Bounded Model Checking of MPL Systems via Predicate Abstractions

FORMATS 2019

Muhammad Syifa'ul Mufid[1,3]
Dieky Adzkiya[2]
Alessandro Abate[1]

[1]Department of Computer Science, University of Oxford, UK
[2]Department of Mathematics, ITS Surabaya, Indonesia
[3]Indonesia Endowment Fund for Education (LPDP)

Amsterdam, 27 August 2019

Outline

- Max-Plus-Linear (MPL) systems and time difference
- Predicate abstractions of MPL systems
- Bounded Model Checking of MPL systems
- Conclusion

Max-Plus-Linear Systems

- Based on max-plus algebra $(\mathbb{R}_{\max}, \oplus, \otimes)$ where $\mathbb{R}_{\max} := \mathbb{R} \cup \{-\infty\}$. For all $a, b \in \mathbb{R}_{\max}$

$$a \oplus b := \max\{a, b\}, \;\; a \otimes b := a + b$$

- The operations can be applied to matrices. For $A \in \mathbb{R}_{\max}^{n \times n}$, $A^{\otimes r}$ to denote $A \otimes \ldots \otimes A$ (r times)
- Defined as $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k),$ where $A \in \mathbb{R}_{\max}^{n \times n}$ and $\mathbf{x}(k) \in \mathbb{R}^n$.
- Applications: transportations, scheduling, biological systems...

Max-Plus-Linear Systems

- The precedence graph of A, denoted by $\mathscr{G}(A)$, is a weighted directed graph with vertices $1, 2 \ldots, n$ and an edge from j to i with weight $A(i,j)$ for each $A(i,j) \neq -\infty$
- The average weight of path $p = i_0 i_1 \ldots i_k$ in $\mathscr{G}(A)$ is equal to

$$\frac{A(i_1, i_0) + \ldots + A(i_k, i_{k-1})}{k}$$

- A matrix $A \in \mathbb{R}_{\max}^{n \times n}$ is called irreducible if $\mathscr{G}(A)$ is strongly connected
- If A is irreducible then there is only one eigenvalue
  λ = the maximum average weight of circuits

Max-Plus-Linear Systems

Transient Condition*

For an irreducible matrix $A \in \mathbb{R}_{\max}^{n \times n}$ and its corresponding eigenvalue λ, there exist $k_0, c \in \mathbb{N}$ such that $A^{\otimes k+c} = \lambda c \otimes A^{\otimes k}$ for all $k \geq k_0$. The smallest such k_0 and c are called the transient and the cyclicity of A, respectively.

* Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and Linearity: An Algebra for Discrete Event Systems. Wiley, Chichester (1992)

Max-Plus-Linear Systems

Transient Condition*

For an irreducible matrix $A \in \mathbb{R}_{\max}^{n\times n}$ and its corresponding eigenvalue λ, there exist $k_0, c \in \mathbb{N}$ such that $A^{\otimes k+c} = \lambda c \otimes A^{\otimes k}$ for all $k \geq k_0$. The smallest such k_0 and c are called the transient and the cyclicity of A, respectively.

Given $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$ and an initial $\mathbf{x}(0)$

$$\mathbf{x}(0),\ \ \mathbf{x}(1),\ \ \mathbf{x}(2),\ \ \ldots$$

is eventually periodic in max-plus algebraic sense. For all $k \geq k_0$,

$$\mathbf{x}(k+c) = \lambda c \otimes \mathbf{x}(k)$$

* Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and Linearity: An Algebra for Discrete Event Systems. Wiley, Chichester (1992)

Max-Plus-Linear Systems

Transient Condition*

For an irreducible matrix $A \in \mathbb{R}_{\max}^{n\times n}$ and its corresponding eigenvalue λ, there exist $k_0, c \in \mathbb{N}$ such that $A^{\otimes k+c} = \lambda c \otimes A^{\otimes k}$ for all $k \geq k_0$. The smallest such k_0 and c are called the transient and the cyclicity of A, respectively.

