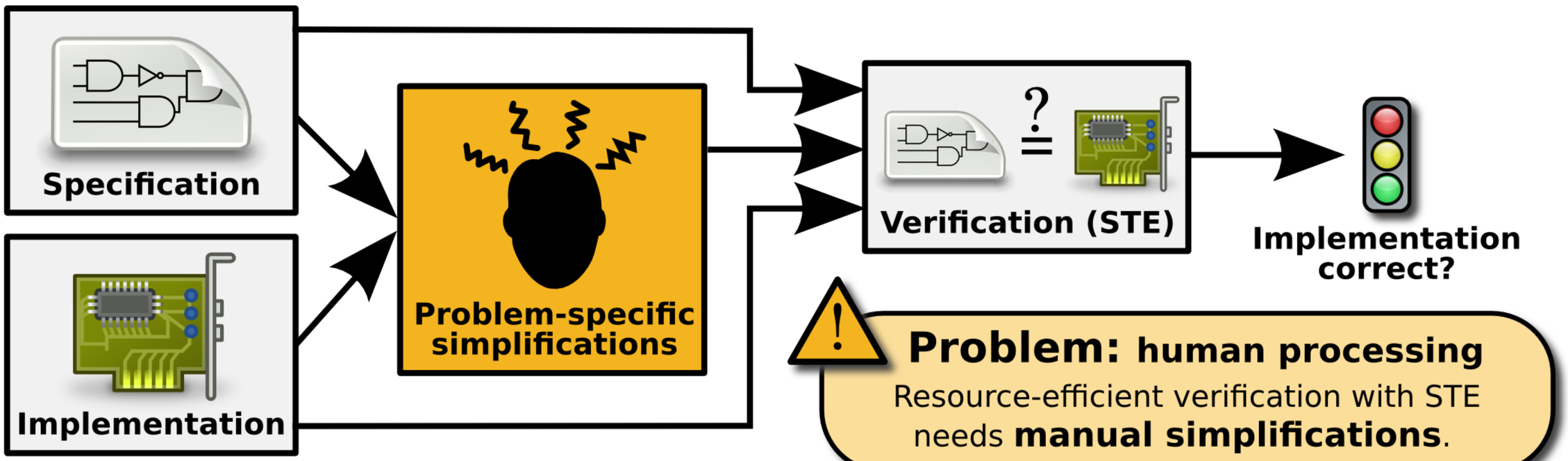
Automating Hardware Verification



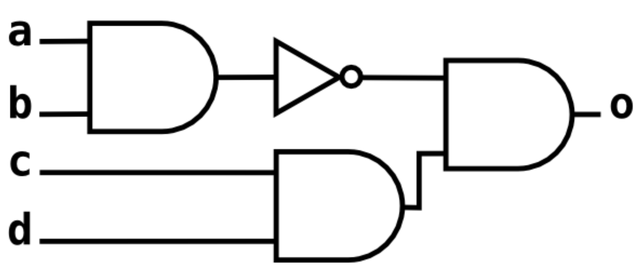
Our Contribution: automatic simplifications

Goal: Reduce number of cases the verification needs to examine

Idea: Determine which partial input combinations lead to a concrete output

Method: Work through the specification from the outputs to the inputs:
determine the different scenarios that lead to a true or false output
by propagating backwards which input stimuli are required in each step

Example:



Specification

a	b	c	d	o
X	X	X	0	0
X	X	0	X	0
1	1	X	X	0
X	0	1	1	1
0	X	1	1	1

before:
16 scenarios
now:
5 scenarios

Partial input combinations

Sufficiency: Partial input combinations cover all possible cases

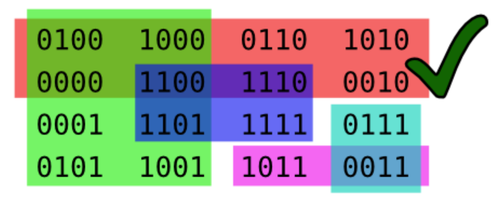
Our automatic simplifications cover all cases by construction. Manual simplifications have to be checked for full coverage.

Results: correct simplifications that enable the verification of more circuits

1. automatic simplifications as good as known manual ones

2. generates simplifications where no manual ones exist

3. automatic simplifications correct by construction



OXFORD UNIVERSITY COMPUTING LABORATORY

Sara Adams, Magnus Björk, Tom Melham, Carl-Johan Seger