Piet Engelke, Bernd Becker, Martin Keim :

A Parameterizable Fault Simulator for Bridging Faults

Albert Ludwigs University Freiburg

Overview

I Bridging faults

- introduction
- expectations from model
- voting model

II The proposed method

- voltage mapping
- determination of bridge values
- bridge value interpretation
- structure of the simulator
- experimental results
- conclusions

I Bridging faults

Introduction

bridging fault : unintentional connection between two lines

classification :

- signal node / internal gate node involved
- feedback / non - feedback
- hard short / resistive short

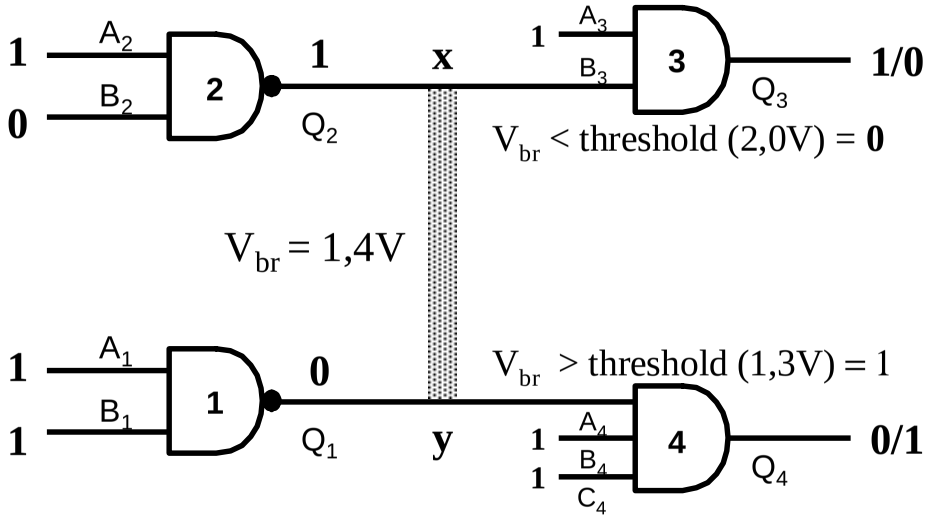
in the following :

- single faults
- combinational circuits
- non - feedback
- hard short
- bridges between gate outputs

Expectations from model (1)

wired - and dominant models cannot model a bridge adequately

e.g. byzantine generals problem



- model should be able to determine a voltage
- closer look at internal gate structure is necessary

Expectations from model (2)

- voltage has to be converted into a logical value
- chips are subject to manufacturing variances

=> determination of voltage is inevitable

  - voltage provides detailed information
  - ranges are sufficient
  - mapping to 0, 1, X means loss of information

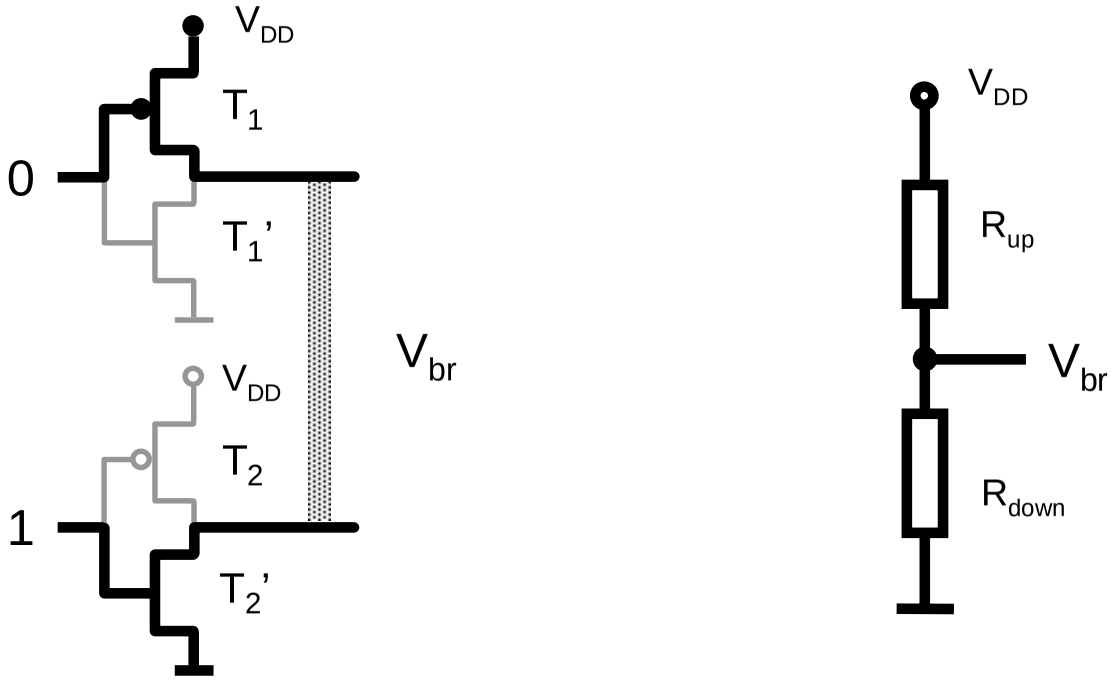
    (X = voltage between logical value 0 and 1)