Given $\mathbf{x}(k+1) = A \otimes \mathbf{x}(k)$ and an initial $\mathbf{x}(0)$

$$\mathbf{x}(0),\ \ \mathbf{x}(1),\ \ \mathbf{x}(2),\ \ \ldots$$

is eventually periodic in max-plus algebraic sense. For all $k \geq k_0$,

$$\mathbf{x}(k+c) = \lambda c \otimes \mathbf{x}(k)$$

$$\begin{bmatrix} x_1(k+c) \\ \vdots \\ x_n(k+c) \end{bmatrix} = \begin{bmatrix} \lambda c \\ \vdots \\ \lambda c \end{bmatrix} + \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix}$$

* Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and Linearity: An Algebra for Discrete Event Systems. Wiley, Chichester (1992)

Max-Plus-Linear Systems

- Time differences

$$x_i(k) - x_j(k) \quad \text{or} \quad x_i(k+1) - x_i(k)$$

Max-Plus-Linear Systems

- Time differences

$$x_i - x_j \quad \text{or} \quad x_i' - x_i$$

Max-Plus-Linear Systems

- Time differences

$$x_i - x_j \quad \text{or} \quad x_i' - x_i$$

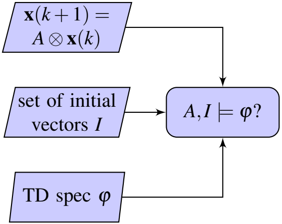
- Time difference propositions

$$x_i{}' - x_i \sim \alpha$$

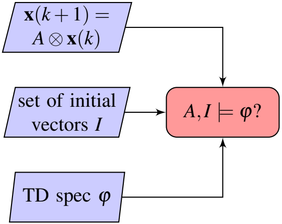
$\sim \in \{<, \leq, \geq, >\}$ and $\alpha \in \mathbb{R}$

- Time difference specifications
  LTL formula over time difference propositions
  - $\bigcirc(x_i{}' - x_i \geq 5) \equiv x_i(2) - x_i(1) \geq 5$
  - $\Diamond\Box(x_i{}' - x_i \leq 8) \equiv \exists k \geq 0$ s.t. $\forall m \geq k \;\; x_i(m+1) - x_i(m) \leq 8$

Max-Plus-Linear Systems



Max-Plus-Linear Systems

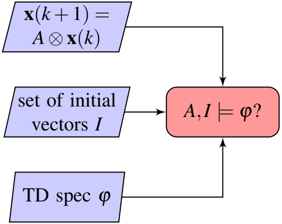


$I = \mathbb{R}^n$

For all $\mathbf{x}(0) \in I$

$$\mathbf{x}(0),\ \mathbf{x}(1),\ \mathbf{x}(2),\ldots \text{ satisfies } \varphi$$

Max-Plus-Linear Systems



$I = \mathbb{R}^n$

For all $\mathbf{x}(0) \in I$

$$\mathbf{x}(0),\ \ \mathbf{x}(1),\ \ \mathbf{x}(2), \ldots\ \ \text{satisfies}\ \ \varphi$$

- Infinite and continuous state space
- The primed variables
- This problem is undecidable
- Solve the problem by applying predicate abstractions (PA) and bounded model checking (BMC)

PA of MPL Systems

- Abstractions: techniques to generate a finite and smaller system from a large or even infinite-space system

$$\hat{S} \models \varphi \to S \models \varphi$$

PA of MPL Systems

- Abstractions: techniques to generate a finite and smaller system from a large or even infinite-space system

$$\hat{S} \models \varphi \rightarrow S \models \varphi$$

- MPL systems $\rightarrow$ Piece-Wise Affine (PWA) System
  Partitioning state space into several convex domains (PWA regions).
  Each region has corresponding affine dynamics
- Given $A \in \mathbb{R}_{\max}^{n \times n}$, the region w.r.t. $\mathbf{g} \in \{1, \ldots, n\}^n$ is

$$R_{\mathbf{g}} = \bigcap_{i=1}^{n} \bigcap_{j=1}^{n} \left\{ \mathbf{x} \in \mathbb{R}^n | x_{g_i} - x_j \geq A(i,j) - A(i,g_i) \right\}$$

