Approximate trace equivalence of real-time linear algebraic transition systems

Hui Zhang[1], Jinzhao Wu[2*], Hongyan Tan[3], Hao Yang[1]

[1]*Chengdu Institute of Computer Application, Chinese Academy of Sciences, China*

[2]*Guangxi Key Laboratory of Hybrid Computational and IC Design Analysis, Guangxi University for Nationalities, China*

[3]*Institute of Acoustics, Chinese Academy of Sciences, China*

Received 1 March 2014, www.tsi.lv

Abstract

In allusion to data error and equivalence relation for software program design, the paper proposes approximate trace equivalence of real-time linear algebraic transition systems. Firstly, it leads real-time algebraic program into transition system and establishes real-time linear algebraic transition system. And then, it uses matrix norm and matrix singular value decomposition to analyse approximation of traces. Afterwards, it obtains approximate trace equivalence of real-time linear algebraic transition systems. Finally, the traffic light control vehicle flow system example shows that approximate trace equivalence of real-time algebraic transition systems can optimize real-time linear algebraic programs and reduce states.

Keywords: transition system, approximate, trace equivalence, algebraic program

1 Introduction

Early formal model of software programs are mostly discrete model of concurrent systems, such as various process algebra on the language level, computation tree logic on the logical level, automatic machine and transition system on the structural level. They are made up of abstract actions, discrete states and transition relations between states. The discrete method is unable to exchange the data stream. The data stream exchange is one of the most important functions of software programs. It needs to be created algebraic programs which can describe programs data stream exchange. In 2004, Z. Manna et al. defined Hybrid Systems program model with polynomial equations data stream is expressed as states transition labels and opened up a new way of program design and verification based on polynomial algebra [1].

How to define and determine the behaviour equivalence of programs is one of the most important issues in the field of software system design and verification analysis. The same model structure as an equivalent definition is too strictly. A class of functional behaviour of different structure program model may be exactly the same. Therefore, the program models with the same function behaviour are called equivalent. Functional equivalence makes program model be simplified by removing the duplicate branches. Glabbeek proposed fourteen kinds of linear time-branching time equivalence relations [2]. Trace equivalence is a basic state space equivalence relation and it has been widely used in the software system program design and verification analysis. But at present there is no research on trace equivalence of real-time linear algebraic transition systems.

In the software program design and verification analysis, experimental data often have errors. We can only take the approximate value and cannot obtain accurate actual value. Within the error range, we can be treated approximate value as actual value. After the value approximation, it makes some same programs and simplifies the structure of real-time linear algebraic transition system.

From what has been discussion above, firstly, it introduces real-time linear algebraic program to transition system and establishes real-time linear algebraic transition system. Secondly, it uses matrix norm and matrix singular value decomposition to judge whether two real-time linear algebraic transition systems are approximate. Thirdly, it simplifies real-time linear algebraic transition system by trace equivalence theory. Finally, the traffic lights control traffic flow system example shows that approximate trace equivalence of real-time linear algebraic transition systems can optimize real-time linear algebraic programs and reduce the states.

2 Real-time linear algebraic transition system

As we know, labelled transition system is a kind of typical model which describes transition relation between states. Many computer system modelling are based on labelled transition system.

* ***Corresponding author*** e-mail: 415360889@qq.com

Definition 1 (Labelled Transition System [3]) A labelled transition system is a tuple M=<S, L, T, s_0>, where

1) S is a finite set of states, $s \in S$ is a state.
2) L is a finite set of labels, $l \in L$ is a label, a label describes an action.
3) $T \subseteq S \times L \times S$ is a set of transitions.
4) s_0 is the initial system state.

Definition 2 (Transition of Labelled Transition System) A transition of labelled transition system is a tuple <s_i, l, s_j>, where s_i represents pre-state of the transition and s_j represents post-state of the transition.

Definition 3 (Trace [3]) In the transition system, a trace is an action sequence $l_1 l_2 \cdots l_n$ and satisfies an execution sequence $\eta = s_0 l_1 s_1 l_2 \cdots l_n s_n$. $trace(\eta) = l_1 l_2 \cdots l_n$.

