Persistent Feasibility in Dual Switched Systems

1st Richard A Hall
Thomas Lord MEMS Department
Duke University
Durham, NC, USA
0000-0003-3512-6941

2nd Leila Bridgeman
Thomas Lord MEMS Department
Duke University
Durham, NC, USA
0000-0002-9367-7218

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

II. System Description

Consider a system with linear dynamics $(A(t), B(t))$ that change according to the external signals $\sigma_s,\ s \in \mathbb{Z}_{[1,S]}$ so that they can be written as $(A(\sigma_1(t),...,\sigma_S(t)), B(\sigma_1(t),...,\sigma_S(t)))$. Let them take the form

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix} = \begin{bmatrix} A^{11}_{\sigma_1(t)} & A^{12}_{\sigma_1(t)} \\ A^{21}_{\sigma_2(t)} & A^{22}_{\sigma_2(t)} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} B^1_{\sigma_1(t)} & 0 \\ 0 & B^2_{\sigma_2(t)} \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}. \tag{1}$$

The critical aspect of this system is the existence of two, independent switching signals, each with its own dwell time and successor constraints. Any system with independent switching sources are better suited for this framework. For example, consider a distributed system with local switching at each node. A switch at node 1 could be followed immediately by a switch at node 2 or after a great deal of time making minimum dwell times invalid. Average dwell time could, perhaps, be used instead but feasibility is difficult to establish under this constraint.

The second important aspect of the system is the block the systems are allowed to switch. This structure switches the dynamics of state x_i according to the respective switching signal, $\sigma_i(\cdot)$.

Our objective is to design safe-set collections that are indexed by the current switching signal states that will ensure persistent feasibility. Recall that every element in a safe-set collection must satisfy be within the one-step preset of all safe-set collection elements indexed by the possible successor switching signal states. The structure of the system under consideration makes this an especially challenging prospect for two reasons. First, the number of successor safe-sets grows exponentially two or more independent switching signals. Second, systems that take this form would tend to be larger than trivial examples. This suggests that set based techniques may not scale well. These concerns will be addressed by splitting the system in two and looking for safe-set collections for each seperatly. Once found, the collections can be merged into a large collection with all possible switching signal states represented.

[1]

Looking only at the first block row, the dynamics can be rewritten as

$$x_1(t+1) = A^{11}_{\sigma_1(t)} x_1(t) + B^1_{\sigma_1(t)} u_1(t) + A^{12}_{\sigma_1(t)} x_2(t). \tag{2}$$

This system has only a single switching signal explicitly appearing, $\sigma_1(t)$ and resembles a locally switched system with additive disturbances. However, the set that x_2 is drawn from is not obvious. The full state constraints could be used but this would lead to a conservative result. If the safe-set collection and switching signal state associated with x_2 were known, then the current safe-set could be used. This however, requires full interaction between the safe-set collections of x_1 and x_2 leading back to the centralized problem. Furthermore, this level of communication may be undesirable in distributed contexts. Alternatively, the convex hull of the safe-set collection union can be used. This only requires acquiring a single set and does not rely on the switching signal.

An import distinction arises at this point. In previous works with an external additive disturbance, the disturbance wasn't known until after it was applied. This meant that the input selected had to work for all disturbances. In the case of the original system, however, it is reasonable to assume a preview of $x_2(t)$ at time t. This implies that the safe-sets need not be within the robust preset of each of its successors but in the previewed robust preset defined below.