  $R_{\mathbf{g}}$ is a Difference-Bound Matrix (DBM)
- If $R_{\mathbf{g}} \neq \emptyset$ then the corresponding affine dynamics

$$x_i{}' = x_{g_i} + A(i,g_i), \;\; i = 1, \ldots, n$$

PA of MPL Systems

- Predicate abstraction: using a set of predicates

$$P = \{p_1, \ldots, p_k\}$$

- Predicates are identified from the (concrete) system and specifications
- Abstract states are generated from all Boolean assignments w.r.t. P

$$|\hat{S}| \leq 2^k$$

- Predicates also serve as atomic propositions*

* Clarke, E., Grumberg, O., Talupur, M., Wang, D.: Making predicate abstraction efficient. In: Hunt, W.A., Somenzi, F. (eds.) CAV 2003. LNCS, vol. 2725, pp. 126-140. Springer, Heidelberg (2003).

PA of MPL Systems

- Predicates from MPL systems?

PA of MPL Systems

- Predicates from MPL systems?

$$R_{\mathbf{g}} = \bigcap_{i=1}^{n} \bigcap_{j=1}^{n} \left\{ \mathbf{x} \in \mathbb{R}^n | x_{g_i} - x_j \geq A(i,j) - A(i,g_i) \right\}$$

PA of MPL Systems

- Predicates from MPL systems?

$$R_{\mathbf{g}} = \bigcap_{i=1}^{n} \bigcap_{j=1}^{n} \left\{ \mathbf{x} \in \mathbb{R}^n | x_{g_i} - x_j \geq A(i,j) - A(i,g_i) \right\}$$

Predicates are in the form of

$$x_k - x_j \sim A(i,j) - A(i,k), \;\; i = 1, \ldots, n, \;\; k < j \in \texttt{fin}_i$$

where $\texttt{fin}_i = \{j | A(i,j) \neq -\infty\}$

WLOG $\sim \in \{>, \geq\}$

PA of MPL Systems

- Predicates from specifications?

$$x_i' - x_i \sim \alpha$$

$$\max_{j \in \mathtt{fin}_i} \{x_j + A(i,j)\} - x_i \sim \alpha$$

PA of MPL Systems

- Predicates from specifications?

$$x_i{}' - x_i \sim \alpha$$
$$\max_{j \in \mathtt{fin}_i} \{x_j + A(i,j)\} - x_i \sim \alpha$$

  Predicates are in the form of $x_j - x_i \sim \alpha - A(i,j)$ for all $j \in \mathtt{fin}_i$

- If $i \in \mathtt{fin}_i$ i.e. $A(i,i) \neq -\infty$, we can ignore $x_i - x_i \sim \alpha - A(i,i)$

PA of MPL Systems

Example:

$$\mathbf{x}' = A \otimes \mathbf{x} = \begin{bmatrix} 2 & 5 \\ 3 & 3 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \;\text{ and }\; t \equiv x_1' - x_1 \leq 5$$

PA of MPL Systems

Example:

$$\mathbf{x}' = A \otimes \mathbf{x} = \begin{bmatrix} 2 & 5 \\ 3 & 3 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \text{ and } t \equiv x_1' - x_1 \leq 5$$

Predicates from MPL system

$$x_k - x_j \sim A(i,j) - A(i,k)$$

Predicates from TD proposition

$$x_j - x_i \sim \alpha - A(i,j)$$

PA of MPL Systems

Example:

$$\mathbf{x}' = A \otimes \mathbf{x} = \begin{bmatrix} 2 & 5 \\ 3 & 3 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \text{ and } t \equiv x_1' - x_1 \leq 5$$

Predicates from MPL system

$$x_k - x_j \sim A(i,j) - A(i,k)$$

$$x_1 - x_2 \geq 3$$

$$x_1 - x_2 \geq 0$$

Predicates from TD proposition

$$x_j - x_i \sim \alpha - A(i,j)$$

$$x_2 - x_1 \leq 0$$

PA of MPL Systems

Example:

$$\mathbf{x}' = A \otimes \mathbf{x} = \begin{bmatrix} 2 & 5 \\ 3 & 3 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \text{ and } t \equiv x_1' - x_1 \leq 5$$