Voting model

the bridge is considered as a resistive divider between V_{DD} and GND

comparison of transistor networks´ strengths

(dependent on number and setting of transistors)



resulting voltage is converted into a logical value

II The proposed method

Voltage mapping

based on : mapping of voltage ranges to n - valued logic

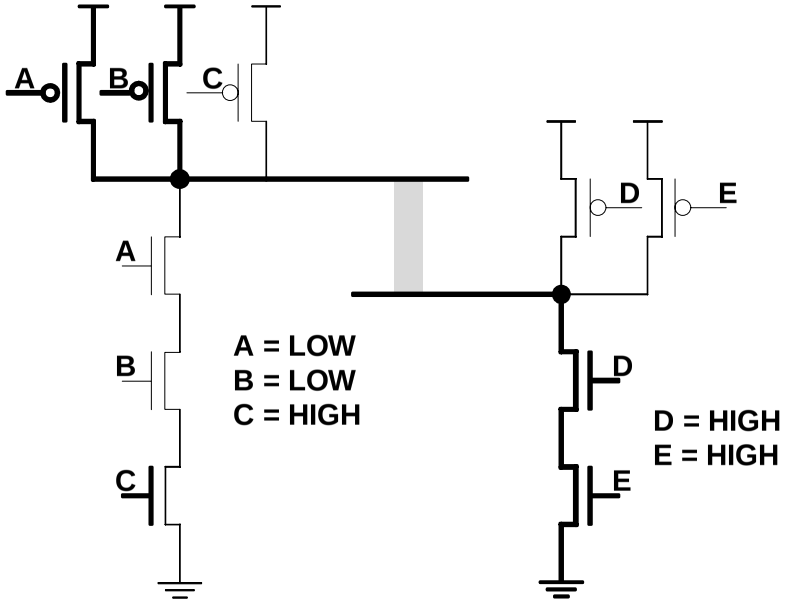
voltage range	logical value		
	5 - valued	3 - valued	
V_{DD} - V_4	4	2	**MAX**
V_4 - V_3	3	⎫	
V_3 - V_2	2	⎬ 1	
V_2 - V_1	1	⎭	
V_1 - V_{GND}	0	0	**MIN**

flexible concerning :

- setting of multivalued logic
- voltage ranges

Determination of bridge values

p - transistor		n - transistor				
		single transistor	parallel network		serial network	
		1	2	3	2	3
single transistor	1	2	1	0	3	4
parallel	2	3	2	1	4	4
network	3	4	3	2	4	4
serial	2	1	0	0	2	3
network	3	0	0	0	1	2



