05-Oct-2011

Dear Dr. Shen:

Manuscript ID TCAD-2011-0407 entitled "Inferring Assertion for Complementary Synthesis" which you submitted to the Transactions on Computer-Aided Design of Integrated Circuits and Systems, has been reviewed. The comments of the reviewer(s) are included at the bottom of this letter.

The reviewer(s) have requested that your paper be shortened to a transactions brief. I invite you to respond to their comments and revise your manuscript.

To upload your revised manuscript, log into http://mc.manuscriptcentral.com/tcad and enter your Author Center, where you will find your manuscript title listed under "Manuscripts with Decisions." Under "Actions," click on "Create a Revision." Your manuscript number has been appended to denote a revision.

You may also click the link below to start the revision process (or continue the process if you have already started your revision) for your manuscript.
If you use this link you will not be required to login to ScholarOne Manuscripts.

http://mc.manuscriptcentral.com/tcad?URL_MASK=jGm4kt98NCSyhfsGhD3x

You will be unable to make your revisions on the originally submitted version of the manuscript. Instead, revise your manuscript using a word processing program and save it on your computer. Please also highlight the changes to your manuscript within the document by using bold, underlined, or colored text.

Once the revised manuscript is prepared, you can upload it and submit it through your Author Center.

When submitting your revised manuscript, you will be able to respond to the comments made by the reviewer(s) in the space provided. You can use this space to document any changes you make to the original manuscript. In order to expedite the processing of the revised manuscript, please be as specific as possible in your response to the reviewer(s).

IMPORTANT: Your original files are available to you when you upload your revised manuscript.
Please delete any redundant files before completing the submission.

Because we are trying to facilitate timely publication of manuscripts submitted to the Transactions on Computer-Aided Design of Integrated Circuits and Systems, your revised manuscript should be submitted by 09-Nov-2011. If it is not possible for you to submit your revision by this date, we may have to consider your paper as a new submission.

Once again, thank you for submitting your manuscript to the Transactions on Computer-Aided Design of Integrated Circuits and Systems and I look forward to receiving your revision.

Sincerely,
Prof. Sachin Sapatnekar
Editor-in-Chief, Transactions on Computer-Aided Design of Integrated Circuits and Systems
tcad@umn.edu

Reviewer(s)' Comments to Author:

Reviewer: 1
Comments to the Author
The contribution makes sense, and the paper is structured well.
The authors should check the work in
Ed Clarke, Reconsidering Cegar: learning good abstractions without refinement
(and related publications) and discuss any connections that exist - conceptual
or technical -- to their work.

REPLY TO REVIEWER: Thank you for your advice. But our algorithm does not have any connection to abstraction, because it does not construct abstract transition relation or remove state variables.

The material added to the ICCAD paper seems significant.

The writing includes occasional glitches in terms of grammar and style.

! So, we propose
 Therefore, we propose
% Do not use "So" as "Therefore" in scholarly publications. It is too colloquial.

REPLY TO REVIEWER: Thank you for your advice. I have changed them.

! that reversely computes
 that inverts
% OR
 that computes an inverse of
% OR "the inverse", if it is unique

REPLY TO REVIEWER: Thank you for your advice. I have changed it to a more appropriate statement: "that uniquely determines".

> To help the user selects
  To help the user select
REPLY TO REVIEWER: Thank you for your advice. I have changed them.

On page 1 line 11, line 18, line 47 (left column) and page 5, line 3, there are missing ligatures "fi"in "configuration", "final" and "defined" -- this may be an issue with PDF processing or cut-and-paste, but needs to be resolved one way or another. Also, I am not giving a full list of these problems -- the authors must carefully check the paper. There's also a spurious spare character in "assertio n" in the abstract. If the authors can't write a clean abstract, this paper can definitely NOT be published.

REPLY TO REVIEWER: Thank you for your advice. I will try my best to avoid this. I use acrobat, evnince and xpdf to view my paper, but can not find such problem in the version downloaded from TCAD website. So if you find such problem again, can you forward it to me through the associate editor? Thank you again.

! One of the most difficult jobs in designing
  One of the most difficult tasks in designing

REPLY TO REVIEWER: Thank you for your advice. I have changed it.

! Manually specifying an assertion needs the user
! to read lots of documentations and try many combinations
 To manually specify an assertion, the user must read extensive documentation and often perform laborious trial-and-error process.
% note that "documentation" is uncountable in English, so there's no plural form

REPLY TO REVIEWER: Thank you for your advice. I have changed it.

% I am not going to proofread the entire paper - it's the authors' job.

REPLY TO REVIEWER: Thank you for your work. I will proofread it more carefully.

Reviewer: 2
Comments to the Author
My main objections are o the problem you are solving and to methods you have chosen.

1) First of all, reading a manual is not such a bad idea. Of course if you had an undocumented chip and wanted to find its working configuration the methods you describe make a perfect sense.
However, even if replacing a lost manual is a tough problem it hardly merits a scientific publication. (Safe-cracking is not easy, but nobody writes papers about it.)

