Managing Large Collections of Benchmarks

An Experiment Report

Hubert Garavel

Inria Grenoble – LIG

Université Grenoble Alpes

http://convecs.inria.fr

Outline

- Building a collection of benchmarks and running a software competition are different problems
- Five collections of benchmarks:
  - VLTS (Very Large Transition Systems)
  - VLPN (Very Large Petri Nets)
  - MCC (Model Checking Contest)
  - REC (Rewrite Engines Competition)
  - MARS (Models for Analysis of Real Systems)
- Conclusion

1. VLTS

Very Large Transition Systems

http://cadp.inria.fr/resources/vlts

Origins of VLTS

- Labelled Transition Systems:
  - the semantic model for process calculi, bisimulations…
  - states and transitions *(explicit-state verification)*
  - all information attached to transitions *(action-based)*
  - can get huge (billions of states and transitions)
- Need for LTS benchmarks:
  - A. Dovier, C. Piazza, A. Politicri (CAV 2001):
    "To the best of our knowledge, there is no 'official' set of benchmarks for testing a [bisimulation] algorithm such as the one we propose in our paper."

The VLTS benchmark collection (1/2)

- A Dutch-French project (INRIA and CWI)
- Designed in 2003 (joint work with Stefan Blom)
- A collection of 40 LTSs:
  - sorted by increasing sizes

    from 300 to 33,000,000 states
  - many of them coming from industrial case studies
    $\Rightarrow$ scrambling (renaming all labels to A1, A2, A3, etc.)
  - deliberate addition of "pathological" examples
    (e.g., disconnected graphs with unreachable states)

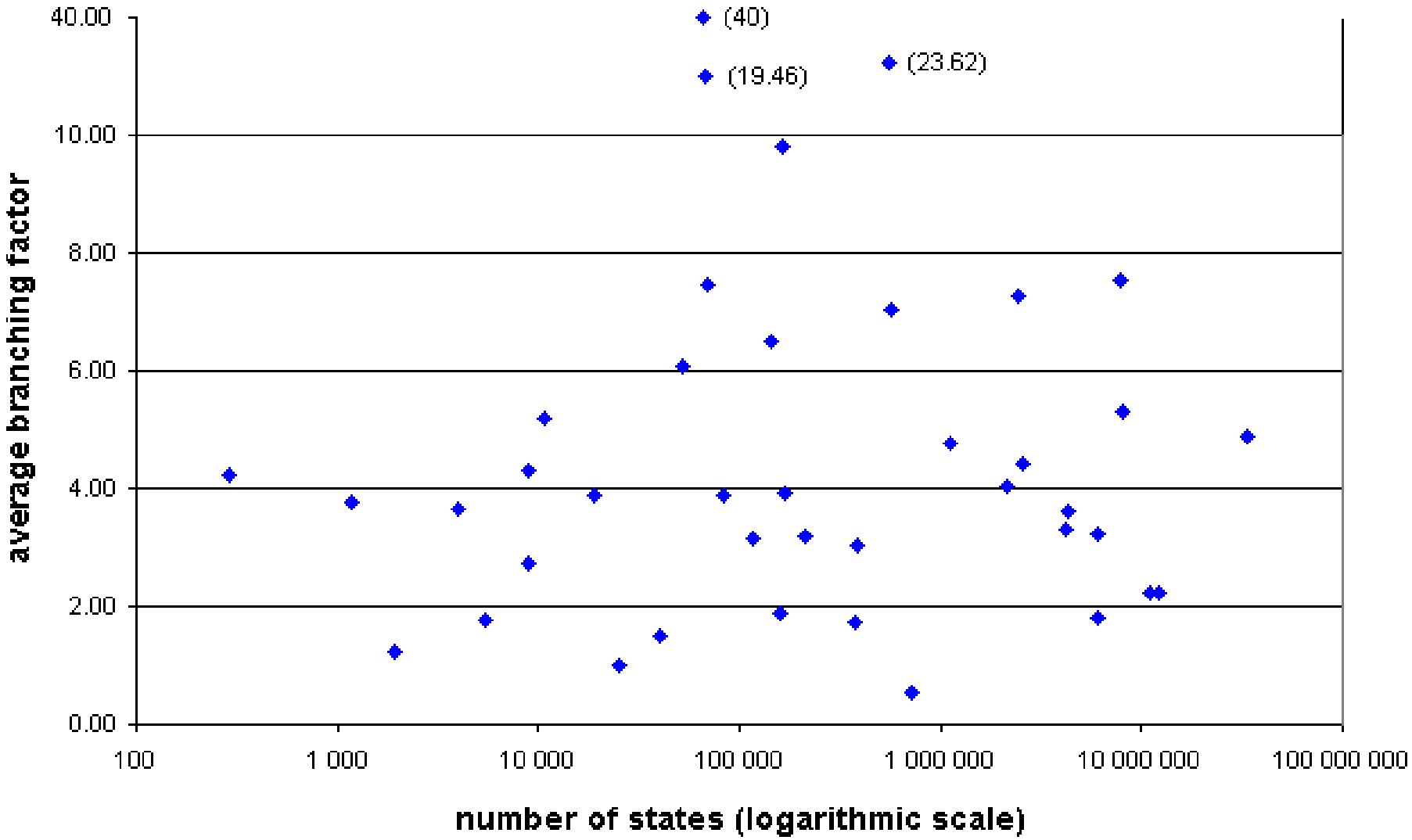
The VLTS benchmark collection (2/2)

