasing, and α is unbounded [5].

Definition 5. *[16] A switched system (12) is QSR dissipative if there exist positive definite storage functions $V_i(x)$, energy supply rates $\omega_i(u,y)$ and cross supply rates $\omega_j^i(u,y,x,t)$ such that the following conditions hold.*

1) *Each subsystem i is QSR dissipative while active, i.e. for $t_{i_k} \leq t_1 \leq t_2 \leq t_{i_k+1}$ and $\forall i, k$,*

$$\int_{t_1}^{t_2} \begin{bmatrix} y \\ u \end{bmatrix}^T \begin{bmatrix} Q_i & S_i \\ S_i^T & R_i \end{bmatrix} \begin{bmatrix} y \\ u \end{bmatrix} dt \geq V_i(x(t_2)) - V_i(x(t_1)). \tag{14}$$

2) *Each subsystem j is dissipative with respect to ω_j^i when the i^{th} subsystem is active, i.e. $\forall j \neq i$, and for $t_{i_k} \leq t_1 \leq t_2 \leq t_{i_k+1}$,*

$$\int_{t_1}^{t_2} \omega_j^i(u,y,x,t)dt \geq V_j(x(t_2)) - V_j(x(t_1)). \tag{15}$$

3) *For all i and j there exist absolutely integrable functions $\phi_j^i(t)$ and some input $u^*(t)$ that may depend on the state $x(t)$ such that the following three conditions hold, $\forall t \geq t_0$,*
   a) *$f_i(0, u^*(t)) \equiv 0$,*
   b) *$\omega_i^i(u^*, y) \leq 0$, and*
   c) *$\omega_j^i(u^*, y, x, t) \leq \phi_j^i(t), \forall j \neq i$.*

Switched systems are passive when $Q_i = 0$, $R_i = 0$, and $S_i(u,y) = \frac{1}{2}I$ for all i. When considering passivity the input $u^*(t) = 0$ satisfies $(3a)$ and $(3b)$, however, the existence of functions ω_j^i and ϕ_j^i in $(3c)$ is not trivial. This final condition may be satisfied for all switching sequences or for state restricted switching. More detail on satisfying $(3c)$ can be found in [13]. Passive switched systems are Lyapunov stable when all storage functions satisfy $V_i(0) = 0$. Additionally, the feedback and parallel interconnections of two passive systems forms a new passive system as expected.

Dissipative switched systems are stable when $Q_i < 0$ for all i. The feedback of two switched systems can be analyzed from the QSR parameters of each system since the feedback forms a new QSR dissipative system. More details on these results can be found in [16].

B. SOS for Storage Functions

For SOS methods to directly apply, the switched system must have polynomial dynamics. For all i the functions $f_i(x,u)$ and $h_i(x,u)$ must be polynomial in x and u. Additionally, all functions involved in showing dissipativity must be polynomial. This includes the energy storage functions V_i and the cross supply rates ω_j^i.

The first step in showing dissipativity for a switched system is in specifying the energy supply rates. When passivity is the property of interest, these are $\omega_i(u,y) = u^T y, \forall i$. For QSR dissipativity, the rates may be chosen in advance or an appropriate rate may be found for each mode by the optimization program. For each mode i, the rates are parameterized by the elements of matrices Q_i, S_i, and R_i. These values may be found by the solver simultaneously with the storage functions.

The next step in showing dissipativity is to find an energy storage function for each mode of the system. These storage functions are dependent on the energy supply rate specified previously (14). A SOS optimization program can be specified to find each storage function assuming that a form is chosen for the storage function. Fortunately, the forms for the storage functions may be generated automatically when the variables of interest $\{x_1, \cdots, x_n\}$ and the desired order of the storage function are specified. For linear systems, a quadratic form can be chosen, $V_i(x) = x^T P_i x$ where $P_i = P_i^T$. For nonlinear systems, a 4^{th} or 6^{th} order storage function may be preferred. In general, the storage function can be written

$$V_i(x) = z^T(x)\Theta_i z(x) \tag{16}$$

where $\Theta_i = \Theta_i^T$. The form is fully specified when $z(x)$ is chosen and the SOS program finds the elements of Θ_i.

