하이브리드 시스템 검증 도구

조재연
건국대학교
컴퓨터공학부 석사과정
(tm77@konkuk.ac.kr)

유준범
건국대학교
컴퓨터공학부 조교수
(jbyoo@konkuk.ac.kr)

1. 서 론

대부분의 산업 분야에서는 시스템이 복잡해지고 있으며, 개별 시스템들의 연동이 매우 활발하게 일어나고 있다. 이런 복잡한 시스템들을 다루기 위해서 Cyber Physical System(CPS) 기술을 사용한다. CPS는 컴퓨팅 요소와 물리적 요소가 강하게 결합되거나, 서로 협동을 하게 되며, 자동차, 장난감, 의료기기 그리고 과학적인 기계 등 여러 산업분야에 적용되는 시스템이다. CPS의 일종인 Hybrid System은 연속적인 요소와 컴퓨팅을 할 수 있는 요소로 이루어져 있으며, 동적인 행위로 기술되어 있다. Hybrid System은 연속적인 행위를 표현하는 Flow와 이산적인 행위를 표현하는 Jump로 이루어져 있다. Flow는 미분 방정식으로 표현할 수 있으며, Jump는 점화식과 컨트롤 그래프로 표현할 수 있다. 일반적인 Hybrid System에서는 연속적인 Flow의 흐름에 의해서 이산적인 상태가 바뀐다. 이러한 하이브리드 시스템의 개발을 위해서 모델을 만들고 이를 시뮬레이션하거나 검증할 도구가 필요하다. 본 논문은 이 도구들을 소개하고, 각 도구들이 가지고 있는 기능에 대해 소개한다. Hybrid System을 다룰 수 있는 13가지의 도구를 조사하였으며, 그 중에 검증할 수 있는 도구를 선정하여 집중적으로 소개한다. 이전 논문[1]에서 Hybrid System을 다룰 수 있는 도구들을 소개하였고 자세한 내용이 나와 있으나, 본 논문에서는 이를 간단하게 요약하였으며, 검증에 초점을 맞추고 도구를 소개한다.

본 논문은 다음과 같은 구성으로 되어 있다. 2장에서는 Hybrid System을 다룰 수 있는 도구들에 관해 소개한다. 상용 도구가 필요 없으며 자동으로 검증가능

한 도구와 그렇지 않은 도구를 분류하였다. 3장에서는 2장에 소개한 도구들을 요약하며 결론을 맺는다.

2. Hybrid System

Hybrid System과 관련된 도구들은 CHARON, CheckMate, d/dt, Ellipsoidal ToolBox, GBT, HSIF, HSolver, HyTech, HyVisual, HybridToolBox, HYSDEL, Keymaera, Ptolemy, MATISSE, PHAVer, SHIFT 그리고 Level Set ToolBox등이 있다. 본 섹션에서는 각 도구를 소개하며, 상용 도구에 의존하지 않으며 검증 가능한 도구의 모임을 따로 분류하여 기술하였다.

2.1 시뮬레이션, 모델링, 및 정리 증명(Theorem proving) 도구들

이번에 소개할 도구들은 자동으로 검증을 하지 않거나, Matlab같은 상용 도구가 필요한 도구들이다.

2.1.1 CHARON

CHARON[2]은 하이브리드 시스템을 동시에 컨트롤이 가능하고, 계층적으로 디자인 할 수 있는 고차원의 언어이다. 펜실베니아 대학에서 개발되었으며, CHARON 도구는 GUI를 제공하며 모델링과 시뮬레이션이 가능하다. CHARON 도구는 [3]에서 자료를 받을 수 있다. 최신 버전은 2003년에 만들어졌다.

2.1.2 CheckMate

CheckMate[4]는 카네기 멜론대학에서 개발되었으며, Hybrid System을 검증할 수 있는 도구이며, Matlab의 Plugin이다. ACTL(Computational Tree Logic) Space에서의 고정 소수점 계산 기법을 사용한다. [5]에서 관련 자료를 받을 수 있지만, CheckMate 도구를 찾을 수 없다. CheckMate의 입력 시스템은 Threshold event-driven Hybrid System이다. 이 시스템에서는 연속적인 식으로 이루어진 입력을 받는다. 어떤 이산 상태에서 특정 값이 어떤 한계에 도달하면 이산적인 상태가 바뀐다. 검증할 모델의 일부 요소에 비선형적인 미분 방정식을 사용할 수 있다.