Definition 4 (Real-Time Linear Algebraic Program) Let R be the set of real numbers, $x_i (i=1,\cdots,n) \in \mathrm{R}$ and $x_i' (i=1,\cdots,n) \in \mathrm{R}$ be variables, $t \in \mathrm{R}$ is a time variable and t>0. An algebraic program $X' = X + (AX_0 + b)t$ is a real-time linear algebraic program, where $X_0 = (x_{01}, \cdots, x_{0n})^{\mathrm{T}}$ is the initial state value of real-time linear algebraic transition system, $X = (x_1, \cdots, x_n)^{\mathrm{T}}$ is the pre-state value of real-time linear algebraic program and $X' = (x_1', \cdots, x_n')^{\mathrm{T}}$ is the post-state value of real-time linear algebraic program. $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$ is the $n \times n$ matrix and $A \neq 0$, $b = (b_1, b_2, \cdots, b_n)^{\mathrm{T}}$ is the n dimension column vector. The elements of A and b are all real numbers.

In the real-time linear algebraic program, the value of time variable t is a fixed real number. t represents the time from one state transition to another state.

Definition 5 (Real-Time Linear Algebraic Transition System) A real-time linear algebraic transition system is a tuple M=<V,TX,S, P, Q, s_0>, where

1) $V = \{x_1, \cdots, x_n\}$ is a finite set of system variables.
2) TX is a finite set of system state values. $X_0 \in TX$ is the initial state value.
3) S is a finite set of states, $s \in S$ is a state.
4) P is a finite set of real-time linear algebraic program, $p \in P$ is a real-time linear algebraic program.
5) $Q \subseteq S \times P \times S$ is a finite set of state transitions, $q \in Q$ is a state transition.
6) $s_0 \in S$ is the initial state.

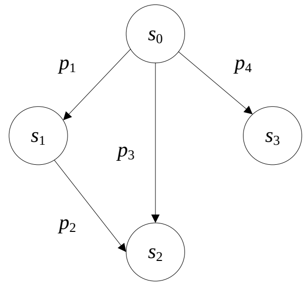

FIGURE 1 An real-time linear algebraic transition system

In the Figure 1, s_0 is the initial state, X_0 is the initial state value. s_1,s_2,s_3 are all system states, X_1, X_2, X_3 respectively represent the state value of s_1, s_2, s_3. p_1, p_2, p_3, p_4 are real-time linear algebraic programs.

3 Approximation of real-time linear algebraic transition systems

In the software program design, matrix elements and vector elements which are in the real-time linear algebraic programs often have errors. Let $A+\delta A$ be actual matrix, A be approximate matrix, $b+\delta b$ be actual vector, b be approximate vector, $X' = X + [(A+\delta A)X_0 + (b+\delta b)]t$ be actual real-time linear algebraic program, $X' = X + (AX_0 + b)t$ be approximate real-time linear algebraic program. In the actual real-time linear algebraic program and corresponding approximate real-time linear algebraic program, time t is the same. We use matrix norm and matrix singular value decomposition to analyse approximation of real-time linear algebraic transition systems.

Definition 6 (Matrix Singular Value Decomposition) Let $A = U \Sigma V^{\mathrm{T}}$ be the matrix singular value decomposition, the main diagonal elements $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ of Σ are called singular value of matrix A, the column vectors $u_1, u_2, \cdots, u_n$ of matrix U are eigenvectors of matrix AA^{T}, $u_1, u_2, \cdots, u_n$ are left singular vectors of matrix A, $v_1, v_2, \cdots, v_n$ are right singular vectors of matrix A.

If $\sigma_{r+1} = \cdots = \sigma_n = 0$, then the matrix A singular value decomposition is $A = \sum_{i=1}^{r} \sigma_i u_i v_i^{\mathrm{T}}$.

Let matrix $A \in \mathrm{R}^{m \times n}$, a norm $\|A\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{\frac{1}{2}}$ of matrix A is called Frobenius norm of matrix A. Matrix norm $\|A\|_F$ and vector norm $\|x\|_2$ are inclusive.

Theorem 1. For any matrix $A \in \mathrm{R}^{m \times n}$, $r = \min\{m, n\}$, $\sqrt{\sum_{i=1}^{r} \sigma_i^2} = \|A\|_F$.