gate type + input vector => transistor combination
= logical value on bridge

only transistors in ”ON-state” are considered

Bridge value interpretation

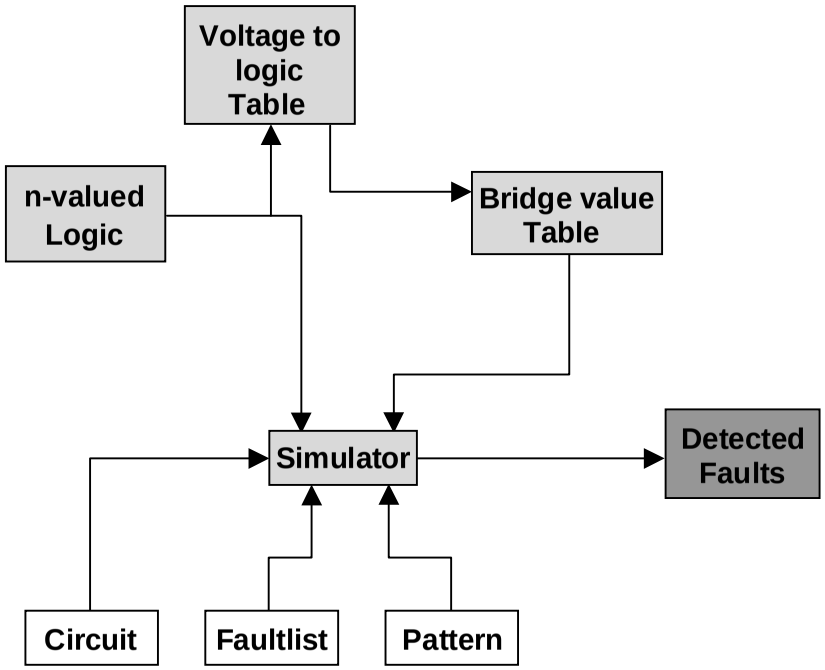
bridge values are interpreted by gate specific logic operators
=> byzantine generals problem is avoided for the most part

NOR	0	1	2	3	4
0	4	4	1	0	0
1	4	3	0	0	0
2	1	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0

adaptable to :

- gate thresholds
- special points of interest

Structure of the simulator



- event driven
- gate level simulation
- operates directly on n - valued logic
- simple algorithms for table lookup

=> fast, small, simple structure

Experimental results

from the definition of the 3 - valued logic follows :

a MIN, MAX value resulting from 3 - valued simulation
will also be determined by the 5 - valued simulation

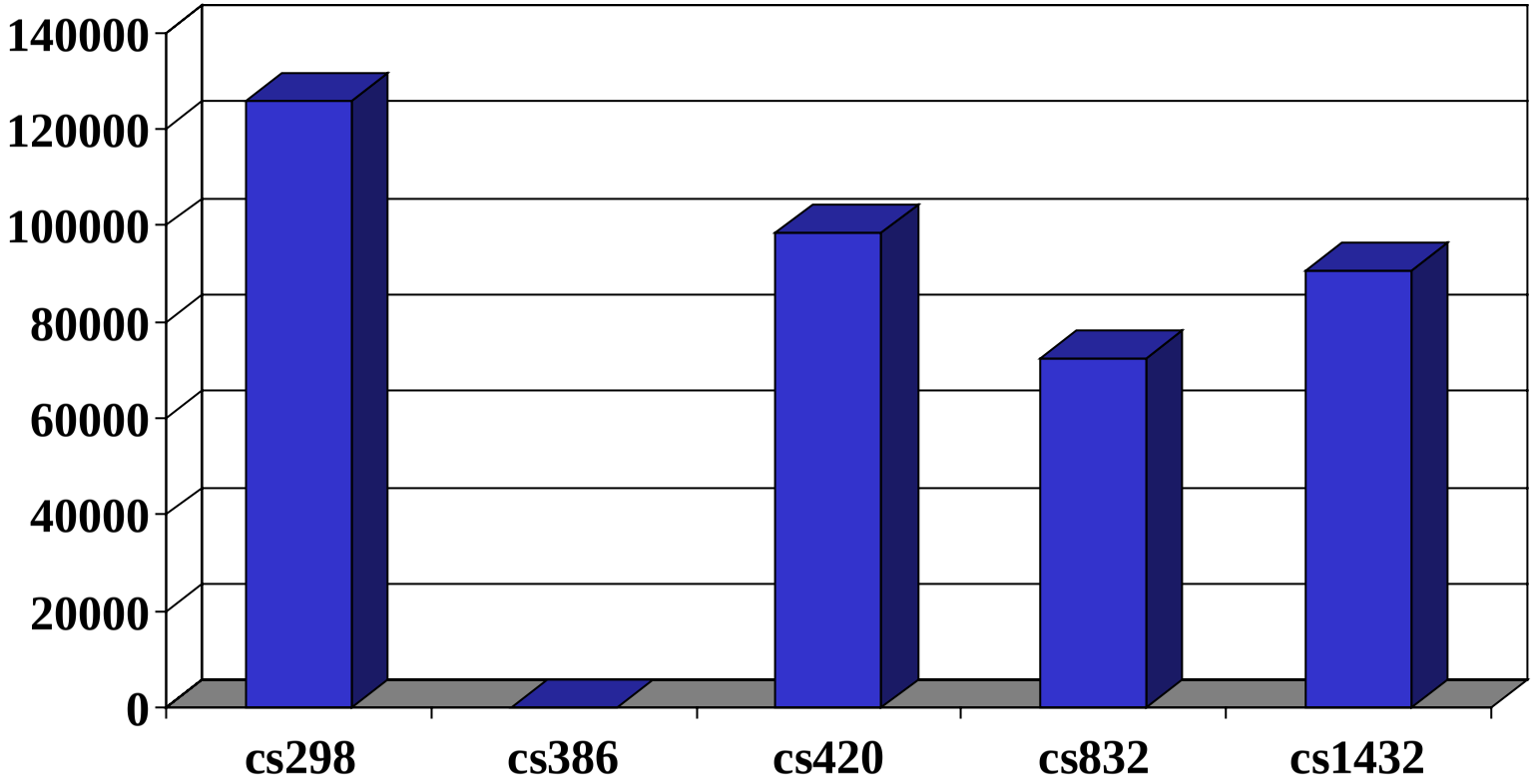
main questions :

