Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Calculation of Critical Fault Recovery Time for Nonlinear Systems based on Region of Attraction Analysis [1]

S. Mojtaba Tabatabaeipour * **Mogens Blanke** *,**

* *Automation and Control Group, Dept. of Electrical Engineering, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark; e-mail: {setaba,mb}@elektro.dtu.dk).*
** *AMOS CoE at Institute for Technical Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway*

Abstract: In safety critical systems, the control system is composed of a core control system with a fault detection and isolation scheme together with a repair or a recovery strategy. The time that it takes to detect, isolate, and recover from the fault (fault recovery time) is a critical factor in safety of a system. It must be guaranteed that the trajectory of a system subject to fault remains in the region of attraction (ROA) of the post-fault system during this time. This paper proposes a new algorithm to compute the critical fault recovery time for nonlinear systems with polynomial vector fields using sum of squares programming. The proposed algorithm is based on computation of ROA of the recovered system and finite-time stability of the faulty system.

1. INTRODUCTION

Dynamics of a system controlled with an active fault tolerant controller subject to a fault can be described by a switched nonlinear system [Blanke et al., 2006, Ch. 7]. One can distinguish between three periods of the operation. The pre-fault period, the fault-on period and the post-fault period. In the pre-fault period, the system is working in the nominal mode with the controller designed for the nominal system. The fault-on period is the time interval from the occurrence of the fault until the fault is detected, isolated, and repaired or the system is reconfigured through a reconfiguration mechanism. During this period the system is controlled by the controller designed for the nominal system. When the fault is recovered or the system is reconfigured, it enters the post-fault period.

Because the nominal controller is not designed to control the faulty situations, depending on the severity of the fault, and the duration of the fault-on period, the system might become unstable after recovery [Zhang and Jiang, 2006].To guarantee that the system remains stable after reconfiguration or recovery, it must be assured that the trajectories of the faulty system during the fault-on period remains in the region of attraction (ROA) of the post-fault system. This is illustrated in Fig 1. Ω_n is ROA of the normal system and Ω_r is the ROA of the post-fault system with x_e^n and x_e^r denoting the corresponding equilibrium points respectively. ϕ_1^f shows a trajectory of the faulty system where the system is recovered at the appropriate time such that the system's state is in the ROA of the post-fault system at the end of the fault-on period. Consequently, the trajectory converges to the equilibrium point of the post-fault system. The post-fault trajectory is depicted by ϕ_1^r. In case the fault is cleared after the system's states exit the ROA of the post-fault system (see ϕ_2^f), the system's trajectory diverges and becomes unstable. ϕ_2^r depicts the trajectory in this situation. This illustrates the criticality of timing in fault recovery.

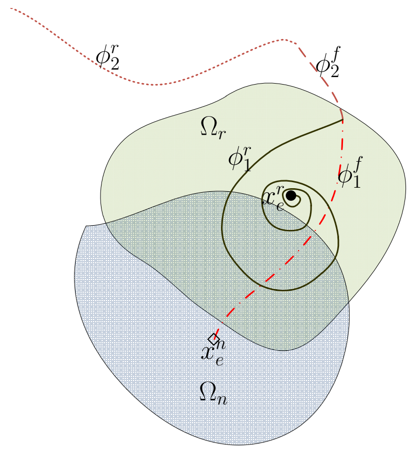

Fig. 1. Illustration of criticality of the recovery time. (ϕ_1^f, ϕ_1^r) is a trajectory where the system is recovered before leaving the ROA of the post-fault system, (ϕ_2^f, ϕ_2^r) is a trajectory where the system is recovered after leaving the ROA of the post-fault system.

This paper suggests a method to compute the maximum permissible time to recover from a fault, referred to as the critical recovery time (CRT) hereafter. To the best of our knowledge this problem is not addressed before in the literature in the general setting that we address in this paper. To compute the CRT, first, the region of attraction (ROA) of the post-fault system is computed. Then, having the ROA of the post-fault system, we propose an algorithm to compute the maximum time that guarantees that the

[1] This work was supported through the SOSPO project by the Danish Council for Strategic Research under grant no. 11-116794.

trajectories of the system initiated from an initial set would remain in it. To compute the ROA and CRT we use sum of squares programming.

Related to the results in paper are the literature in power system which consider the problem of fault-clearing for transient stability analysis [Kundur, 1994]. The methods are mainly based on Monte-Carlo simulations or numerical integration based the nonlinear model of the system [Pavella et al., 2000]. Alternative methods are based on finding analytical energy function for the power system [Chiang, 2011]. Simulation based method cannot provide us with guaranteed transient stability analysis due to the uncertainty in the set of initial conditions at the occurrence of fault. Also, energy based function methods are based on simplified models where the transfer conductances are neglected [Chiang, 2011].