Let s_0 be the initial state, $p'_1 p'_2 \cdots p'_k$ be the trace of actual real-time linear algebraic transition system and it satisfies a finite execute sequence $\eta = s_0 p'_1 s_1 p'_2 \cdots p'_k s_k$, $p_1 p_2 \cdots p_k$ be the trace of approximate real-time linear algebraic transition system and it satisfies a finite execute sequence $\eta = s_0 p_1 s_1 p_2 \cdots p_k s_k$. $p'_1, p'_2, \cdots, p'_k$ are actual real-time linear algebraic programs and $p_1, p_2, \cdots, p_k$ are approximate real-time linear algebraic programs. The actual value of state s_j is $X'_j = [E + jt(A + \delta A)] X_0 + jt(b + \delta b)$, the approximate value of state s_j is $X_j = (E + jtA) X_0 + jtb$. The actual value X'_j can be written as $X'_j = \begin{pmatrix} E + jt(A + \delta A) & jt(b + \delta b) \end{pmatrix} \begin{pmatrix} X_0 \\ 1 \end{pmatrix}$, the approximate value X_j can be written as $X_j = \begin{pmatrix} E + jtA & jtb \end{pmatrix} \begin{pmatrix} X_0 \\ 1 \end{pmatrix}$. Approximate of actual value X'_j and approximate value X_j is equal to approximate of matrix $\begin{pmatrix} E + jt(A + \delta A) & jt(b + \delta b) \end{pmatrix}$ and matrix $B_j = \begin{pmatrix} E + jtA & jtb \end{pmatrix}$ .Let $1 \le j \le k$, p'_j is $X' = X + [(A_j + \delta A_j) X_0 + (b_j + \delta b_j)] t_j$, p_j is $X' = X + (A_j X_0 + b_j) t_j$, the actual value of state s'_j is $X'_j = \left\{ E + \sum_{i=1}^{j} [(A_i + \delta A_i) t_i] \right\} X_0 + \sum_{i=1}^{j} [(b_i + \delta b_i) t_i]$, the approximate value of state s_j is $X_j = \left[E + \sum_{i=1}^{j} (A_i t_i) \right] X_0 + \sum_{i=1}^{j} (b_i t_i)$. The actual value X'_j can be written as $X'_j = \begin{pmatrix} E + \sum_{i=1}^{j} [(A_i + \delta A_i) t_i] & \sum_{i=1}^{j} [(b_i + \delta b_i) t_i] \end{pmatrix} \begin{pmatrix} X_0 \\ 1 \end{pmatrix}$, the approximate value X_j can be written as $X_j = \begin{pmatrix} E + \sum_{i=1}^{j} (A_i t_i) & \sum_{i=1}^{j} (b_i t_i) \end{pmatrix} \begin{pmatrix} X_0 \\ 1 \end{pmatrix}$. The approximation of actual value X'_j and approximate value X_j is equal to approximation of matrix $\begin{pmatrix} E + \sum_{i=1}^{j} [(A_i + \delta A_i) t_i] & \sum_{i=1}^{j} [(b_i + \delta b_i) t_i] \end{pmatrix}$ and matrix $\begin{pmatrix} E + \sum_{i=1}^{j} (A_i t_i) & \sum_{i=1}^{j} (b_i t_i) \end{pmatrix}$.

$$B_j + \delta B_j = \begin{pmatrix} E + \sum_{i=1}^{j} [(A_i + \delta A_i) t_i] & \sum_{i=1}^{j} [(b_i + \delta b_i) t_i] \end{pmatrix},$$

$$B_j = \begin{pmatrix} E + \sum_{i=1}^{j} (A_i t_i) & \sum_{i=1}^{j} (b_i t_i) \end{pmatrix}.$$

The singular value decomposition of matrix δB_j is $\delta B_j = U_j \Sigma_j V_j^{\mathrm{T}}$ and $\left\| \delta B_j \right\|_F = \sqrt{\sum_{i=1}^{r_j} \sigma_{ji}^2}$. $\left\| (B_j + \delta B_j) - B_j \right\|_F = \left\| \delta B_j \right\|_F = \sqrt{\sum_{i=1}^{r_j} \sigma_{ji}^2} = W_j$. For a given positive number ε, if it holds $W_j < \varepsilon$, then matrix $B_j + \delta B_j$ and matrix B_j are approximate. If matrix $B_j + \delta B_j$ and matrix B_j are approximate, then actual value X'_j and approximate value X_j are approximate, actual real-time linear algebraic program and approximate real-time linear algebraic program are approximate.

