ITMO UNIVERSITY

LULEÅ UNIVERSITY OF TECHNOLOGY

Aalto University

Reconstruction of Function Block Controllers Based on Test Scenarios and Verification

Daniil Chivilikhin
PhD
ITMO University

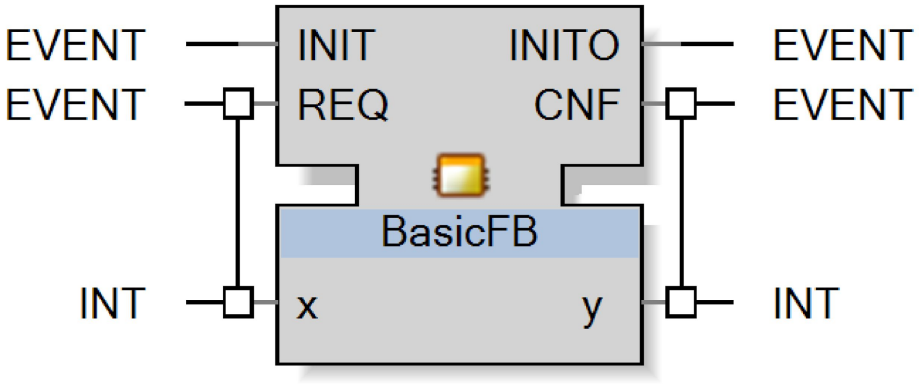
Ilya Ivanov
Undergrad student
ITMO University

Anatoly Shalyto
Dr. Sci., professor
ITMO University

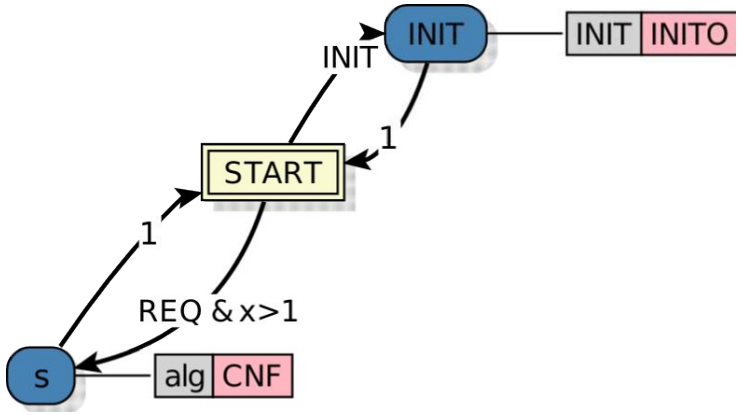
Valeriy Vyatkin
Dr. Eng., professor,
Aalto University,
Luleå University
of Technology