- Main issues:
  - Each LTS is potentially a large file
  - Problem #1: disk storage
  - Problem #2: network bandwidth
- Approch followed:
  - BCG file format of CADP (specific data compression) + bzip2 tool (generic data compression)
  - the VLTS collection takes 500 Mbytes
  - distribution via anonymous FTP
- VLTS used and cited in 47 publications so far

The 40 VLTS benchmarks

Name (.bcg.bz2)	#States	#Transitions	#Tau-transitions	#Labels	Branching factor avg [min - max]	Deadlocks	Livelocks	Deterministic
vasy_0_1	289	1,224	0	2	4.24 [4 - 8]	-	-	-
cwi_1_2	1,952	2,387	2,215	26	1.22 [1 - 16]	-	-	-
vasy_1_4	1,183	4,464	1,213	6	3.77 [2 - 5]	-	-	-
cwi_3_14	3,996	14,552	14,551	2	3.64 [0 - 6]	X	-	-
vasy_5_9	5,486	9,676	2,094	31	1.76 [0 - 6]	X	-	-
vasy_8_24	8,879	24,411	8,534	11	2.75 [1 - 5]	-	-	-
vasy_8_38	8,921	38,424	2,916	81	4.31 [0 - 10]	X	-	X
vasy_10_56	10,849	56,156	2,680	12	5.18 [4 - 6]	-	-	X
vasy_18_73	18,746	73,043	39,217	17	3.90 [1 - 6]	-	-	-
...								
cwi_7838_59101	7,838,608	59,101,007	22,842,122	20	7.54 [3 - 13]	-	X	-
vasy_8082_42933	8,082,905	42,933,110	2,535,944	211	5.31 [0 - 48]	X	-	X
vasy_11026_24660	11,026,932	24,660,513	2,748,559	119	2.24 [0 - 13]	X	-	-
vasy_12323_27667	12,323,703	27,667,803	3,153,502	119	2.25 [0 - 13]	X	-	-
cwi_33949_165318	33,949,609	165,318,222	74,133,306	31	4.87 [1 - 17]	-	X	-

