RERS 2018 – Parallel CTL track

Towards the solution of problems 101,102,103

Franco Mazzanti

ISTI - CNR

Pisa, Italy

WHY-1

- **We have a family of model checkers (KandISTI), developed “in house”, targeting verification of CTL-like properties.**

  KandISTI/FMC specifications are based on a simple process algebra (CCS/CSP like), and it is very simple to translate the .dot designs into FMC specifications.

  FMC is still “experimental”, useful feedback expected.

WHAT-1

- **Automatic translation from DOT design into FMC/LNT/nuXmv specifications**

  DIVERSITY allows to check the correctness of the translation.
  (test fragments of the problems should generate LTS of exactly the same size)

- **Manual translation of properties**

  Almost immediate in the case of FMC / CADP
  (but errors are still possible)

  Rather complex in the case of nuXmv
  missing weak until in the logics, complexity introduced by the need to handle also the verification of finite paths.

WHY-2

- **We are interested in experimenting DIVERSITY in formal verification:**

CADP
Event based framework, of industry-ready maturity,
allowing imperative style LNT specifications,
supporting alternation-free mu-calculus (and much more),
allowing efficient on the fly verification,
compositional verification, partial model checking,
very powerful set of LTS manipulation features (SVL).

nuXmv (state based framework, industry-ready maturity,
allowing verification of CTL/LTL properties,
based on symbolic (BDD) and SMT based verification techniques.

Problem 101 Properties #21 #22 #23

- **Small problem size: just 118.584 states**

FMC #21: AG [a21][a23][a4][true] false **result: FALSE**

CADP #21: AG (([A21] [A23] [A4][true] false) **result: FALSE**

NuXmv #21: **result: FALSE**

AG ((last=21) -> (AX ((last=23) -> (AX (last=4) -> (AX FALSE)))))

(just for infinite paths)

!E[final=0 U EX (last=21 & final=0 &
EX (last=23 & final=0 &
EX (last=4 & final=0 &
EX final=0)))] *(including finite paths)*

All three problems easily (exhaustively) verified with all the three frameworks

#21 FALSE, #22 FALSE #23 TRUE

Problem 102   Property #22

FMC: EG [a35] E[([a23] false) U (<a35> true)]   **result: FALSE**

Already the initial state can perform an a35 action, after which
E[([a23] false) U (<a35> true)] does not hold.
Counter-example found after observing just 20000 states (dfs traversal).

Early attempts to deduce the validity of the formula without full system model checking led to ***WRONG*** *conclusions!*

CADP: EG (["A35"] EU([("A23"] false),(<"A35"> true))   **result: FALSE**

The full LTS generated for property #21 has been reused.

nuXmv: killed *after 12 hours* ...

Problem 102   Property #23

FMC: AG [a22] A[([a8] false) U (<a22> true)]   **result: FALSE**

Counter-example generated after the analysis of just 706 states.

CADP: AG (["A35"] AU([("A8"] false),(<"A22"> true))   **result: FALSE**

Counter-example generated

nuXmv: *unable to build the full statespace in a reasonable time*

Problem 103 Property #21

FMC:

Model too big,
FMC not able to find a response with the available reasources.

CADP:

Model too big for plain verification
CADP functionalities not fully exploited before the RERS deadline.

AG((["A11"] AW(["A2"] false, <"A6"> true)) implies
(["A11"] AW(["A5"] false,<"A6"> true)))

*After the deadline, several appraches taking advantage of problem decomposion, divergence sensitive branching minimizations and/or partial model checking approaches suggest a **TRUE** result*

nuXmv: *not tried*

Problems 103 Properties #22 #23

FMC:

#22 result: TRUE
#23 result: FALSE

CADP:

#22 result: TRUE
#23 result: FALSE

Both CADP and FMC, with just their on-the-fly approach can easily find the result (and show the counter-example/proof) without any particular strategy.

nuXmv: *not tried*

Problem 102 Property #21

FMC: EF(AG([a5] false)) **result: TRUE**

(not necessary to generate the full statespace to check the property)

EF FINAL *(lucky shortcut!)*

CADP: *full statespace generated for further uses* **result: TRUE**

(273.103.932 states / 2.507.025.655 trans)

nuXmv: *skipped …*

Conclusions

ON THE FLY (model generation + evaluation) is OK

EXPLICIT is not BAD (when onthe fly)

DIVERSITY is GOOD (for trustness and best feature selection)

OUT-OF-THE BOX reasoning sometimes helps (but dangerous).

COMPOSITIONAL/PARTIAL model checking can be a silver bullet.

BRUTE-FORCE approaches for really BIG systems require
extreme knowledge of the framework details.
(naive uses of symbolic approaches not successful)