2.1.3 Ellipsoidal Toolbox

Ellipsoidal Toolbox[6]는 Matlab의 Plugin이며, 타원체를 계산할 수 있으며 연속적, 이산적, 또는 제약사항이 있는 선형 시스템을 분석할 수 있다. 명시적으로 Hybrid System을 분석할 수는 없지만, Hybrid System의 몇몇 요소들을 계산할 수 있다. Ellipsoidal Toolbox는 [7]에서 받을 수 있다.

2.1.4 GBT

GBT(Geometric bounding toolbox)[8]는 polytope와 1차원 이상의 유클리디언 공간의 타원체를 계산할 수 있다. 타원체들의 내/외부를 Approximation을 할 수 있다. GBT는 현재 상용 도구로 제작되었으며, [37]에서 구할 수 있다.

2.1.5 HSIF

HSIF[9]는 Hybrid automata로 이루어진 Network를 모델링할 수 있는 도구이며, 각 하이브리드 오토마타는 신호와 공유변수를 이용하여 통신한다. 신호는 동기적인 통신과 모델의 예측 가능한 실행을 위해 쓰이며, 공유 변수는 비동기적으로 느슨하게 오토마타들을 연결하는 데 쓰인다. HSIF와 관련된 도구들 중에서 모델링과 Dynamic analysis를 할 수 있으며, 윈도우에서 실행 될 수 있다. HSIF는 [29]에서 받을 수 있다.

2.1.6 HSolver

HSolver[10]는 Theorem proving을 이용하여 검증하는 도구이며 연속적인 시간을 다루는 하이브리드 시스템이다. 비선형 순차적 미분등식(Non-linear temporal differential equation)을 분석할 수 있다. 하이브리드 시스템뿐만 아니라, 입력 시스템의 추상적인 모델까지

계산할 수 있다. HSolver는 아직 어떤 하이브리드 시스템의 종류에도 최적화되어 있지 않다. HSolver가 사용하는 도구는 abstraction refinement에 기반한 간격적 제약 전파 기법(Interval constraint propagation based abstraction refinement)이다. 이 도구는 점진적으로 입력 시스템의 추상화를 refine하는 것이다. HSolver는 이산적인 제약사항을 추론하고 계산하는 RSolver[11]에 기반하여 알고리즘을 구현하였다.

2.1.7 HyVisual

HyVisual[12]은 GUI를 제공하는 Modeling Tool이다. HyVisual은 동적 시스템과 하이브리드 시스템을 모델링할 수 있는 Block-diagram editor를 제공하며, 시뮬레이션을 할 수 있다. HyVisual은 도메인 특성 기반의 도구인 Ptolemy 위에서 작동한다. [30]에서 다운로드 받을 수 있다.

2.1.8 HYSDEL

HYSDEL[13](Hybrid Systems DEscription Language)는 Discrete Hybrid Automata(DHA)을 표현한다. DHA는 Finite State Machine, Switched Affine system, Reset map, Event generator 등의 요소로 이루어져 있다. [31]에서 다운받을 수 있다.

2.1.9 KeYMaera

KeyMaera[14]는 Differential Dynamic Logic formula를 입력언어로 사용한다. 검증 기법은 KeY[15](Interactive theorem prover)이다. Java를 이용하여 실행할 수 있으며 현재도 업데이트 되고 있다.

2.1.10 Ptolemy

Ptolemy[16]는 종합적인 Hybrid System 및 Embedded System을 시뮬레이션, 검증, 모델링 할 수 있는 framework로써 검증 시 REDLIB[17], SMV[18], Real Maude[19]등과 연동하여 검증 또는 분석을 실행할 수 있다. Java 언어로 작성되었다.

2.1.11 MATISSE

MATISSE[20]는 Hybrid System의 일종인 Transition System을 Bisimulation기법을 이용하여 분석할 수 있다. Multi-Parametrinc Toolbox[21]와 Matlab이 필요하다.

2.1.12 Multi-Parametric Toolbox

Multi-Parametric Toolbox는 HYSDEL언어로 기술된 DHA를 사용할 수 있으며, Linear Time-Invariant dynamics와 Piecewise-Affine dynamics를 입력 언어로 사용할 수 있다. Reachability analysis 기법을 이용하여 계산한다.