Contributions of this paper are the followings: we propose a new method for estimation of CRT for nonlinear system based on the estimation of ROA. In estimating ROA, we modify the algorithm in [Jarvis-Wloszek, 2003] in two ways. We add safety constraints and we estimate an invariant subset of ROA that is contained in the safe set given by safety constraints. Also, instead of using a fixed shape to enlarge the estimate of ROA, we update the shape factor iteratively. In computing CRT, we use our recent results on finite time stability of nonlinear systems Tabatabaeipour and Blanke [2014] and we show how to use the proposed method to compute the CRT.

This paper is organized as follows. Preliminaries and basic definitions used throughout the paper are introduced in Section 2. Section 3, formulates the problem. Section 4 proposes an algorithm to compute an invariant subset of ROA contained in a given set describing safety constraints. Section 5, develops an algorithm to compute CRT based on the estimate of ROA. Throughout the paper, a single machine infinite bus system subject to a three phase short circuit fault, is used as an example to demonstrate steps of the proposed method. Conclusions are finally given.

2. PRELIMINARIES

This section provides basic definitions and concepts used throughout the paper.

Definition 1. **Monomial:** A monomial m_α is a function $m_\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$ which is defined as: $m_\alpha(x) = x^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \dots x_n^{\alpha_n}$. The degree of a monomial is defined as $deg\ m_\alpha = \Sigma_{i=1}^n \alpha_i$.

Definition 2. [Parrilo, 2003] **Polynomial:** a polynomial $p(x)$ is a linear combination of a finite number of monomials: $p(x) := \sum_{j=1}^k c_j m_{\alpha_j}(x)$. The degree of a polynomial is defined as $p := max_j(deg\ m_{\alpha_j})$.

The set of all polynomials with n variables is denoted by $\mathcal{R}_n$. The set of positive semidefinite polynomials denoted by $\mathcal{P}_n$ are the set of polynomials that are nonnegative on all $\mathbb{R}^n$ which is defined by: $\mathcal{P}_n := \{p \in \mathcal{R}_n : p(x) \geq 0, \forall x \in \mathbb{R}^n\}$.

Definition 3. **Sum of squares polynomial:** A polynomial p is said to be sum of squares (SOS) if it can be decomposed to a sum of squares of M polynomials $p_1, \dots, p_M$ i.e $p = \sum_{i=1}^M p_i(x)^2$.

The set of all SOS polynomials in n variables is denoted by Σ_n which is defined as: $\Sigma_n := \{s \in \mathcal{R}_n : \exists M, p_i, \in \mathcal{R}_n, i = 1, \cdots, M$ such that $s = \sum_{i=1}^M p_i^2\}$.

Proposition 1. A polynomial $p(x) \in \mathcal{R}_n$ of degree $2d$ is SOS if and only if there exist a positive semidefinite matrix $Q \geq 0$ and a vector of monomials $z(x)$ in n variables up to degree d such that $p(x) = z^T(x) Q z(x)$.

Theorem 1. [Parrilo, 2003] The existence of a SOS decomposition of a polynomial system in n variables of degree $2d$ can be formulated as a linear matrix inequality (LMI) feasibility problem test.

The following lemma is used to check conditions of the form $g_0(x) \geq 0$ whenever $g_1(x) \geq 0, \cdots, g_m(x) \geq 0$ by converting them into sum of squares programming [Jarvis-Wloszek, 2003].

Lemma 1. (Generalized S-procedure) Given functions $g_0(x), g_1(x), \cdots, g_m(x) \in \mathcal{R}_n$, if there exist $s_1, s_2, \cdots, s_m \in \Sigma_n$ such that $g_0 - \Sigma_{i=1}^m s_i g_i \in \Sigma_n$ then, it holds that:

$$\{x \in \mathbb{R}^n : g_1(x) \geq 0, \cdots, g_m(x) \geq 0\} \subseteq \{x \in \mathbb{R}^n : g_0(x) \geq 0\}. \tag{1}$$

Closure of a given set D is denoted by $\bar{D}$.

