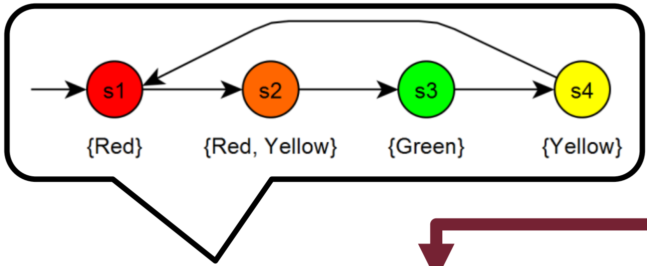
A Preliminary Analysis on the Effect of Randomness in a CEGAR Framework

Ákos Hajdu[1,2], Zoltán Micskei[1]

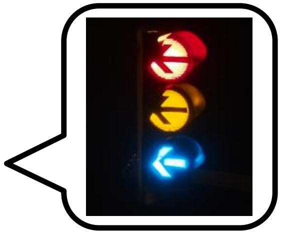
[1]Budapest University of Technology and Economics,
Department of Measurement and Information Systems
[2]MTA-BME Lendület Cyber-Physical Systems Research Group

25th Minisymposium of DMIS, 29.01.2018.

Background – Formal verification



Real-life system



Formal model

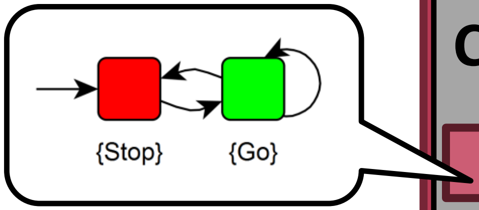
Formal requirement

¬(Red ∧ Green)

Verification: explore states

CEGAR

Counterexample



Abstraction

Refinement

Refutation

Safe

Counterexample

Motivation

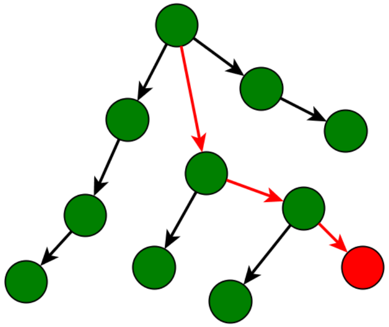
- Theta verification framework
  - Abstraction refinement-based algorithms
  - Easy development, evaluation, combination
  - Many strategies and configurations
  - Open source: github.com/FTSRG/theta

Θ

- Strategies are becoming more advanced
  - Difficult to evaluate performance of a configuration
  - Performance influenced by unintentional factors

The experiment

- Randomize some external factors
  - Search strategy
    - Good configuration may be guided in a bad direction



  - Variable naming
    - Algorithms rely on logic (SMT)
    - Affect their order in collections
    - Influence inner heuristics of solvers

$A \land (\lnot B \lor C)$

⇓

$(X \lor \lnot Y) \land Z$

Variables of the experiment

- Input variables: model
  - Category (hw, sw (locks, eca, ssh), plc)
  - Model name

- Input variables: configuration
  - Domain of abstraction (Pred./Expl.)
  - Refinement strategy (Binary/Sequence itp.)
  - Randomized factor (Search/Variables/Deterministic)

  → 2 x 2 x 3 = 12 configurations

Variables of the experiment

- Output metrics
  - Did the configuration terminate successfully
  - Execution time
  - Number of refinement iterations
  - Size of the ARG (Abstract Reachability Graph)
  - Depth of the ARG
  - Length of the counterexample (cex)

Measurement procedure

- 30 input models
  - 10 hardware (benchmarks from HWMCC)
  - 15 software (benchmarks from SV-COMP)
  - 5 PLCs (from CERN)
- 12 algorithm configurations
- Repeated 30 times with different seeds
- Timeout 180s

→ 10 800 runs, 7 080 successful, ~10 days CPU time

→ Raw data, analysis scripts and detailed report is available at doi.org/10.5281/zenodo.1117853

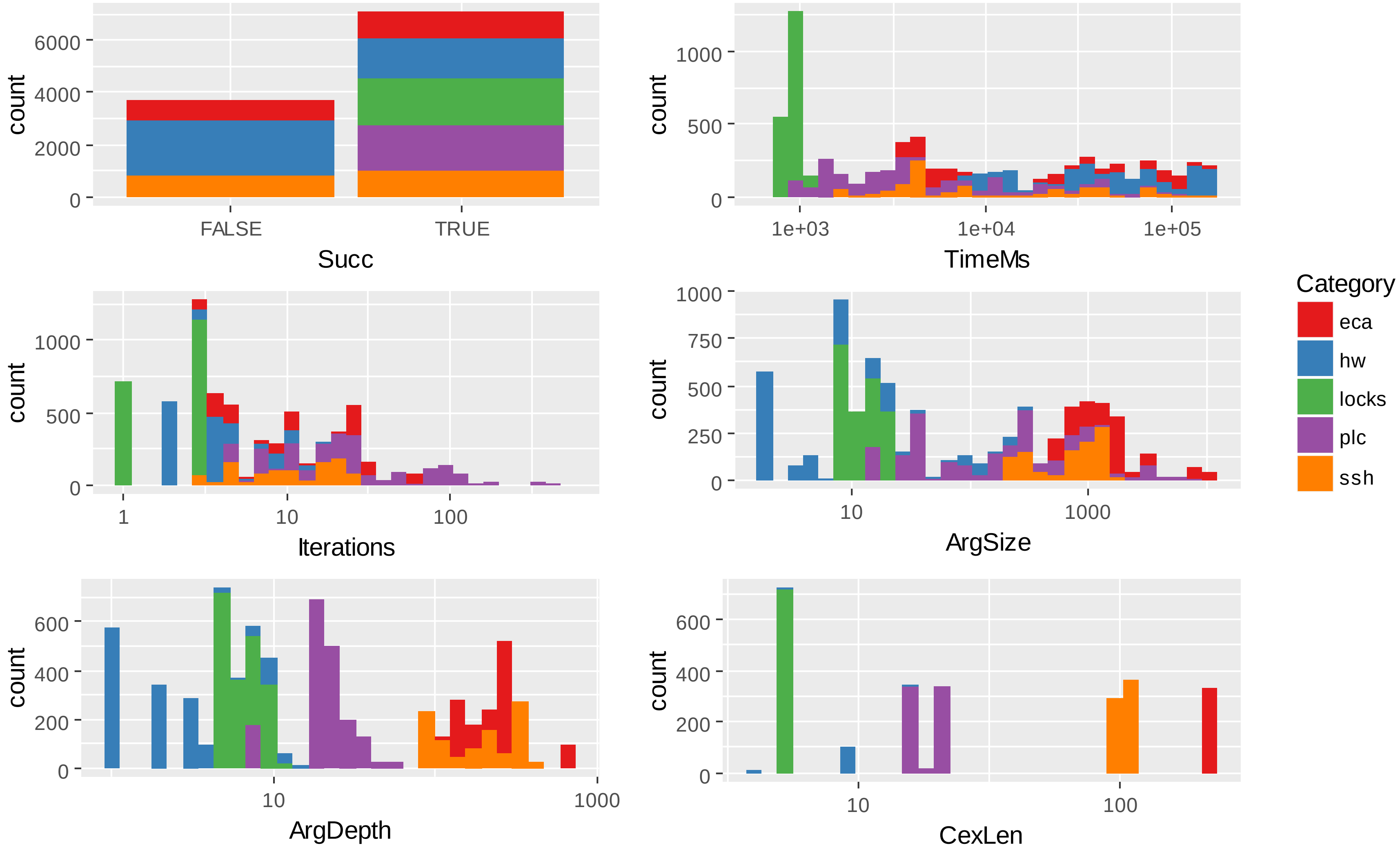
Research questions

- **RQ1**: success rates of randomized configurations
  - Can randomized configurations verify models that the deterministic ones cannot?
- **RQ2**: variation of output metrics
  - How does randomization affect variation?
  - Which yields greater variation (search/variable names)?