INDIN ’16, Futuroscope-Poitiers, France, July 19, 2016

IEC 61499 function blocks

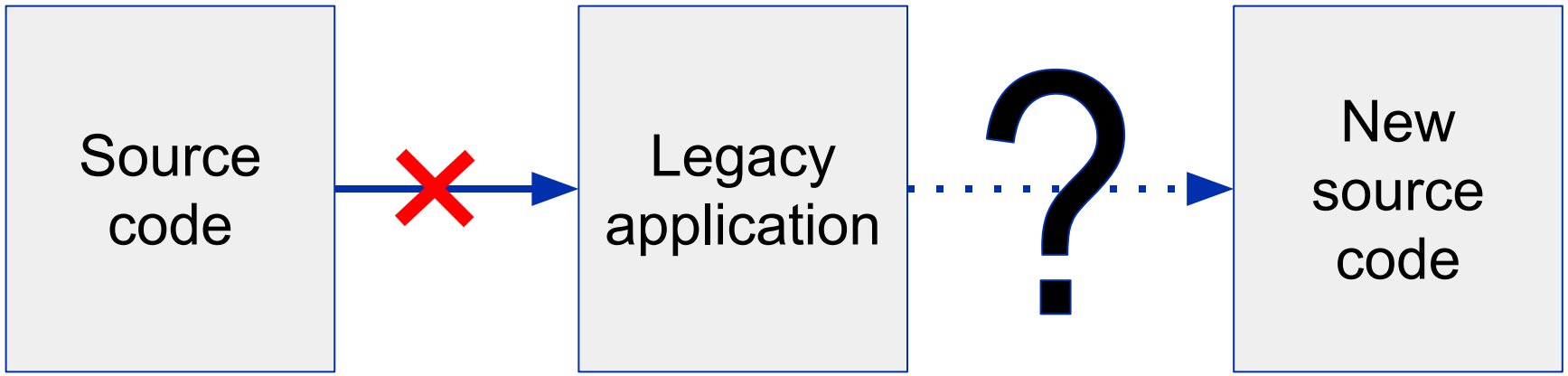


Function block interface

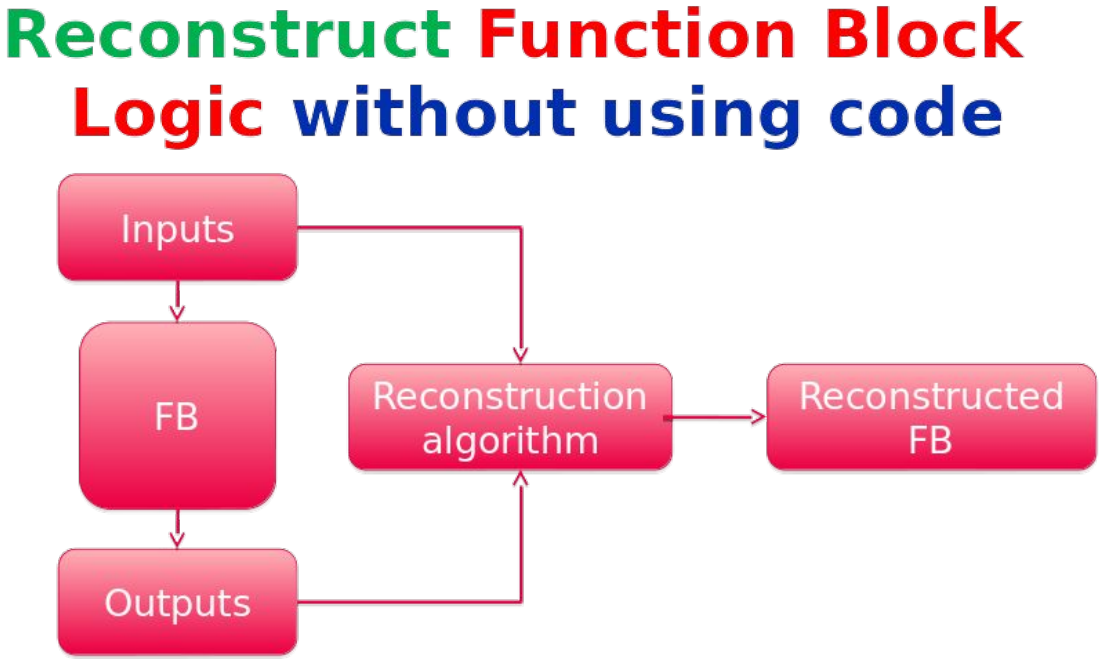


Execution Control Chart (ECC)