3. PROBLEM FORMULATION

We consider a controlled system with a fault recovery or a reconfiguration mechanism. A fault occurs at $t = t_f$ and the system is recovered or reconfigured at $t = t_r$. Dynamic of the overall closed-loop system can be described by the following switched nonlinear system:

$$\dot{x} = \begin{cases} f_n(x) \text{ for } 0 \leq t < t_f, \\ f_f(x) \text{ for } t_f \leq t < t_r, \\ f_r(x) \text{ for } t_r \leq t < \infty, \end{cases} \tag{2}$$

where $x \in \mathbb{R}^n$ is the state of the system, t_f is the time that the fault occurs, t_r is the time that the system is reconfigured or recovered from the fault. The dynamics of the system during nominal operation is described by $\dot{x} = f_n(x)$. When a fault occurs at t_f the dynamic of the system switches to $\dot{x} = f_f(x)$ and when the system recovers from the fault, its dynamic switches to $\dot{x} = f_r(x)$. Generally, the dynamic of the reconfigured system might be different from that of the nominal system, but if the fault is totally repaired then it could be the case that $f_r = f_n$. Also, note that the equilibrium point of the system in the post-fault period might be different from that of the nominal operation. We assume that f_n, f_f, and f_r are nonlinear functions. We also assume that the nominal and the post-fault closed-loop system are designed such that they are locally stable.

To investigate the stability of the system, we use the concept of region of attracktion (ROA), which is defined by the largest set of initial conditions whose trajectories would converge to the equilibrium point of the system.

Definition 4. **Region of Attraction:** Given the nonlinear system $\dot{x} = f(x)$ with the equilibrium point of $x_e = 0$, the ROA of the origin is defined as:

$$\Omega := \{x_0 : \lim_{t \rightarrow \infty} \phi(t, x_0) = 0\}, \tag{3}$$

where $\phi(t, x_0)$ is a solution of the nonlinear system with the initial state x_0.

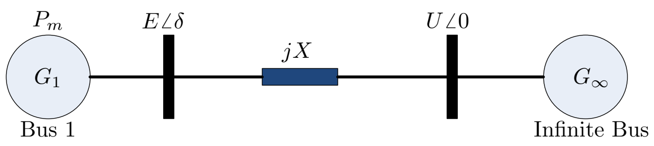

Fig. 2. Single machine infinite bus system

To guarantee the stability of the system, the trajectories of the faulty system initiated at time t_f must not exit the ROA of the post-fault system during the fault-on period $[t_f, t_r)$.

Example To demonstrate the idea, consider a single machine infinite bus (SMIB) system. SIMB is widely used in the power system literature, to investigate transient stability of a multi-machine system using extended transient stability approach where the generators of the system are divided into two groups. The system is equivalent to a two machine system which is modeled as a SMIB, see e.g. [Pavella et al., 2000], [Xue et al., 1988].

The dynamic of the system is given by:

$$\begin{cases} \dot{\delta} = \omega \\ M\dot{\omega} = P_m - P_e \sin(\delta) - D\omega \end{cases} \tag{4}$$

where δ is the machine rotor angle, ω is the relative angular velocity of the rotor, P_m is the mechanical power input of the machine, P_e is the maximum electrical power output of the machine, and M is the inertia of the machine. The values of these parameters are chosen as: $P_m = 1$ per unit, $P_e = EU/X = 1.35$ per unit, $M = 0.2$, $D = 0.12$.

We consider a three phase short circuit fault at a point in the line such that the power transmitted to the bus becomes $P_e = 0.5 \sin\delta$. The dynamic of the machine in the faulty condition is given by:

$$\begin{cases} \dot{\delta} = \omega, \\ M\dot{\omega} = P_m - 0.5 \sin\delta - D\omega. \end{cases} \tag{5}$$

We assume that the fault is detected and cleared at $t = t_r$ so that the dynamic of the system switches back to that of the nominal condition. Whather the machine losses synchronism with the infinite bus depends on the fault clearing time. If the fault is not cleared at the right time, then the state of the machine exits the ROA of the post-fault condition and the machine looses synchronism.

Figure 3 shows the system initiated with $w_0 = -2, \delta_0 = 2$, the fault occurs at $t_f = 10$, and it is cleared at $t = 10.48$. The black line shows the boundary of ROA for the pre-fault system which is the same as that of the post-fault system. As we can see, when the fault is cleared the states of the system are already outside the ROA of the post-fault system and the machine looses synchronism. This emphasizes the importance of critical clearing time.

Computation of the critical clearing time includes two main steps: estimation of ROA of the post-fault system Ω_r, and then computation of the time that it takes the fault-on trajectories to leave the Ω_r. In the following section we address the first step.

