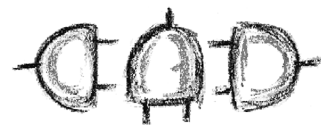


COTEST meeting

Torino April, 26th 2002

Outline

- Experiment description
- Results
- Conclusions.

Goal

- Assess the feasibility of high-level test vectors generation based on high-level fault models.

Activities

- Different versions of the adopted benchmarks have been developed (using `Behavioral Compiler`)
- Prototypical environments have been developed (supporting semi-automated fault list generation, high-level fault simulation and gate-level fault simulation)
- High-level fault models have been compared.

Assumptions

- Behavioral-compiler (BC) like model:
  - One entity
  - One process
  - Explicit clocking strategy via (multiple) `wait`
  - Global synchronous reset
- BC+DC Synopsys synthesis flow
- Simple gate-level library (and, or, ffd…).

Metrics

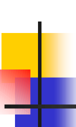
- Behavioral level:
  - Statement coverage: SC
  - Bit coverage: BC
  - Condition coverage: CC
- Gate level (GL):
  - Single permanent stuck-at: GL stuck-at

Benchmarks

Benchmark	Scheduling	VHDL lines	Gates	FFs
BIQUAD 1	Fixed IO speed optimized	93	6,043	125
BIQUAD 2	Fixed IO area optimized	93	3,252	257
TLC	Fixed IO	168	241	23

Behavioral Fault Simulation

- Performed via
  - Manual instrumentation of circuit description for fault injection and faulty effects observation
  - Commercial VHDL/SystemC simulator.



Analysis of available fault models

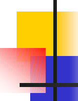
Random Test Sequences

Benchmark	SC	BC	BC+CC
BIQUAD 1	0.63	0.97	0.97
BIQUAD 2	0.64	0.97	0.98
TLC	0.83	0.45	0.80

Figures represent the **correlation** between high-level coverage and gate-level one.

Observation (I)

- BIQUAD:
  - SC is useless
  - CC is not applicable (due to the benchmark nature)
  - Good correlation between BC and GL stuck-at.
  - Predictions about the testability are implementation-independent: same trends no matter the scheduling mode and the optimization constraints.



Observation (II)

- TLC:
  - SC is somewhat useful
  - BC has poor correlation with GL stuck-at
  - BC+CC has much better (but still rather low) correlation with GL stuck-at.

A new fault model

- State coverage (STC): it measures the percentage of `wait` statements that are reached
- It is related with the percentage of states of the model control unit that are traversed.

