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

Hybrid systems are a broad and important class of systems with both discrete and continuous (or discrete approximations of continuous) dynamics. A subclass of these are systems whose discrete dynamics are purely time dependent and unknown to the system. Such discrete, external dynamics can be viewed as an external signal switching the system dynamics at discrete, unforeseen times.

Many systems experience changes to its dynamics beyond the input of the controller. For example, user input and component failures can both cause the system to suddenly change. If the system is constrained, then special care should be taken to make the system robust to these switches so that constraints are not violated.

If the system were allowed to be switched arbitrarily, then satisfying the constraints would be restrictive indeed. This would require the controller to always keep the state within a control invariant set common to every modes. If the modes are not extremely similar, this may be a very difficult requirement meet.

Instead of arbitrary switching, dwell time and successor constraints are often imposed on the switching signal. Minimum dwell time constraints give the system time to recover after a switch and prepare for the next one while maximum dwell time constraints ensure that the system will not dwell in "poorly" behaved modes for too long. These constraints can either be generated by the physical implementation and used by the controller or generated by the controller and enforced in the physical implementation.

Dwell time and successor constraints have been used in previous works to create time-varying control invariant sets that are robust to all possible switching signals **CITE**. An issue with these previous methods is that they can be very computationally expensive and suffer greatly from the curse of dimensionality. This precludes their use in large systems.

A further shortcoming of the current literature is that only a single switching signal is considered. It is not difficult to imagine cases where there are multiple sources of switching that are independent from each other. Consider a distributed system where the individual agents are switching according to local switching signals. This independence makes a common switching signal with shared dwell time and successor constraints much more difficult to motivate. Seeking to enumerate the possible states of even just two switching signals and define their dwell time and successor constraints leads quickly back to arbitrary switching but with many more modes.

Symbol	Description
C_a	Number of switching signals/agents
$\mathcal{I}^a$	Set of agent indicies, $\mathbb{Z}_{[1,C_a]}$
C_m^α	Number of modes in agent $\alpha \in \mathcal{I}^a$
$\mathcal{I}_\alpha^m$	Set of mode indicies in agent α, $\mathbb{Z}_{[1,C_m^\alpha]}$

This work develops an algorithm with applications to both reduce the computational expense of previous algorithms and analyze of systems with multiple sources of switching. The algorithm's most computationally expensive steps are parallelizable, greatly improving its scalability. In the next section, notation and concepts are introduced that will be used throughout the reminder of the paper. Next, the general form of the system is described and the associated challenges are discussed further. Finally, several algorithms are introduced as key contributions of this work and their theoretical properties and implementation are explored. These are then applied to a numerical example that demonstrates the effectiveness of the results.

II. Preliminaries

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 $\mathcal{S}_{(i)}$.

B. Set Operations

Set operations provide tools to analyze how a system can evolve under allowable inputs. Given a disturbed linear system with state and input constraints, $\mathcal{M}_\mu \triangleq \{A_\mu, B_\mu, E_\mu, \mathcal{X}_\mu, \mathcal{U}_\mu\}$ where $x(t+1) = A_\mu x(t) + B_\mu u(t) + E_\mu w(t)$, the following important set operations are introduced.

Definition 1 (Robust Preset). *The k-step, robust preset of a set $\mathcal{S}$ under the constrained dynamics $\mathcal{M}_\mu$ and disturbance $w \in \mathcal{W}$ is given by*

$$\mathit{Pre}_\mu^0(\mathcal{S}, \mathcal{W}) \triangleq \mathcal{S}, \tag{1}$$

$$\begin{aligned}\mathit{Pre}_\mu^k(\mathcal{S}, \mathcal{W}) \triangleq \{&x \in \mathcal{X} \mid \exists u \in \mathcal{U} \text{ s.t. } \forall w \in \mathcal{W},\\ &A_\mu x + B_\mu u + E_\mu w \in \mathit{Pre}_\mu^{k-1}(\mathcal{S}, \mathcal{W})\}.\end{aligned} \tag{2}$$

Definition 2 (Previewed Robust Preset). *The k-step, previewed robust preset of a set $\mathcal{S}$ under the constrained dynamics $\mathcal{M}_\mu$ and disturbance $w \in \mathcal{W}$ is given by*

$$\mathit{Pre}_\mu^{\circledast,0}(\mathcal{S}, \mathcal{W}) \triangleq \mathcal{S}, \tag{3}$$

$$\begin{aligned}\mathit{Pre}_\mu^{\circledast,k}(\mathcal{S}, \mathcal{W}) \triangleq \{&x \in \mathcal{X} | \forall w \in \mathcal{W}, \exists u \in \mathcal{U} \text{ s.t.}\\ &A_\mu x + B_\mu u + E_\mu w \in \mathit{Pre}_\mu^{\circledast,k-1}(\mathcal{S}, \mathcal{W})\}.\end{aligned} \tag{4}$$

The previewed robust preset is a superset or equal to the standard robust presets. They can be found using basic set operation such as the Minkowski sum and difference as shown below [1].