Motivation

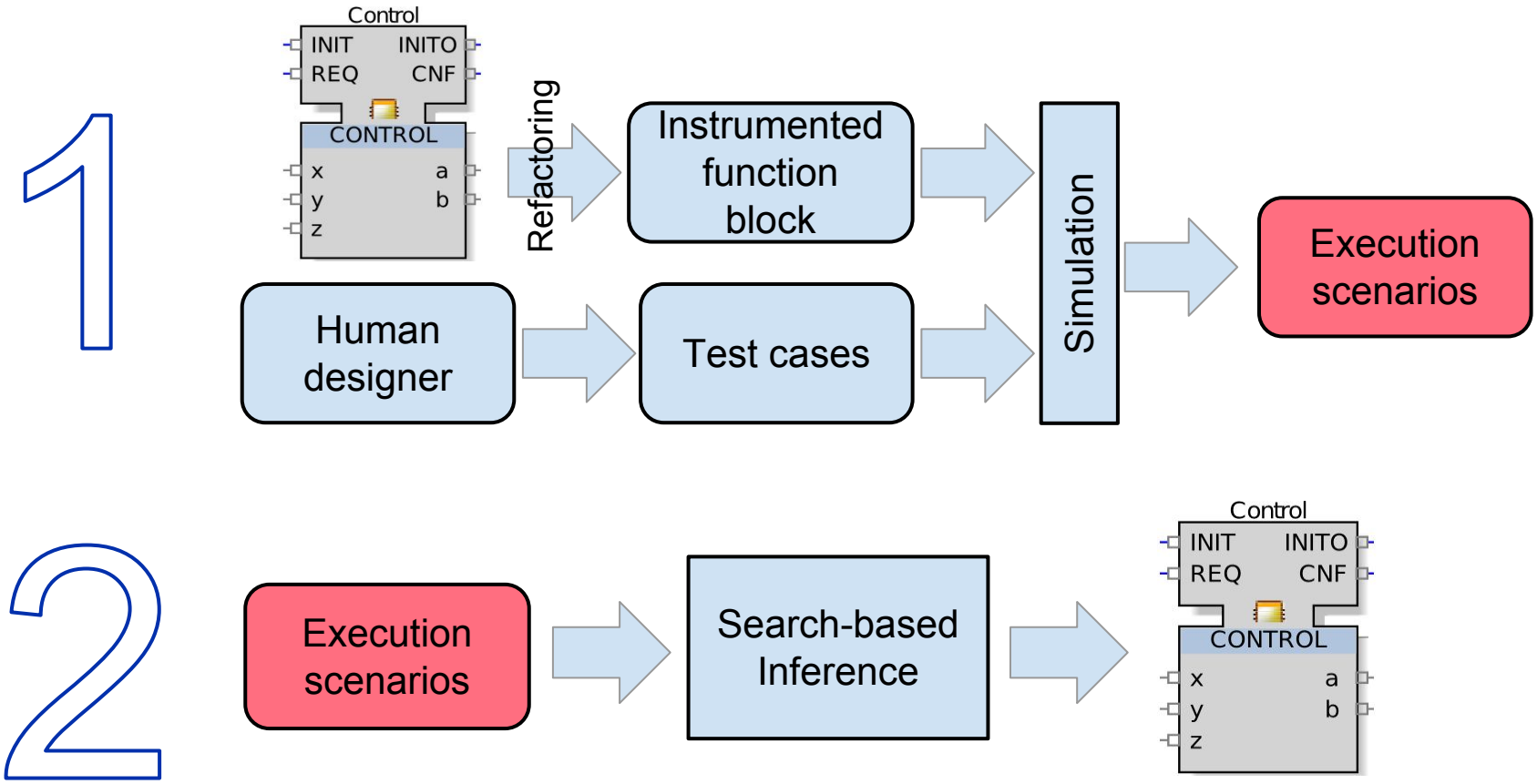


Previous work: Test-based FB reconstruction

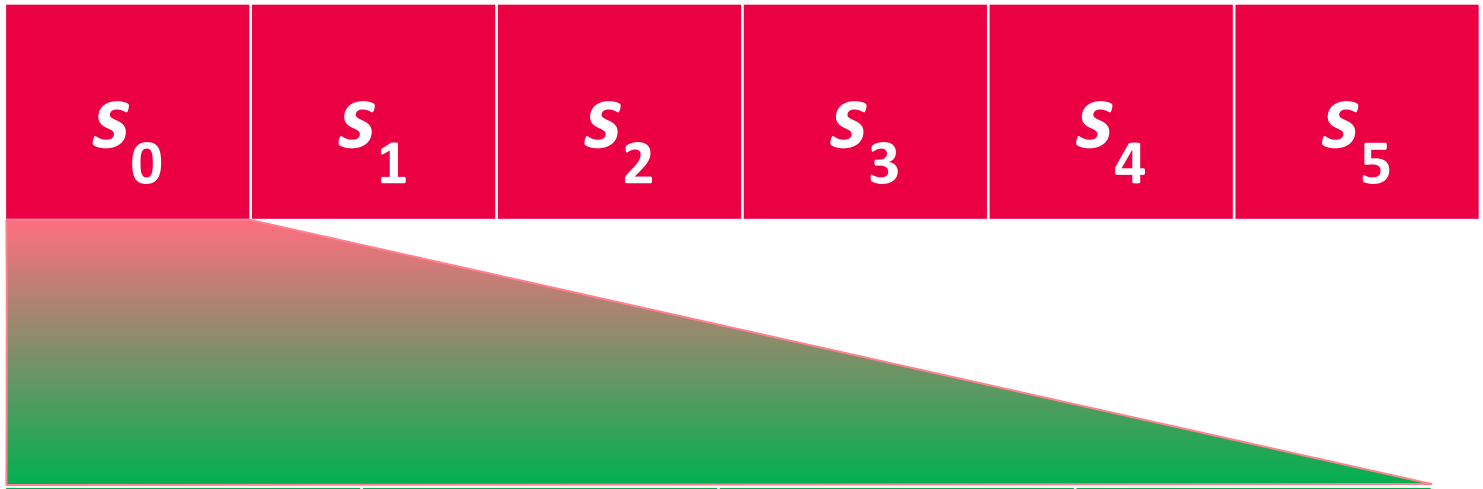


Chivilikhin D. et al. Reconstruction of Function Block Logic using Metaheuristic Algorithm: Initial Explorations / In Proceedings of INDIN'15