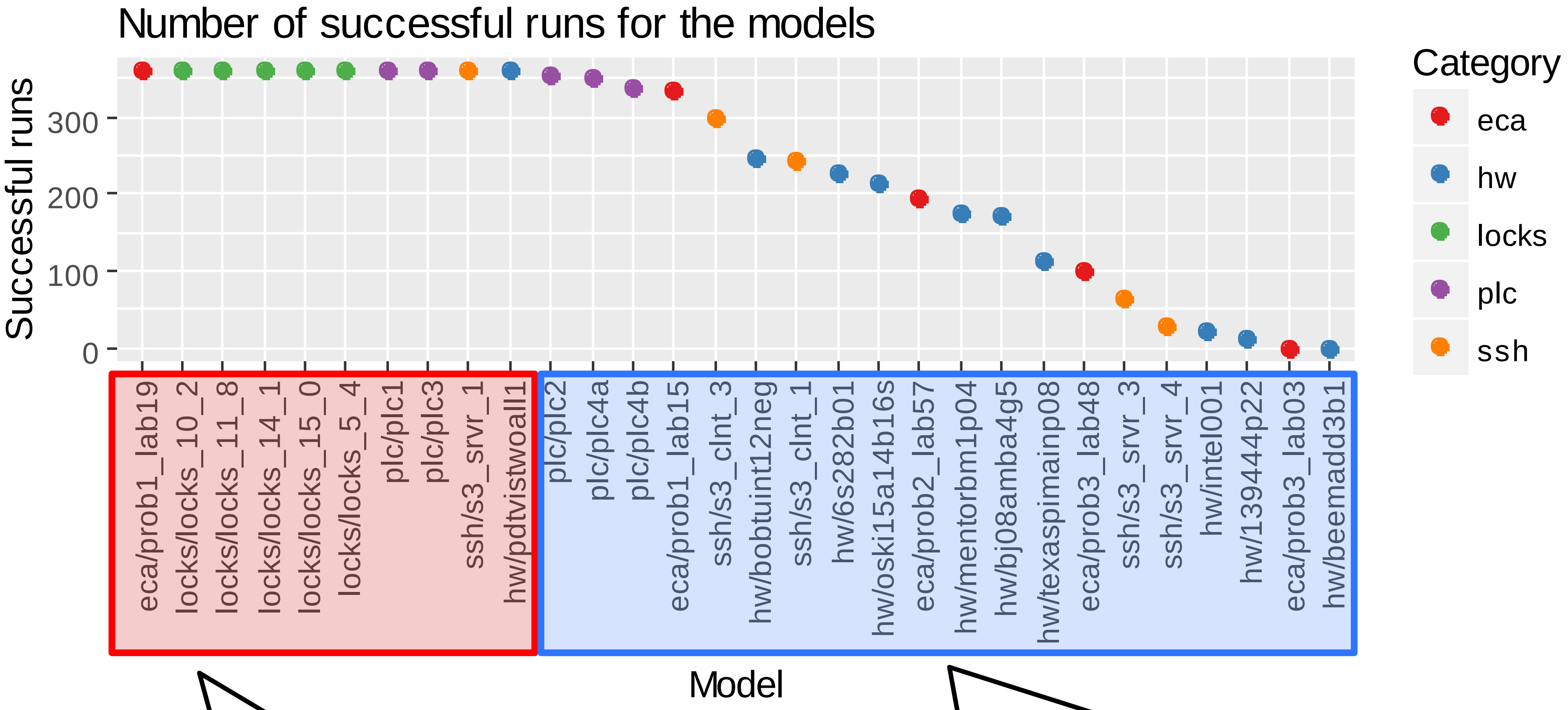
- Validity
  - External: representative input models
  - Internal: repetitions, dedicated machines