Definition 1 (Previewed Robust Preset)**.** *The k-step, previewed robust preset of a set $\mathcal{S}$ under the constrained dynamics $x^+ = Ax + Bu + w$, $x \in \mathcal{X}$, $u \in \mathcal{U}$, $w \in \mathcal{W}$ is given by*

$$\textit{Pre}^{\circledast,0}(\mathcal{S}) \triangleq \mathcal{S}, \tag{3}$$

$$\textit{Pre}^{\circledast,k}(\mathcal{S}) \triangleq \{x \in \mathcal{X} | \forall w \in \mathcal{W}, \exists u \in \mathcal{U} \ \textit{s.t.}\ Ax + Bu + w \in \textit{Pre}^{\circledast,k-1}(\mathcal{S})\}. \tag{4}$$

The previewed robust preset (PRP) is a superset or equal to the standard robust presets. Both can be found using the following set operations

$$\text{Pre}^1(\mathcal{S}) = (((\mathcal{S} \ominus \mathcal{W}) \oplus (-B \circ \mathcal{U})) \circ A) \ \cap \ \mathcal{X}, \tag{5}$$

$$\text{Pre}^{\circledast,1}(\mathcal{S}) = (((\mathcal{S} \oplus (-B \circ \mathcal{U})) \ominus \mathcal{W}) \circ A) \ \cap \ \mathcal{X}. \tag{6}$$

A circular dependency arose in the previous discussion – the safe-set collection for x_1 depend on the collection for x_2 which depends on the collection for x_1. This suggests an iterative algorithm with parallel elements. The basic steps are laid out below.

1) Assume trivial safe-sets
2) Share the convex hulls of the current safe-set collections.
3) Compute the resulting safe-set collection (large and invalid).
4) Share the convex hulls of the invalid safe-set collections.
5) Compute the resulting safe-set collection (conservative and valid?).
6) If no change, terminate. Else repeat from (2).

Every node can run steps (3) and (4) in parallel. This means that the algorithms scale very will with the number of nodes so long as their state dimensions remain relatively small.

There are still questions that need to be answered. When will the algorithm converge? Can it be terminated early at step (6)? How conservative is this approach compared with centralized methods when there is a common switching signal (can it be used to solve large dimension switched systems)? Can it be generalized to an arbitrary number of block-rows and switching signals? I will look into these questions in the coming weeks.

A. Notation

Let $\mathcal{S} = \{\{\{\mathcal{S}_{(i,j,k)}\}_{k\in\mathcal{I}_K(i,j)}\}_{j\in\mathcal{I}_J(i)}\}_{i\in\mathcal{I}_I}$, $\mathcal{S} \subseteq \mathbb{R}^n$ be a nested collection of subsets of the n-dimensional, real-valued numbers. The first and second level sub-collections are denoted as $\mathcal{S}_{(i)} = \{\{\mathcal{S}_{(i,j,k)}\}_{k\in\mathcal{I}_K(i,j)}\}_{j\in\mathcal{I}_J(i)}$ and $\mathcal{S}_{(i,j)} = \{\mathcal{S}_{(i,j,k)}\}_{k\in\mathcal{I}_K(i,j)}$. Operations preformed between nested collections require the collections be of equal size and are preformed elementwise. For example, $\mathcal{S}_{(i)}$ equals $\tilde{\mathcal{S}}_{(i)}$ if they are the same size and every element in both are equivalent. Operations preformed between a nested collection and a single element behave as if the single element where an appropriate sized collection with elements equal to the single element. For example, setting $\mathcal{S} = \underline{0}$ sets every element of $\mathcal{S}$ to $\underline{0}$. Finally, union operations are preformed on every element of the nested collection. For example, $\cup\mathcal{S}_{(i)}$ is the union of all the elements of $\mathcal{S}_{(i)}$.