Dispersion of VLTS benchmarks

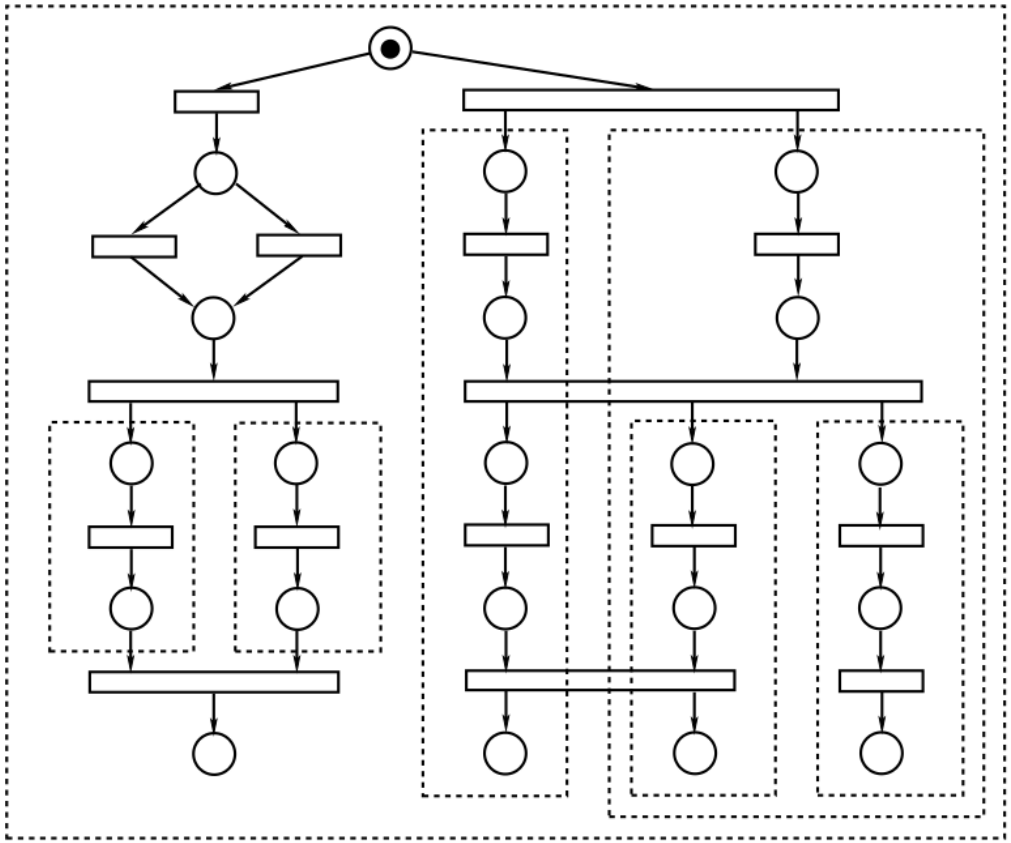


2. VLPN

Very Large Petri Nets

http://cadp.inria.fr/resources/vlpn

Nested-Unit Petri Nets (NUPNs)



- A new model for concurrency
- Extension of Petri nets with hierarchy and concurrent processes
- Easy to generate from higher-level textual languages

NUPNs in software competitions

- NUPNs make verification easier:
  - logarithmic reduction (-40%) in bits to encode markings
  - already implemented in 13 tools
- Two file formats for NUPNs:
  - .pnml XML-based format, with "toolspecific" section
  - .nupn text format (30 times smaller than PNML)
- The Model Checking Contest uses NUPNs
  $\Rightarrow$ 1/3 of MCC benchmarks are NUPNs
- The RERS challenge (parallel track) uses NUPNs
  $\Rightarrow$ benchmarks are either NUPNs or Promela

A collection of NUPN benchmarks

- VLPN:
  - collection under construction
  - 350 carefully chosen NUPNs among 13,000+ examples
  - generated from large (often industrial) examples
- Various sources:
  - CHP, EXP (communicating automata), FCR (Fiacre), LOT(OS), LNT, PIC(alculus), and MCC
- Organization:
  - the VLPN collection is divided into 8 groups