- more MIN, MAX values in simulation based on 5 - valued logic ?
- does the fault detection increase ?

parameters of the simulations :

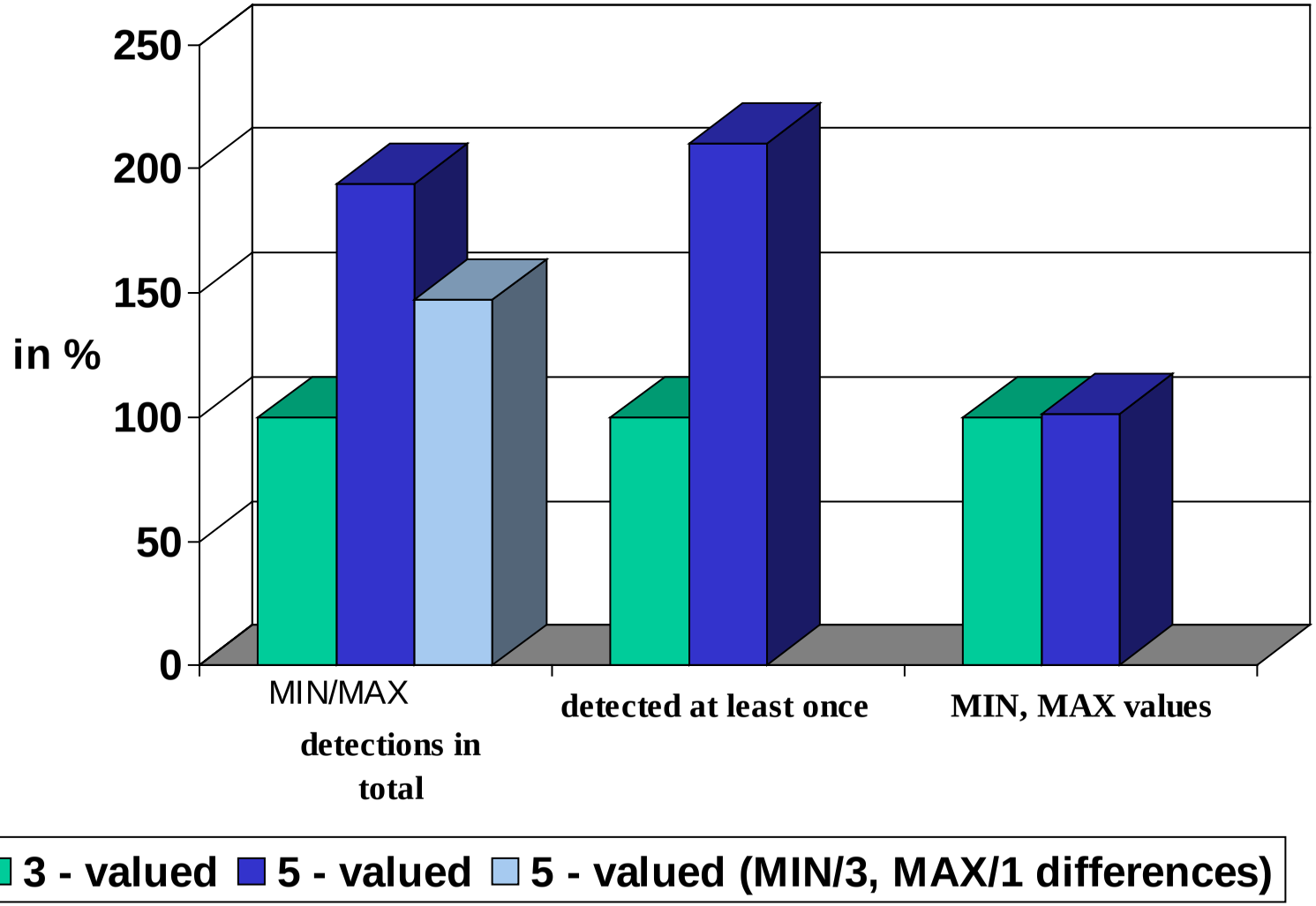
- 100 random patterns, 10.000 randomly picked faults
- no fault dropping
- propagate all differences to the primary outputs