The optimization problem to find each storage function can be written in as a semi-definite program (7) subject to

$$V_i(x) = \sum_k a_k p_k(x)^2 \tag{17}$$

$$\omega_i(u,y) - \frac{\partial V_i}{\partial x} f_i(x,u) = \sum_l b_l p_l(x)^2 \tag{18}$$

This optimization problem can be solved in MATLAB using SOSTOOLS. In this case, $i = 1$ and $\omega_i = u^T y$.

```
ineq1 = V1-.001*(x'*x);
ineq2 = u'*h1-dV1dx*f1;
```

The SOS program can be repeated for each mode of the system to find all storage functions or the problem can be combined into a single large SOS program. This guarantees that condition (14) of dissipativity for switched systems holds.

C. SOS for Cross Supply Rates

In addition to storage functions, showing dissipativity for switched systems requires finding as many as $M(M-1)$ different cross supply rates (15). Like storage functions, a form must be chosen for the cross supply rates and the coefficients may be determined by the optimization problem. For two modes i and j, a good form for the cross supply rates can be chosen by comparing the dynamics of the two modes and the storage functions V_i and V_j determined in the previous SOS program. Since these are both known quantities, the forms can be specified computationally.

For example, consider the cross supply rate ω_1^2 which captures the rate energy is supplied to mode 1 when mode 2 is active. The rate depends on the energy storage for mode 1 ($V_1(x)$) and it depends on the energy dissipation of mode 2 which includes $V_2(x)$, ω_2^2, and the dynamics of mode 2. The quantity of interest is the difference between the energy being stored for subsystem 2 and for subsystem 1.

$$\omega_2^2(u, h_2(x,u)) - \frac{\partial V_2}{\partial x} f_2(x,u) + \frac{\partial V_1}{\partial x} f_2(x,u) \tag{19}$$

All functions in this expression are known so the expression is fully specified. Additionally, the functions are all polynomial. The resulting polynomial terms in the expression can be extracted and used in the candidate for the cross supply rates with unknown coefficients. The coefficients can be found by running a SOS program. Additional terms may be added to the cross supply rate if desired. When an appropriate form is chosen, the optimization program can be executed with the condition

$$\omega_j^i(u,y,x,t) - \frac{\partial V_j}{\partial x} f_i(x,u) = \sum_k a_k p_k(x)^2. \tag{20}$$

This constraint can be implemented in SOSTOOLS when $i = 1$ and $j = 2$.

```
ineq = omega_12-dV2dx*f1;
```

Overall, finding cross supply rates is more involved than finding storage functions. For one, there are as many as $M(M-1)$ cross supply rates so there are many more functions to determine. Additionally, each function is dependent on more terms so each SOS program may take longer to execute. However, this analysis method is expected to run off-line. Once the storage functions and cross supply rates are determined, further analysis can use these functions without running additional SOS programs.

The third condition of dissipativity involves the search for an input $u(t)$ that has certain properties. These properties with such an input has a strong parallel with control Lyapunov functions that are used to find stabilizing inputs. In some cases the existence of such an input is obvious from the dynamics and the cross supply rates. In other cases it may be very difficult to find such an input. However, for passivity of switched systems the input $u(t) = 0$ often satisfies these conditions. For now, there does not appear to be a SOS method to find such an input which is a limitation to this approach.

V. Switched Systems Example

The SOS methods discussed above is applied to a nonlinear switched system with two modes to demonstrate that the system is passive. Mode 1 has dynamics given by

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -0.6x_1^3 - 2x_1 + 2x_2 \\ -1.2x_1 - 3x_2 + u \end{bmatrix} \\ y &= \begin{bmatrix} x_2 \end{bmatrix}, \end{aligned} \tag{21}$$

and mode 2 has dynamics given by

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -2x_1^3 + 0.5x_2 \\ -0.6x_1 - 3x_2 - x_2^3 + u \end{bmatrix} \\ y &= \begin{bmatrix} x_2 \end{bmatrix}. \end{aligned} \tag{22}$$