Definition 7 (Approximation of Traces) For a given positive number ε, $\forall j, 1 \le j \le k$, if it holds $W_j < k$, then actual trace $p'_1 p'_2 \cdots p'_k$ and approximate trace $p_1 p_2 \cdots p_k$ are approximate. If there exists $W_j \ge k$, then actual trace $p'_1 p'_2 \cdots p'_k$ and approximate trace $p_1 p_2 \cdots p_k$ are not approximate.

Definition 8 (Approximation of Real-Time Algebraic Transition Systems) If all traces of two real-time linear algebraic transition systems are approximate, then these two real-time linear algebraic transition systems are approximate.

Approximation of real-time linear algebraic transition systems can optimize real-time linear algebraic programs and reduce bits of matrix elements and vector elements. It can improve computation speed of real-time linear algebraic transition system. For the actual real-time linear algebraic transition system RS_1, it gets approximate real-time linear algebraic transition system RS_2 by approximate algorithm.

4 Approximation trace equivalence of real-time linear algebraic transition systems

Definition 9 (Trace Equivalence [2]) If there exists a process q and $p \xrightarrow{\sigma} q$, then $\sigma \in Act$ is a trace of a process p. Let $T(p)$ be a set of traces of process p. If $T(p) = T(q)$, then two processes p and q are trace equivalence, it can be written as $p =_T q$. In trace semantics two processes are identified if they are trace equivalence.

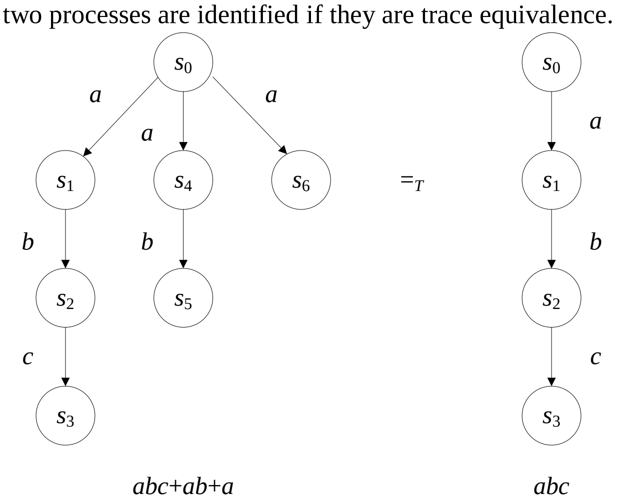

FIGURE 2 A trace equivalence example

In the Figure 2, *a*, *b*, *c* are real-time linear algebraic programs, left real-time linear algebraic transition system has three traces *abc*, *ab*, *a*, right real-time linear algebraic transition system has only one trace *abc*. The left system and right system have the same function.

We have the approximate real-time linear algebraic transition system RS_2, through trace equivalence theory we obtain approximate trace equivalence system RS_3 of actual real-time linear algebraic transition system.

Approximate trace equivalence of real-time linear algebraic transition systems has a very important significance. It can optimize real-time linear algebraic programs and reduce states.

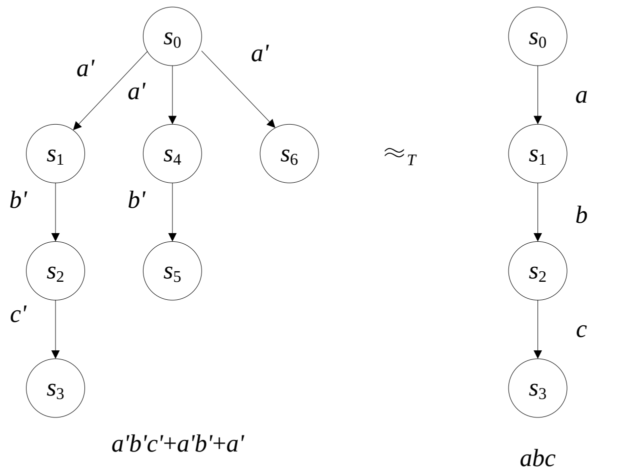

FIGURE 3 An approximate trace equivalence of real-time linear algebraic transition systems example

In the Figure 3, *a'*,*b'*,*c'* are actual real-time linear algebraic programs, *a*,*b*,*c* are approximate real-time linear algebraic programs. The left system has seven states, the right system has four states.