VLPN groups G1–5

VLPN group 1: nets containing redundant units							
vlpn_001	52 units	330 places	512 transitions	3–49–162	≥ 6.75209e+10 states	LOT	NUPN PNML PDF
vlpn_002	55 units	152 places	3213 transitions	8–28–91	1.52668e+14 states	LOT	NUPN PNML PDF
vlpn_003	69 units	176 places	134 transitions	16–35–91	1.13293e+13 states	LOT	NUPN PNML PDF
vlpn_004	72 units	218 places	215 transitions	5–62–143	≥ 9.14574e+09 states	LOT	NUPN PNML PDF
vlpn_005	81 units	199 places	219 transitions	19–51–120	2.46905e+12 states	LOT	NUPN PNML PDF
VLPN group 2: nets containing disconnected places or transitions							
vlpn_006	25 units	572 places	588 transitions	6–13–74	2.36483e+12 states	LOT	NUPN PNML PDF
vlpn_007	35 units	223 places	1088 transitions	18–18–75	3.29769e+13 states	LOT	NUPN PNML PDF
vlpn_008	37 units	644 places	660 transitions	7–19–104	3.85119e+14 states	LOT	NUPN PNML PDF
vlpn_009	200 units	199 places	699 transitions	––199	1.1418e+46 states	MCC	NUPN PNML PDF
vlpn_010	486 units	486 places	776 transitions	2–485–486	9.79474e+21 states	MCC	NUPN PNML PDF
VLPN group 3: unsafe nets							
vlpn_011	76 units	75 places	56105 transitions	––75	unknown state space	MCC	NUPN PNML PDF
vlpn_012	103 units	102 places	136662 transitions	––102	unknown state space	MCC	NUPN PNML PDF
vlpn_013	1409 units	1408 places	2400 transitions	––1408	unknown state space	MCC	NUPN PNML PDF
vlpn_014	2458 units	2457 places	5400 transitions	––2457	unknown state space	MCC	NUPN PNML PDF
vlpn_015	9154 units	9153 places	25110 transitions	––9153	unknown state space	MCC	NUPN PNML PDF
VLPN group 4: nets having one single unit							
vlpn_016	1 unit	2816 places	11520 transitions	1–1–12	2816 states	LOT	NUPN PNML PDF
vlpn_017	1 unit	5121 places	8961 transitions	1–1–13	5121 states	LOT	NUPN PNML PDF
vlpn_018	1 unit	6144 places	25600 transitions	1–1–13	6144 states	LOT	NUPN PNML PDF
vlpn_019	1 unit	11521 places	20225 transitions	1–1–14	11521 states	LOT	NUPN PNML PDF
vlpn_020	1 unit	25601 places	45057 transitions	1–1–15	25601 states	LOT	NUPN PNML PDF
VLPN group 5: trivial nets (i.e., having one more unit than the number of places)							
vlpn_021	118 units	117 places	176 transitions	––117	≥ 4.53408e+07 states	MCC	NUPN PNML PDF
vlpn_022	121 units	120 places	111160 transitions	––120	unknown state space	MCC	NUPN PNML PDF
vlpn_023	140 units	139 places	87 transitions	––139	≥ 4.07308e+07 states	MCC	NUPN PNML PDF
vlpn_024	150 units	149 places	1280 transitions	––149	≥ 1.5251e+11 states	MCC	NUPN PNML PDF
vlpn_025	175 units	174 places	318 transitions	––174	≥ 863546 states	MCC	NUPN PNML PDF
vlpn_026	195 units	194 places	2205 transitions	––194	unknown state space	MCC	NUPN PNML PDF
vlpn_027	207 units	206 places	325 transitions	––206	≥ 3.69099e+08 states	MCC	NUPN PNML PDF

- G1 to G4: collection of "corner" cases
- G5: "trivial" NUPNs (Petri nets without NUPN structure)

VLPN groups G6–G8

VLPN group 6: nets being communicating automata							
vlpn_079	3 units	2432 places	193936 transitions	1–2–21	786432 states	EXP	NUPN PNML PDF
vlpn_080	3 units	7636 places	13566727 transitions	1–2–25	unknown state space	EXP	NUPN PNML PDF
vlpn_081	3 units	12413 places	56777 transitions	1–2–25	1.452e+06 states	EXP	NUPN PNML PDF
vlpn_082	3 units	13664 places	16967720 transitions	1–2–26	unknown state space	EXP	NUPN PNML PDF
vlpn_083	5 units	1157 places	377368 transitions	1–4–33	1.22235e+06 states	EXP	NUPN PNML PDF
vlpn_084	6 units	4944 places	836184 transitions	1–5–44	unknown state space	EXP	NUPN PNML PDF