Predicates from MPL system

$$x_k - x_j \sim A(i,j) - A(i,k)$$

$$x_1 - x_2 \geq 3$$

$$x_1 - x_2 \geq 0$$

Predicates from TD proposition

$$x_j - x_i \sim \alpha - A(i,j)$$

$$x_2 - x_1 \leq 0$$

There are two predicates, $P = P_{mat} \cup P_{time} = \{p_1, p_2\}$ where

$$p_1 \equiv x_1 - x_2 \geq 3$$

$$p_2 \equiv x_1 - x_2 \geq 0$$

PA of MPL Systems

Example:
There are four possible Boolean assignments

$$
\begin{aligned}
\neg p_1 \neg p_2 &\equiv (x_1 - x_2 < 3) \wedge (x_1 - x_2 < 0) \\
\neg p_1 p_2 &\equiv (x_1 - x_2 < 3) \wedge (x_1 - x_2 \geq 0) \\
p_1 \neg p_2 &\equiv (x_1 - x_2 \geq 3) \wedge (x_1 - x_2 < 0) \quad \text{empty set} \\
p_1 p_2 &\equiv (x_1 - x_2 \geq 3) \wedge (x_1 - x_2 \geq 0)
\end{aligned}
$$

but only three abstracts states:

$$
\begin{array}{ll}
\hat{s}_0 \equiv \neg p_1 \neg p_2 & \mathtt{DBM}(\hat{s}_0) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 < 0\} \\
\hat{s}_1 \equiv \neg p_1 p_2 & \mathtt{DBM}(\hat{s}_1) = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 \leq x_1 - x_2 < 3\} \\
\hat{s}_2 \equiv p_1 p_2 & \mathtt{DBM}(\hat{s}_2) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \geq 3\}
\end{array}
$$

PA of MPL Systems

Example:
There are four possible Boolean assignments

$$
\begin{aligned}
\neg p_1 \neg p_2 &\equiv (x_1 - x_2 < 3) \wedge (x_1 - x_2 < 0) \\
\neg p_1 p_2 &\equiv (x_1 - x_2 < 3) \wedge (x_1 - x_2 \geq 0) \\
p_1 \neg p_2 &\equiv (x_1 - x_2 \geq 3) \wedge (x_1 - x_2 < 0) \quad \text{empty set} \\
p_1 p_2 &\equiv (x_1 - x_2 \geq 3) \wedge (x_1 - x_2 \geq 0)
\end{aligned}
$$

but only three abstracts states:

$$
\begin{aligned}
&\hat{s}_0 \equiv \neg p_1 \neg p_2 && \texttt{DBM}(\hat{s}_0) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 < 0\} \\
&\hat{s}_1 \equiv \neg p_1 p_2 && \texttt{DBM}(\hat{s}_1) = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 \leq x_1 - x_2 < 3\} \\
&\hat{s}_2 \equiv p_1 p_2 && \texttt{DBM}(\hat{s}_2) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \geq 3\}
\end{aligned}
$$

Next step: generate the abstract transition system

PA of MPL Systems

- Concrete transition systems

Definition (Trans. sys. associated with MPL system)

A transition system for an MPL system is a tuple $TS = (S, T, I, \mathsf{AP}, L)$ where

- the set of states S is $\mathbb{R}^n$,
- $(\mathbf{x}, \mathbf{x}') \in T$ if $\mathbf{x}' = A \otimes \mathbf{x}$,
- $I \subseteq \mathbb{R}^n$ is a set of initial conditions, (we use $I = \mathbb{R}^n$)
- AP is a set of time-difference propositions,
- the labelling function $L : S \to 2^{\mathsf{AP}}$ is defined as follows: a state $\mathbf{x} \in S$ is labelled by '${x_i}' - x_i \sim \alpha$' if $[A \otimes \mathbf{x} - \mathbf{x}]_i \sim \alpha$, where $\sim \in \{>, \geq, <, \leq\}$.

