



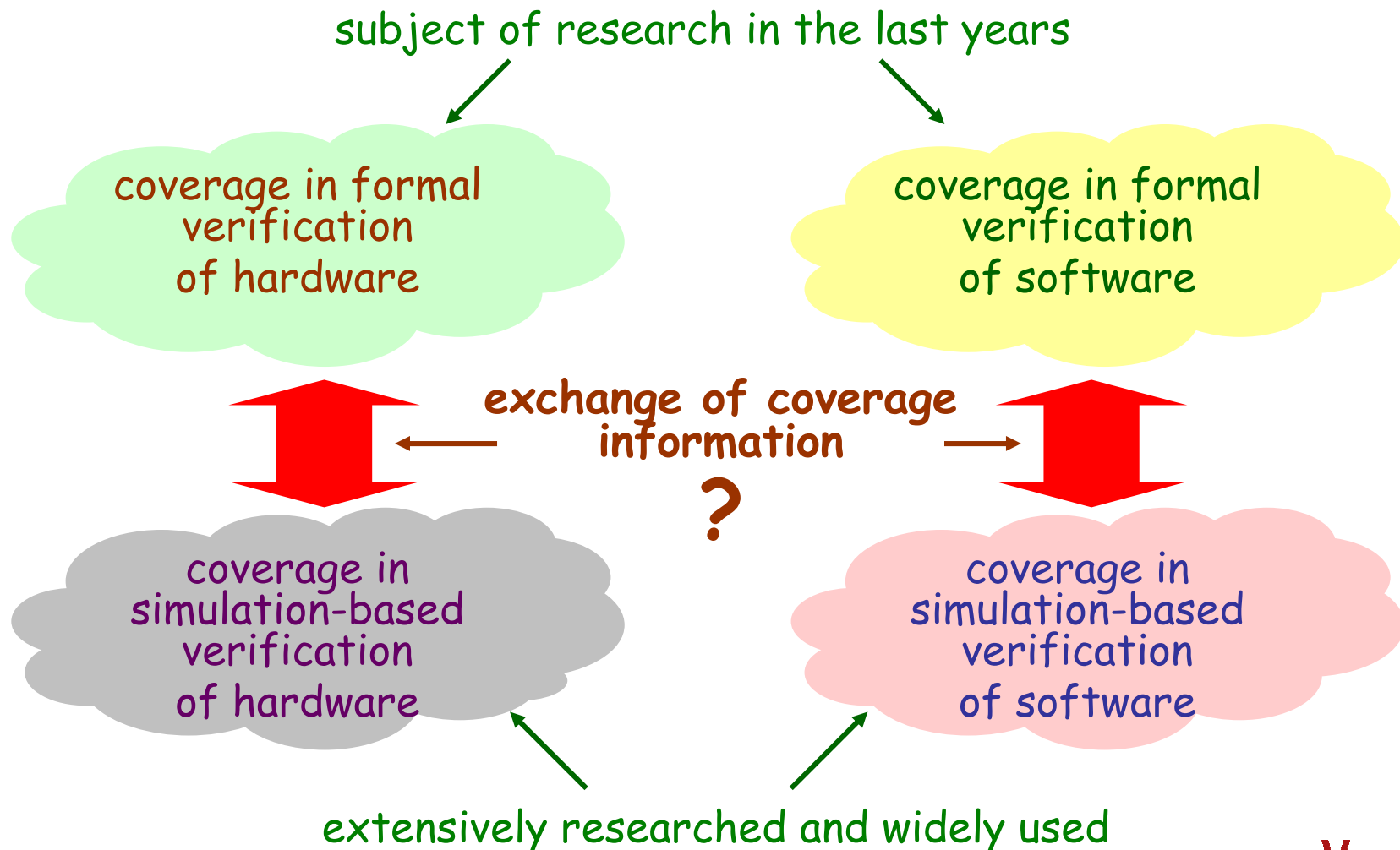
IBM Research

How to exchange coverage information between formal verification and simulation?

Hana Chockler
IBM HRL



Coverage in verification - current situation

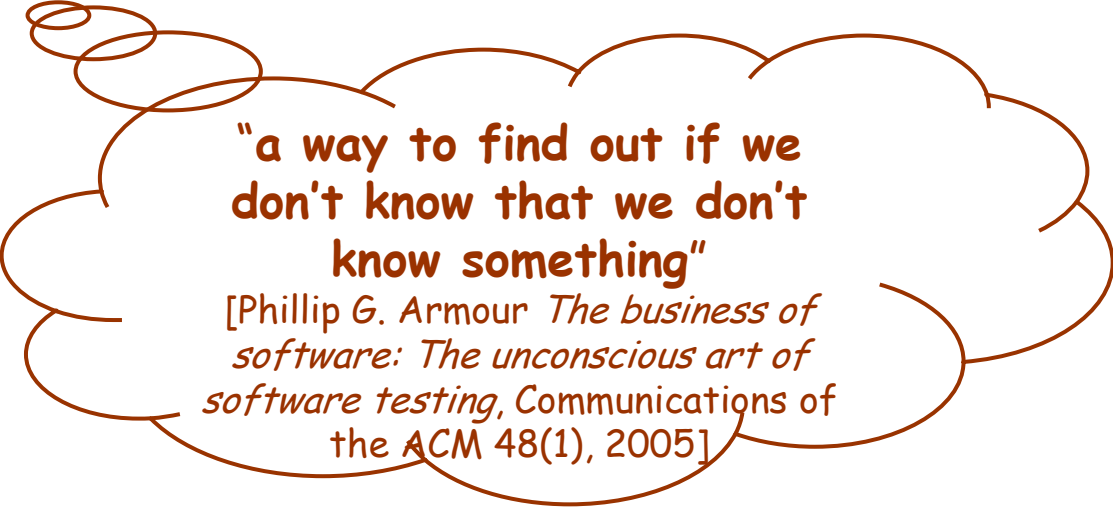


Simulation-based Verification

Coverage metrics and coverage models are heuristic measures of exhaustiveness of a test suite:
high coverage ➡ fewer bugs