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my motivation more clearly.
According to my working experiment on complementary synthesis, reading documentation and find out the correct combination is a very tedious work.
Please refer to page 1, right column paragraph 4, which starts with "For example", and the subsection V.C. They describe my experience on XFI encoder with 120 configuration pins. Finding the meaning and correct combination of them take me a very long time. But under the guidance of the precondition formulas inferred by our new algorithm, I only need to find out the meaning of two pins instead of 120 pins. Obviously this can significantly save the human effort.

2) But let us assume that indeed the problem at hand is to save the user's time

I still have some doubts about the methods you employ. The largest assertion among the four examples you give in the experimental section has 6 variables. I assume that any reasonably good user should be able to find the correct configuration by making a sensible assignment to 6 variables that satisfies this assertion. If necessary this user may take a look at the manual to find out what these 6 variables specify.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my motivation more clearly.
Yes, you are right, these assertions has very small number of variables. But if you refer to the last row of table 1, you can find that these benchmarks have 16, 26 or even 120 configuration pins. Therefore, the ability to infer an assertion with small number of variables from large number of configuration pins indicates exactly the power of our algorithm.
Under the guidance of these inferred formulas, the user can focus on those variables mentioned by these formulas, which can significantly save their time.

Instead, you generate all possible encoders (that have decoders) for the configurations satisfying this simple assertion and then let the user pick the right encoder based on the assertion specifying the configurations corresponding to this encoder. Note that the user still has to be able to distinguish between wrong assignments to those 6 variables.
It beats me, why it cannot be done without generating all possible encoders.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my motivation more clearly.
Yes, you are right, reading the documentation can not be totally avoided. But with the formulas inferred by our algorithm, the user can focus on only 6, instead of 120 variables. That is a significantly saving.
Actually, although they mention up to 6 variables, the user don't need to inspect them all. They only need to inspect those with different values. According to subsection V.C, no more than two pins are needed to be inspected.

3) Of course, hypothetically, one can have a very complex assertion eliminating wrong configurations that involves many variables. But then, it will be hard for the user to manually identify the correct configuration due to shear complexity of this communication chip. Besides, in this case the number of encoders may be very large.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea more clearly.
Yes, you are right, so many configuration pins (up to 120) may corresponds to many many many encoders, but most of them do not have corresponding decoders, which will be ruled out by the first step of our algorithm.

4) One more technical comment. You build an assertion excluding wrong configurations iteratively repeating two steps:
a) Find an C counter-example proving that under particular assignment of configuration variables no decoder exists.

b) Then you build an interpolant to increase the set of excluded configurations.

What you do at step b) is frequently used in algorithms of quantifier elimination based on enumerating satisfying assignments.
There are more than a few papers on this topic, for example "M.Ganai, A.Gupta, and P. Ashar, "Efficient SAT-based unbounded symbolic model checking using circuit cofactoring", in Proc. ICCAD-2004, pp.510-517.

This paper describes a trick (circuit cofactoring) with the same objective you have. Namely, one satisfying assignment is used to remove many more.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea more clearly.
Yes, you are right, that paper describes exactly what we do in our paper, except that we use craig interpolant to characterize the formula.
So we remove the detail description and refer the reader to our ICCAD' 11 paper.
At the same time we add a citation to the cofactoring paper to show the connection between it and ours.

Reviewer: 3
Comments to the Author
This paper addresses the problem of computing assertions in complementary circuit synthesis. The computation consists of three phases: 1) computing assertion configurations, 2) computing the set of Boolean relations, and 3) computing the assertion configuration of each Boolean relation. The problem formulation and its corresponding computation have practical applications.

The presentation in several ways is not clear and should be improved.

1. Boolean relation and its defined function(s) are never explicitly defined. Please provide the definitions in Section III with proper references.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea more clearly.
Yes, you are right. As my paper has been shorten to 5 pages, I don' t have the space to define it separately, so I change the statement a little in place. Please refer to page 3 left column, section IV.A, the two paragraphs in bold font are changed according to your advice.

2. As Theorem 2 is just a slight extension of [16], the appended proof can be completely removed by simply referring the reader to [16].

REPLY TO REVIEWER: Thank you for your advice. I have changed it.

3. In the discussion of Section IV.C, the o variables seem to be incorrectly omitted. For example, f depends on o, o should be asserted in Equation (7), etc. Please correct related formulations. Otherwise the interpolation may be incorrect.
REPLY TO REVIEWER: Thank you for your advice.
Yes, you are right, the o variables are omitted intentionally. Because these o variables are the output of the transition relations, their values are uniquely determined by the

input and configuration variables. So they do not need to be asserted.
On the other hand, asserting these o variables may significantly narrow down the size of space covered by the interpolation, and then slow down our algorithm.
So we chose to omit o variables. Of course, our paper has be shorten to 5 pages, so we can only remove this subsection and refer the reader to our iccad 11 paper.