PA of MPL Systems

- Concrete transition systems

Definition (Trans. sys. associated with MPL system)

A transition system for an MPL system is a tuple $TS = (S, T, I, \mathsf{AP}, L)$ where

- the set of states S is $\mathbb{R}^n$,
- $(\mathbf{x}, \mathbf{x}') \in T$ if $\mathbf{x}' = A \otimes \mathbf{x}$,
- $I \subseteq \mathbb{R}^n$ is a set of initial conditions, (we use $I = \mathbb{R}^n$)
- AP is a set of time-difference propositions,
- the labelling function $L : S \to 2^{\mathsf{AP}}$ is defined as follows: a state $\mathbf{x} \in S$ is labelled by '${x_i}' - x_i \sim \alpha$' if $[A \otimes \mathbf{x} - \mathbf{x}]_i \sim \alpha$, where $\sim \in \{>, \geq, <, \leq\}$.

- The (abstract) transition system for MPL system is $\hat{TS} = (\hat{S}, \hat{T}, \hat{I}, P_{mat} \cup P_{time}, \hat{L})$

PA of MPL Systems

- Concrete transition systems

Definition (Trans. sys. associated with MPL system)

A transition system for an MPL system is a tuple $TS = (S, T, I, \mathsf{AP}, L)$ where

- the set of states S is $\mathbb{R}^n$,
- $(\mathbf{x}, \mathbf{x}') \in T$ if $\mathbf{x}' = A \otimes \mathbf{x}$,
- $I \subseteq \mathbb{R}^n$ is a set of initial conditions, (we use $I = \mathbb{R}^n$)
- AP is a set of time-difference propositions,
- the labelling function $L : S \to 2^{\mathsf{AP}}$ is defined as follows: a state $\mathbf{x} \in S$ is labelled by '$x_i{}' - x_i \sim \alpha$' if $[A \otimes \mathbf{x} - \mathbf{x}]_i \sim \alpha$, where $\sim \in \{>, \geq, <, \leq\}$.

- The (abstract) transition system for MPL system is $\hat{TS} = (\hat{S}, \hat{T}, \hat{I}, P_{mat} \cup P_{time}, \hat{L})$

$$\forall \hat{s} \in \hat{S},\ p \in \hat{L}(\hat{s}) \text{ iff } p \text{ is true in } \hat{s}$$

PA of MPL Systems

- Concrete transition systems

Definition (Trans. sys. associated with MPL system)

A transition system for an MPL system is a tuple $TS = (S, T, I, \mathsf{AP}, L)$ where

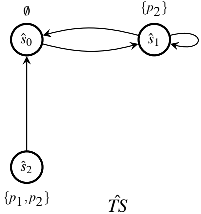
- the set of states S is $\mathbb{R}^n$,
- $(\mathbf{x}, \mathbf{x}') \in T$ if $\mathbf{x}' = A \otimes \mathbf{x}$,
- $I \subseteq \mathbb{R}^n$ is a set of initial conditions, (we use $I = \mathbb{R}^n$)
- AP is a set of time-difference propositions,
- the labelling function $L : S \to 2^{\mathsf{AP}}$ is defined as follows: a state $\mathbf{x} \in S$ is labelled by '$x_i{}' - x_i \sim \alpha$' if $[A \otimes \mathbf{x} - \mathbf{x}]_i \sim \alpha$, where $\sim \in \{>, \geq, <, \leq\}$.

- The (abstract) transition system for MPL system is $\hat{TS} = (\hat{S}, \hat{T}, \hat{I}, P_{mat} \cup P_{time}, \hat{L})$

$$(\hat{s}_i, \hat{s}_j) \in \hat{T} \text{ if } \mathtt{Im}(\mathtt{DBM}(\hat{s}_i)) \cap \mathtt{DBM}(\hat{s}_j) \neq \emptyset$$

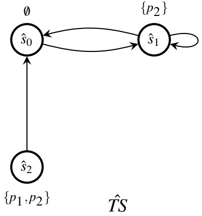
where $\mathtt{Im}(\mathtt{DBM}(\hat{s}_i)) = \{A \otimes \mathbf{x} \mid \mathbf{x} \in \mathtt{DBM}(\hat{s}_i)\}$ (by DBM manipulation)