Overview

Distribution and range of output metrics



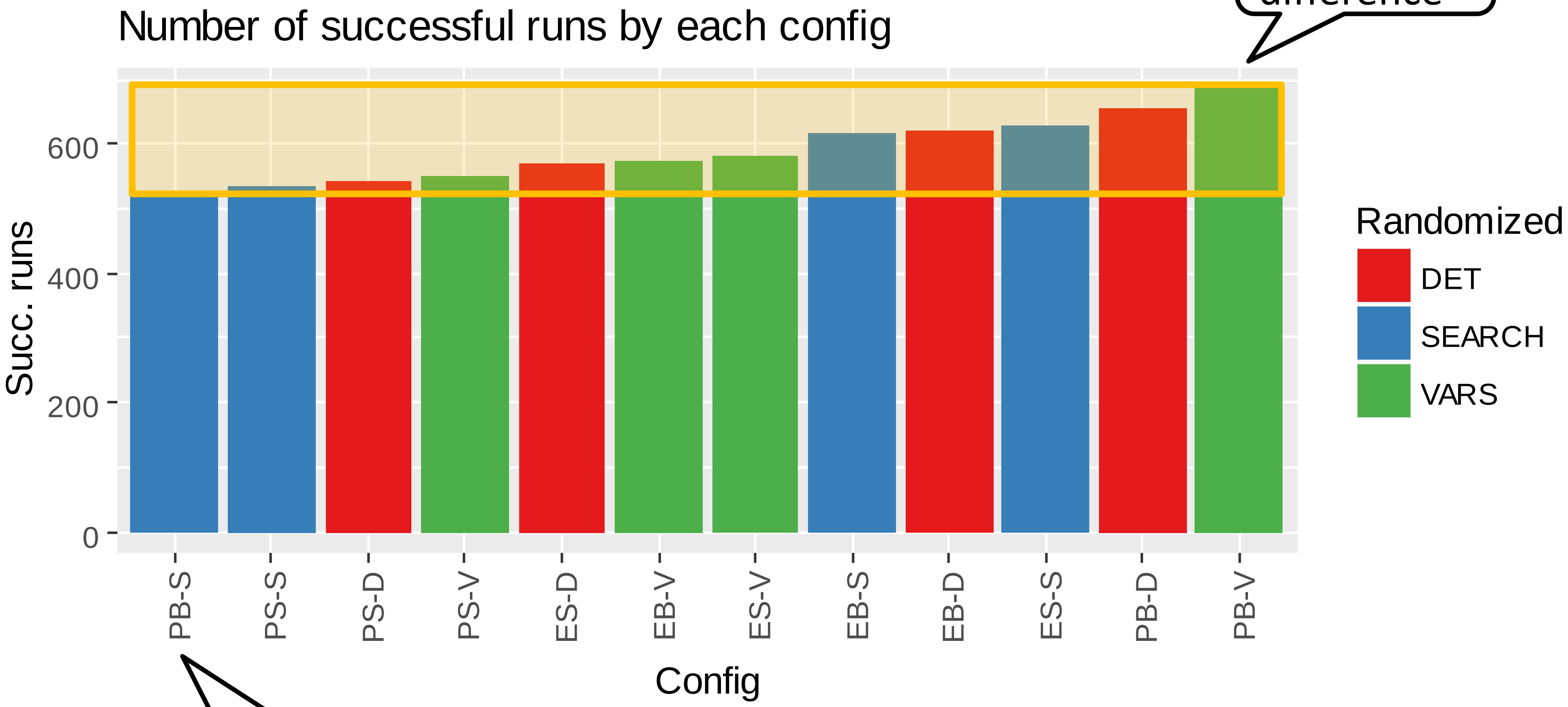
Overview


Number of successful runs for the models

Easy models

Gradually increasing difficulty

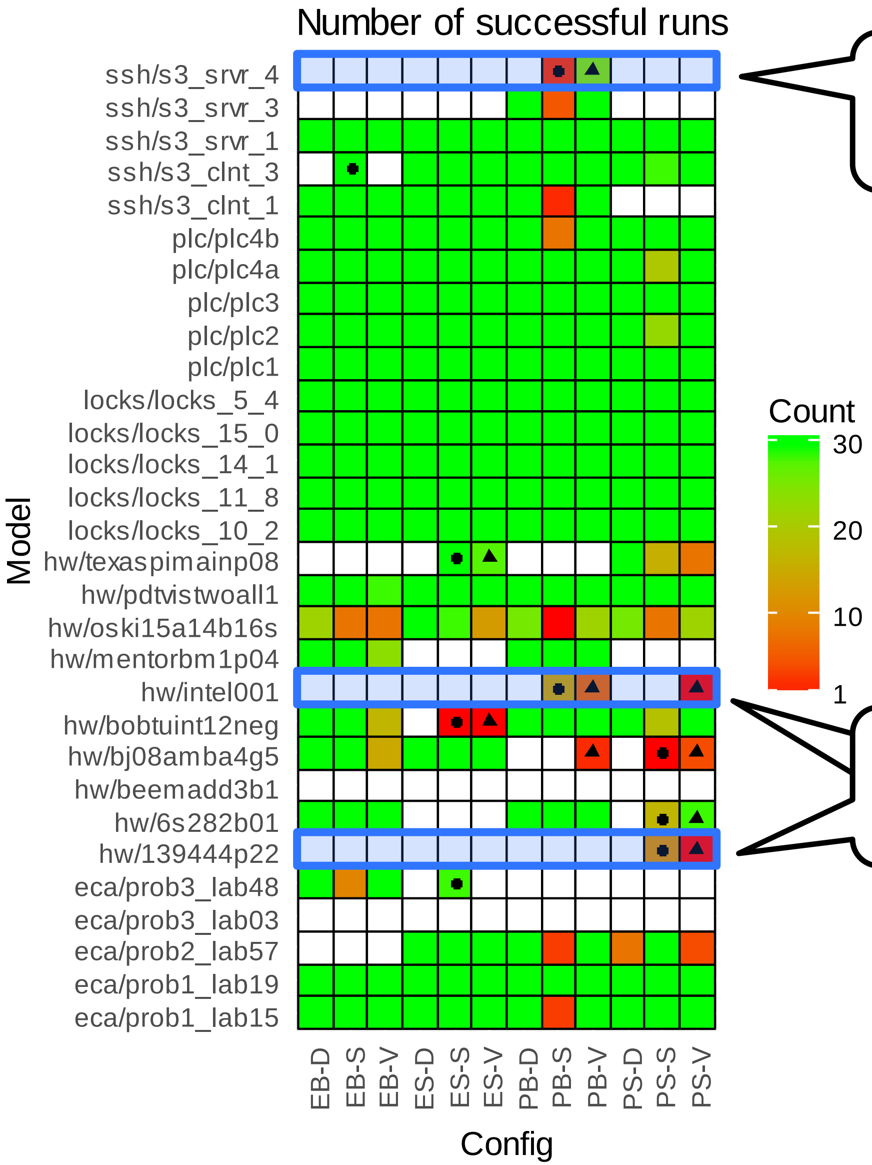
Overview



Not a huge difference

Pred.
Binary itp.
Search

RQ1: success rates



Only verified by randomized

Only verified by randomized

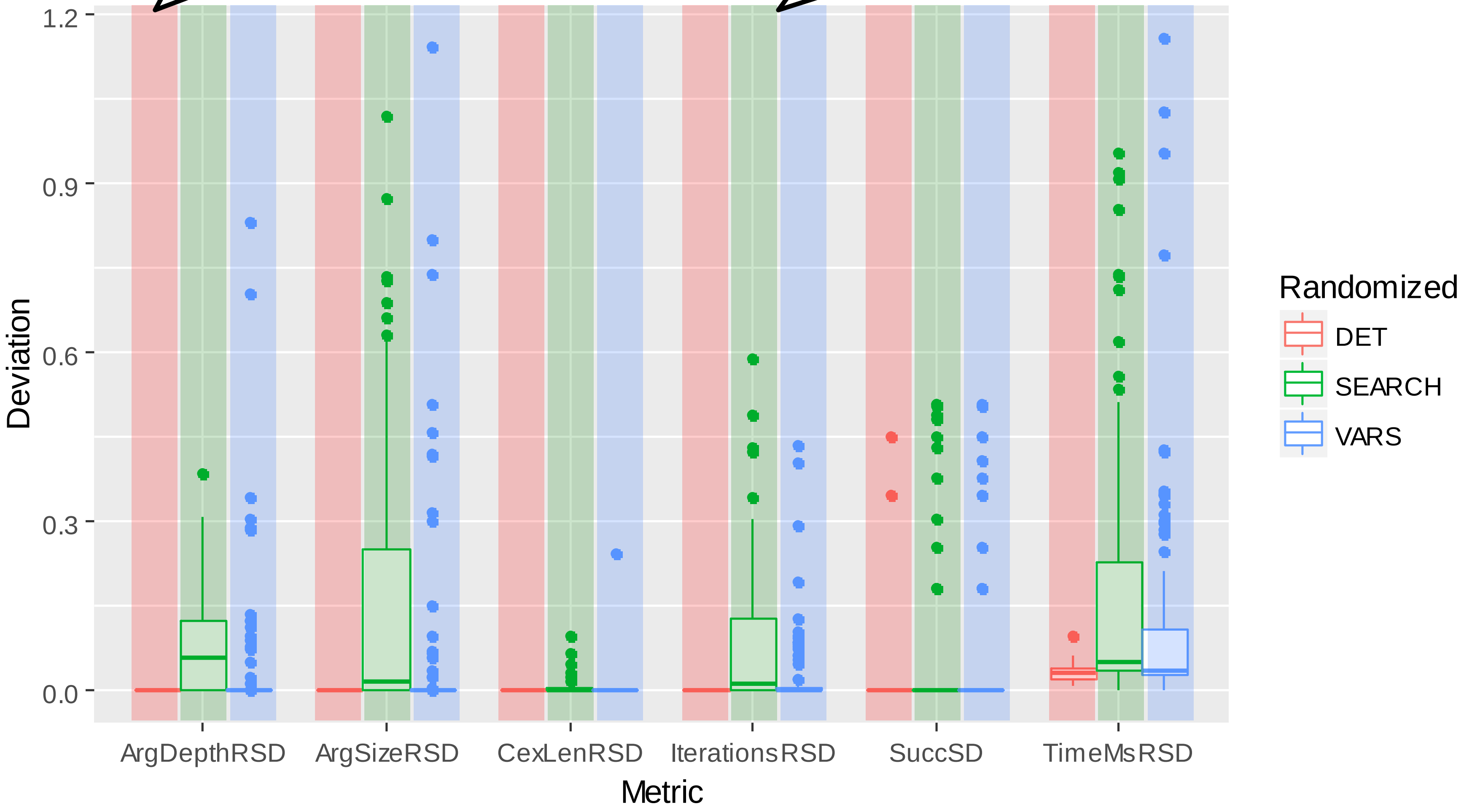
RQ1: success rates

- Analysis of individual cases
  - 1394444p22
    - Large model, 8600 variables, formula with size 1.6x10^5
    - Deterministic fails to prove infeasibility of a counterexample
    - Randomized quickly finds feasible one in some runs
  - intel001
    - Counterexample refutation formulas become large (4.4x10^4)
    - Randomized can find smaller refutations
  - s3_srvr_4
    - Deterministic runs slightly out of time
    - Unnecessary refutation formulas discovered