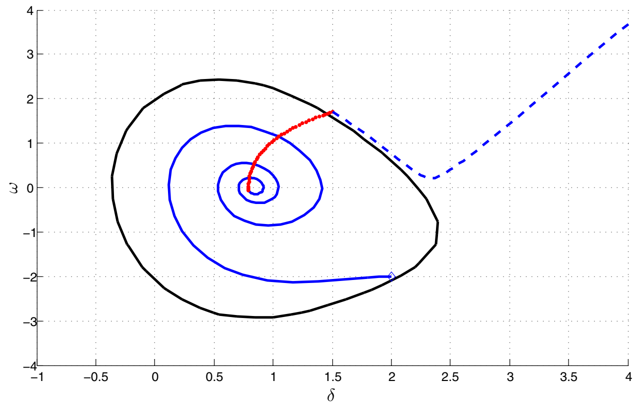

Fig. 3. ROA of the post-fault system (solid black). Trajectory of the system in pre-fault condition (solid blue), faulty condition (solid red), and post-fault condition (dashed blue), The fault is cleared at $t = 10.48$

4. ESTIMATION OF REGION OF ATTRACTION

Computation of ROA of a nonlinear system is generally a very difficult problem, see [Chesi, 2011]. In our work, we are interested in computing an inner estimate of the ROA, namely a set that is contained in the ROA of the system. The reason for this is as follows. Assume that we find an outer estimate of the ROA. If we can guarantee that the trajectories of the faulty system remain inside the outer estimate, we cannot guarantee that they remain inside the ROA itself.

In this section we give a numerical method to compute an inner estimate of the ROA based on the algorithm proposed in [Jarvis-Wloszek, 2003]. The idea of the algorithm is to expand a set contained in the interior of a level set of the Lyapunov function (LF) of the system. The algorithm in [Jarvis-Wloszek, 2003], do not consider any constraints on the state of the system, but here we include some constraints on the states of the system. These constraints could be due to safety reasons. Moreover, we introduce an inner loop in our algorithm to update the shape of the set that is used for expansion of the estimate of the ROA. We assume the system is safe as long as the states are inside the given set:

$$\mathcal{X}_s = \{x : g_s(x) \geq 0\}. \tag{6}$$

The goal is to compute an inner estimate of ROA that is also contained in the safe set $\mathcal{X}_s$. This set is a level set of a Lyapunov function of the system that is contained in the safe set. The method is based on finding invariant subsets of the ROA.

Theorem 2. [Jarvis-Wloszek, 2003] If there exist a continuously differentiable function V such that:

$$\begin{cases} V \text{ is positive definite,} \\ \Omega_V := \{x \in \mathbb{R}^n : V(x) \leq 1\} \text{ is bounded,} \\ \dot{V} = \dfrac{\partial V}{\partial x} f < 0, \forall x \in \Omega_V \setminus \{0\}, \end{cases} \tag{7}$$

then Ω_V is an invariant subset of the ROA.

To find an invariant subset of ROA that is also contained in the safe set, $\mathcal{X}_s$, the following conditions must be satisfied:

$$\begin{cases} V \text{ is positive definite,} \\ \Omega_V := \{x \in \mathbb{R}^n : V(x) \leq 1\} \text{ is bounded,} \\ \dot{V} = \dfrac{\partial V}{\partial x} f < 0, \forall x \in \Omega_V \setminus \{0\}, \\ \Omega_V \subseteq \mathcal{X}_s, \end{cases} \tag{8}$$

To expand the ROA, the authors in [Jarvis-Wloszek, 2003] and [Topcu et al., 2010] defined a fixed shape region with a variable size given by:

$$P_\beta := \{x \in \mathbb{R}^n : p(x) \leq \beta\} \tag{9}$$

with the constraint that $P_\beta \subseteq \Omega_V$ where p is a positive definite polynomial. The idea is to expand the the ROA by expanding the set P_β that is contained in it. In other words, we are expanding the ROA by expanding its interior. Then, the problem is cast as the following optimization problem:

$$\max \ \beta$$
$$\text{s. t.} \begin{cases} V(x) > 0 \ \forall x \in \mathbb{R}^n \setminus \{0\} \text{ and } V(0) = 0, \\ \{x \in \mathbb{R}^n : V(x) \leq 1\} \text{ is bounded,} \\ \{x \in \mathbb{R}^n : V(x) \leq 1\} \setminus \{0\} \subseteq \{x \in \mathbb{R}^n : \dfrac{\partial V}{\partial x} f < 0\}, \\ \{x \in \mathbb{R}^n : p(x) \leq \beta\} \subseteq \{x \in \mathbb{R}^n : V(x) \leq 1\}, \\ \{x \in \mathbb{R}^n : V(x) \leq 1\} \subseteq \{x \in \mathbb{R}^n : g_s(x) \geq 0\}, \end{cases} \tag{10}$$