III. Algorithm Design

We begin with an algorithm to find the safe-sets of a single agent, $\mathcal{M}^\alpha$, assuming the the other agents' safe-sets are fixed. Let $\mathcal{W}_{\tilde{\alpha}}$, $\tilde{\alpha} \in \mathbb{Z}_{[1,C_a]} \setminus \alpha$ be the convex hull of the union of the safe-sets for each of the other agents and define

$$\mathcal{W}^\alpha_\mu = \bigoplus_{\tilde{\alpha}\in\mathbb{Z}_{[1,C_a]}\setminus\alpha} A^{\alpha,\tilde{\alpha}}_\mu \mathcal{W}_{\tilde{\alpha}}. \tag{7}$$

This represents the set of additive errors agent $\mathcal{M}^\alpha$ can experience while in mode μ. Denote the collection of these sets as $\mathfrak{W}^\alpha = \{\mathcal{W}^\alpha_\mu\}_{\mu=1}^{C^\alpha_m}$. With these definitions, the following algorithm is introduced.

Algorithm 1 Nodal safe-sets with previewed disturbances

procedure $\textsc{NodalSafeSets}(\mathcal{M}^\alpha,\ \mathcal{W})$
  $k \leftarrow 0$
  $\Omega^k_{(\mu,\tau)} \leftarrow \mathcal{X}_\mu$ for all $\mu \in \mathbb{Z}_{[1,C^\alpha_m]},\ \tau \in \mathbb{Z}_{[1,\overline{\tau}_\mu]}$.
  repeat
    $k \leftarrow k + 1$
    for $\mu \in \mathbb{Z}_{[1,C^\alpha_m]}$ **do**
      for $\tau \in \mathbb{Z}_{[1,\overline{\tau}_\mu]}$ **do**
        $\Omega^k_{(\mu,\tau)} \leftarrow \Omega^{k-1}_{(\mu,\tau)}$
        for $(\tilde{\mu},\tilde{\tau}) \in \Lambda((\mu,\tau))$ **do**
          $\Omega^k_{(\mu,\tau)} \leftarrow \Omega^k_{(\mu,\tau)} \cap \mathrm{Pre}^{\circledast,1}_{\tilde{\mu}}\Big(\Omega^{k-1}_{(\tilde{\mu},\tilde{\tau})}, \mathcal{W}^\alpha_\mu\Big)$
        end for
      end for
    end for
  until $\Omega^k_{(\mu,\tau)} = \Omega^{k-1}_{(\mu,\tau)}\ \forall\ \mu \in \mathbb{Z}_{[1,C^\alpha_m]},\ \tau \in \mathbb{Z}_{[1,\overline{\tau}_\mu]}$
  $\mathcal{S}_{(\mu,\tau)} \leftarrow \Omega^k_{(\mu,\tau)}$ for all $\mu \in \mathbb{Z}_{[1,C^\alpha_m]},\ \tau \in \mathbb{Z}_{[1,\overline{\tau}_\mu]}$.
end procedure

Algorithm 2 Distributed safe-set collection

procedure $\textsc{SystemSafeSets}(\mathcal{M})$
  $\Omega^0 \leftarrow \{\{\{\underline{0}\}_{\tau\in\mathcal{I}_\tau(\alpha,\mu)}\}_{\mu\in\mathcal{I}_\mu(\alpha)}\}_{\alpha\in\mathcal{I}_\alpha}$
  $k \leftarrow 0$
  repeat
    $\Phi^k \leftarrow \{\textsc{ConHull}(\bigcup\Omega^k_{(\alpha)})\}_{\alpha\in\mathcal{I}_\alpha}$
    for $\alpha \in \mathcal{I}_\alpha$ **do**
      $\Omega^{k+1}_{(\alpha)} \leftarrow \textsc{NodeSafeSets}(\mathcal{M}^\alpha, \Phi^k)$
    end for
  until $\Omega^k_{(\alpha)} = \Omega^{k-1}_{(\alpha)}\ \forall\ \alpha \in \mathcal{I}_\alpha$
  $\mathcal{S} \leftarrow \Omega^k$
end procedure

References

[1] R. M. Schaich, M. A. Müller, and F. Allgöwer, "A distributed model predictive control scheme for networks with communication failure," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 19, pp. 12 004–12 009, 2014, ISSN: 14746670. DOI: 10.3182/20140824-6-za-1003.01507.