COMMENTS TO THE AUTHOR:

Reviewer #1: I would like to thank the authors for carefully taking into account my previous comments. The running example is now more clear and the detailed descriptions dispersed across examples 1,2,3 and 4 make the paper more readable.

However, the paper still needs a carefully pass for typos, especially in the newly added text. E.g., :

 \* page 6, lines 25-29 have multiple typos (spaces are also missing)

 \* page 27 line 20: compsitionality

[Corrected.]

Here are some additional comments/questions:

1. I am not sure I fully understand the pruning mechanism used in Algorithm 1. I am referring to step 11 (filterUnsatTransitions). According to page 11, lines 10-11: all the unsatisfiable transitions are pruned before the generation of the first open automaton. But then again in page 13 lines 27-28: you write that "while building the open automaton, …., it is not logically incorrect to keep potentially unsatisfiable transitions". My understanding is that the removal of unsatisfiable transitions happens before the generation of the first open automaton in step 12 in Algorithm 1.  Am I missing something? Also, do I understand correctly that after the pruning of unsatisfiable transitions, any non-final generated open automata are not minimum since they might contain unreachable transitions? Also, how do you prune states?

2. Can you provide the sub algorithms for steps 12, and 13 of Algorithm 1?

[We add two new algorithms (Algorithm 3 and 4) to describe the process of makeReachableSubAutomaton() and refineReachableSubAutomaton() in Algorithm 1.]

3. Page 21, line 50: how does the described modification enforce the same properties since the sets of acceptable behaviors between the two architectures are different?

4. Page 24, line 17: why/how did you pick 3 out of the 6 possible reachable global states?

[After pruning the unsatisfiable transitions, we figure out the states those who can be reached through satisfiable transitions.]

5. Please add reference 3 in the conclusions section (page 27, lines 19-20), since the system safety and liveness results are not a contribution of the current paper, and in this particular paragraph you are summarizing the paper results.

[Added.]

Reviewer #3: The authors took into account the comments and suggestions on the previous version of the paper, improving the structure of the paper and providing more examples of the notions and constructions introduced. Overall, this improved the paper, which is now much more readable.

The style of exposition still mixes formal and informal content (e.g., the definition of some notions is informally given in the text, while some definitions include also an informal explanation of what is going to be defined). I would have avoided this mixture as much as possible, especially in such a technically complex setting. However, this could well be also a matter of taste.

All in all, I think that the paper is now acceptable. I have some minor comments listed below, which I would ask the authors to take into account in the preparation of the final version.

MINOR COMMENTS

\*\*\*\* LITERATURE:

I think that the literature about symbolic approaches to the specification and verification of transition systems could be better accounted for.

For instance, concerning LTS generation and bisimilarity checking one could include the following papers:

- K.G. Larsen and L. Xinxin. Compositionality through an operational semantics of contexts, ICALP'90, LNCS 443, 1990

- A. Rensink. Bisimilarity of open terms. Information and Computation, 156(1-2), 2000.

- P. Baldan, A. Bracciali, R. Bruni. Bisimulation by unification. AMAST 2002, LNCS 2422, 2002

Also all the line of work about deriving bisimulation congruences, started with Leifer and Milner essentially falls in this setting, and could be referenced.

J.J. Leifer, R. Milner. Deriving bisimulation congruences for reactive systems. Proc. CONCUR 2000, LNCS 1877, 2000.

\*\*\*\* TYPOS and MINOR THINGS

page 3, line 12: in the course -> during the execution of?

[Modified according to the advice.]

page 3, line 14: the following sentence sounds strange:

"and show the correctness of the optimised algorithm w.r.t. the original one."

I guess that the algorithm is correct (and complete) and the "w.r.t. to the original one" part refers to the way things are proven, i.e., by showing that the two algorithms produce the same results. Is this right? If so, please rephrase.

[It means the result of the “smart” version is equivalent to the original version. Rephrased.]

pages 5/6

I am not sure it is a good idea to have "2.2 Running example: the Enable operator" as a separate section. Moreover, at page 6, the first sentence "... we give two possible pNet encodings of the "Enable" operator ..."  seems to contradict the one immediately before, i.e., "In the next section we will define pNet encodings of the Enable operator.”

[We encode the example in the state-oriented style and the data-oriented style which are both feasible.]

page 7, line 20: I think that parametrised actions should be defined (the term is discussed only informally in text, as far as I can see)

page 11, line 10 : naturally -> clearly?

[Modified.]

page 13, line 42: I would avoid references to pages (here and after)

[All the \pageref are removed.]

page 20:, line 18: I cannot properly understand the sentence

"This translation ensures that no runtime error will occur in the SMT engine.”

[The translation is defined to ensure the correctness of the inputs.]

page 22, line 32: synchrons  -> synchronous

page 26, line 46: ".. before submitting them to the Z3 solver." -> "and we submit them to the Z3 solver”?

[Modified according to the advice.]

page 27, line 5:

"the reachable computation" -> "the computation of the reachable states”?

[Modified according to the advice.]

page 27, line 20 "compsitionality"

[Corrected.]