Debugging Functional Coverage Models

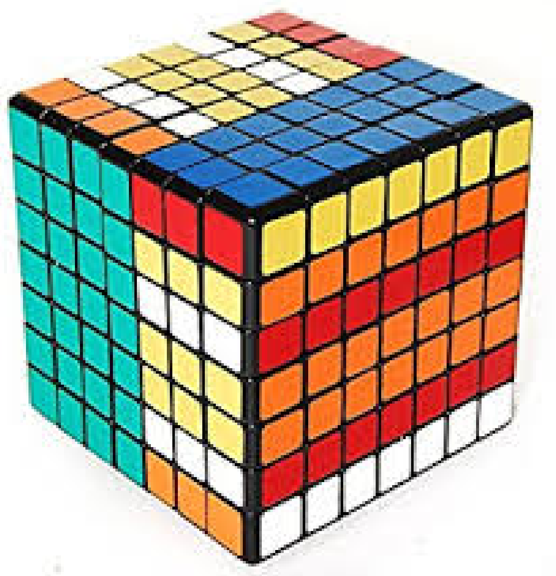
Get The Most Out of Your Cover Crosses

Mennatallah Amer, Amr Hany

Mentor, A Siemens Business

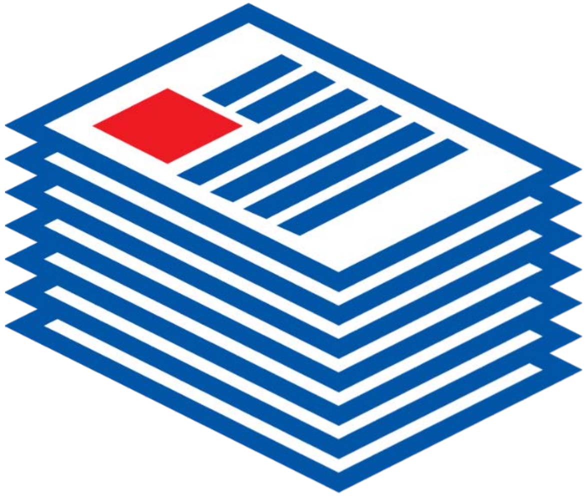
Problem

- Functional Coverage
  - Coverage from a system point of view
  - Cover **variables**, **expressions** and their **combinations**
  - Specify testplan in a **simple** and **concise** way
- But
  - Large number of cross bins
  - Covergroups contain hundreds of cover-crosses
  - Reports become too tedious to analyze
- How to identify **large holes** effectively?



Real Case Study

- 138 coverpoints
- 202 crosses
- Coverage: 72.9%
- Hit Percent: 30.79%
- There are a lot of holes
- From where to start?



2018 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Least Covered Cross

- **Coverage: 3.3%, 9 coverpoints**
- **Missed/Total Bins: 464/ 480**

Rows only differ in a single coverpoint value

Opcode	priv_mode	debug	reg	usr_en	hdcr	tpmcr	hstr	excp	hit
MCR	ns_user	0	pmcr	1	0	0	0	null_req	0
MRC	ns_user	0	pmcr	1	0	0	0	null_req	0
MCR	ns_user	0	pmcr	0	0	0	0	undef_req	0
MCR	ns_user	0	pmcr	1	0	0	0	undef_req	0
MRC	ns_user	0	pmcr	0	0	0	0	undef_req	0
MRC	ns_user	0	pmcr	1	0	0	0	undef_req	0
MCR	ns_user	0	pmcr	0	0	1	0	undef_req	0
MCR	ns_user	0	pmcr	1	0	1	0	undef_req	0
MRC	ns_user	0	pmcr	0	0	1	0	undef_req	0
MRC	ns_user	0	pmcr	1	0	1	0	undef_req	0