RQ2: variations

Zero or small variation

Greater variation for random search



Lessons learned & plans

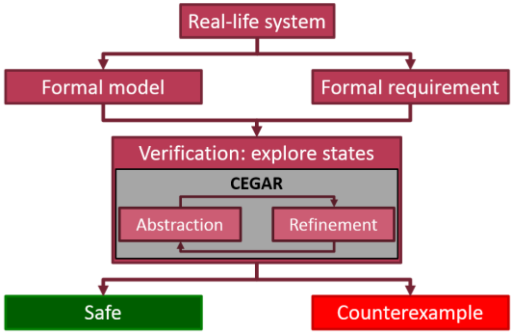
- Lessons learned
  - Randomized factors introduce great variations
  - Might even influence the success of verification

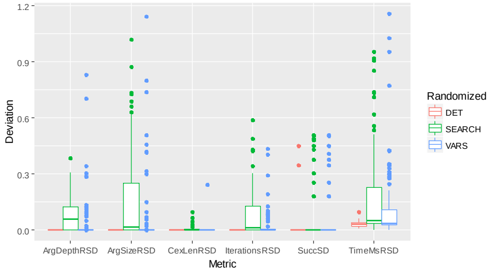
- Plans
  - Consider randomized strategies as viable option
    - E.g., random search besides BFS/DFS
  - Improve shortcomings of the algorithm
    - E.g., consider multiple counterexamples for refinement

Conclusions

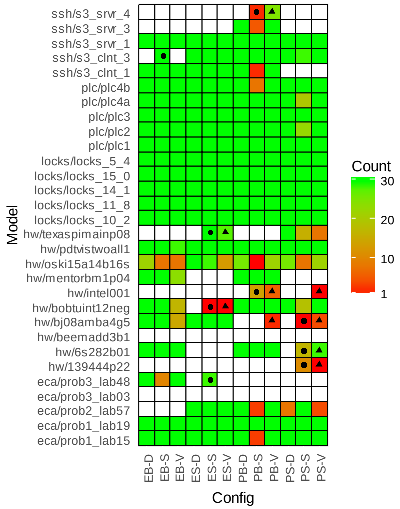
- Theta framework
  - Abstraction refinement-based algorithms
  - Various configurations
- Randomized search/variable names
  - Great variation in the output metrics
  - Influence the success of verification
  - Analysis of specific cases
- Future work
  - More models and repetitions
  - More factors to randomize