2.1.13 SHIFT

SHIFT[22]는 Hybrid System을 모델링할 수 있는 언어로써 재사용가능한 시뮬레이션 프레임 워크를 구현하는 것에 초점을 두고 개발되었다. 관련 도구는 Shift 언어를 C로 변환할 수 있다.

2.1.14 STeP

STeP[23]의 입력 언어는 Reactive system이다. STeP을 검증할 Property는 Temporal logic formula로 표현될 수 있다. STeP으로 Hybrid System을 검증할 수 있다. [32]에서 Educational Version을 다운받아서 실행 할 수 있다.

2.1.15 Level Set Toolbox

Level Set Toolbox[24]는 시뮬레이션과 검증을 할 수 있으며, MATLAB의 플러그인이다. Dynamic implicit surface들을 시뮬레이션 할 수 있다. 해밀턴 야코비 방정식의 해를 Approximation하는 기법을 사용하여 동적인 공간을 시뮬레이션 할 수 있다.

2.2 선정된 검증 도구들

다음에 소개할 도구들은 모델 체킹 기법을 이용하여

Hybrid System을 자동으로 검증할 수 있다. 도구들은 Matlab의 Plugin이 아니기 때문에 상용 도구가 아니며 실행 가능하다.

2.2.1 d/dt

d/dt[25]는 Thao Dang에 의해 개발되었다. 입력 언어는 선형 미분식이 포함된 하이브리드 시스템이다. f(x)=Ax+Bu의 형태에서 u는 유계 볼록면체의 값이고, 불확실한 값이다. 모든 Invariant와 전이 조건이 볼록 다면체에 의해 정의된다. 볼록 다면체는 선형 부등식의 조합이다. reset값은 이산적인 전이들이 R(x)=Ax+B의 형태로 제시된다. d/dt는 리눅스에서 실행할 수 있으며, 검증 기법은 On-the-fly approximating, Over-approximation등의 기법을 사용한다. d/dt는 [33]에서 다운받을 수 있다. 최신 버전은 2000년도에 개발되었다.

2.2.2 HyTech

HyTech[26]는 Approximation기법을 사용하여 Linear Hybrid Automata의 도달가능성을 검증할 수 있다. HyTech의 입력언어는 선형 하이브리드 오토마타로써, 하이브리드 오토마타의 대부분의 요소들이 선형이어야 하며, Flow independence를 만족시켜야 한다. Flow는 오토마타 내에서 간단한 Dynamic으로 이루어진 Hybrid System을 계산하는데 적합하다. [34]에서 다운받을 수 있다. Linux 및 window(cygwin을 사용)에서 실행이 가능하다.

그림 1의 왼쪽의 Train-gate-controller모델은 [34]에서 다운받을 수 있는 예제 중 하나로 기차가 역 안으로 들어올 때 게이트를 열고 닫는 것을 조절하는 시스템으로써 x는 교차로부터 train까지의 거리, y는 게이트의 각도, t는 컨트롤러의 타이머이며, train과, controller, gate등의 오토마타로 이루어져 있다. 그림 1의 오른쪽은 x가 10보다 작을 때, gate가 closed 상태에 없으면 안전성을 위반하는지 검증한 예제이다.

2.2.3 Phaver

PHAVer[27]는 Hytech의 입력 언어를 공유할 수 있으며, Hytech와 유사하게 사용할 수 있다. Input/Output 변수를 가진 구조의 Linear Hybrid Automata를 입력언어로 받아들이며, Affine dynamics를 계산할 수 있다. Reachability analysis, Approximation등을 할 수 있다. 검증 기법은 On-the-fly Over approximation이다. [35]에서 다운받을 수 있다.

그림 2의 왼쪽은 PHAVer의 입력언어로 작성한 Bouncing Ball 모델 예제로 [35]에서 다운받을 수 있다. Bouncing Ball 모델은 공이 떨어지는 상황을 모델링한 것이며, x변수는 위치, z변수는 속도를 나타낸다. 그림 2의 오른쪽 그림은 Phaver로 분석한 결과를 Graph로 그려놓은 것으로써, 가로축은 위치를, 세로축은 속도를 나타낸다. 붉은 색 부분은 해당 모델이 가질 수 있는 값들의 모임이다.