Using the S-procedure, then the best inner estimate of ROA can be find by solving the following SOS programming:

$$\max_{V(0)=0, s_1, s_2, s_3, s_4 \in \Sigma_n} \beta$$
$$\text{s. t.} \begin{cases} V - l_1 \in \Sigma_n \\ -[(\beta - p)s_1 + (V - 1)] \in \Sigma_n \\ -[(1 - V)s_2 + \dfrac{\partial V}{\partial x} f s_3 + l_2] \in \Sigma_n \\ g_s - (1 - V)s_4 \in \Sigma_n \end{cases} \tag{11}$$

where l_1 and l_2 are of the form:

$$l_i = \sum_{i=1}^{n} \epsilon_{ij} x_j^2,$$

where ϵ_i are small positive scalars. The above problem is bilinear in its variable. To solve it we use the following iterative algorithm given in Algorithm 1.

The algorithm is initialized with a linearized model of the system. A LF for the linearized system is obtained by solving the LMI (12). This LF is used to initiate the optimization problem. At the γ step, the largest level set of the LF that is contained in the safe set $\mathcal{X}_s$ such that $\dot{V} < 0$ is obtained. Then, the largest set P_β that is contained in the γ level set of the LF is computed. Having P_{β^*}, we update the candidate LF. In the initialization step, an LF of degree two ($x^T P x$) is used. In this step, we can use LFs with higher degrees to improve the estimation. After scaling, the LF is used to update the shape of the P_β by choosing $p^i = \tilde{V}$. The shape factor has an important role in enlarging the estimation. If the shape of the P_β is fixed and its level set does not aligned with the level sets of ROA, then we would have a poor estimation of ROA [Chakraborty et al., 2011]. Therefore, in our algorithm at each iteration p^i is chosen as the latest update of LF. In this way the level sets of P_β are closely aligned with the level sets of ROA which yields a better estimate of ROA.

Algorithm 1 Computation of ROA with safety constraints

Initialization.1: Use the linearized model of the system for initialization: $A = \frac{\partial f}{\partial x}|_{x=0}$. Find a positive definite P that solves:

$$A^T P + PA = -I. \tag{12}$$

$V^0 = x^T P x$ and $p^0(x) = V^0(x)$, $i = 0$.

Initialization.2:, Fix $\tilde{V} = V^0$ and solve the following optimization problem by bisection on γ:

$$\gamma^* := \max \gamma$$
$$s.t. \begin{cases} -[(\gamma - \tilde{V})s_2 + \dfrac{\partial \tilde{V}}{\partial x} f s_3 + l_2] \in \Sigma_n \\ g_s - (\gamma - \tilde{V})s_4 \in \Sigma_n \end{cases} \tag{13}$$

repeat

$i \leftarrow i + 1$

P_β **step** : Fix $\tilde{V} = V^{i-1}$, $\tilde{p} = p^{i-1}$, and solve the following optimization problem by bisection on β:

$$\beta^* := \max \beta$$
$$s.t. \begin{cases} -[(\beta - \tilde{p})s_1 + (\tilde{V} - \gamma^*)] \in \Sigma_n \\ -[(\gamma^* - \tilde{V})s_2 + \dfrac{\partial \tilde{V}}{\partial x} f s_3 + l_2] \in \Sigma_n \\ g_s - (\gamma^* - \tilde{V})s_4 \in \Sigma_n \end{cases} \tag{14}$$

V **step:** Fix s_2, s_3, s_4 and find $\tilde{V}$ by solving the following problem by bisection on γ:

$$\gamma^* := \max \gamma$$
$$s.t. \begin{cases} \tilde{V} - l_1 \in \Sigma_n, \tilde{V}(0) = 0 \\ -[(\beta^* - \tilde{p})s_1 + (\tilde{V} - \gamma)] \in \Sigma_n \\ -[(\gamma - \tilde{V})s_2 + \dfrac{\partial \tilde{V}}{\partial x} f s_3 + l_2] \in \Sigma_n \\ g_s - (\gamma - \tilde{V})s_4 \in \Sigma_n \end{cases} \tag{15}$$

$V^i \leftarrow \frac{\tilde{V}}{\gamma^*}$

update p^i**:** $p^i \leftarrow \tilde{V}$.

until the largest difference (in absolute value) between the coefficients of p^i and p^{i-1} is less than a small given value.