→ inf.mit.bme.hu/en/members/hajdua





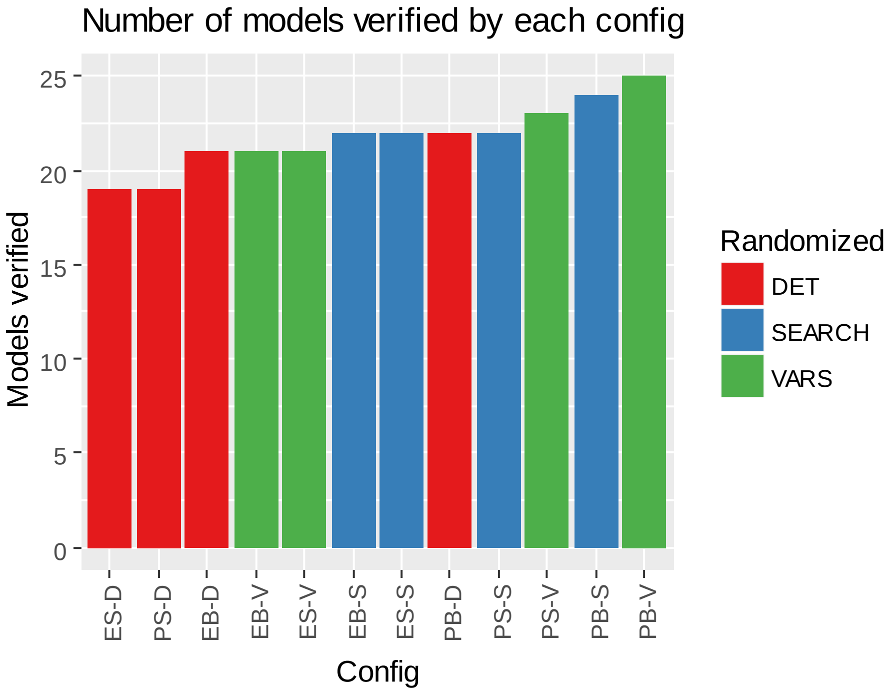
Number of successful runs

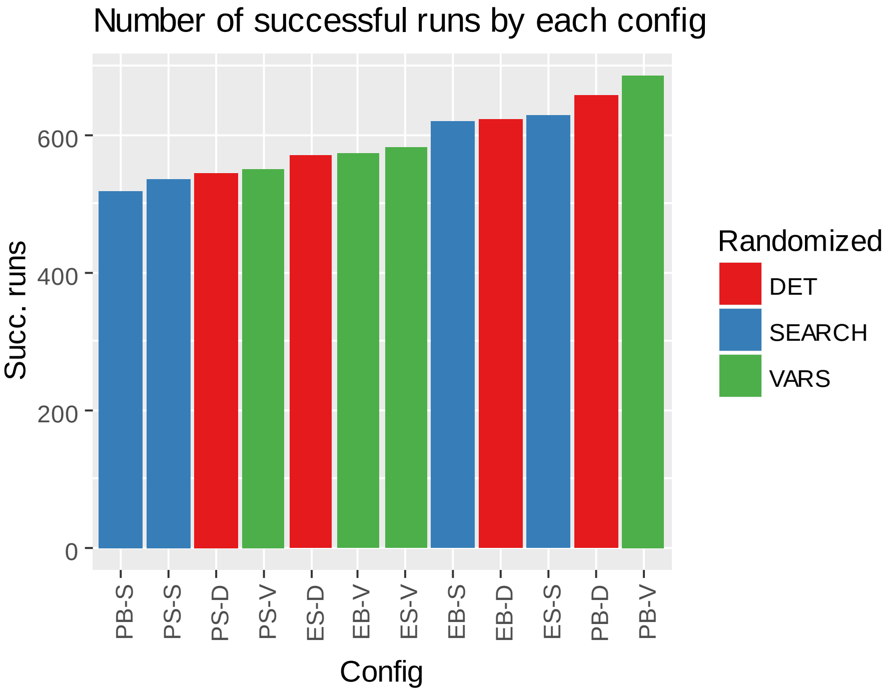


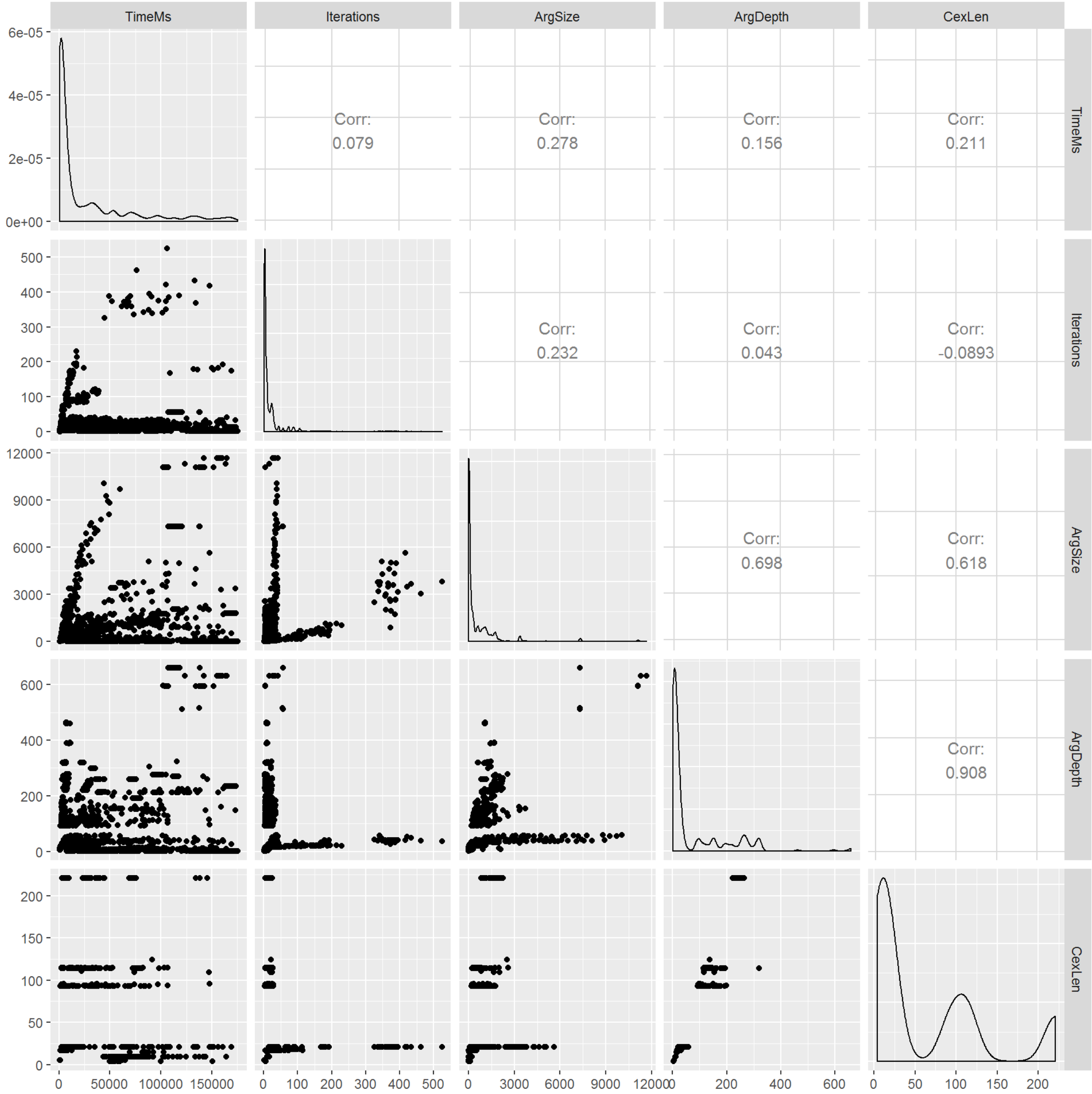
- Models

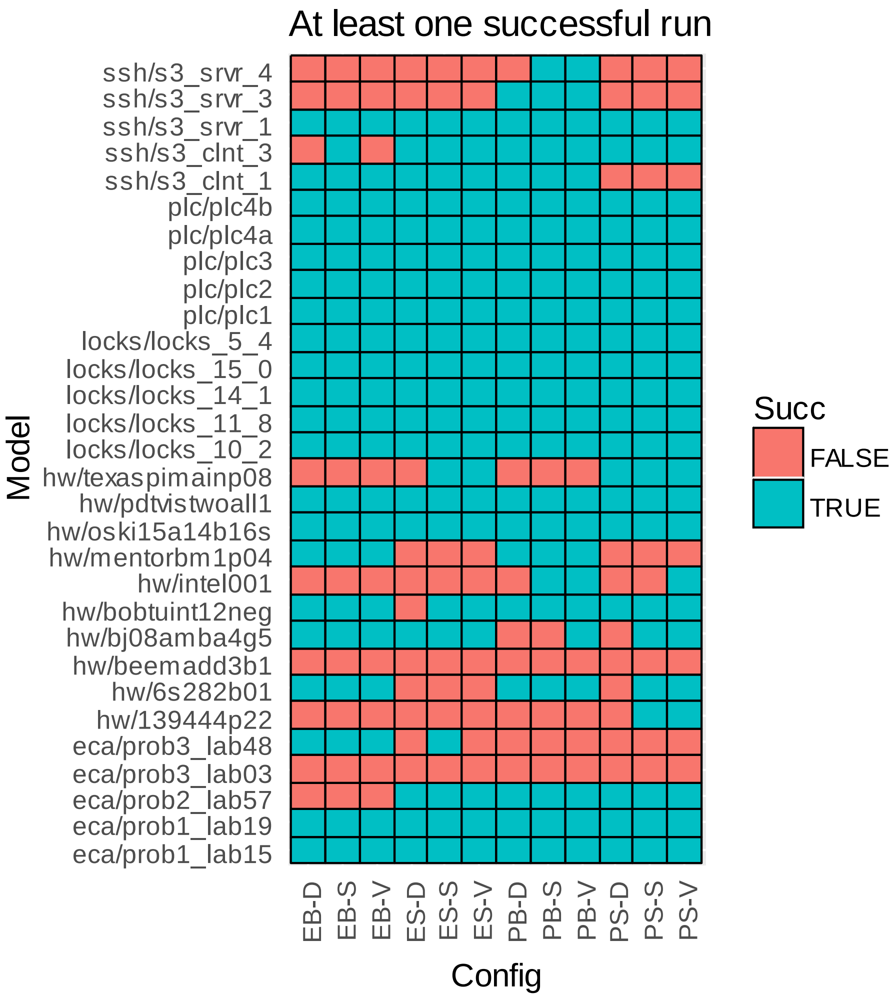
Model	Inputs	Latches	AndGates
hw/139444p22	244	322	5549
hw/6s282b01	44	637	3185
hw/beemadd3b1	60	56	876
hw/bj08amba4g5	11	33	13585
hw/bobtuint12neg	212	207	1937
hw/intel001	31	23	240
hw/mentorbm1p04	100	2373	17508
hw/oski15a14b16s	1023	3451	33367
hw/pdtvistwoall1	6	31	725
hw/texaspimainp08	14	42	1955

Model	Locs	Edges	Assigns	Assumes	Havocs
eca/prob1_lab15	317	393	236	156	1
eca/prob1_lab19	322	403	236	166	1
eca/prob2_lab57	312	408	211	196	1
eca/prob3_lab03	1261	1436	1081	354	1
eca/prob3_lab48	1280	1474	1081	392	1
locks/locks_10_2	26	36	10	24	2
locks/locks_11_8	16	21	5	14	2
locks/locks_14_1	9	10	2	6	2
locks/locks_15_0	9	10	2	6	2
locks/locks_5_4	13	16	4	10	2
plc/plc1	66	70	51	11	8
plc/plc2	175	196	135	40	21
plc/plc3	175	196	135	40	21
plc/plc4a	175	196	135	40	21
plc/plc4b	175	196	135	40	21
ssh/s3_clnt_1	187	262	79	154	29
ssh/s3_clnt_3	193	268	85	154	29
ssh/s3_srvr_1	233	323	102	184	37
ssh/s3_srvr_3	230	320	100	184	36
ssh/s3_srvr_4	230	320	100	184	36