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

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

C. Feasibility Analysis

In constrained systems, it is critically important that the controller can satisfy the state constraints using only feasibly inputs. If a feasible input exists such that the resulting state is also feasible, the system is feasible. If feasibility at the current time implies feasibility at all future times, then the system is persistently feasible.

One way to ensure persistent feasibility is by constraining the system to be within a control invariant set that is a subset of the state constraints. Since the definition of control invariance implies that a feasible input exists that will keep the system in the set if it starts in the set, then at least this single input, and possibly others, can serve as a feasible input. This will continue for all time establishing persistent feasibility.

Relying on constant, control invariant sets become difficult in externally switched systems, however. The set must be common to all system modes are persistent feasibility is lost. This concern can be addressed using time varying, control invariant sets that take advantage of constraints on the switching signal. For example, in [2], [3], time varying, control invariant sets where developed that force the system to move during the minimum dwell time to a region that will be safe for any successor mode.

The ideas presented in the previous literature can be described using the concepts of safe-set collections. These are collections of sets, indexed by some time varying signal, that serve as the active state constraints of the system. To establish persistent feasibility, they require that every set is witching the 1-step preset of all possible successor sets. This is presented formally in the following definition.

Definition 3 (Safe-set collection). *Given the time varying dynamic system $\mathcal{M}(t) = \{f(x(t), u(t), t), \mathcal{X}(t), \mathcal{U}(t)\}$, a collection of sets, $\mathcal{S} = \{\mathcal{S}_{i_1}, ..., \mathcal{S}_{i_N}\}$ is a safe-set collection if there exists a mapping between time and the current mode index, $g : \mathbb{Z}_{\geq 0} \to \{i_1, ..., i_N\}$ and another mapping between the current time and possible indicies at the next step, $g^+ : \mathbb{Z}_{\geq 0} \to \mathcal{P}(\{i_1, ..., i_N\})$ such that*

$$\mathcal{S}_{g(t-1)} \subseteq \mathit{Pre}^1_{\mathcal{M}(t)}\left(\mathcal{S}_{g(t)}\right) \ \forall \ g(t) \in g^+(t-1)$$

III. System Description

Consider consider the collection of external signals

$$\sigma_\alpha : \mathbb{Z}_{\geq 0} \to \mathcal{I}^m_\alpha \triangleq \mathbb{Z}_{[1, C^\alpha_m]}, \ \alpha \in \mathcal{I}^a \triangleq \mathbb{Z}_{[1, C_a]}.$$

Then the dynamics of the system being studied take the following form

$$A(t) = \begin{bmatrix} A^{11}_{\sigma_1(t)} & A^{12}_{\sigma_1(t)} & \cdots & A^{1C_a}_{\sigma_1(t)}, \\ A^{21}_{\sigma_2(t)} & A^{22}_{\sigma_2(t)} & \cdots & A^{2C_a}_{\sigma_2(t)}, \\ \vdots & \vdots & \ddots & \vdots \\ A^{C_a 1}_{\sigma_{C_a}(t)} & A^{C_a 2}_{\sigma_{C_a}(t)} & \cdots & A^{C_a C_a}_{\sigma_{C_a}(t)} \end{bmatrix}, \tag{7}$$

$$B(t) = \begin{bmatrix} B^1_{\sigma_1(t)} & 0 & \cdots & 0, \\ 0 & B^2_{\sigma_2(t)} & \cdots & 0, \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B^{C_a}_{\sigma_{C_a}(t)} \end{bmatrix}. \tag{8}$$

This structure is motivated by dynamically coupled distributed systems. Two features of these dynamics equations are immediately apparent. First, note how each block-row is governed by a single switching signal. This makes intuitive sense because local switching signals are more likely to effect how neighboring states impact the local agent rather then how local states will effect neighboring agents. Second, any dwell time or successor restrictions of the individual switching signals do not transfer to the well to the full system. Any attempt to apply previous analysis techniques to this system will be futile.

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.

[4]

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

$$x_1(t+1) = A^{11}_{\sigma_1(t)} x_1(t) + B^{1}_{\sigma_1(t)} u_1(t) + A^{12}_{\sigma_1(t)} x_2(t). \quad (9)$$

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 4 (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*

$$\mathit{Pre}^{\text{👁},0}(\mathcal{S}) \triangleq \mathcal{S}, \quad (10)$$

$$\mathit{Pre}^{\text{👁},k}(\mathcal{S}) \triangleq \{x \in \mathcal{X} | \forall w \in \mathcal{W}, \exists u \in \mathcal{U} \ \ \textit{s.t. } Ax + Bu + w \in \mathit{Pre}^{\text{👁},k-1}(\mathcal{S})\}. \quad (11)$$

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

$$\mathrm{Pre}^1(\mathcal{S}) = (((\mathcal{S} \ominus \mathcal{W}) \oplus (-B \circ \mathcal{U})) \circ A) \ \cap \ \mathcal{X}, \quad (12)$$

$$\mathrm{Pre}^{\text{👁},1}(\mathcal{S}) = (((\mathcal{S} \oplus (-B \circ \mathcal{U})) \ominus \mathcal{W}) \circ A) \ \cap \ \mathcal{X}. \quad (13)$$

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).