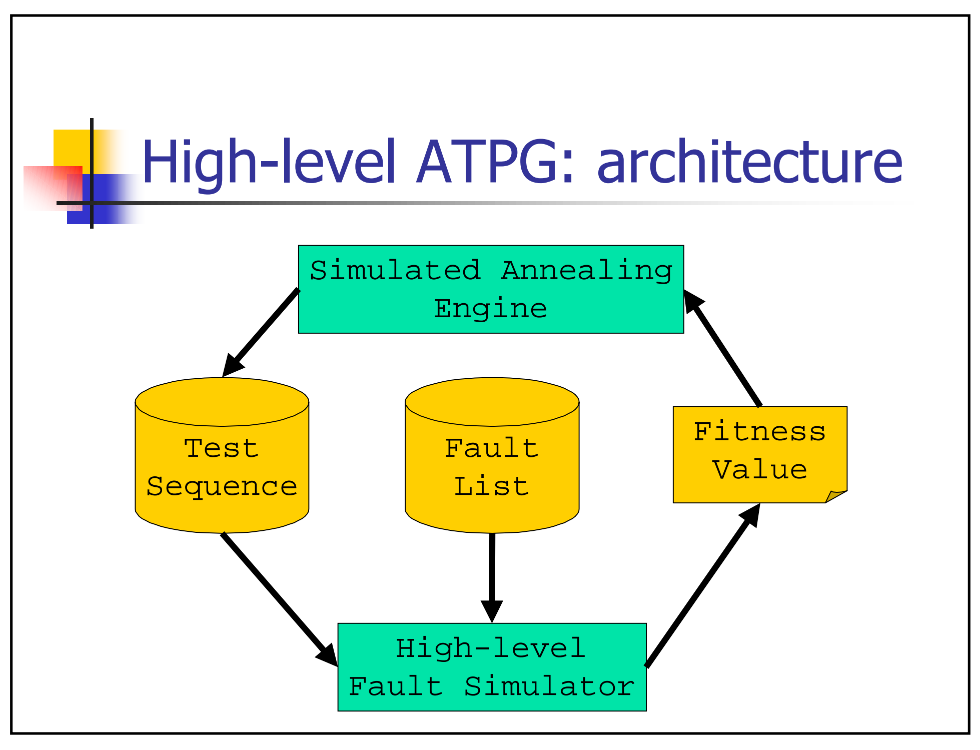
Analysis of the new fault model

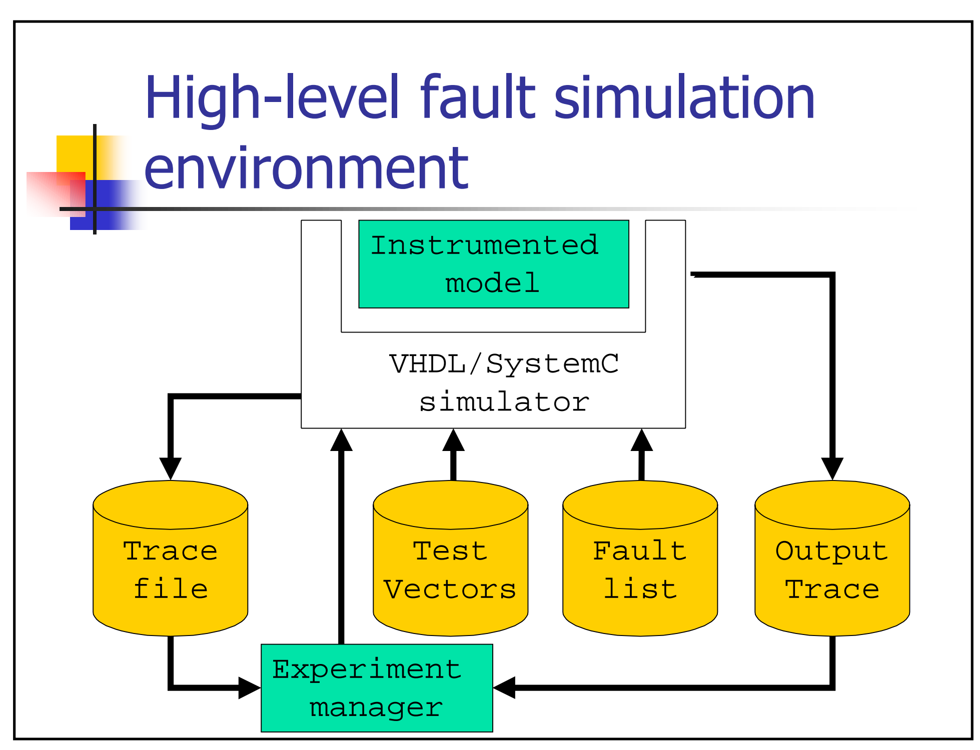
Benchmark	BC+CC	BC+CC+STC
BIQUAD 1	0.97	0.97
BIQUAD 2	0.98	0.98
TLC	0.80	0.86

Figures represent the **correlation** between high-level coverage and gate-level one.

High-level ATPG

- Intended to preliminary assess the feasibility of generating test vectors on behavioral models
- It is based on the BC+CC+STC high-level fault model
- It exploits a Simulating Annealing algorithm.