Experimental results - MIN, MAX values



increase of MIN, MAX values with 5 - valued simulation
(with respect to 3 - valued simulation)

Experimental results - cs420



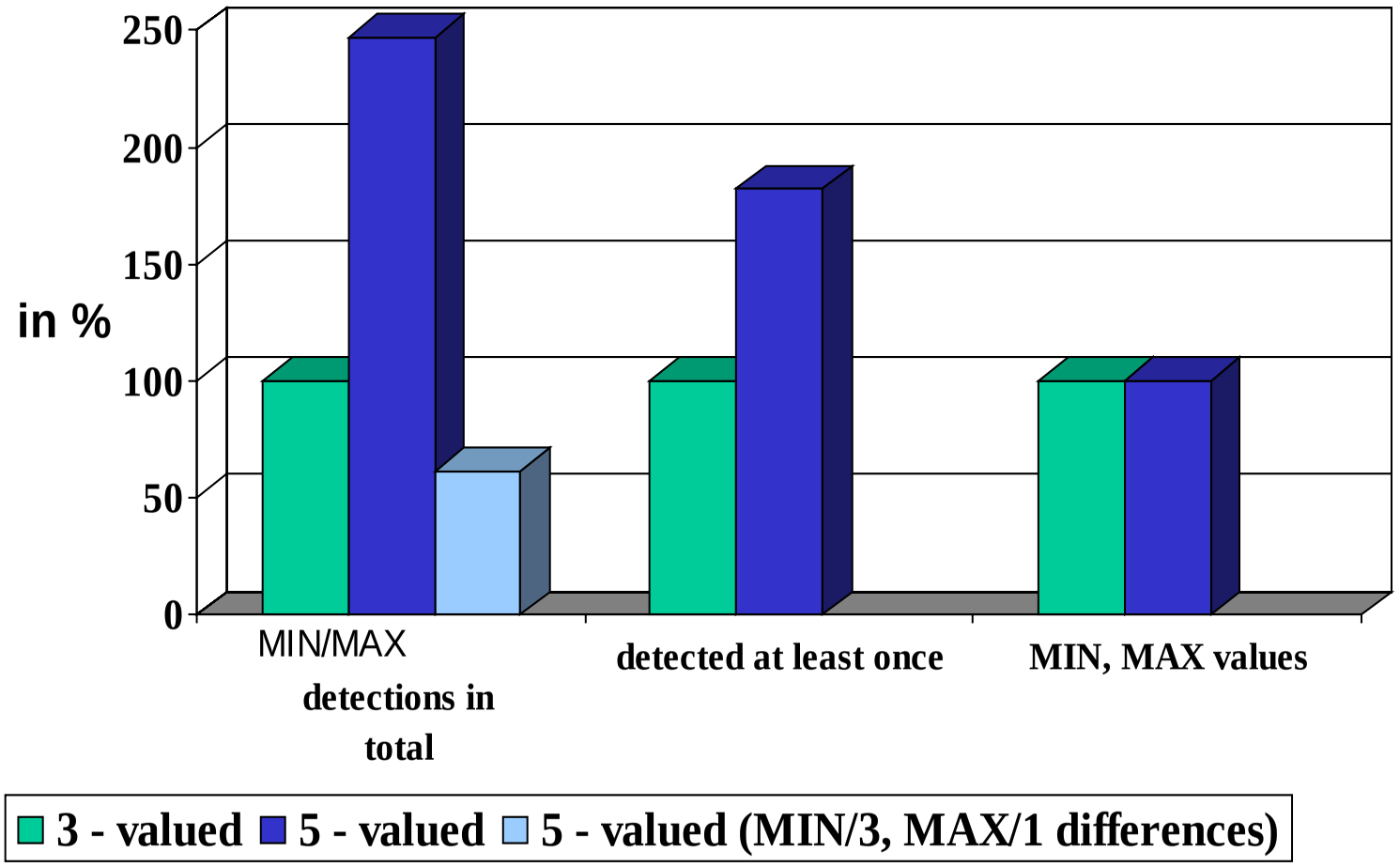
Conclusions

- small and fast simulator
- circuit behaviour accurately determined
- manufacturing variances are taken into account
- more faults detected
- flexible