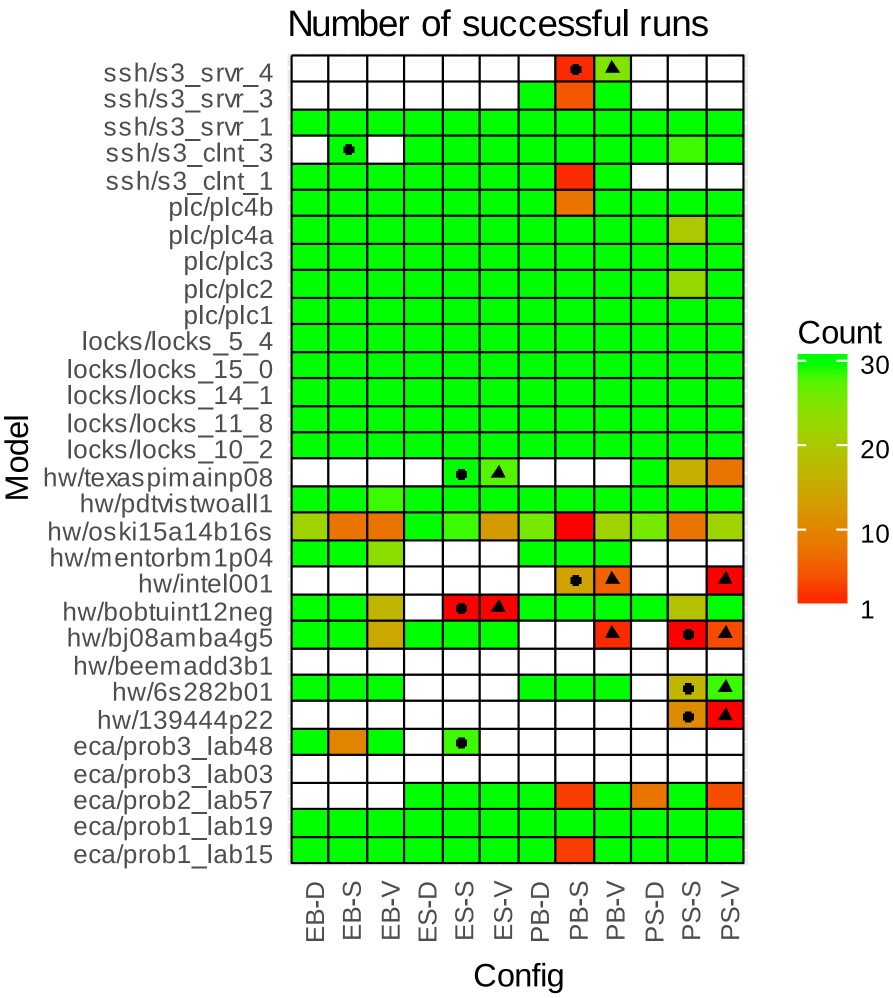

Number of models verified by each config
Models verified
25
20
15
10
5
0
ES-D
PS-D
EB-D
EB-V
ES-V
EB-S
ES-S
PB-D
PS-S
PS-V
PB-S
PB-V
Config
Randomized
DET
SEARCH
VARS


Number of successful runs by each config
Succ. runs
600
400
200
0
PB-S
PS-S
PS-D
PS-V
ES-D
EB-V
ES-V
EB-S
EB-D
ES-S
PB-D
PB-V
Config
Randomized
DET
SEARCH
VARS

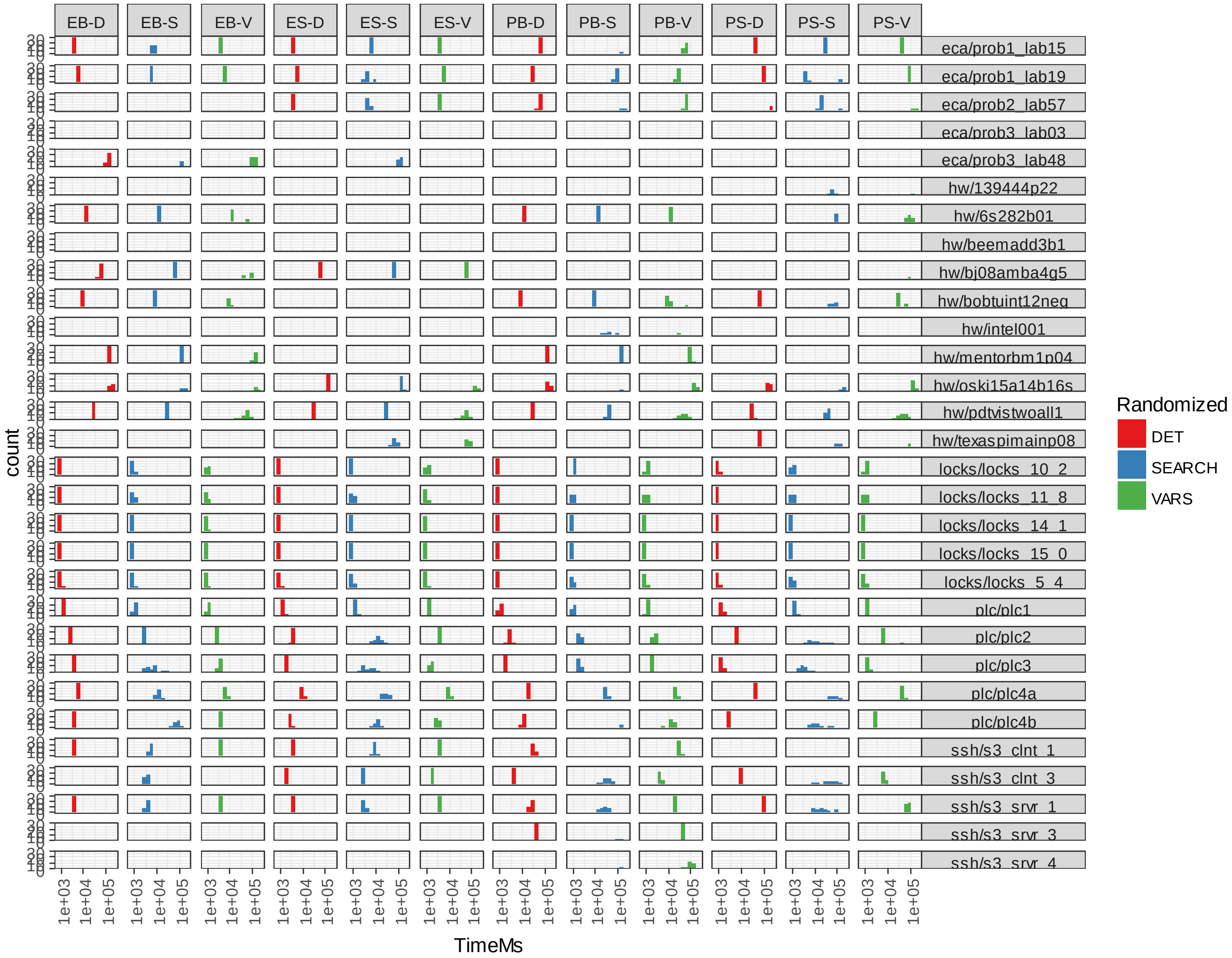

TimeMs
Iterations
ArgSize
ArgDepth
CexLen
6e-05
4e-05
2e-05
0e+00
Corr:
0.079
Corr:
0.278
Corr:
0.156
Corr:
0.211
TimeMs
500
400
300
200
100
0
Corr:
0.232
Corr:
0.043
Corr:
-0.0893
Iterations
12000
9000
6000
3000
0
Corr:
0.698
Corr:
0.618
ArgSize
600
400
200
0
Corr:
0.908
ArgDepth
200
150
100
50
0
CexLen
0
50000
100000
150000
0
100
200
300
400
500
0
3000
6000
9000
12000
0
200
400
600
0
50
100
150
200