VLPN group 7: nets being pseudo-communicating automata (with a root unit containing the initial place)							
vlpn_106	6 units	116 places	3900 transitions	2–5–26	5.59263e+06 states	LNT	NUPN PNML PDF
vlpn_107	7 units	130 places	413 transitions	2–6–28	1.27121e+07 states	LOT	NUPN PNML PDF
vlpn_108	8 units	108 places	512 transitions	2–7–28	2.81183e+06 states	LNT	NUPN PNML PDF
vlpn_109	9 units	82 places	129 transitions	2–8–33	9.00532e+06 states	LOT	NUPN PNML PDF
vlpn_110	9 units	216 places	977 transitions	2–8–37	≥ 3.76759e+09 states	LNT	NUPN PNML PDF
vlpn_111	9 units	265 places	423 transitions	2–8–43	1.24613e+06 states	LOT	NUPN PNML PDF

VLPN group 8: nets combining concurrency and hierarchy (i.e., with a "genuine" NUPN structure)							
vlpn_195	9 units	1088 places	7363 transitions	5–5–45	≥ 1.01421e+08 states	LNT	NUPN PNML PDF
vlpn_196	11 units	353 places	717 transitions	6–6–42	≥ 4.23911e+08 states	LOT	NUPN PNML PDF
vlpn_197	11 units	550 places	1728 transitions	5–6–43	≥ 1.66852e+09 states	LNT	NUPN PNML PDF
vlpn_198	11 units	607 places	870 transitions	5–6–39	≥ 1.17931e+08 states	LOT	NUPN PNML PDF
vlpn_199	11 units	923 places	7197 transitions	5–6–44	≥ 1.59852e+09 states	PIC	NUPN PNML PDF
.....							
vlpn_343	579 units	2479 places	6124 transitions	30–340–1154	unknown state space	CHP	NUPN PNML PDF
vlpn_344	700 units	700 places	947 transitions	6–671–700	1.25971e+06 states	LOT	NUPN PNML PDF
vlpn_345	700 units	2357 places	3924 transitions	6–671–1372	unknown state space	LOT	NUPN PNML PDF
vlpn_346	739 units	3099 places	8124 transitions	26–440–1419	unknown state space	CHP	NUPN PNML PDF
vlpn_347	799 units	5276 places	6556 transitions	9–720–1552	unknown state space	LOT	NUPN PNML PDF
vlpn_348	1061 units	3016 places	2866 transitions	4–1005–2121	unknown state space	CHP	NUPN PNML PDF
vlpn_349	1107 units	3551 places	3012 transitions	77–598–1722	unknown state space	CHP	NUPN PNML PDF
vlpn_350	1197 units	7904 places	9825 transitions	10–1079–2324	unknown state space	LOT	NUPN PNML PDF

- G6: communicating automata
- G7: pseudo communicating automata
- G8: genuine NUPNs with hierarchy

3. MCC

Model Checking Contest

http://mcc.lip6.fr/models.php

The MCC collection of benchmarks

- Each benchmark is a Petri net encoded in PNML
  - P/T nets (33% being NUPNs) or colored nets
- A companion PDF file gives history and properties
- Support for scalable benchmarks of increasing size
- Already cited in 76 papers

Year	2011	2012	2013	2014	2015	2016	2017	2018
New models	7	12	9	15	13	11	10	13
All models	7	19	28	43	56	67	77	90
New instances, among which:	95	101	70	138	121	139	153	139
– *new colored nets*	43	37	24	33	27	9	16	0
– *new P/T nets*	52	64	46	105	94	130	137	139
– *new NUPNs (among P/T nets)*	0	0	1	5	15	62	64	77
All instances	95	196	266	404	525	664	817	956