PA of MPL Systems



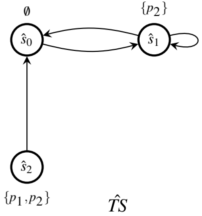
$\hat{TS}$

PA of MPL Systems



$$(x_1' - x_1 \leq 5) \Leftrightarrow p_2$$

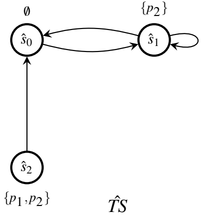
PA of MPL Systems



$(x_1' - x_1 \leq 5) \Leftrightarrow p_2$

Specs: $\Diamond\Box(x_1' - x_1 \leq 5) \equiv \Diamond\Box p_2$

PA of MPL Systems



$(x_1' - x_1 \leq 5) \Leftrightarrow p_2$

Specs: $\Diamond\Box(x_1' - x_1 \leq 5) \equiv \Diamond\Box p_2$

- One TD proposition may correspond to more than one predicates

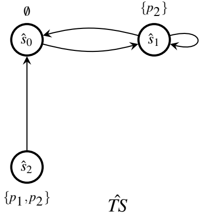
Proposition

Suppose $p_1, \ldots, p_k$ are the predicates corresponding to a TD proposition $t \equiv x_i' - x_i \sim \alpha$.

i. For $\sim \{>, \geq\}$, $t \Leftrightarrow (p_1 \vee \ldots \vee p_k)$

ii. For $\sim \{<, \leq\}$, $t \Leftrightarrow (p_1 \wedge \ldots \wedge p_k)$

PA of MPL Systems



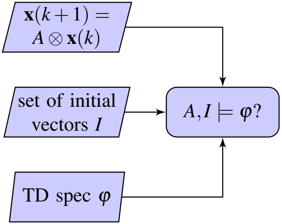
$(x_1' - x_1 \leq 5) \Leftrightarrow p_2$

Specs: $\Diamond\Box(x_1' - x_1 \leq 5) \equiv \Diamond\Box p_2$

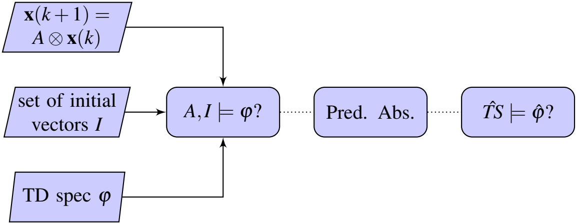
$\hat{TS} \not\models \Diamond\Box p_2 \rightarrow TS \not\models \Diamond\Box(x_1' - x_1 \leq 5)$?

dont know yet

PA of MPL Systems



PA of MPL Systems



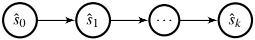
- ~~Infinite and continuous state space~~
- ~~The primed variables~~
- This problem is undecidable

BMC of MPL Systems

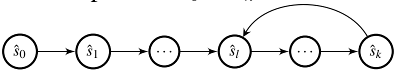
- Find a counterexample with length k
- Increase the length until a pre-known completeness threshold is reached or the problem becomes intractable
- To find completeness threshold is at least as hard as solving the original model-checking problem

BMC of MPL Systems

- Find a counterexample with length k
- Increase the length until a pre-known completeness threshold is reached or the problem becomes intractable
- To find completeness threshold is at least as hard as solving the original model-checking problem
- Two types of k-length bounded counterexample $\pi = \hat{s}_0 \ldots \hat{s}_k$



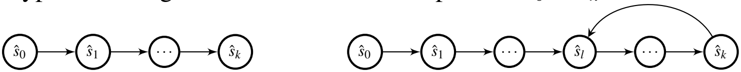
no-loop path



lasso-shaped path

BMC of MPL Systems

- Find a counterexample with length k
- Increase the length until a pre-known completeness threshold is reached or the problem becomes intractable
- To find completeness threshold is at least as hard as solving the original model-checking problem
- Two types of k-length bounded counterexample $\pi = \hat{s}_0 \ldots \hat{s}_k$