Both modes are passive when active so $\omega_i(u,y) = u^T y$ for $i = 1, 2$. SOS methods can be used to find storage functions to demonstrate passivity when active. The form of storage function is assumed to contain all terms quadratic or quartic in x_1 and x_2. The optimization problem results in the storage functions,

$$\begin{aligned} V_1(x) &= 0.3x_1^2 + 0.5x_2^2 \\ V_2(x) &= 0.6x_1^2 + 0.5x_2^2. \end{aligned} \tag{23}$$

Now SOS methods can be used to find cross supply rates. The following form was chosen for the cross supply rates,

$$\begin{aligned} \omega_i^j(u,y,x,t) &= u^T y + a_1 x_1^2 + a_2 x_1 x_2 + a_3 x_2^2 + a_4 x_1^4 \\ &\quad + a_5 x_1^3 x_2 + a_6 x_1^2 x_2^2 + a_7 x_1 x_2^3 + a_8 x_2^4. \end{aligned} \tag{24}$$

The optimization problem yields the cross supply rates,

$$\begin{array}{rcl}\omega_2^1(u,y,x,t) & = & u^Ty - 1.2x_1^4 - x_2^4 + 0.0075x_1^2 - 3x_2^2\\ \omega_1^2(u,y,x,t) & = & u^Ty - 0.7x_1^4 - 2.4x_1^2 + 1.2x_1x_2 - 3x_2^2.\end{array} \tag{25}$$

Some terms are not present in the final cross supply rates due to the optimization problem returning a zero value for those coefficients. For both cross supply rates, the input $u(t) = 0$ is considered for condition $(3c)$ of Definition 5. There are many functions $\phi(t)$ that can bound ω_1^2 by substituting $u(t) = 0$ and studying the remaining terms, e.g. $\phi(t) = ||x(t_0)||\, e^{-t}$ for all $x \in \mathbb{R}^2$. It is not possible to bound the rate ω_2^1 for all x. Considering $u(t) = 0$, a region in $\mathbb{R}^2$ may be defined where ω_2^1 is positive. A switching rule may be defined to avoid this region. For this example, the set is small and defined by $\mathcal{X} = \{x \in \mathbb{R}^2 |\, ||x_1|| < 0.0866,\ ||x_1|| > 20\,||x_2||$ For this set, only mode 2 may be active. With this restriction to the switching, the system is passive and also stable for zero input.

This process can be repeated as needed to determine passivity, passivity indices, or more general forms of dissipativity. The only restriction is that the energy supply rates and cross supply rates are polynomials that are linear in the decision variables. As SOS methods are sufficient only, the failure of a test for a specific energy supply rate does not imply that the system is not dissipative with respect to that supply rate. It only implies that the property cannot be shown using SOS methods for the chosen forms of storage functions and cross supply rates.

VI. Conclusions

This paper discussed SOS methods for automating the process of demonstrating that a given dynamical system is dissipative or passive. This included showing dissipativity for nonlinear systems with emphasis on the special case of passivity indices. Then the case of switched systems was covered where SOS methods can be used to find both energy storage functions and cross supply rates. As a given switched system has many modes, there may be several energy storage functions and even more cross supply rates. These automated methods allow results from dissipativity theory to be more readily applicable to real systems. Examples and relevant pieces of MATLAB code were provided to illustrate and facilitate using these methods.

References

[1] J. C. Willems, "Dissipative dynamical systems, Part I: General theory," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 321–351, 1972.

[2] ——, "Dissipative dynamical systems Part II: Linear systems with quadratic supply rates," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 352–393, 1972.

[3] D. J. Hill and P. J. Moylan, "The Stability of Nonlinear Dissipative Systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 5, pp. 708–711, 1976.

[4] ——, "Stability Results of Nonlinear Feedback Systems," *Automatica*, vol. 13, pp. 377–382, 1977.

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

[6] J. Bao and P. L. Lee, *Process Control*. London: Springer-Verlag, 2007.