A collection growing every year

Name	PN Type	Scale Parameter(s)	Data on the Model	form	PNML
ASLink	P/T	yes	**Places:** up to 4 410, **Transitions:** up to 5 405, **Arcs:** up to 16 377		
BusinessProcesses	P/T	Yes	**Places:** 782, **Transitions:** 697, **Arcs:** 2 011		
DLCflexbar	P/T	yes	**Places:** up to 47 560, **Transitions:** up to 76 160, **Arcs:** up to 216 499		
DiscoveryGPU	P/T	yes	**Places:** up to 436, **Transitions:** up to 464, **Arcs:** up to 1 214		
DoubleExponent	P/T	None	**Places:** up to 10 604, **Transitions:** up to 9 998, **Arcs:** up to 28 194		
EGFr	P/T	Yes	**Places:** up to 208, **Transitions:** up to 378, **Arcs:** up to 3 198		
HospitalTriage	P/T	None	**Places:** 228, **Transitions:** 680, **Arcs:**		
MAPKbis	P/T	Yes	**Places:** 106, **Transitions:** 173, **Arcs:** 986		
NQueens	P/T	Yes	**Places:** up to 1 080, **Transitions:** up to 900, **Arcs:** up to 4 500		
RERS17pb113	P/T	Yes	**Places:** 639, **Transitions:** 31 353, **Arcs:** 125 418		
RERS17pb114	P/T	Yes	**Places:** 1 446, **Transitions:** 151 085, **Arcs:** 604 252		
RERS17pb115	P/T	Yes	**Places:** 1 399, **Transitions:** 144 369, **Arcs:** 577 414		
RefineWMG	P/T	Yes	**Places:** up to 504, **Transitions:** up to 403, **Arcs:** up to 1 208		

4. REC

Rewrite Engines Competition

http://rec.gforge.inria.fr

The REC competitions

- A satellite event of WRLA *(Workshop on Rewriting Logic and Applications)*
- Past editions: 2006, 2008, 2010, 2018
- Focus on term rewriting problems
- 4 different categories:
  - unconditional and conditional term rewrite systems
  - rewriting modulo equations and modulo strategies
- Benchmarks expressed in a generic language REC
  - 2 versions: REC-2008 and REC-2017
  - translators from REC to the tools' input languages

Simple REC-2017 benchmark

```
REC-SPEC simple
SORTS % abstract data domains
    Bool Nat
CONS      % primitive operations
    true : -> Bool
    false : -> Bool
    zero : -> Nat
    succ : Nat -> Nat
OPNS          % defined functions
    and : Bool Bool -> Bool
    plus : Nat Nat -> Nat
```

```
VARS                % free variables
  A B : Bool
  M N : Nat
RULES       % function definitions
  and (A, B) -> B  if A -><- true
  and (A, B) -> false if A -><- false
  plus (zero, N) -> N
  plus (succ (M),N) -> succ (plus
(M,N))
EVAL     % terms to be evaluated
  and (true, false)
  plus (succ (zero), succ (zero))
END-SPEC
```

18 tools involved in REC competitions

language (tool)	web site	REC1	REC2	REC3	REC4
ASF+SDF	http://www.meta-environment.org	×	×	×	
CafeOBJ	http://cafeobj.org				×
Clean	http://clean.cs.ru.nl				×
Haskell (GHC)	http://www.haskell.org				×
LNT (CADP)	http://cadp.inria.fr				×
Lotos (CADP)	http://cadp.inria.fr				×
Maude	http://maude.cs.illinois.edu	×	×	×	×
mCRL2	http://www.mcrl2.org				×
OCaml	http://www.ocaml.org				×
Opal (OCS)	http://github.com/TU-Berlin/opal				×
Rascal	http://www.rascal-mpl.org				×
Scala	http://www.scala-lang.org				×
SML (MLton)	http://www.mlton.org				×
SML (SML/NJ)	http://www.smlnj.org				×
Stratego/XT	http://www.metaborg.org		×	×	×
TermWare	http://gradsoft.ua/index_eng.html		×		
Tom	http://tom.loria.fr		×	×	×
TXL	http://txl.ca			×	