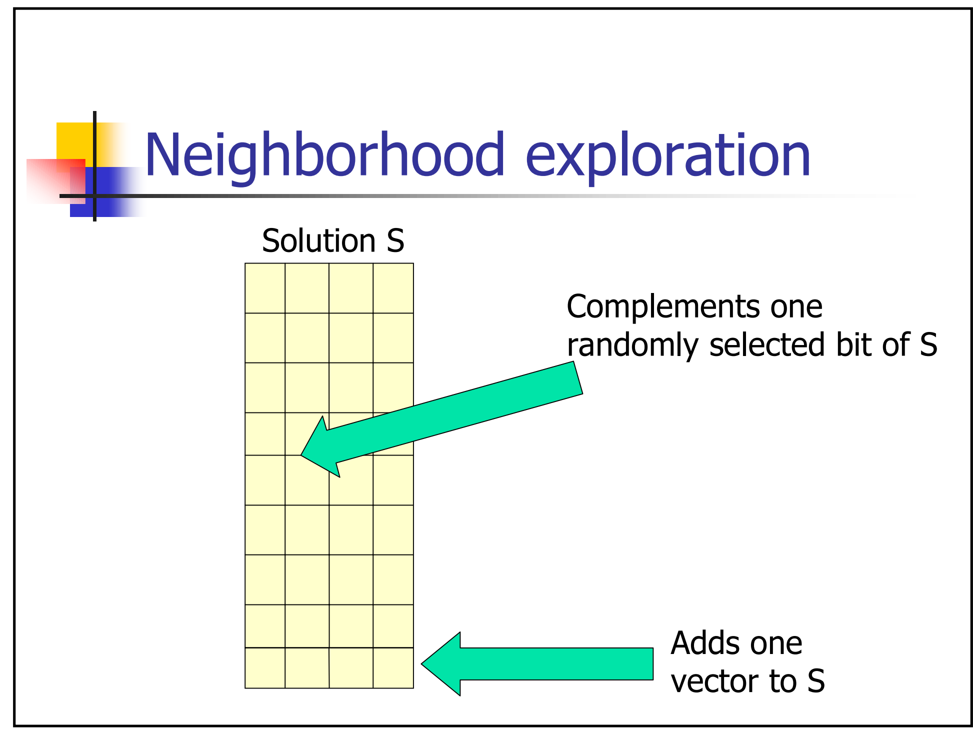

High-level ATPG: architecture
Simulated Annealing Engine
Test Sequence
Fault List
Fitness Value
High-level Fault Simulator


High-level fault simulation environment
Instrumented model
VHDL/SystemC simulator
Trace file
Test Vectors
Fault list
Output Trace
Experiment manager

Simulated Annealing

- The solution is a sequence of vectors
- The evaluation function measures the BC+CC+STC coverage
- The temperature is gradually decreased to a *freeze* temperature, where no further change occurs.