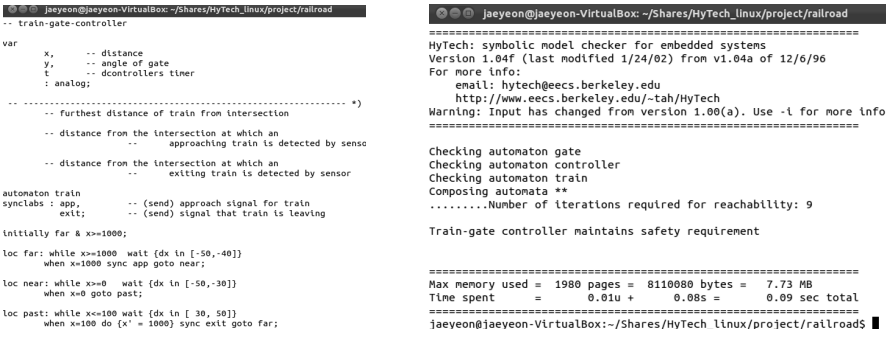

그림 1. HyTech언어로 작성된 Train-gate-controller 모델(왼쪽)과 검증 결과(오른쪽)

```
jaeyeon@jaeyeon-VirtualBox: ~/Shares/phaver/src
// ------------------------------------------------------------
// Hybrid System Example: Bouncing Ball
// ------------------------------------------------------------

// ------------------------------------------------------------
// Constants
// ------------------------------------------------------------
g:=1; // constant for gravity

// ------------------------------------------------------------
// System Description
// ------------------------------------------------------------
automaton bouncing_ball
contr_var: x, v;
synclabs: jump;
loc state: while x>=0 & x<=10 & v<=10 & v>=-10 wait {x'==v & v'==-g}
       when x==0 & v<0 sync jump do {v'==-v*0.5 & x'==x} goto state;
initially: state & x==2 & v==0;
end

// ------------------------------------------------------------
// Define Partitioning
// ------------------------------------------------------------
// Add a new label for the transitions that are introduced
// between partitions
bouncing_ball.add_label(tau);

// Define the directions of the partitions
// here: in both axes with a min. threshold of partition size 0.2
//bouncing_ball.set_refine_constraints((x, 0.2),(v,0.2),tau);
```

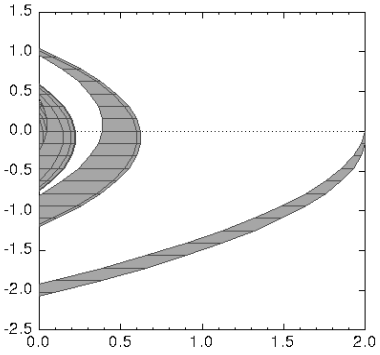

그림 2. PHAVer를 이용한 bouncing ball 예제 및 분석 결과

2.1.4 SpaceEx

SpaceEx[28]는 Hybrid Automata를 SX라는 언어를 이용하여 입력하고 이를 검증한다. SX라는 언어는 Hybrid System을 xml의 형태로 기술할 수 있는 것이다. SpaceEx는 Hybrid Automata를 입력받을 수 있는데, Hybrid Automata SpaceEx의 최근 버전은 2011년 현재 지속적으로 업데이트 되고 있으며, 기본적인 Simulator 시나리오를 사용할 수 있도록 확장되었다. LeGuernic-Girard Algorithm이란 것을 사용하여 검증한다. 그러나 아직 미분 등식에서 비결정적인 경우에 대해서 지원하지 않는다. 또한 이것은 가드와 불변에서의 변수를 조절하지 못한다. [36]에서 다운받을 수 있다. SpaceEx는 모델링을 할 수 있는 GUI와, 웹 인터페이스와, Core executable을 제공한다.

그림 3은 GUI환경에서 Hybrid System을 모델링할 수 있는 예제이다. SpaceEx의 모델 구조는 Component들로 이루어져 있고, Component 안에 Variable, Transition, Location등으로 이루어져 있다.

3. 결 론

본 논문에서는 Hybrid System을 모델링하고, 시뮬레이션 또는 검증을 할 수 있는 도구들을 소개하였다.