Test-based FB reconstruction

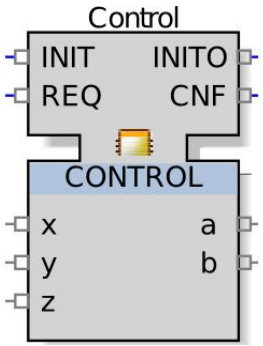


Execution scenario



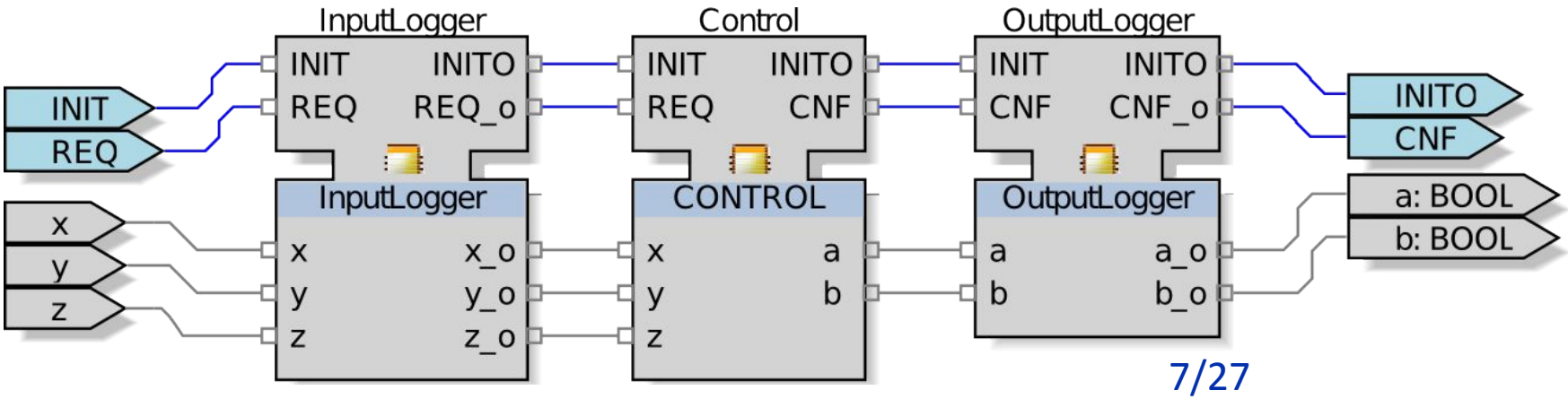
Input event	Input variables	Output variables	Output event
REQ	01011010	101010	CNF