Example (continued) Using the above algorithm, we compute the ROA of the SMIB system. The algorithm is initialized with the the Lyapunov function obtained from linearization of the system around the the equilibrium point of the system which is $(\delta_0, \omega_0) = (0.8342, 0)$. The trigonometric terms are approximated using Taylor expansion. The safe set is given by: $\mathcal{X}_s = \{-\frac{\pi}{2} \leq \delta - \delta_0 \leq \frac{\pi}{2}\}$. The degree of the Lyapunov function V as well as the degrees of multipliers s_1, s_2, s_3, s_4 are chosen as 4. We use the YALMIP toolbox [Löfberg, 2004] and SeDuMi [Sturm, 1999] to solve the optimization problems. The algorithm is stopped when largest difference (in absolute value) between the coefficients of p^i and p^{i-1} are less than 0.005. In this example the algorithm stops after 20 iterations. Figure 4 shows how the estimate of ROA is expanded iteratively. The ROA is finally given as: $\Omega_r = \{x : V(x) \leq 1\}$ with:

$$\begin{aligned} V(x) = &-0.1018x_1^3 + 0.5315x_1^2 + 0.1197x_1^4 + 0.0910x_1x_2 \\ &+0.1135x_2^2 + 0.1810x_1^2x_2 + 0.0575x_1x_2^2 + 0.0201x_2^3 + \\ &0.1054x_1^3x_2 + 0.0963x_1^2x_2^2 + 0.0192x_1x_2^3 + 0.0086x_2^4 \end{aligned} \tag{16}$$

where $x_1 = \delta - \delta_0$ and $x_2 = \omega - \omega_0$.

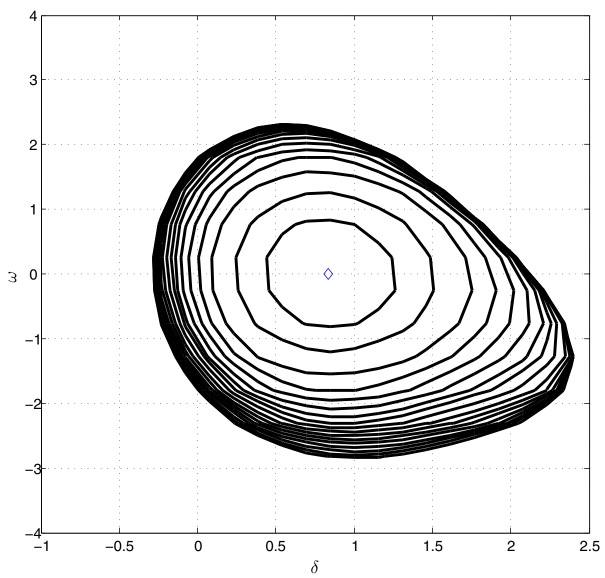

Fig. 4. Expansion of estimate of ROA of the SMIB system using the proposed algorithm(20 iterations)

5. COMPUTATION OF CRITICAL RECOVERY TIME

To compute the CRT, we use the concept of finite-time stability. A system is finite-time stable if the the states of the system remain in a given bounded set in a finite-time interval. We note that the concept is different from Lyapunov stability or asymptotic stability as in the latter concepts the behavior of the system over an infinite interval of time is studied, see [Amato et al., 2010] and references therein. This means that a system that is FTS might not be asymptotically stable and vice versa. Consider a nonlinear system given by:

$$\dot{x} = f(x), \tag{17}$$

where $x \in D \subseteq \mathbb{R}^n$ and $f : D \to \mathbb{R}^n$ is Lipschitz on D.

Definition 5. **Finite-time Stability** The nonlinear system $\dot{x} = f(x),\ t \in [0, T]$ is said to be finite-time stable (FTS) with respect to (D_1, D_2, T), where $D_1 \subset D_2 \subseteq D$ iff:

$$x(0) \in D_1 \Rightarrow x(t) \in D_2 \text{ for all } t \in [0, T], \tag{18}$$

where D_1 and D_2 are given sets.

It is assumed that D, D_1 and D_2 are given as semi-algebraic sets:

$$D = \{x \in \mathbb{R}^n : g_0(x) \geq 0\}, \tag{19}$$
$$D_1 = \{x \in \mathbb{R}^n : g_1(x) \geq 0\}, \tag{20}$$
$$D_2 = \{x \in \mathbb{R}^n : g_2(x) < 0\}. \tag{21}$$

Therefore, verifying whether the states of the system initiated form a set D_1, remain in the ROA of the post-fault system (Ω_r) during a given recovery time (RT), is equivalent to verifying if the faulty system is FTS w.r.t (D_1, D_2, RT) with $D_2 = \Omega_r$. To find the maximum allowable recovery time or the CRT, we must find the maximum of RT such that the system is FTS w.r.t the given sets. The general problem to be solved has the following form,

Problem 1. Given the nonlinear system (17), and the sets D_1, D_2, find the maximum T_M such that the nonlinear system (17) is FTS with respect to (D_1, D_2, T) for all $T < T_M$.

