



**THE CHIPS
TO SYSTEMS
CONFERENCE**

SHAPING THE NEXT GENERATION OF ELECTRONICS

JUNE 23-27, 2024

MOSCONE WEST CENTER
SAN FRANCISCO, CA, USA

A novel formal verification technique to System verification using contract refinement

Surinder Sood*/ Scott Meeth¹/Nirmal Jose*

¹ ARM limited, US

* ARM limited, Manchester UK



Motivation and Problem statement

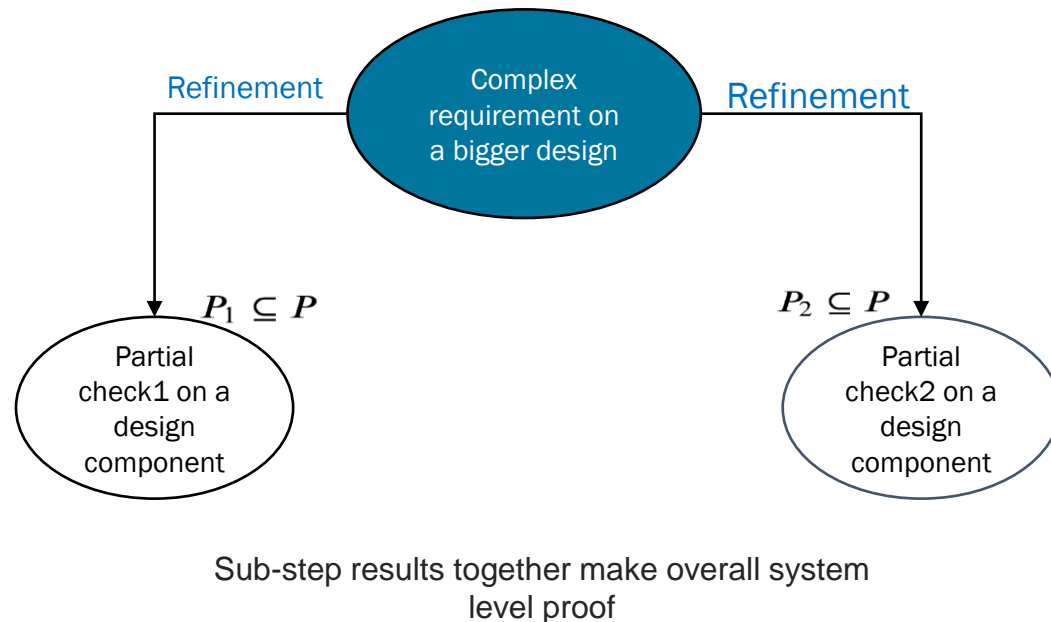
- There is no technique available to verify specific system² behavior which should guarantee
 - Completeness
 - Correctness
 - Consistency
- Existing formal techniques are limited only to component level designs, but guarantee 3Cs (mentioned above)¹
- Proof convergence on larger designs is difficult to achieve using available state-of-the-art formal techniques

¹<https://dvcon-proceedings.org/document/lets-be-formal-while-talking-about-verification-quality-a-novel-approach-to-qualify-assertion-based-vips/>

² A system is defined as an integration of two or more components

Main Idea: Proof convergence using decomposition and refinement

$P = \text{System level property having assumptions}(A) \text{ and guarantees}(G)$