Recording execution scenarios

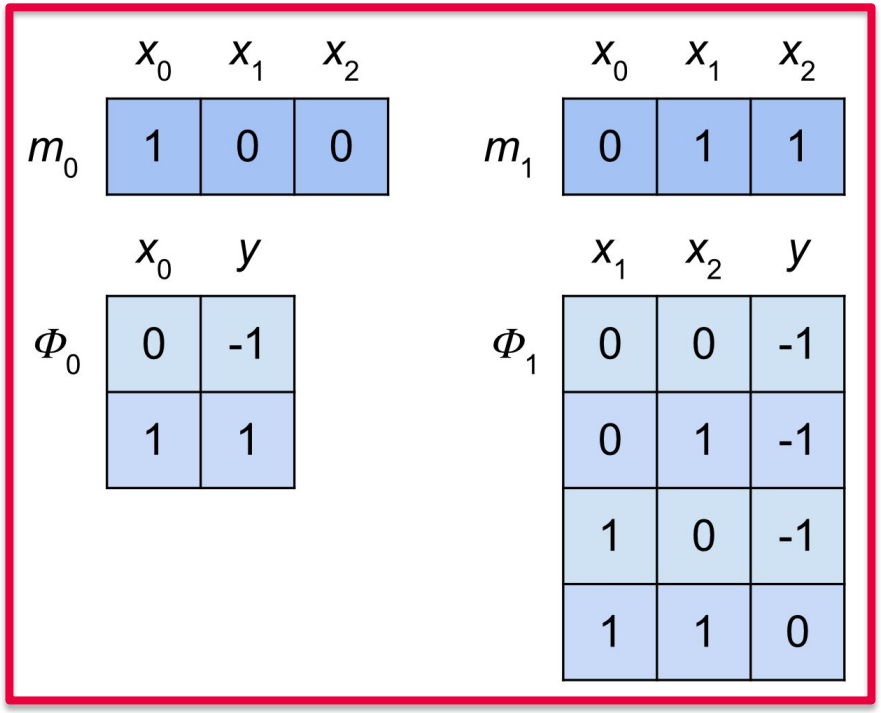


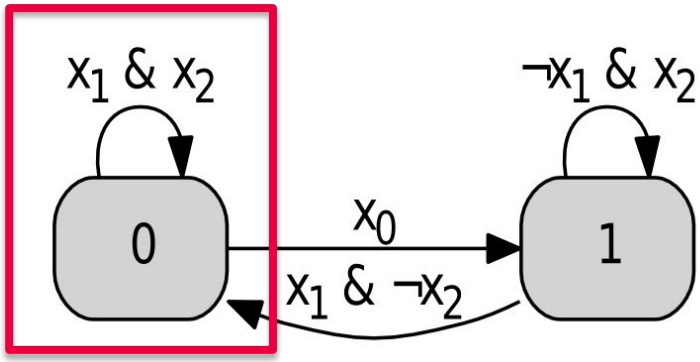
Original function block

Automated refactoring