a lot of
different
metrics



"a way to find out if we
don't know that we don't
know something"

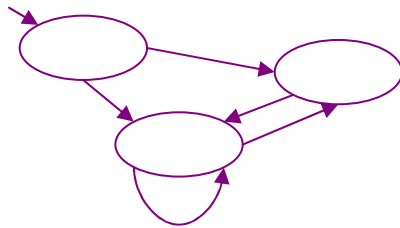
[Phillip G. Armour *The business of
software: The unconscious art of
software testing*, Communications of
the ACM 48(1), 2005]

- ◆ Simple: parts (lines, statements, etc.) that are visited during the execution.
- ◆ More complex: coverage models, functionality coverage, mutation coverage.

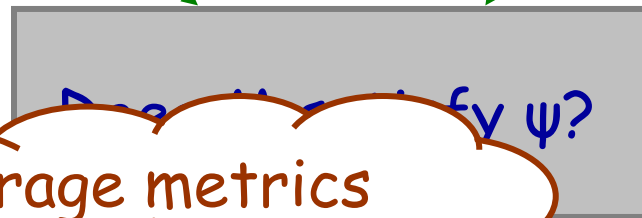
Model Checking

Is the system correct?

A mathematical model
of the system M (an FSM):



A formal specification ψ



coverage metrics
based on
mutations

counter example

yes

the system
is correct!

is the
specification
complete?



Is it enough to compute coverage
separately for formal
verification and for simulation?

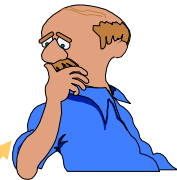


Is it enough to compute coverage separately for formal verification and for simulation?



Who defines coverage goals and interesting mutations?

what does it mean? is it good or bad?



Moshe Vardi

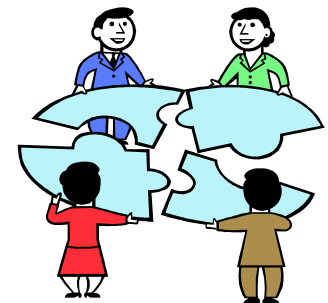
Simulation coverage:
80% of coverage goals reached
FV coverage: 40% of mutations covered

What can be the reasons for low coverage in formal verification?

- ♦ Specification is not exhaustive enough
- ♦ System contains redundancy
 - o Not necessarily a problem - redundancy may be needed to ensure fault tolerance
- ♦ Over-restrictive environment
 - o May be a reason for low coverage if mutations are masked by environment restrictions
 - o **Can we check this by running simulation?**
- ♦ The set of interesting mutations does not reflect real coverage goals
 - o **How do we determine which mutations are interesting? The decision should be connected to simulation!**

Exchange of coverage information between formal verification and simulation - benefits

- ♦ Division of work between formal verification and simulation based on the coverage goals
 - o Reduces double work - a coverage goal needs to be reached by one method
- ♦ Goals uncovered by one method can be covered by another
- ♦ Blocks that are verified by model checking can nevertheless be not fully covered with respect to coverage goals - need more checks using simulation
- ♦ Some coverage goals can be fully reached using formal verification - no need for running simulation



Exchange of coverage information based on coverage models

- ♦ The set of interesting mutations can be derived from the coverage model; then we can model-check which mutations falsify assertions or affect the output:
 - o Mutations that have no effect are uncovered.
 - o Mutations that affect the output are covered.
 - o If no assertion is falsified by a specific mutation:
 - We need more assertions, or
 - check whether coverage model should contain this mutation.
 - o If there is an assertion that is not falsified by any mutation:
 - Enhance the coverage model with more mutations.
- ♦ Mutating the boundaries of the environment - does it falsify assertions or affect the output?
 - o According to the result, we might need to tighten the environment, loosen the environment, or add more assertions.

How do we know which covered mutations can happen in reality?

- ♦ Mutations of the design:
 - o In the design we are more interested in uncovered mutations, because they represent unchecked parts of the design.
 - o Mutations do not happen in the current design.
 - o The knowledge of covered mutations might be useful “for future use”, in case the design is modified.
- ♦ Mutations of the environment:
 - o **Can happen in reality!**
 - o We can run simulation on mutations of the boundaries of the environment that have some effect on the specification - either falsify it or cause vacuous satisfaction - in order to check which are possible in reality.

The goal

- ◆ Coverage model is built for simulation-based and for formal verification together
- ◆ Coverage is computed based on the combined results and feedback between simulation-based and formal verification

60% coverage after running formal and simulation according to the coverage model X →
need more checks!

OR

80% coverage after running formal and simulation according to the coverage model Y →
we are done!