4. In Section IV.C, computing f using Craig interpolation may seem unnecessary as cofactoring f already yields a desired solution. Having Craig interpolation in the computation may seem just the matter of speedup as stated in the last sentence of Section IV. If this is the case, please provide experimental evaluation comparing the computation with and without Craig interpolation.

REPLY TO REVIEWER: Thank you for your advice.
Yes, you are right, the formula obtained after cofactoring is already the set of enlarged invalid configuration letter. But it contains the whole relation F_{LN}, which is often very large. So we use interpolant to characterize it and then use BDD to minimize it. After that, their sizes are significantly reduced. You can find out this by looking at the inferred assertion shown in subsection V.B.
I am sorry that I do not have enough space to compare the computation with and without interpolation, because our paper has been shorten to 5 pages.

5. The presentation of Section V.A seems to hint that the function defined by R is unique. Please comment on this.

REPLY TO REVIEWER: Thank you for your advice.
Yes, you are right, the function defined by R is unique. Because R can uniquely determine Y from c and X, which means it defines a mapping from c and X to Y, which is actually a function. Please refer to subsection IV.A, the 4th paragraph in bold font.

6. For Section VI, there are no additional experiments compared to [2]. Essentially only Phase 1 is experimented. Please experiments on Phase 2 and Phase 3 computations as well. Showing statistics of the computed Boolean relations and their assertions (in terms of formula size, CPU time, etc.) provides the reader a better idea how the proposed algorithms work in practice.

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea more clearly.
Actually the paper [2] only includes the phase 1, while the phase 2 and 3 are newly added in this paper.

7. Please compare the computed assertions of [2] and this work. It seems to the reviewer that the assertions of XGXS, PCI-E, scrambler, and T2 ethernet in [2] and this work are functionally the same, whereas those of XFI are functionally different. Please explain why the computed assertions can differ.
REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not checked it carefully.
For the assertion inferred for XFI, I compare the following three versions
1) ICCAD11

2) TCAD initial submitted version
3) The assertion inferred by current tool implementation
These three are all different from each other in syntax, but the 1) and 3) are equal, while the 2) is not. I think this is caused by preparing the TCAD initial submitted version with latex \texttt, which requires me to replace _ with _, & with \&, and also need to insert space in proper position to break too long symbol to two lines. In this process, it is very likely that I delete an ! unintentionally. I am really sorry about it.
But in this revised version, I update all experimental result, including these assertions, with the result generated by new tool implementation, which correct some bugs in previous implementation. So if you find any further inconsistency, it may be caused by the bug in the old implementation. But I still welcome your comment, because it may give me another opportunity to fix further bugs.
Please refer to www.ssypub.org to download the source code of new implementation, and try it to verify my experimental result.

8. XFI and scrambler have 2 decoders each. Please list their respective two assertions, rather than just one each.
REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea clearly.
Actually, each encoder has one inferred assertion, which represents the cases with decoders.
But for each decoder, they only have a precondition formula, which has been shown in subsection V.C.

9. How can scrambler have 2 decoders with its assertion equals true ? It seems that the assertions of different decoders must be disjoint.
REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not checked it carefully.
Actually this is a bug in my initial implementation of my algorithm. Now it is corrected.
Please refer to subsection V.C.
Please refer to www.ssypub.org to download the source code of new implementation, and try it to verify my experimental result.

10. In introduction, it was said that for other benchmarks with multiple decoders, the user can easily select the correct one, by inspecting the precondition formulas and finding out the meaning of no more than one pin, instead of up to 120 pins like the original approach [1].?Which benchmark circuit and which pin do the authors refer to in the experiment?

REPLY TO REVIEWER: Thank you for your advice. It is my fault that has not described my idea clearly.
This benchmark is XFI, I have rewritten the 4 th paragraph on page 1 right column.

11. It was said that the user can easily select the correct decoder by inspecting these preconditions. ?Please elaborate how the selection can be done. It may seem to the reviewer that the selection cannot be done unless the user already know the right configuration, but in this case there is no need to compute inferred assertions.
REPLY TO REVIEWER: Thank you for your advice.
Yes, you are right. The user still needs to read the documentation or the source code, but under the guidance of the inferred precondition formula, they only need to find out

the meaning of those variables with different value in IA_0 and IA_1.
For XFI, the two inferred precondition formula refer to only 3 instead of all 120 configuration pins, and only 2 pins have different values. Therefore, the user only needs to figure out the meaning of these two pins, instead of 120.
Please refer to subsection V.C for more details.

Associate Editor: Kunz, Wolfgang
Comments to the Author:
The paper is considered to make a worthwhile contribution although there are doubts about its practical importance. In the revision as Transaction Brief please give a better motivation for the practical relevance of your work.