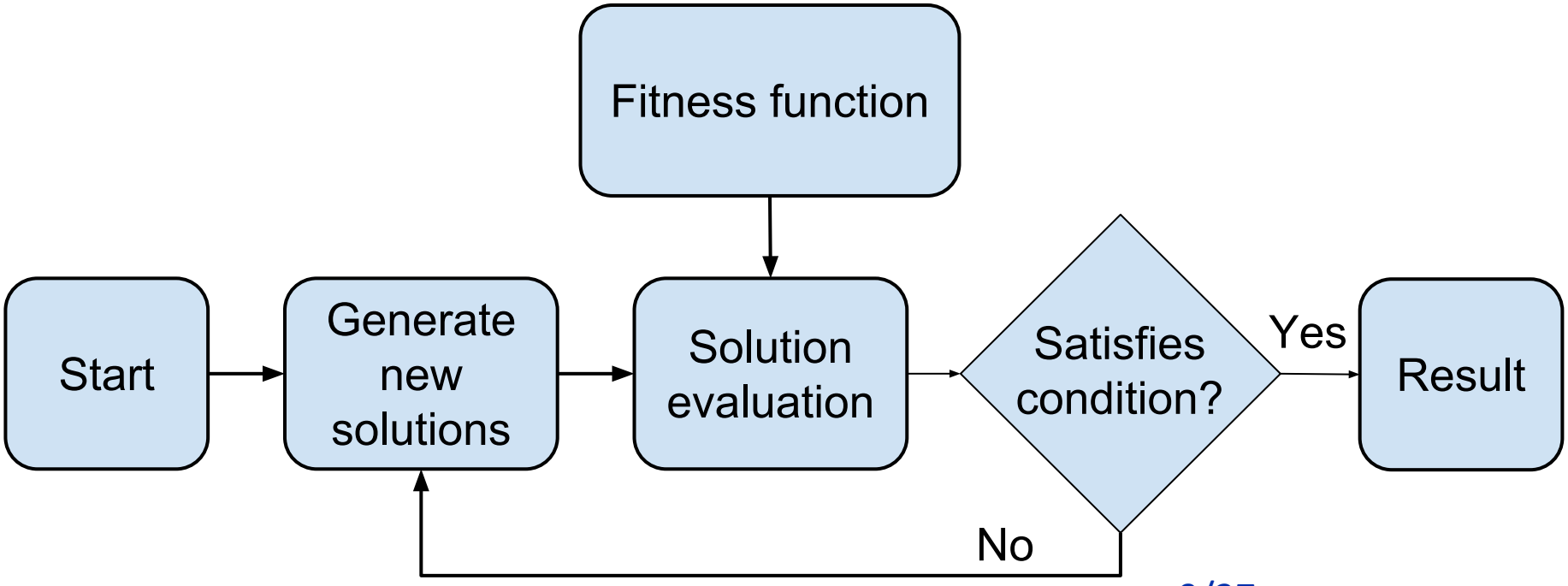
Solution representation





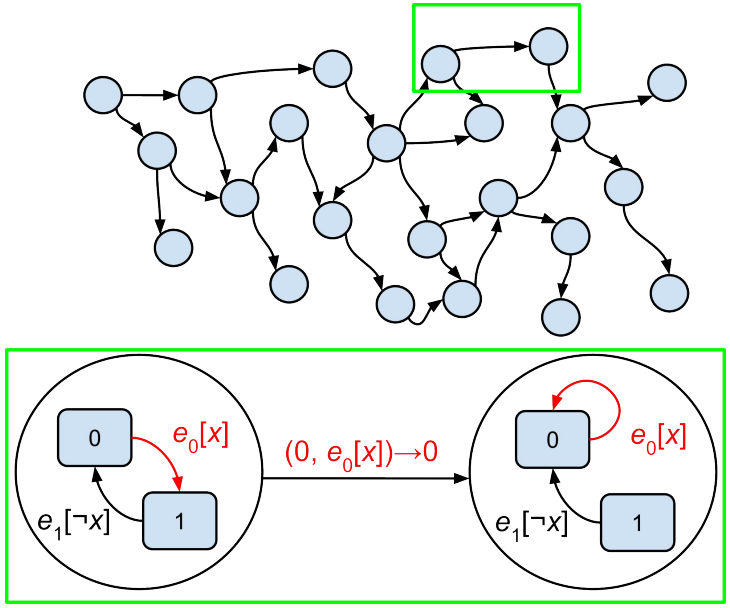
Inference algorithm (1)

- Parallel MuACO algorithm [Chivilikhin et al, 2014]