Neighborhood exploration

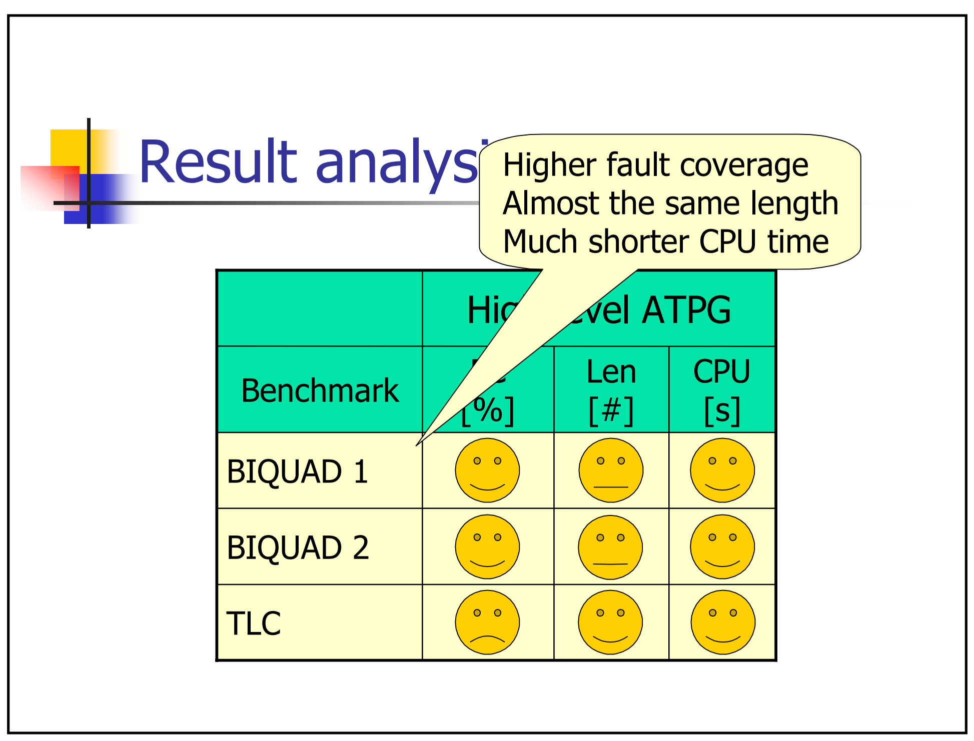


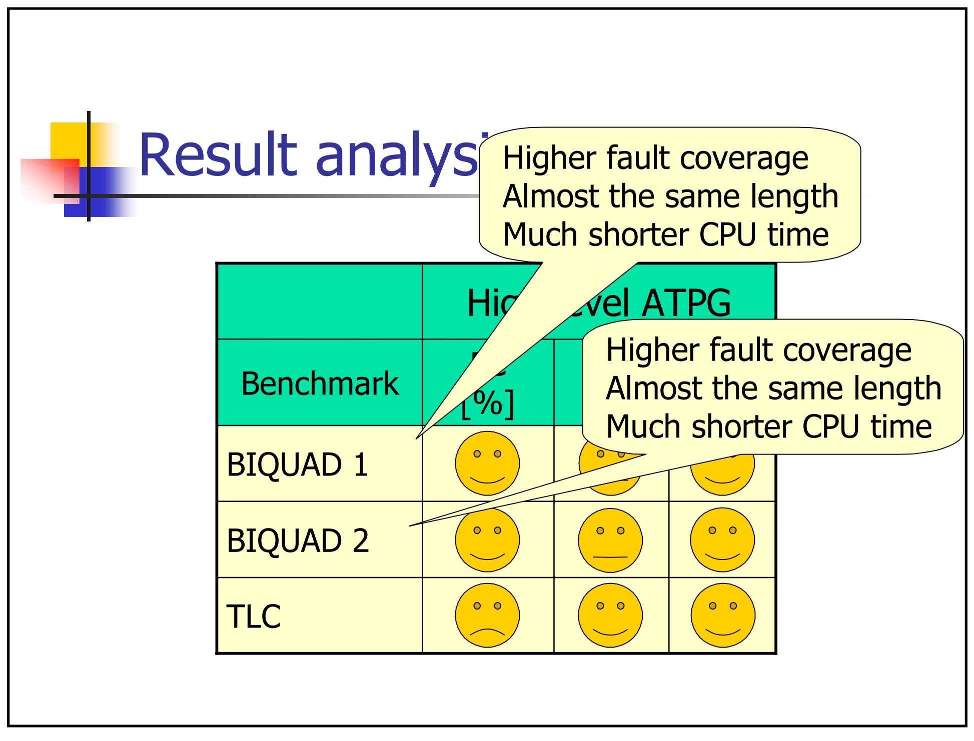
Experimental results

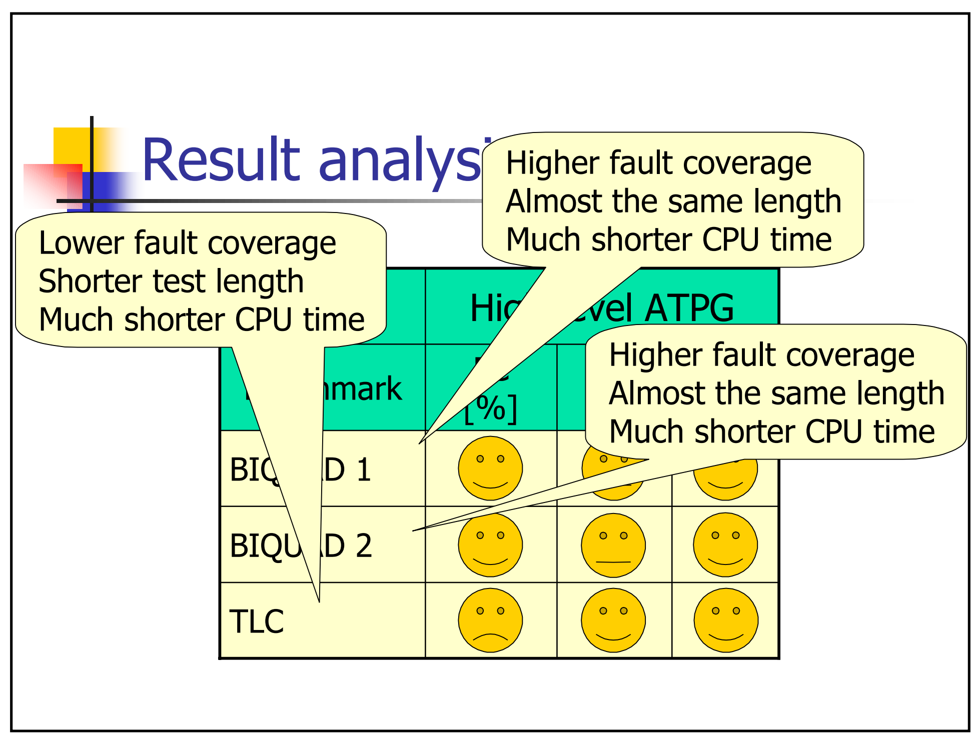
	High-level ATPG			testgen		
Benchmark	FC [%]	Len [#]	CPU [s]	FC [%]	Len [#]	CPU [s]
BIQUAD 1	68.27	287	2,139	37.06	154	10,817
BIQUAD 2	86.94	287	2,139	70.75	245	11,060
TLC	74.48	23	9.71	80.67	77	98.77

Result analysis

	High-level ATPG		
Benchmark	FC [%]	Len [#]	CPU [s]
BIQUAD 1			
BIQUAD 2			
TLC			


Result analysi
Higher fault coverage
Almost the same length
Much shorter CPU time
vel ATPG
Benchmark
[%]
Len [#]
CPU [s]
BIQUAD 1
BIQUAD 2
TLC


Result analysi
Higher fault coverage
Almost the same length
Much shorter CPU time
vel ATPG
Benchmark
[%]
Higher fault coverage
Almost the same length
Much shorter CPU time
BIQUAD 1
BIQUAD 2
TLC



Conclusion

- According to the parameters defined in the COTEST technical annex:
  - High-level ATPG is well-suited for data-dominated applications.
  - Research is still required for effective testing of control-dominated applications, although preliminary results are encouraging.

Publications

- O. Goloubeva, M. Sonza Reorda, M. Violante,
  “Experimental analysis of fault models for behavioral-level test generation”, IEEE DDECS 2002, Design and Diagnostics of Electronic Circuits and Systems Workshop, pp. 416-419

Submitted

- O. Goloubeva, M. Sonza Reorda, M. Violante, ”Behavioral-level fault models comparison: An experimental approach”, ICAM
- O. Goloubeva, M. Sonza Reorda, M. Violante, ”Behavioral-level test vector generation: fault model selection and preliminary test generation results”, DCIS