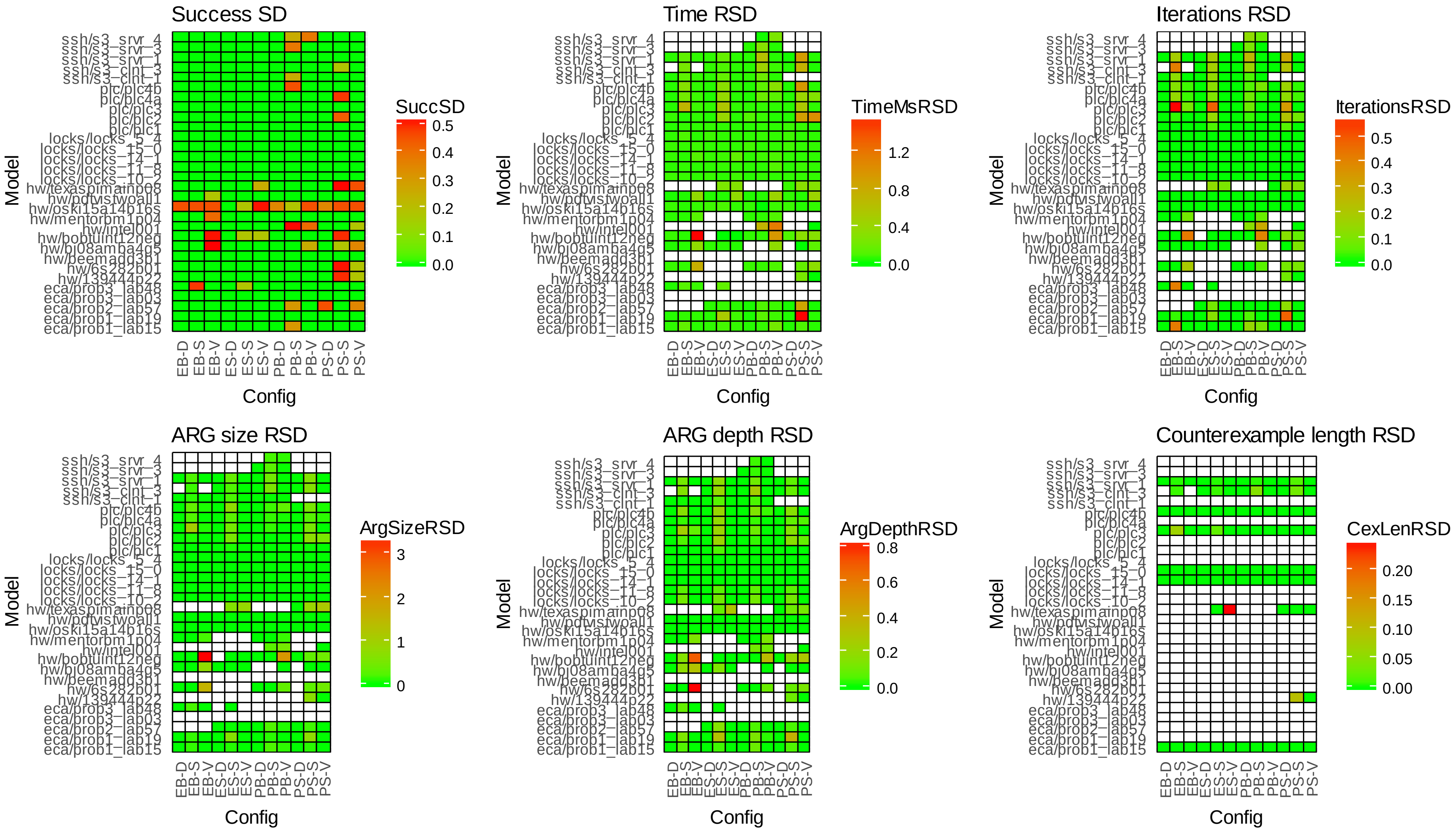
At least one successful run

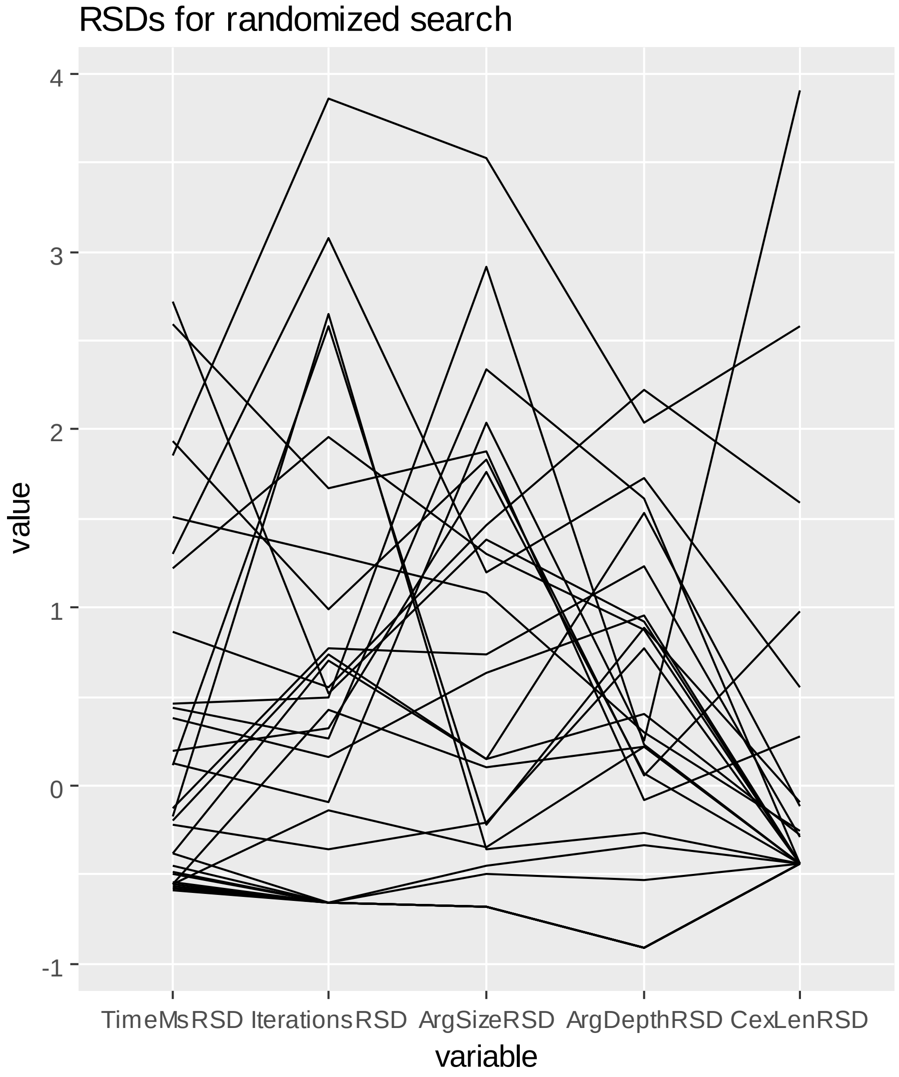


Number of successful runs

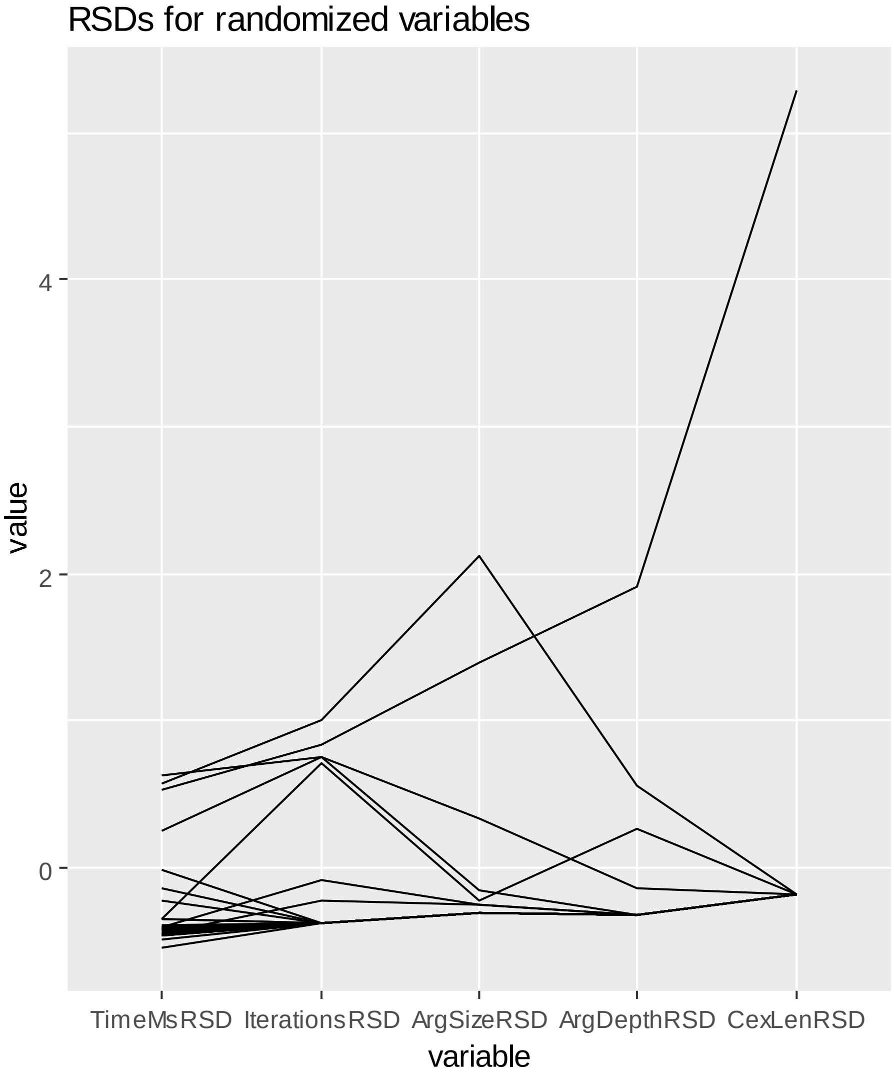


Distributions of execution time


Success SD
SuccSD
0.5
0.4
0.3
0.2
0.1
0.0
Time RSD
TimeMsRSD
1.2
0.8
0.4
0.0
Iterations RSD
IterationsRSD
0.5
0.4
0.3
0.2
0.1
0.0
ARG size RSD
ArgSizeRSD
3
2
1
0
ARG depth RSD
ArgDepthRSD
0.8
0.6
0.4
0.2
0.0
Counterexample length RSD
CexLenRSD
0.20
0.15
0.10
0.05
0.00
Model
ssh/s3_srvr_4
ssh/s3_srvr_3
ssh/s3_srvr_1
ssh/s3_clnt_3
ssh/s3_clnt_1
plc/plc4b
plc/plc4a
plc/plc3
plc/plc2
plc/plc1
locks/locks_5_4
locks/locks_15_0
locks/locks_14_1
locks/locks_11_8
locks/locks_10_2
hw/texaspimainp08