Automated Hole Analysis

- **Coverage: 3.3%, 9 coverpoints**
- **Missed/Total Bins: 464/ 480**

- Azatchi et al, 2006
- **26 distinct bucketed holes!**

Top Hole, has 256 missed bins, when usr_en is 1

Holes can be sorted by missed bins

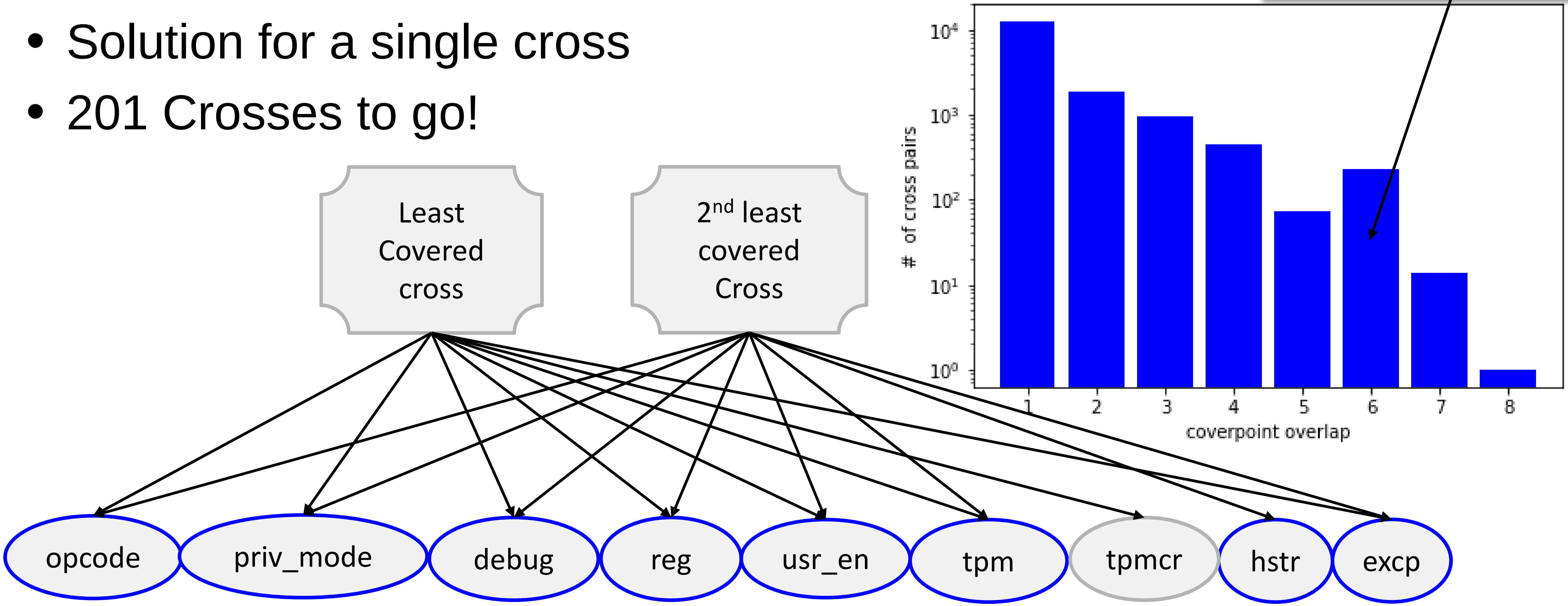
Opcode	priv_mode	debug	reg	usr_en	hdcr	tpmcr	hstr	excp	Missed
*	*	*	*	1	*	*	*	*	256
*	*	1	*	*	*	*	*	*	224
*	ns_user,ns_krnl,…	*	*	*	*	*	1	*	176
*	ns_user,s_usr,…	*	*	*	1	*	*	*	136
		*	*	*	1	*	1	*	120
		*	*	*	*	*	*	null_req	118
		*	*	*	1	0	*	null_req	92
		*	*	*	*	*	*	*	64
*	ns_krnl,s_mon	*	*	*	*	0	*	*	64
MCR	*	*	*	*	*	1	1	*	60

Larger holes, formed by aggregation of smaller projected hole. priv_mode has multiple values ns_usr,ns_krnl

Cool, But….

- Solution for a single cross
- 201 Crosses to go!