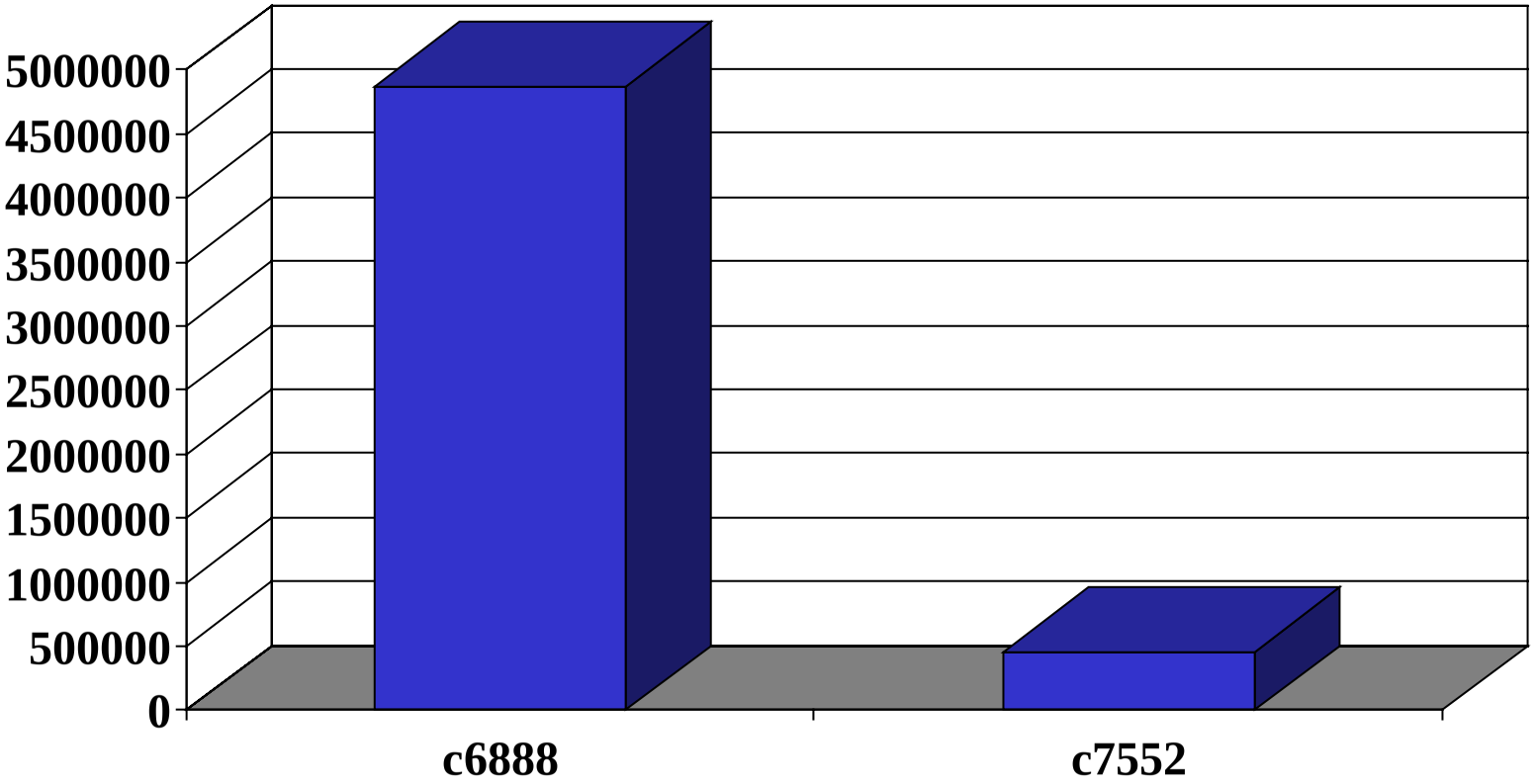
=> accurate bridging fault simulation at high speed

Additional transparencies

Experimental results - cs832



Experimental results - MIN, MAX values



increase of MIN, MAX values with 5 - valued simulation
(with respect to 3 - valued simulation)

Experimental results - summary