hw/pdtvisfwoall1
hw/oski15a14b16s
hw/mentorbm1p04
hw/intel001
hw/bobtuint12neg
hw/bj08amba4g5
hw/beemaddd3b1
hw/6s282b01
hw/139444p22
eca/prob3_lab48
eca/prob3_lab03
eca/prob2_lab57
eca/prob1_lab19
eca/prob1_lab15
EB-D
EB-S
EB-V
ES-D
ES-S
ES-V
PB-D
PB-S
PB-V
PS-D
PS-S
PS-V
Config

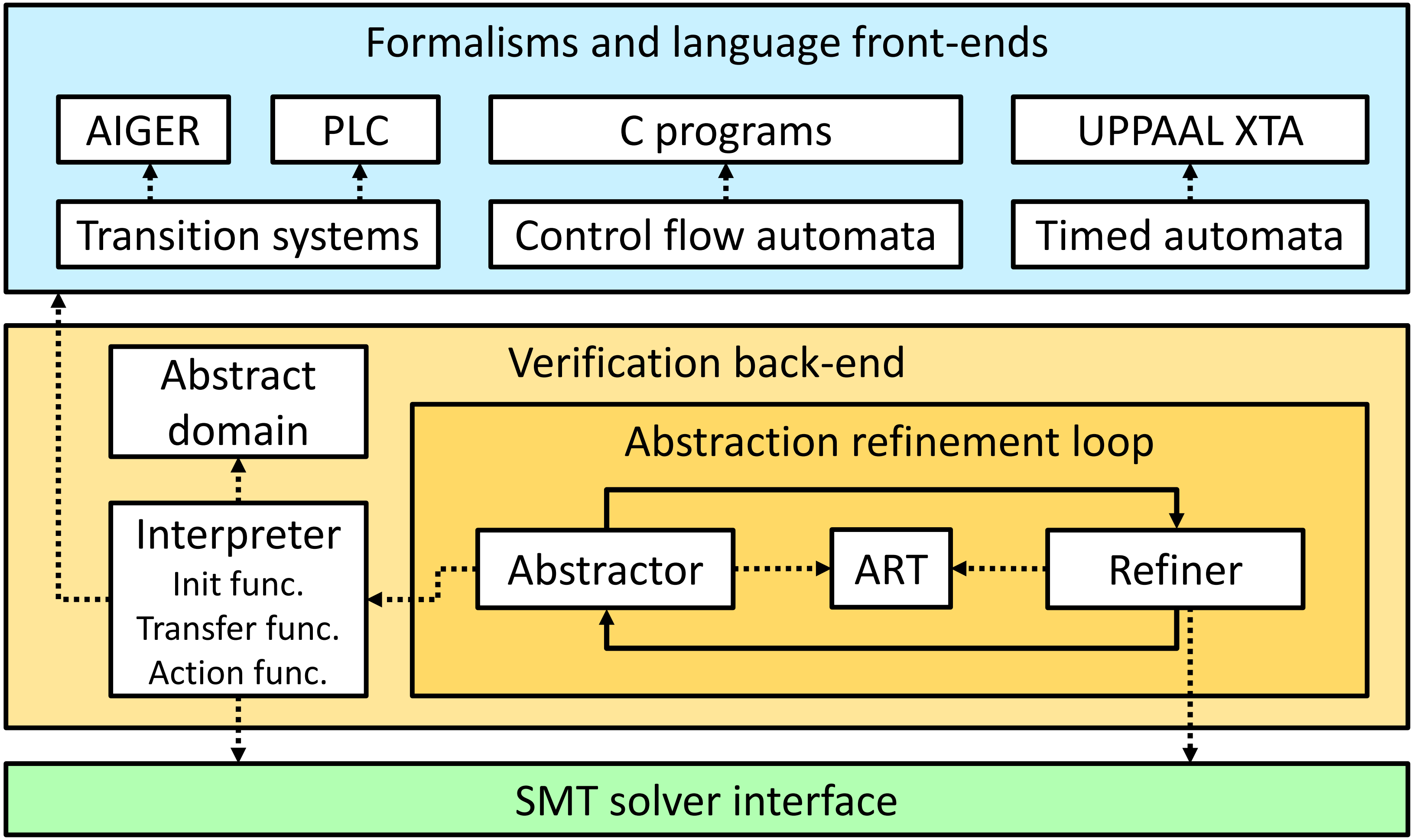


RSDs for randomized search



RSDs for randomized variables

- Architecture of Theta


Formalisms and language front-ends
AIGER
PLC
C programs
UPPAAL XTA
Transition systems
Control flow automata
Timed automata
Verification back-end
Abstract domain
Interpreter
Init func.
Transfer func.
Action func.
Abstraction refinement loop
Abstractor
ART
Refiner
SMT solver interface