Algorithm 1 Nodal safe-sets with previewed disturbances

procedure 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**
          $\Omega^k_{(\mu,\tau)} \leftarrow \Omega^k_{(\mu,\tau)} \cap \mathrm{Pre}^{\text{👁},1}_{\tilde{\mu}}\Big(\Omega^{k-1}_{(\tilde{\mu},\tilde{\tau})}, \mathcal{W}^\alpha_\mu\Big)$
        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

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.

IV. 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

$$\mathcal{W}^\alpha_\mu = \bigoplus_{\tilde{\alpha} \in \mathbb{Z}_{[1,C_a]} \setminus \alpha} A^{\alpha,\tilde{\alpha}}_\mu \mathcal{W}_{\tilde{\alpha}}. \quad (14)$$

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

Lemma 1. *Given any agent, $\mathcal{M}_{(\alpha)}$, and constraint set $\mathcal{W}$,* AgentSafeSets$(\mathcal{M}_{(\alpha)}, \mathcal{W})$ *returns the maximal safe-set collection.*

Proof. Follows the logic of [2, Theorem 2] but with previewed preset operations. $\square$

Algorithm 2 Distributed safe-set collection

procedure $\textsc{SystemSafeSets}(\mathcal{M})$
  $\Omega^0 \leftarrow \{\{\{\underline{0}\}_{\tau\in\mathcal{I}_\tau(\alpha,\mu)}\}_{\mu\in\mathcal{I}_\mu(\alpha)}\}_{\alpha\in\mathcal{I}_\alpha}$
  $\Phi^0 \leftarrow \{\underline{0}\}_{\alpha\in\mathcal{I}_\alpha}$
  $k \leftarrow 0$
  repeat
    parfor $\alpha \in \mathcal{I}_\alpha$ **do**
      $\Omega^{k+0.5}_{(\alpha)} \leftarrow \textsc{NodeSafeSets}(\mathcal{M}^\alpha, \Phi^k)$
      $\Phi^{k+0.5}_{(\alpha)} \leftarrow \textsc{ConHull}(\bigcup \Omega^{k+0.5}_{(\alpha)})$
    end parfor
    parfor $\alpha \in \mathcal{I}_\alpha$ **do**
      $\Omega^{k+1}_{(\alpha)} \leftarrow \textsc{NodeSafeSets}(\mathcal{M}^\alpha, \Phi^{k+0.5})$
      $\Phi^{k+1}_{(\alpha)} \leftarrow \textsc{ConHull}(\bigcup \Omega^{k+1}_{(\alpha)})$
    end parfor
    $k \leftarrow k+1$
  until $\Omega^k_{(\alpha)} = \Omega^{k-1}_{(\alpha)} \ \forall\ \alpha \in \mathcal{I}_\alpha$
  $\mathcal{S} \leftarrow \Omega^k$
end procedure

Lemma 2. *Given an agent, $\mathcal{M}_{(\alpha)}$, and the sets $\hat{\mathcal{W}} \subseteq \tilde{\mathcal{W}}$, the relationship*

$$\textsc{AgentSafeSets}(\mathcal{M}_{(\alpha)}, \hat{\mathcal{W}}) \supseteq \textsc{AgentSafeSets}(\mathcal{M}_{(\alpha)}, \tilde{\mathcal{W}})$$

holds element-wise.

Proof. □

Lemma 3. *The algorithm* $\textsc{SystemSafeSets}(\mathcal{M})$ *produces a valid safe-set collection for every $k \in \mathbb{Z}_{\geq 0}$.*

Proof. This holds trivially for Ω^0. The following induction steps complete the proof.

1) Assume Ω^k is valid under the disturbance constraints Φ^k.
2) By Lemma 1, $\Omega^k \subseteq \Omega^{k+0.5}$.
3) By Lemma 2, $\Phi^k \subseteq \Phi^{k+0.5}$ implies that $\Omega^{k+1} \subseteq \Omega^{k+0.5}$.
4) Since Ω^{k+1} is valid for $\Phi^{k+0.5}$, it will also be valid for $\Phi^{k+1} \subseteq \Phi^{k+0.5}$.

□

REFERENCES

[1] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017. DOI: 10.1017/9781139061759. [Online]. Available: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Predictive+Control+for+linear+and+hybrid+systems#0.

[2] C. Danielson, L. J. Bridgeman, and S. Di Cairano, “Necessary and sufficient conditions for constraint satisfaction in switched systems using switch-robust control invariant sets,” *International Journal of Robust and Nonlinear Control*, vol. 29, no. 9, pp. 2589–2602, 2019, ISSN: 10991239. DOI: 10.1002/rnc.4509.

[3] E. D. Santis, M. D. Di Benedetto, and L. Berardi, “Computation of Maximal Safe Sets for Switching Systems,” *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, vol. 49, no. 2, pp. 184–195, 2004. DOI: 10.1109/TAC.2003.822860.

[4] 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.