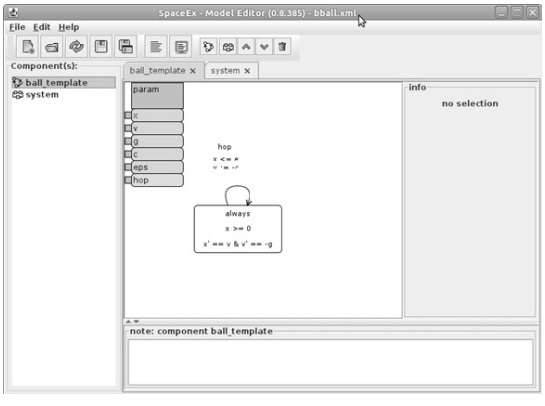

그림 3. SpaceEx 모델링 예제

각 도구들은 모델링, 시뮬레이션 및 검증 기능을 가지고 있다. 관련 도구들 중에서 상용 도구를 필요로 하지 않으면서 Hybrid System을 설계하고 이를 검증할 수 있는 도구들을 선정하였으며, 이 도구들의 기능을 살펴보았다.

표 1은 본 논문에서 소개한 도구들을 정리한 표이다. 정형 검증의 자동화 정도가 자동인 것은 대부분 Automatic model checking 기법을 사용하며, 수동인 것은 Theorem proving 기법이나 Computation을 수행한다. 특히 SpaceEx는 VMWare를 이용하면 웹 인터페이스를 사용할 수 있다. 향후에는 위의 각 도구들을 이용하여 Hybrid System개발과정에서 어떻게 활용할 수 있을지에 대해 연구할 것이다.

참고문헌

[1] Carloni, L.P. and Passeron, R. and Pinto, A, "Languages and tools for hybrid systems design", 2006.

[2] Alur, R. and Grosu, R. and Hur, Y. and Kumar, V. and Lee, I, "Modular specification of hybrid systems in CHARON", Hybrid Systems : Computation and Control, p. 6–19, 2000.

[3] CHARON Hompage : http://www.cis.upenn.edu/mobies/charon.

[4] Silva, B.I. and Krogh, B.H, "Formal verification of hybrid systems using CheckMate : A case study" in American Control Conference, 2000. Proceedings of the 2000, IEEE, vol 3, pp. 1679–1683, 2000.

[5] Checkmate introduction page : http://www.mathworks.com/matlabcentral/fx_files/15441/3/content/doc/main.htm.

[6] Kurzhanskiy, A.A. and Varaiya, P, "Ellipsoidal toolbox (ET)" in Decision and Control, 2006 45th IEEE

표 1. 도구 비교(*cygwin을 이용하면 Windows에서도 실행 가능)

이름	최신 업데이트	실행 환경	주요 검증 기법	정형 검증의 자동화 정도
CHARON	2001	Java	없음	없음
CheckMate	–	Matlab	approximation	자동
d/dt	2001	Linux*	approximation	자동
Ellipsoidal Toolbox	2011	Matlab	approximation	수동
GBT	2003	Matlab	convex hull computation	수동
HSIF	2004	Windows	없음	없음
HSolver	2005	Linux	approximation	수동
HyTech	2000	Linux	approximation	자동
HyVisual	2005	Java	없음	없음
Hybrid Toolbox	2011	Matlab	LP/QP Solver이용	수동
HYSDEL	2011	Linux*	없음	없음
KeYmaera	2011	Java	Key(Theorem proving)알고리즘 이용	수동
Level Set Toolbox	2011	Matlab	approximation	수동
MATISSE	2005	Matlab	bisimulation	수동
Multiparametric Toolbox	2006	Matlab	approximation	수동
PHAVer	2007	Linux*	approximation	자동
Ptolemy	2010	Java	SMV(Symbolic model checker)이용	자동(다른 도구를 이용함)
SHIFT	1999	Linux*	없음	없음
SpaceEx	2011	VMWare, Linux*	approximation	자동
STeP	1998	Linux*	theorem proving	수동