229 Cross pairs sharing up to 6 coverpoints!

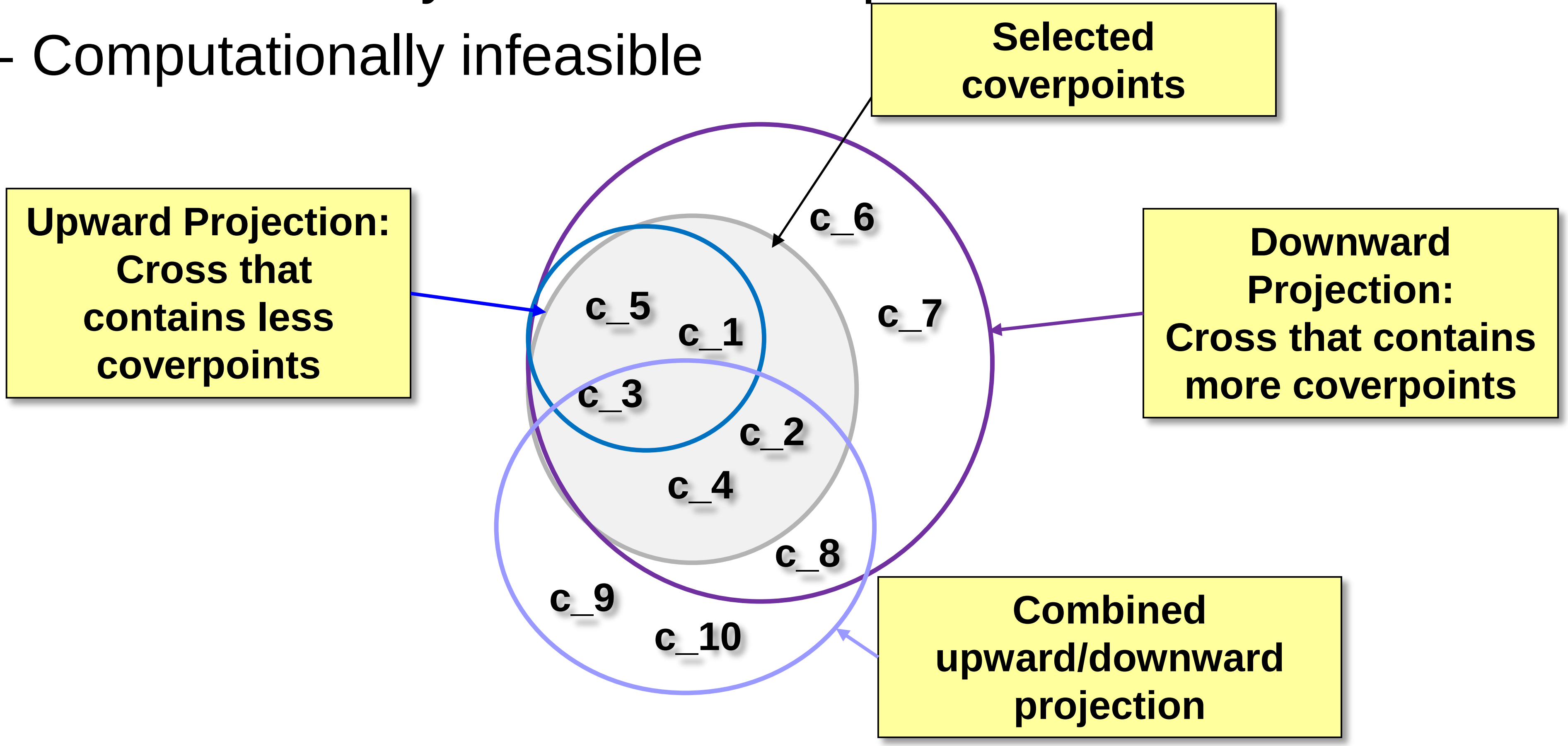


Architecting Cross-Coverage

- Based on extensive discussions/iterations
- Each cross captures a desired behavior
- Overlaps exist due to:
  - Bins in each cross grouped in a different way
  - Marking various illegal combinations
  - Whole cross is banking another signal
- Crosses can be analyzed simultaneously:
  - Multi-Hole Analysis
  - Rank new aggregated reports

Projection

- Can not run analysis on all coverpoints of the selected crosses:
  – Computationally infeasible



Multi- Hole Analysis

- 9 Crosses, Coverage: 11.08%, 5 coverpoints
- Missed/Total Bins: 15850/17570
- A single Report/ 34 buckets

