mL: Shrinking the Verification volume using Machine Learning

Yash Phogat (yash.phogat@arm.com), Patrick Hamilton (Patrick.Hamilton@arm.com)
Arm Inc., Austin TX, USA

accellera SYSTEMS INITIATIVE

2024 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES
SAN JOSE, CA, USA
MARCH 4-7, 2024

INTRODUCTION

Hardware design verification uses Constrained Random Testing (CRT) [1], where the test inputs are generated randomly. These tests may end up wasting compute in scouring healthy regions of design, that have already been tested in past regressions.

Using Machine Learning (ML), we classify randomly generated tests on their likelihood of failing/passing pre-simulation. Based on this classification we run more efficient regressions.

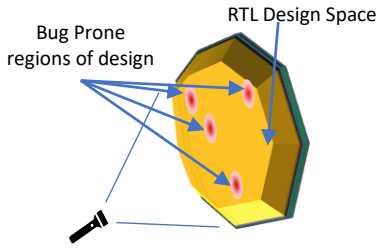
Train Data	Retrain Frequency	Input	Output
2 weeks of regression data Input	Weekly	Test knobs	Test Fail/Pass Decision

Continually changing design necessitates ML retraining periodically to keep the models relevant to the underlying RTL.

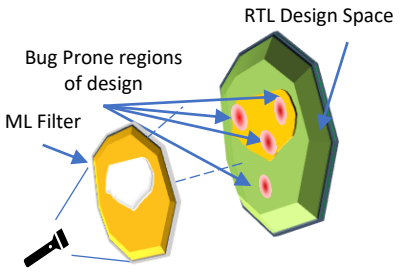
Test #	*Knob-1*	*Knob-2*	*….*	*Knob-p*	*Fail*
Test_1	Val-1_1	Val-1_2	….	Val-1_p	True
Test_2	Val-2_1	Val-2_2	….	Val-2_p	False
:	:	:	….	:	:
Test_n	Val-n_1	Val-n_2	….	Val-n_p	False

Table 1: Simulation test data structure for training. n=1 million, p=1000

OBJECTIVES



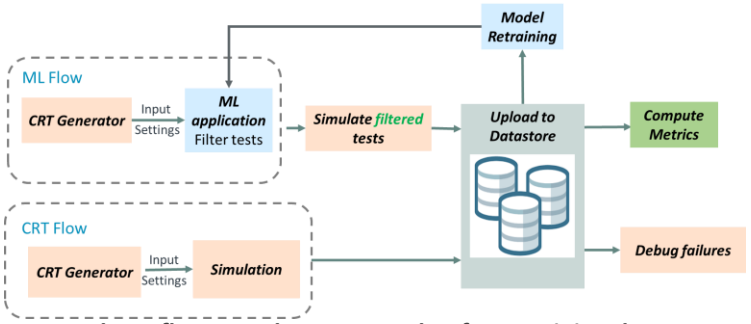
CRT searches the entire RTL design space, even the healthy design space. Which means testing the yellow space again and again is a waste of compute



ML based filter prunes inefficient tests that waste compute (green area)

The objective is to use an ML based predictive filter, before simulating tests on the design. As illustrated in the above example, the ML filter would prevent from running tests classified to the healthy region of the design. In this way we run smaller, more efficient regressions and save compute.

Architecture and Metrics



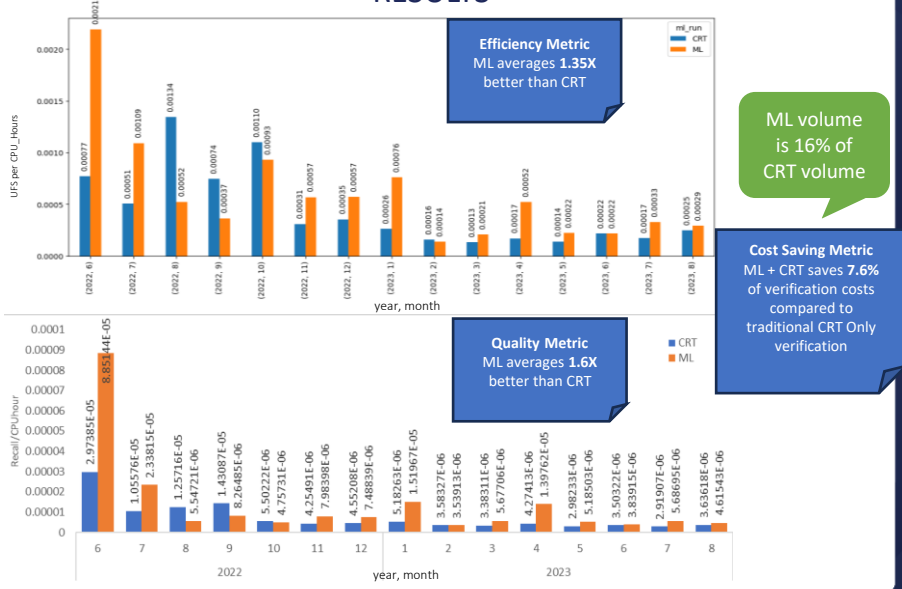
Note: ML and CRT flows need to run together for retraining data generation and exploring unexplored design regions

Efficiency Metric (UFS per CPU_Hour)
Number of Unique Fail signatures found in 1 CPU Hour spent by either flow.

Quality Metric (UFS Recovery Rate)
Given 1 CPU hour, what portion of design UFS's are found by ML and CRT flows.

Cost Saving Metric
Compute saved by ML in finding the UFS's it found, which in an ML absent scenario would have to be found by CRT.

RESULTS



CONCLUSIONS

CHALLENGES

- **Class Imbalance** - The number of fails and passes present in our training data are highly imbalanced (5:95). Hence, we use gradient boosted training algorithms which handle class imbalances well.
- ML development during live projects **get limited budget for experiments**.
- **Adaptability to project phases and the verification strategy**.
- ML is a science of probability and hence comes with a margin of approximation. It's hard to reason for ML picking a test stimulus.

FUTURE WORK

Deployment Improvements:

1. Find ML:CRT tests ratio for best savings by the phase of the project? 16%
2. Find the best ratio of generated test to the no. of filtered tests for ML? 12%

Machine Learning Improvements:

1. Tune ML models - feature selection, other training algorithms, creation of better Cross Validation scoring functions etc.
2. Use ML to find efficient test set for quantifying RTL coverage.
3. Using generative AI to control knob generation and hit bug prone test knobs.

REFERENCES

[1] Constrained Random Verification, Yuan J.; Pixley, C.; Aziz, A.; 2006, XII, 254p. 72 Illus., Hardcover, ISBN: 978-0-387-25974-5

ML - "Case study: real-world machine learning application for hardware failure detection", H. Shin; Proc. 18th Python in Science Conf. (SciPy), pp. 5-12, 2019.

Preprocessing - H. Shin, "Data-Centric Machine Learning Pipeline for Hardware Verification," 2022 IEEE 35th International System-on-Chip Conference (SOCC), Belfast, United Kingdom, 2022, pp. 1-2, doi: 10.1109/SOCC56010.2022.9908095.

Acknowledgements

I would like to acknowledge my employer's ML & CPU team for development, budgets, and review of this paper.

Encino Trace,
5707 Southwest
Pkwy, Bld 1, Suite 100
Austin, TX 78735

arm