Conference on, IEEE, pages 1498–1503, 2006.
[7] Ellipsoidal Toolbox hompage : http://www.eecs.berkeley. edu/Pubs/TechRpts/2006/EECS-2006-46.html.
[8] Veres, SM. "Geometric bounding toolbox(GBT) for Matlab", Official website : http://www.sysbrain.com, 2003.
[9] Group, M. "Hsif semantics(version 3, synchronous edition)." Internal document, The University of Pennsylvania 2002.
[10] Ratschan, S. and She, Z, "HSolver", 2004.
[11] Ratschan, S. and others, "RSolver", 2004.
[12] Brooks, C. and Cataldo, A. and Lee, E.A. and Liu, J. and Liu, X. and Neuendorffer, S. and Zheng, H. and ERL, UCB, "Hyvisual: A hybrid system visual modeler", Electronics Research Laboratory, College of Engineering, University of California, 2005.
[13] Torrisi, F.D. and Bemporad, A. and Bertini, G. and Hertach, P. and Jost, D. and Mignone, D, "Hysdel 2.0.5-user manual", Eidgenossische Technische Hochschule. Zurich, Switzerland, pp 7–24, 2002.
[14] Platzer, A, and Quesel, J.D, "Keymaera: A hybrid theorem prover for hybrid systems(system description)", Automataed Reasoning, pp171–178, Springer, 2008.
[15] Beckert, B. and H"ahnle, R. and Schmitt, P.H, "Verification of object-oriented software: The KeY approach", Springer-Verlag, 2007.
[16] Davis, J. and Goel, M. and Hylands, C. and Kienhuis, B. and Lee, E. and Liu, J. and Liu, X. and Muliadi, L. and Neuendorffer, S. and Reekie, J. and others, "Ptolemy II: Heterogeneous concurrent modeling and design in Java", University of California, Berkeley, Tech. Rep. UCB/ERL M, vol 99, 1999.
[17] Wang, F. "REDLIB for the formal verification of embedded systems", Leveraging Applications of Formal Methods, Verification and Validation, 2006. ISoLA 2006. Second International Symposium on, pages pp. 341–346, IEEE, 2006.
[18] McMillan, K, "Cadence smv", Cadence Berkely Labs, CA, 2000.
[19] CsabaOlveczky, P. and Meseguer, J. "Real-Time Maude: A tool for simulating and analyzing real-time and hybrid systems", Proceedings Third International Workshop on Rewriting Logic and its Applications, WRLA, pp. 18–20, 2000.
[20] Girard, A. and Pappas, G.J. "Approximation metrics for discrete and continuous systems", Automatic Control, IEEE Transactions on, vol 52, n.5, page 782–798, IEEE, 2007.
[21] Kvasnica, M. and Grieder, P. and Baotic, M. and Morari, M, "Multi-parametric toolbox(MPT)", Hybrid Systems : Computation and Control, pp. 121–124, 2004.
[22] Deshpande, A. and Gollu, A. and Semenzato, L. "SHIFT reference manual", 1997.
[23] Bjorner, N. and Browne, A. and Chang, E. and Colon, M. and Kapur, A. and Manna, Z. and Sipma, H. and Uribe, T, "STeP: Deductive-algorithmic verification of reactive and real-time systems", Computer Aided Verification, Computer Aided Verification, pp 415–418, 1996.
[24] Mitchell, I.M, "A toolbox of level set methods version 1.0", 2004.
[25] Asarin, E. and Dang, T. and Maler, O, "The d/dt tool for verification of hybrid systems", Computer Aided Verification, pp 746–770, Springer, 2002.
[26] Henzinger, T.A. and Ho, P.H. and Wong-Toi, H, "HyTech: A model checker for hybrid systems", International Journal on Software Tools for Technology Transfer(STTT), vol 99, num 1, pp 110–122, 1997.
[27] Frehse, G, "PHAVer: Algorithmic verification of hybrid systems past HyTech", Hybrid Systems: Computation and Control, pp 258–273, 2005.
[28] Frehse, G. and Le Guernic, C. and Donze, A. and Cotton, S. and Ray, R. and Lebeltel, O. and Ripado, R. and Girard, A. and Dang, T. and Maler, O, "Spaceex: Scalable verification of hybrid systems, Computer Aided Verification, pp 379–395, 2011.
[29] HSIF homepage http://w3.isis.vanderbilt.edu/Projects/mobies/downloads.asp#HSIF.
[30] ptolemy homepage : http://ptolemy.eecs.berkeley.edu.
[31] Hysdel hompage :http://control.ee.ethz.ch/~hybrid/hysdel/hysdel.php.
[32] STeP hompage : http://www-step.stanford.edu.
[33] d/dt hompage : http://www-verimag.imag.fr/~tdang/ddt.html.
[34] HyTech homepage : http://embedded.eecs.berkeley.edu/research/hytech.
[35] phaver hompage : http://www-verimag.imag.fr/~frehse/phaver_web.
[36] SpaceEx homepage : http://spaceex.imag.fr.
[37] GBT homepage : http://www.sysbrain.com/gbt.