In the following, we explain how to use the results from [Tabatabaeipour and Blanke, 2014] to solve the above problem for a polynomial system. Compared to the results in the literature the method of [Tabatabaeipour and Blanke, 2014] is not restricted to systems with quadratic vector fields like [Amato et al., 2010] and can handle nonlinear systems with polynomial vector fields.

The following theorem we gives sufficient conditions to check FTS of a nonlinear system.

Theorem 3. [Tabatabaeipour and Blanke, 2014] The system (17) is FTS with respect to (D_1, D_2, T) if there exist a continuously differentiable function $B(x)$, a positive scalar α, and $0 < \epsilon < 1$ such that the following conditions are satisfied:

$$B(x) \leq \epsilon \ \forall x \in D_1 \tag{22}$$
$$B(x) \geq 1 \ \forall x \in D_2^c \tag{23}$$
$$\dot{B}(x) - \alpha B(x) \leq 0 \quad \forall x \in \bar{D_2} \tag{24}$$
$$\alpha < -\frac{1}{T} \ln \epsilon \tag{25}$$

Using the generalized S-procedure (Lemma 1), the conditions of theorem 3 are satisfied if the following SOS programming is feasible:

$$\begin{cases} -B(x) + \epsilon - s_1 g_1(x) \in \Sigma_n \\ (B(x) - 1) - s_2 g_2(x) \in \Sigma_n \\ -\dfrac{\partial B(x)}{\partial x} f(x) + \alpha B(x) + s_3 g_2(x) \in \Sigma_n \\ \alpha < -\dfrac{1}{T} \ln \epsilon. \end{cases} \tag{26}$$

where $B(x)$ is a polynomial, s_1, s_2 are SOS polynomials, α is a a positive scalar, and ϵ is positive such that $0 < \epsilon < 1$.

In words, $B(x)$ is a function that maps the trajectories of the system, $x(t)$, such that the corresponding value for trajectories that are initiated within D_1 is always bounded from above by $\epsilon e^{\alpha t}$ in time.

Due to the presence of the constraint $T \leq -\frac{1}{\alpha} \ln \epsilon$, the conditions given in (26) form a nonlinear SOS programming. To solve the problem 1, we need to calculate the maximum value of T such that given D_1, D_2, we can verify that all trajectories initiated in D_1 would remain in D_2. Maximizing T is actually the same as minimizing the rate of growth of the function B in time. Accordingly, to maximize T we search for the minimum α. Therefore, to solve the problem 1, the following optimization problem is solved:

$$\alpha^* := \min_{B \in \mathcal{R}_n, s_1, s_2, s_3 \in \Sigma_n} \alpha$$
$$s.t. \begin{cases} -B(x) + \epsilon - s_1 g_1(x) \in \Sigma_n \\ (B(x) - 1) - s_2 g_2(x) \in \Sigma_n \\ -\dfrac{\partial B(x)}{\partial x} f(x) + \alpha B(x) + s_3 g_2(x) \in \Sigma_n \end{cases} \tag{27}$$

Then, the system is FTS for all $T < T_M = -\frac{1}{\alpha^*} \ln \epsilon$. For a given ϵ, because of the presence of the term $\alpha B(x)$ the above problem is bilinear. Since α is scalar, the problem can be solved by bisection on α. Therefore, to find the minimum of α, a line search on ϵ is made where, for each ϵ, a bisection is performed on α , see algorithm 2

To apply the FTS framework to compute the CRT, we find T^* for the nonlinear faulty system $\dot{x} = f_f$ where D_1 is the set describing the initial state of the faulty system and D_2 describes the ROA of the post-fault system as computed by Algorithm 1. The procedure is:

- Given the set $\mathcal{X}_s$, f_r, compute the Ω_r of the post-fault system using Algorithm 1.
- Given f_f, and D_1, set $D_2 = \Omega_r$ and calculate T^* using Algorithm 2.

Algorithm 2

Given $\epsilon = [\epsilon_1, \cdots, \epsilon_m]$, $(0 < \epsilon_1, \epsilon_m < 1)$
for $i = 1$ to m **do**
  $\epsilon \leftarrow \epsilon_i$
  Solve the optimization problem (27) by bisection on α
  $T_i \leftarrow \frac{1}{\alpha^*} \ln \epsilon$
end for
$T^* \leftarrow \min T_i$

Example (Continued) The CRT for the SMIB system is now computed, using the ROA from above with $D_2 = \Omega_r$. The initial state of the system before fault is in the set $D_1 = \{x : (\delta - \delta_0)^2 + \omega^2 \leq 0.1\}$. Then, using the procedure given above and solving (27), $CRT = 0.671$. Extensive simulations of 5×10^4 trajectories initiated randomly in D_1 gives $CRT = 0.6742$, which shows the tightness of the bound. Figure 5 shows the result of 1000 of these trajectories where the fault is cleared at $t = 0.671$. All of the fault-on trajectories remain in Ω_r and then converge to the equilibrium point when the fault is cleared, which verifies our theoretical result.