5 Experiments

Assume that vehicles on a section road are divided into two types of motor vehicle and non-motor vehicle. Exit of road has traffic lights. Traffic light basic transformation sequence is yellow→red→blue→yellow.

When the traffic light is yellow, vehicles which are beyond the stop line can forward pass. When the traffic light is red, vehicles are prohibited passage. When the traffic light is blue, vehicles are allowed passage. Let x_1 be motor vehicle flow, x_2 be non-motor vehicle flow, $X=(x_1,x_2)^{\mathrm{T}}$ be vehicle flow. The initial state value of traffic light control vehicle flow system is $X_0=(5,5)^{\mathrm{T}}$. Figure 4 is the actual traffic light control vehicle flow system.

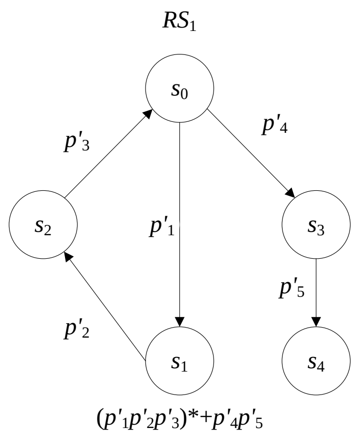

FIGURE 4 Actual traffic light control vehicle flow system

In the Figure 4, actual real-time linear algebraic program p'_1 is $X'=X+\left[\begin{pmatrix}2.01 & \\ & 2.01\end{pmatrix}X_0+\begin{pmatrix}1.01\\1.01\end{pmatrix}\right]\times 2$, p'_2 is $X'=X+\left[\begin{pmatrix}4.02 & \\ & 4.02\end{pmatrix}X_0+\begin{pmatrix}2.02\\2.02\end{pmatrix}\right]\times 10$, p'_3 is $X'=X+\left[\begin{pmatrix}-4.02 & \\ & -4.02\end{pmatrix}X_0+\begin{pmatrix}-2.02\\-2.02\end{pmatrix}\right]\times 11$, p'_4 is $X'=X+\left[\begin{pmatrix}1.99 & \\ & 1.99\end{pmatrix}X_0+\begin{pmatrix}0.99\\0.99\end{pmatrix}\right]\times 2$, p'_5 is $X'=X+\left[\begin{pmatrix}3.98 & \\ & 3.98\end{pmatrix}X_0+\begin{pmatrix}1.99\\1.99\end{pmatrix}\right]\times 10$.

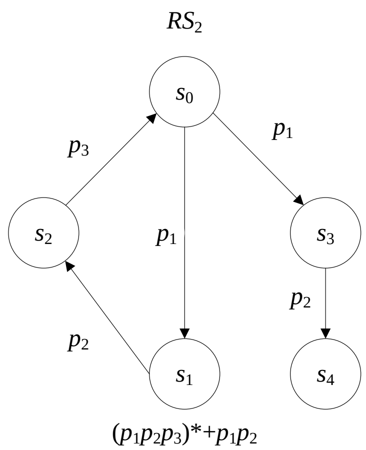

FIGURE 5 Approximate traffic light control vehicle flow system

In the Figure 5, approximate real-time linear algebraic program p_1 is $X'=X+\left[\begin{pmatrix}2 & \\ & 2\end{pmatrix}X_0+\begin{pmatrix}1\\1\end{pmatrix}\right]\times 2$, p_2 is $X'=X+\left[\begin{pmatrix}4 & \\ & 4\end{pmatrix}X_0+\begin{pmatrix}2\\2\end{pmatrix}\right]\times 10$, p_3 is $X'=X+\left[\begin{pmatrix}-4 & \\ & -4\end{pmatrix}X_0+\begin{pmatrix}-2\\-2\end{pmatrix}\right]\times 11$.