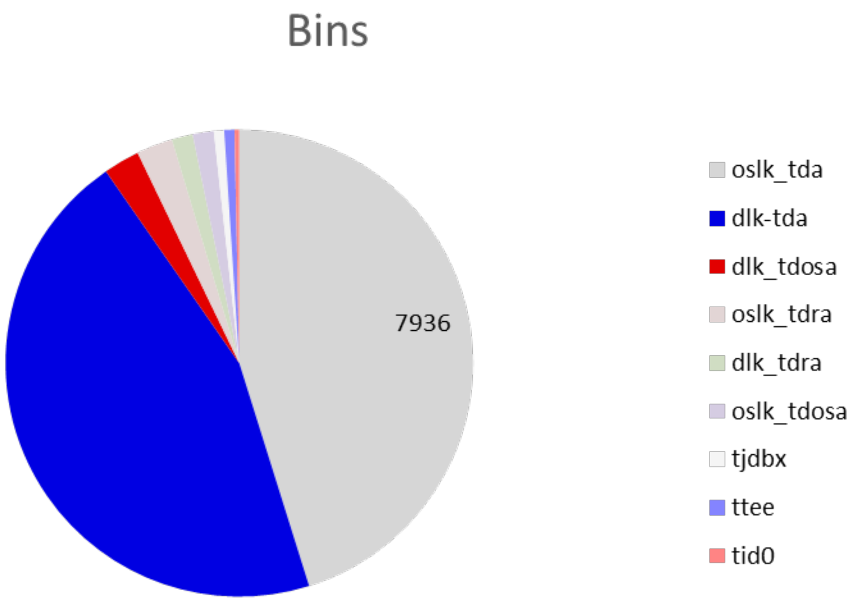
Top common hole having 3976 missed bins

Is this the best ranking?

debug	reg	priv_mode	excp	lock	Missed
*	*	*	*	1	3976
1	*	ns_user,ns_krnl…	*	*	3788
1	osdlr,prcr,…	*	*	*	2640
*	osdlr,prcr,…	ns_mon	*	*	2313
1	vcr,dsmcr,..	*	null_req	*	1344
1	*	ns_hyp,s_user	undef_req	*	1188
1	jidr,dttrx,…	*	undef_req	*	1022
1	teecr,teehbr,…	ns_hyp	*	*	844

Hole Effect

- 9 Crosses, Coverage: 11.08%, 5 coverpoints
- Missed/Total Bins: 15850/17570
- Cross bins [56- 7936]
- Hole Effect proportional to coverage increase



$$covInc_c = \frac{miss_c}{bin_c}$$

$$HoleEffect_C = \frac{\sum_{c \epsilon C} w_c \cdot covInc_c}{\sum_{c \epsilon C} w_c}$$

Multi-Hole Analysis

- 9 Crosses, Coverage: 11.08%, 5 coverpoints
- Missed/Total Bins: 15850/17570

- A single Report/ 34 buckets
- Hole Effect

New top hole would cause a coverage increase of 26.17%

debug	reg	priv_mode	excp	lock	Missed	Hole Effect
1	osdlr,prcr,…	*	*	*	2640	26.17
1	*	ns_user,ns_krnl…	*	*	3788	20.10
*	*	*	*	1	3976	15.00
*	osdlr,prcr,…	ns_mon	*	*	2313	12.68
[illegible]	…lr,prcr,…	ns_krnl	*	*	701	7.19
[illegible]	…,dttrx,…	*	undef_req	*	1022	4.19
[illegible]	…r, joscr,…	ns_user	null_req	*	550	3.93
1	*	ns_hyp,s_user	undef_req	*	1188	3.21

Hole having largest number of bins ranked 3rd, having hole effect to 15%

Summary

- Aggregated report
  - Easy to rank/analyze
  - Identify global holes
- Computationally feasible
  - Independent of number of crosses
  - Cost of a single hole analysis run
- Hole Effect for ranking top holes
- Save time and effort of analyzing complete coverage model

What's next?

- Automate the start of the analysis:
  - Automatically select cover point/crosses
  - Leverage clustering algorithms
  - No human-driven factor
- Bridge the gap between complex cover groups and analysis engine

References

- H. Azatchi, L. Fournier, E. Marcus, S. Ur, A. Ziv and K. Zohar, "Advanced Analysis Techniques for Cross-Product Coverage," IEEE Transactions on Computers, Nov. 2006
- S. Asaf, E. Marcus, and A. Ziv, “Defining Coverage Views to Improve Functional Coverage Analysis,” DAC, June 2004
- O. Lachish, E. Marcus, S. Ur, and A. Ziv, “Hole Analysis for Functional Coverage Data,” DAC, June 2002
- A. Yehia, "Faster coverage closure: Runtime guidance of Constrained Random stimuli by collected," Electronics, Communications and Photonics Conference (SIECPC), May 2013
- E. E. Mandouh and A. G. Wassal, "Guiding intelligent testbench automation using data mining and formal methods," International Design & Test Symposium (IDT), 2015

Questions?