Evolution of REC benchmarks

- Growing number of benchmarks
- Evolving REC language (versions 2008 and 2017)
- Support for scalable benchmarks of increasing size
  - 8-bit, 16-bit, 32-bit binary adders and multipliers
  - MAA cryptographic algorithm (13 sorts, 684 rewrite rules)

category	REC1	REC2	REC3	REC4
source language	tool-specific	REC-2008	REC-2008	REC-2017
unconditional term rewrite systems	(5) 7	(5) 12	(7) 26	(19) 43
conditional term rewrite systems	(9) 25	(8) 18	(6) 17	(24) 42
rewriting modulo equations	(4) 9	(4) 6	(4) 6	(0) 0
rewriting modulo strategies	(0) 0	(1) 1	(1) 3	(0) 0
TOTAL	(18) 41	(18) 37	(18) 52	(43) 85

5. MARS

Models for Analysis of Real Systems

http://mars-workshop.org/repository.html

The MARS workshops

- A new series of events in computer science
- Past workhops: 2015, 2017 (ETAPS), 2018 (ETAPS)
- Focus on case studies done using formal methods
  - 12-page papers published in the EPTCS journal (arXiv)
  - each paper comes with source files
  - Creative Commons 4.0 license *(Attribution, NonCommercial, ShareAlike)*
- Repository of case studies (joint work with Peter Hoefner)
  - multiple languages allowed for the same case study
  - multiple versions supported (SVN on INRIA Gforge)

Current MARS repository: 21 models

	Description	Formalism(s)/Tool(s)	Event
021	Distributed Integrated Modular Avionics (DIMA)	Uppaal	MARS'18
020	CBTC Automatic Train Supervision System	UMC; Promela (Spin); NuSMV/nuXmv; mCRL2; CPN tools; FDR4; ProB; LNT (CADP); TLA toolbox; ProB; Uppaal; ProB	MARS'18
019	Tera-Scale ARchitecture (TSAR)	Promela (Spin); Divine (DiVine, ITS-tools); ITS-tools	MARS'18
018	TLS Handshake Model	LNT (CADP)	MARS'18
017	Robotic Cell Injection	Discrete-Time Markov Chain (PRISM)	MARS'17
016	Production Cell	LNT (CADP); LOTOS (CADP)	MARS'17
015	A Model-Derivation Framework for Software Analysis	Uppaal	MARS'17
014	Emergency Power Supply	BDMP(KB3)	MARS'17
013	AUTOSAR	Erlang QuickCheck	MARS'17
012	Message Authenticator Algorithm	REC tool; AProVE tool; Clean; Haskell; LNT (CADP); LOTOS (CADP); Maude; mCRL2; OCaml; Opal; Rascal; Scala; Standard ML; Stratego/XT; Tom; LOTOS (CADP); LNT (CADP)	MARS'17 MARS'18
011	IEEE 802.15.4 TCSH MAC	mCRL2 tool suite	MARS'17
010	Memory Access and Interrupts	Isabelle/HOL	MARS'17
009	B.A.T.M.A.N.	Uppaal	MARS'17
008	Fragmentation and Reassembly for CAN	AWN	MARS'17
007	Stream Control Transmission Protocol	Uppaal	MARS'17
006	DES Standard	LNT (CADP); LOTOS (CADP)	MARS'15
005	Bitcoin Protocol	Uppaal	MARS'15
004	Caches and Pipelines	Uppaal	MARS'15
003	Self-Balancing Unicycles	SpaceEx; Mathematica	MARS'15
002	eChronos	Isabelle/HOL	MARS'15
001	Bilby File System	Isabelle/HOL	MARS'15

Conclusion

Conclusion

- Large collections of benchmarks raise specific issues
- Taxonomy of benchmarks
  - produced by humans vs. randomly generated
  - anonymous/scrambled (VLTS, VLPN) vs. with an history (MARS, MCC, REC)
  - normal-size files (MARS) vs. large-size files (VLTS, MCC)
- Taxonomy of collections
  - homogeneous (VLTS, MCC) vs. heterogeneous (MARS)
  - sorted (VLTS, VLPN) vs. unsorted (MCC, REC, MARS)
  - stable (VLTS) vs. growing (MCC, REC, MARS)
  - encoded in fixed language vs. evolving language (REC)