[7] M. J. McCourt and P. J. Antsaklis, "Stability of Networked Passive Switched Systems," in *Conference on Decision and Control*, 2010, pp. 1263–1268.

[8] M. J. McCourt, E. Garcia, and P. J. Antsaklis, "Model-based dissipative control of nonlinear discrete-time systems over networks," *University of Notre Dame, ISIS Technical Report, isis-2012-005*, 2012.

[9] P. J. Antsaklis, B. Goodwine, V. Gupta, M. J. McCourt, Y. Wang, P. Wu, M. Xia, H. Yu, and F. Zhu, "Control of cyberphysical systems using passivity and dissipativity based methods," *European Journal of Control*, vol. 19, no. 5, pp. 379–388, 2013.

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

[11] C. Ebenbauer and F. Allgöwer, "Analysis and design of polynomial control systems using dissipation inequalities and sum of squares," *Computers & Chemical Engineering*, vol. 30, no. 10-12, pp. 1590–1602, 2006.

[12] M. Zefran, F. Bullo, and M. Stein, "A notion of passivity for hybrid systems," in *Conference on Decision and Control*, 2001, pp. 768–773.

[13] J. Zhao and D. J. Hill, "Dissipativity Theory for Switched Systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 941–953, 2008.

[14] A. Bemporad, G. Bianchini, and F. Brogi, "Passivity Analysis and Passification of Discrete-Time Hybrid Systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 4, pp. 1004–1009, 2008.

[15] B. Liu and D. J. Hill, "Decomposable dissipativity and related stability for discrete-time switched systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1666–1671, 2011.

[16] M. J. McCourt and P. J. Antsaklis, "Stability of interconnected switched systems using QSR dissipativity with multiple supply rates," in *American Control Conference*, 2012, pp. 4564 – 4569.

[17] ——, "Control Design for Switched Systems Using Passivity Indices," in *American Control Conference*, 2010, pp. 2499–2504.

[18] M. S. Branicky, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[19] N. Kottenstette and P. J. Antsaklis, "Relationships Between Positive Real, Passive Dissipative, & Positive Systems," in *American Control Conference*, 2010, pp. 409–416.

[20] R. E. Kalman, "Lyapunov functions for the problem of Lur'e in automatic control," *Proceedings of the National Acadamy of Science USA*, vol. 49, no. 2, pp. 201–205, 1963.

[21] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, 2000.

[22] A. Papachristodoulou, "Scalable analysis of nonlinear systems using convex optimization," Ph.D. dissertation, California Institute of Technology, 2005.

[23] S. Prajna, "Optimization-based methods for nonlinear and hybrid systems verification," Ph.D. dissertation, California Institute of Technology, 2005.

[24] A. Papachristodoulou, M. M. Peet, and S. Lall, "Analysis of polynomial systems with time delays via the sum of squares decomposition," *IEEE Transactions on Automatic Control*, vol. 54, no. 5, pp. 1058–1064, May 2009.

[25] M. D. Choi, T. Y. Lam, and B. Reznick, "Sums of squares of real polynomials," *Proceedings of Symposia in Pure mathematics*, vol. 58, no. 2, pp. 103–126, 1995.

[26] A. Packard, U. Topcu, P. Seiler, and G. Balas, "Help on SOS," *Control Systems IEEE*, vol. 30, no. 4, pp. 18–23, 2010.

[27] S. Prajna, A. Papachristodoulou, and P. A. Parrilo, "SOSTOOLS for MATLAB, 2004," 2004. [Online]. Available: http://www.cds.caltech.edu/sostools/

[28] C. Ebenbauer and F. Allgöwer, "Minimum-phase property of nonlinear systems in terms of a dissipation inequality," in *American Control Conference*, 2004, pp. 1737–1742.

[29] ——, "A dissipation inequality for the minimum phase property," *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 821–826, Apr. 2008.

[30] M. J. McCourt and P. J. Antsaklis, "Demonstrating passivity and dissipativity using computational methods," *University of Notre Dame, ISIS Technical Report, isis-2013-008*, 2013.