no-loop path

lasso-shaped path

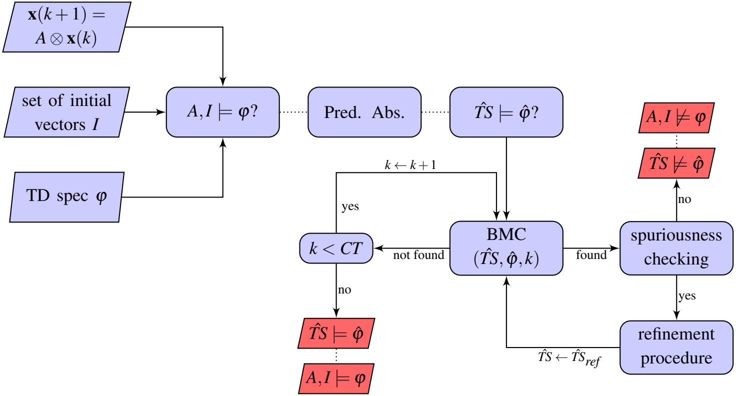
lasso-shaped:

$$\pi = \pi_{stem}(\pi_{loop})^{\omega}$$

where $\pi_{stem} = \hat{s}_0 \ldots \hat{s}_{l-1}$ and $\pi_{loop} = \hat{s}_l \ldots \hat{s}_k$

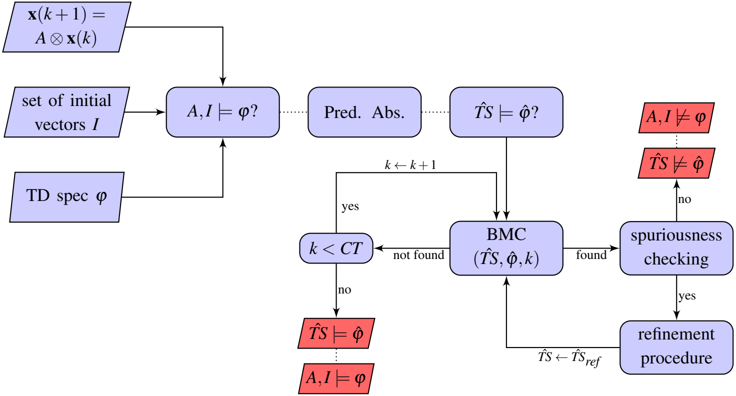
BMC of MPL Systems

- The framework



BMC of MPL Systems

- The framework



- BMC by NuSMV 2.6

BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)

BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure
  - Lazy abstraction*: find pivot state, a state in which the spuriousness starts

* Henzinger, T.A., Jhala, R., Majumdar, R., Sutre, G.: Lazy abstraction. In: Proceedings of the ACM Symposium on Principles of Programming Languages (POPL 2002), pp. 58-70 (2002).

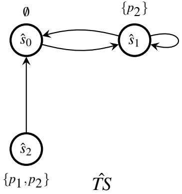
BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure
  - Lazy abstraction: find pivot state, a state in which the spuriousness starts
  - Splitting procedure in `VeriSiMPL` 2*
    splitting a state with more than one outgoing transitions

* Adzkiya, D., Zhang, Y., Abate, A.: `VeriSiMPL` 2: an open-source software for the verification of max-plus-linear systems. Discrete Event Dyn. Syst. 26(1), 109-145 (2016).

BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure



$\hat{TS} \not\models \Diamond\Box(p_2)$

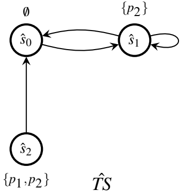
$\pi = (\hat{s}_1)(\hat{s_0}\hat{s}_1)^\omega$

π is spurious

pivot state is $\hat{s}_1$

BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure



$\hat{TS} \not\models \Diamond\Box(p_2)$

$\pi = (\hat{s}_1)(\hat{s}_0\hat{s}_1)^{\omega}$

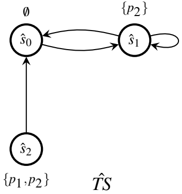
π is spurious

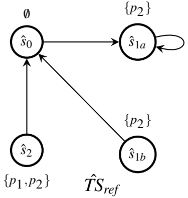
pivot state is $\hat{s}_1$

$\mathtt{DBM}(\hat{s}_0) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 < 0\}$
$\mathtt{DBM}(\hat{s}_1) = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 \leq x_1 - x_2 < 3\}$
$\mathtt{DBM}(\hat{s}_2) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \geq 3\}$

BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure





$\mathtt{DBM}(\hat{s}_0) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 < 0\}$
$\mathtt{DBM}(\hat{s}_1) = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 \leq x_1 - x_2 < 3\}$
$\mathtt{DBM}(\hat{s}_2) = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \geq 3\}$

Partition of $\mathtt{DBM}(\hat{s}_1)$ is

$$\mathtt{DBM}(\hat{s}_{1a}) = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 \leq x_1 - x_2 \leq 2\} \text{ and } \mathtt{DBM}(\hat{s}_{1b}) = \{\mathbf{x} \in \mathbb{R}^2 \mid 2 \leq x_1 - x_2 < 3\}$$

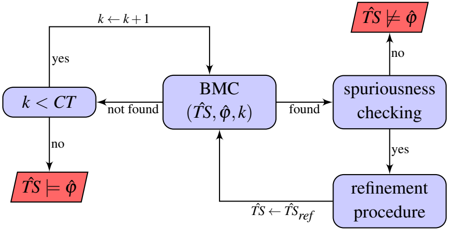
BMC of MPL Systems

- Spuriousness checking
  Algorithms via forward-reachability analysis. Completeness:
  - For no-loop paths
  - For lasso-shaped paths (irreducible MPL systems only)
- Refinement procedure
  - Lazy abstraction: find pivot state, a state in which the spuriousness starts
  - Splitting procedure in `VeriSiMPL` 2
    splitting a state with more than one outgoing transitions
- Upper bound of completeness thresholds

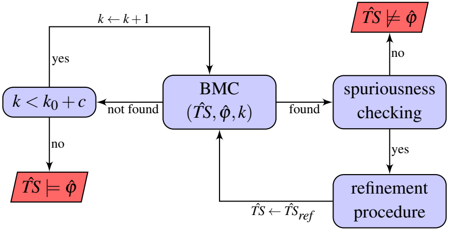
Lemma

Consider an irreducible $A \in \mathbb{R}_{\max}^{n \times n}$ with transient k_0 and cyclicity c and the resulting abstract transition system $\hat{TS} = (\hat{S}, \hat{T}, \hat{I}, P_{mat} \cup P_{time}, \hat{L})$. The completeness threshold for $\hat{TS}$ and for any LTL formula $\hat{\varphi}$ over $P_{mat} \cup P_{time}$ is bounded by $k_0 + c$.

BMC of MPL Systems

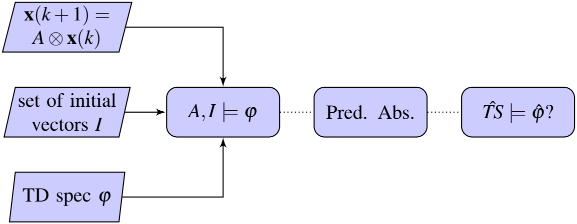


BMC of MPL Systems



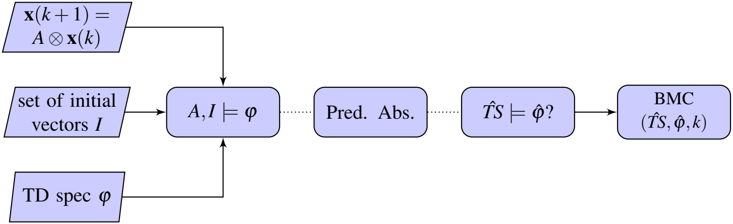
BMC for irreducible MPL systems is complete

BMC of MPL Systems



- ~~Infinite and continuous state space~~
- ~~The primed variables~~
- This problem is undecidable

BMC of MPL Systems



- ~~Infinite and continuous state space~~
- ~~The primed variables~~
- This problem is decidable for irreducible MPL systems

Conclusions

- New abstraction technique of MPL systems
  via a set of predicates.
- BMC of MPL systems w.r.t. TD specifications is decidable
  for irreducible ones.
- The completeness thresholds are related to the transient
  and cyclicity of MPL systems