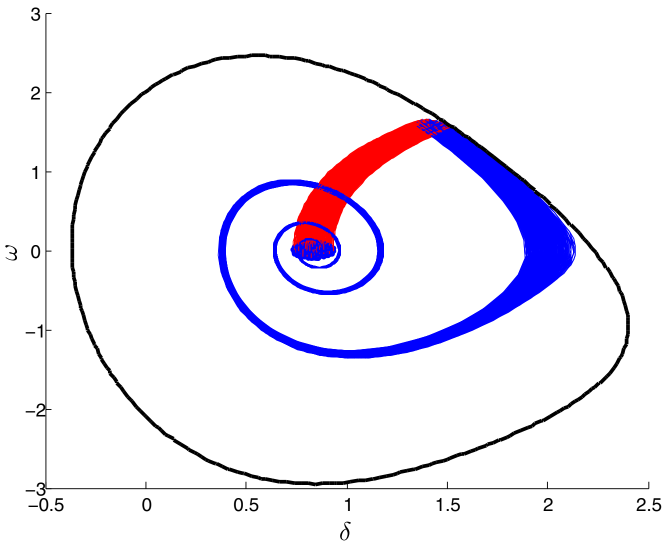

Fig. 5. 1000 trajectories of the SMIB example initiated randomly in D_1. The fault is cleared at $t = 0.6710$ s. Fault-on trajectories (red), post-fault trajectories (blue).

6. CONCLUSION

A new method for calculation of critical recovery time for safety critical nonlinear systems was presented. The method is based on computation of region of attraction of the post-fault system and finite-time stability analysis of the faulty system. Using sum of squares programming, we showed how to use finite-time stability analysis with region of attraction estimation to compute the critical recovery time. A simulation example showed tightness of the bound obtained from the method.

REFERENCES

F. Amato, C. Cosentino, and A. Merola. Sufficient conditions for finite-time stability and stabilization of nonlinear quadratic systems. *Automatic Control, IEEE Transactions on*, 55(2):430–434, 2010.

M. Blanke, M. Kinnaert, J. Lunze, and J. Staroswiecki. *Diagnosis and Fault-Tolerant Control.* Springer, 2nd edition, 2006.

Abhijit Chakraborty, Peter Seiler, and Gary J. Balas. Nonlinear region of attraction analysis for flight control verification and validation. *Control Engineering Practice*, 19(4):335 – 345, 2011. ISSN 0967-0661.

Graziano Chesi. *Domain of attraction: analysis and control via SOS programming.* Springer, 2011.

H. D. Chiang. *Direct Methods for Stability Analysis of Electric Power Systems.* Wiley, New York, NY, USA, 2011.

Zachary William Jarvis-Wloszek. *Lyapunov based analysis and controller synthesis for polynomial systems using sum-of-squares optimization.* PhD thesis, University of California, 2003.

Prabha Kundur. *Power System Stability and Control.* McGraw-Hill, 1994.

J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the IEEE Conference on Computer Aided Control Systems Design*, pages 284–289, 2004.

Pablo A Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical programming*, 96 (2):293–320, 2003.

Mania Pavella, Damien Ernst, and Daniel Ruiz-Vega. *Transient stability of power systems: a unified approach to assessment and control.* Kluwer Academic Publishers, 2000.

J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

S. M. Tabatabaeipour and M. Blanke. Compositional finite-time stability analysis of nonlinear systems. In *Accepted for presentation at the 2014 American Control Conference*, Portland, USA, June 2014.

Ufuk Topcu, Andrew K Packard, Peter Seiler, and Gary J Balas. Robust region-of-attraction estimation. *Automatic Control, IEEE Transactions on*, 55(1):137–142, 2010.

Yusheng Xue, Thierry Van Cutsem, and M Ribbens-Pavella. A simple direct method for fast transient stability assessment of large power systems. *Power Systems, IEEE Transactions on*, 3(2):400–412, 1988.

Y. M. Zhang and J. Jiang. Issues on integration of fault diagnosis and reconfigurable control in active fault-tolerant control systems. In *Proceedings of the IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processed*, pages 1513–1524, 2006.