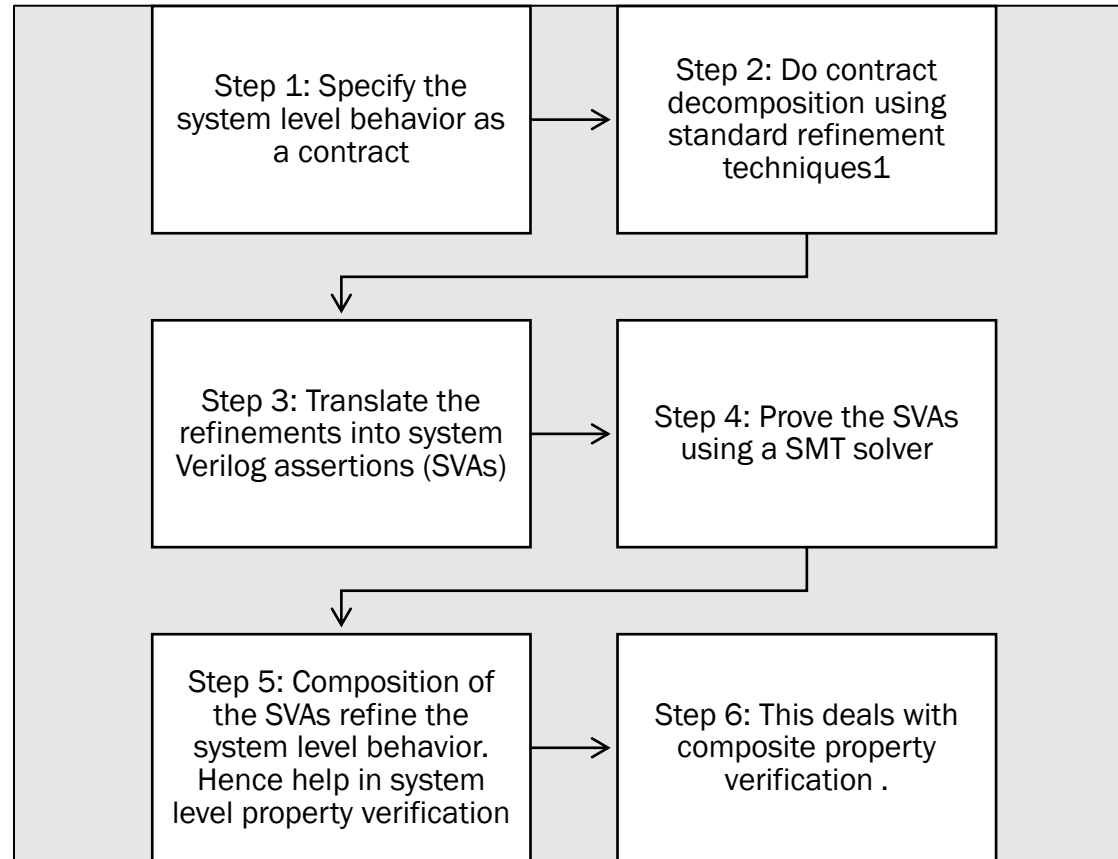


But how to select the partial checks ??