For a given positive number ε=1, it uses formula $\|(B_j+\delta B_j)-B_j\|_F = \|\delta B_j\|_F = \sqrt{\sum_{i=1}^{r_j}\sigma_{ji}^2} = W_j$ to obtain $\|\delta B_j\|_{F\max} = 0.44 < \varepsilon$. Because the corresponding traces of RS_1 and RS_2 are approximate, actual traffic light control vehicle flow system RS_1 and approximate traffic light control vehicle flow system RS_2 are approximate.

Through the trace equivalence theory, we get the approximate trace equivalence system RS_3.

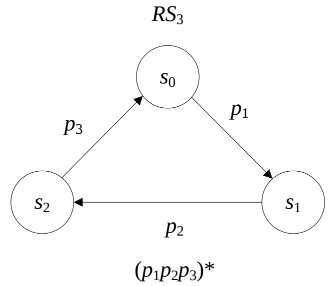

FIGURE 6 Approximate trace equivalence system

Form what has been discussion above, approximate trace equivalence of traffic light control vehicle flow systems can optimize real-time linear algebraic programs and reduce system states.

6 Conclusion

In this paper, the approximate trace equivalence of real-time linear algebraic transition system is proposed. It can optimize real-time linear algebraic programs and reduce system states. In the future work, we will use approximate trace equivalence of real-time linear algebraic transition systems to study approximate trace equivalence of real-time linear algebraic Hybrid Systems.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No.11371003, the Science and Technology Foundation of Guangxi under Grant No.10169-1, the Natural Science Foundation of Guangxi under Grant No.2011GXNSFA018154 and N0.2012GXNSFGA060003, and the Scientific Research Project No.201012MS274 from Guangxi Education Department.

References

[1] Sankaranarayanan S, Sipma H, Manna Z 2004 Non-Linear Loop Invariant Generation Using Groebner Bases *POPL04* 318-29
[2] van Glabbeek R J 1990 The Linear Time-Branching Time Spectrum *Inst. für Informatik* 1-86
[3] Clarke E, Grumberg O, Peled D 2000 *Model Checking* MIT Press
[4] Chen Y 2000 *Matrix Theory* WestNorth Industry University Press
[5] Hierons R M 2012 Overcoming controllability problems in distributed testing from an input out transition system *Distributed Computing*. **25**(1) 63-81
[6] Bauer S S, et al 2012 Weighted modal transition systems *Formal Methods in System Design* 1-28
[7] Shen M 2013 Hinfin filtering of continuous Markov jump linear system with partly known Markov modes and transition probabilities *Journal of the Franklin Institute* **350**(10) 3384-99
[8] Fu J, Tanner H, Heinz J, Chandlee J 2014 Adaptive symbolic control for finite-state transition systems with grammatical inference *IEEE Transactions on Automatic Control* **59**(2) 505-11
[9] Bernardo M, Nicola N D, Loreti M 2014 Revisiting trace and testing equivalences for nondeterministic and probabilistic processes *Logical Methods in Computer Science* **10**(1)
[10] Cheval V, Cortier V, Delaune S 2013 Deciding equivalence-based properties using constraint solving *Theoretical Computer Science* **492** 1-39
[11] Bonchi F et al 2012 Final semantics for decorated traces *Electronic Notes in Theoretical Computer Science* **286** 73-86
[12] Weidich M, Mendling J 2012 Perceived consistency between process models *Information Systems* **37**(2) 80-98

Authors

Zhang Hui, China

Current position, grades: doctorial student
University studies: Chengdu Institute of Computer Application, Chinese Academy of Sciences
Scientific interest: formal verification and network security

Wu Jinzhao, China

Current position, grades: professor
University studies: Institute of System Science, Chinese Academy of Sciences
Scientific interest: formal verification and network security

Tan Hongyan, China

Current position, grades: associate professor
University studies: Lanzhou University
Scientific interest: formal verification and network security

Yang Hao, China

Current position, grades: doctorial student
University studies: Chengdu Institute of Computer Application, Chinese Academy of Sciences
Scientific interest: formal verification and network security