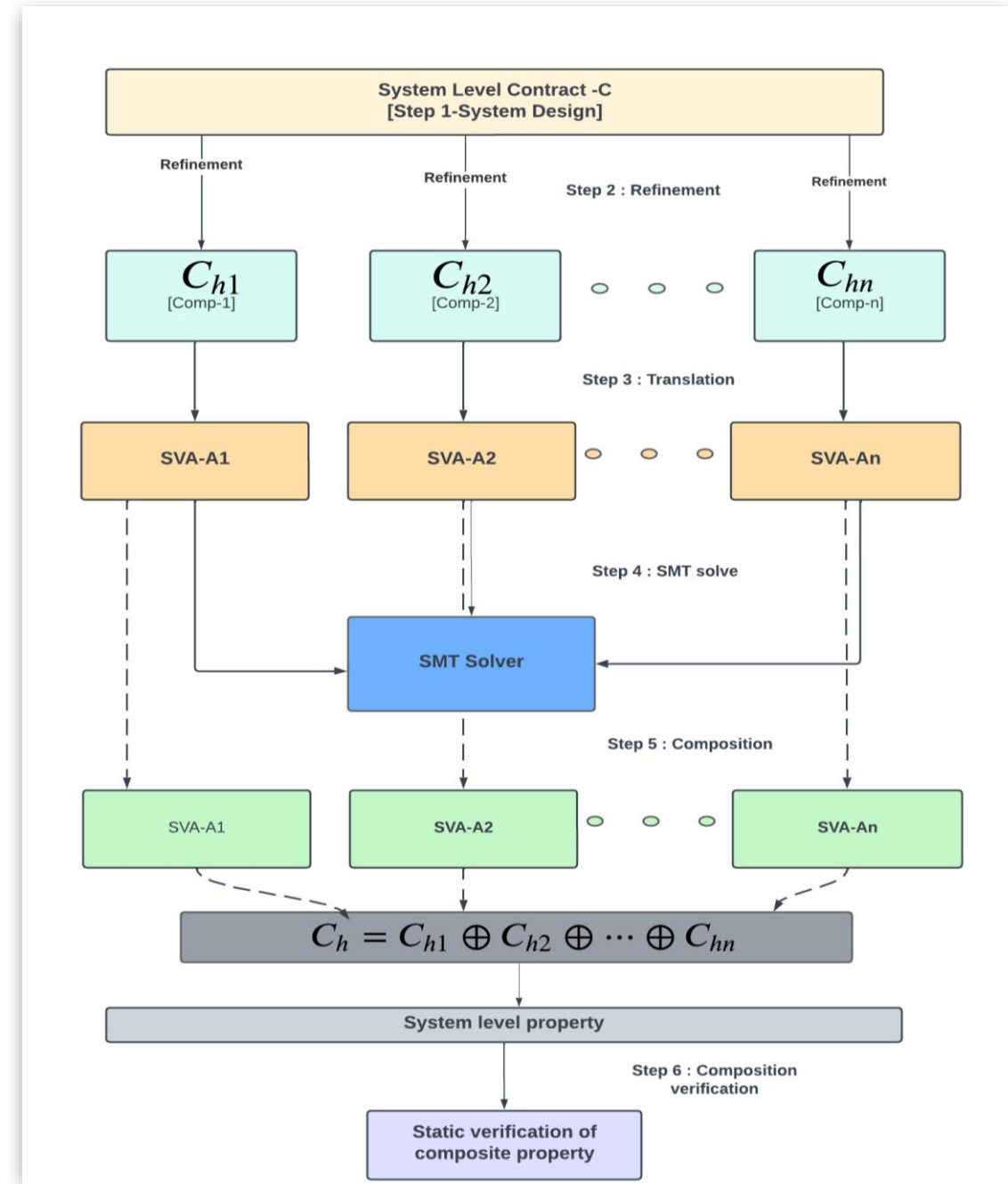
Refinements and composition

$$\begin{aligned} P_1 &\subseteq P & A_1 &\supseteq A \\ & & G_1 &\subseteq G \\ P_2 &\subseteq P & A_2 &\supseteq A \\ & & G_2 &\subseteq G \\ A &= A_1 \cup A_2 \\ G &= G_1 \cap G_2 \end{aligned}$$

Main Idea

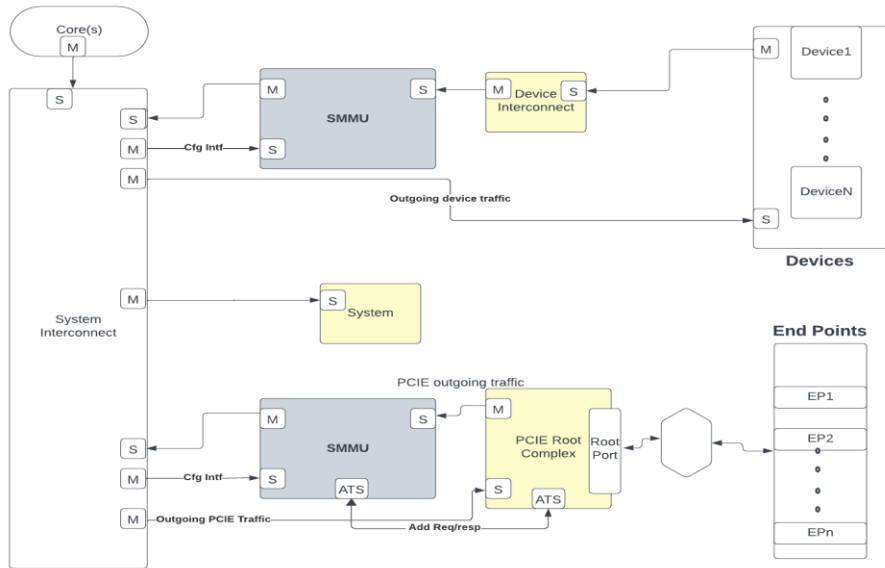


¹A. Cimatti and S. Tonetta, "Contracts-refinement proof system for component-based embedded systems," Science of computer programming, vol. 97, pp. 333–348, 2015.



Evidence: Formal verification of a System memory management unit

- A SMMU has in build TLBs and caches which provide faster access to various DMA requests from I/O devices before they are passed to the system interconnect .
- These caches behave in a similar way as the processor caches



SMMU in an example system

- Example system level behaviors for SMMU
 - Any invalidation request also invalidate the corresponding caches [E1]
 - Any system level invalidation/Sync request is eventually Acknowledged [E2]
 - A design Bug introduced in E2[E3]

System level behavior	Time taken (conventional formal technique/Proposed technique)	Property converging: Conventional/Proposed technique	Comments
E1	122271/106541 seconds	No/Yes	The system level property did not converge the property
E2	5200/4100 seconds	Yes/yes	Proposed technique converged property faster
E3	49588/48849 seconds	No/Yes	Proposed solution caught the bug, while conventional solution did not converge at all

Summary

1. Standalone properties consume more resources, their probability of convergence is still less
2. Compositional system level properties are automatically proven if their corresponding component level refinement properties are proven
3. Better refinement strategy gives faster convergence
4. If the system level property does not converge with the proposed technique, it still gives a better bound as compared to standalone properties
5. Further research is required to create more efficient refinement techniques and deployment of AI can also be done for faster convergence