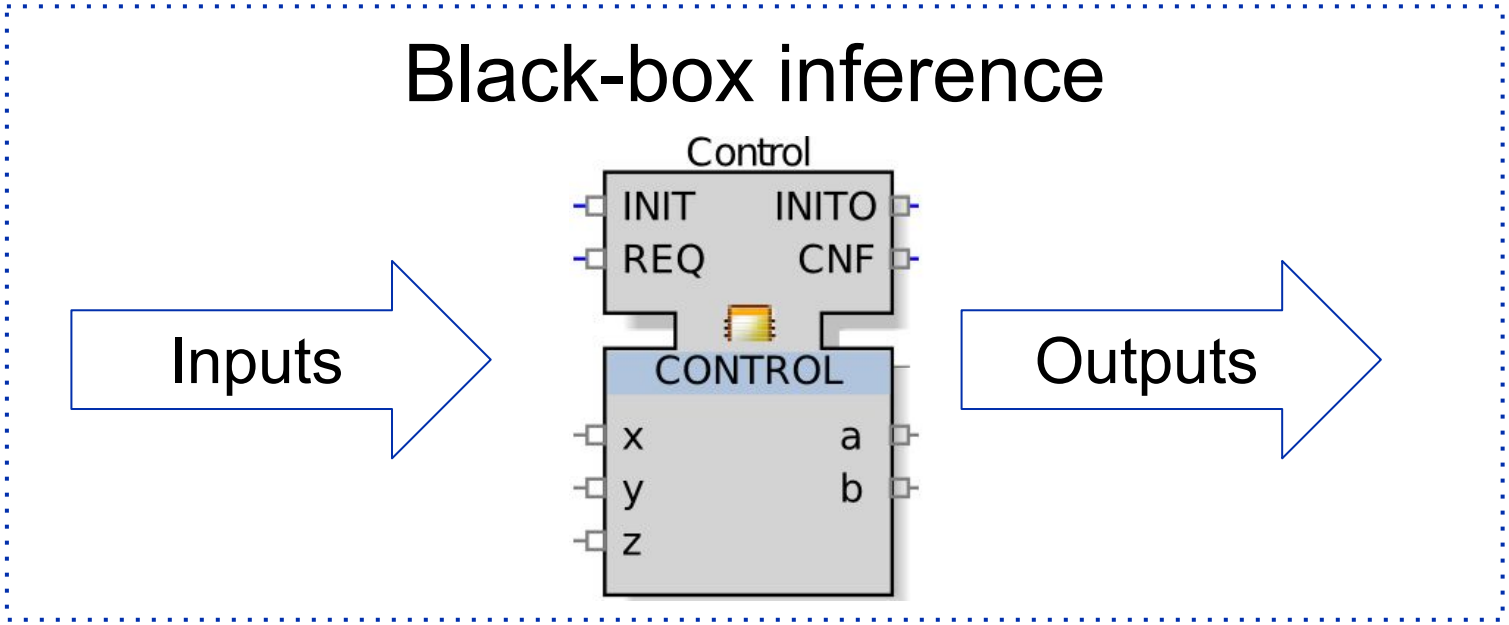


Inference algorithm (2)

1. Start with **random** solution
2. **Build** new solutions with **mutation operators**
3. **Evaluate** new solutions with **fitness function**



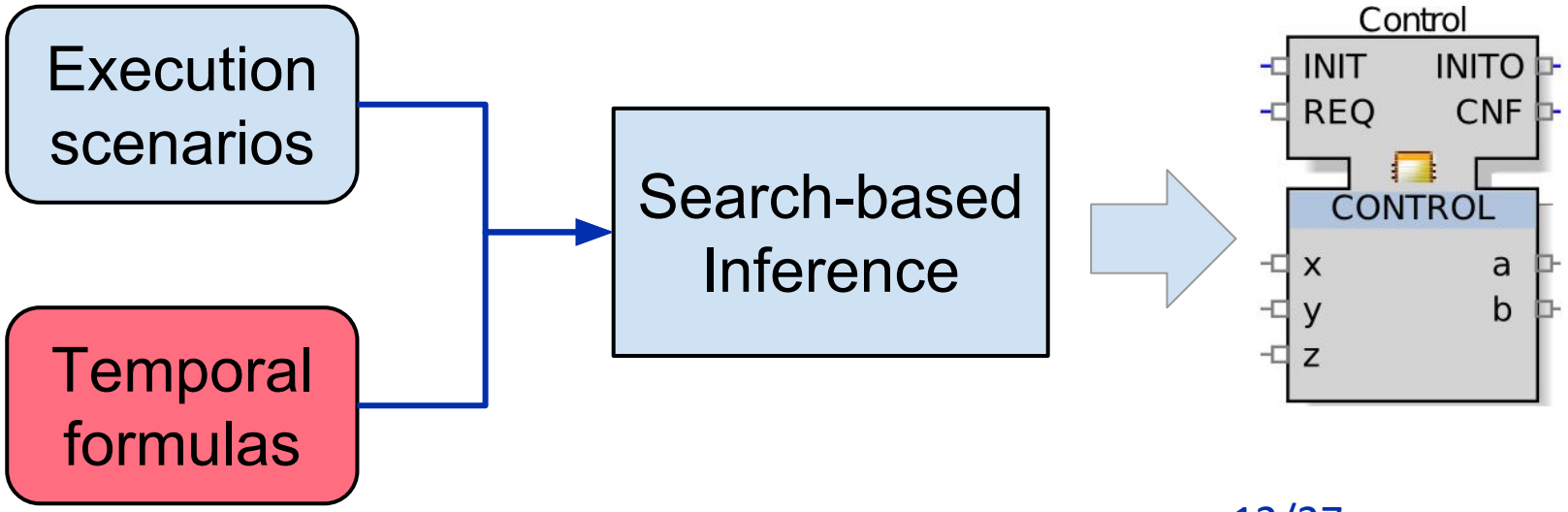
Issue with previous approach



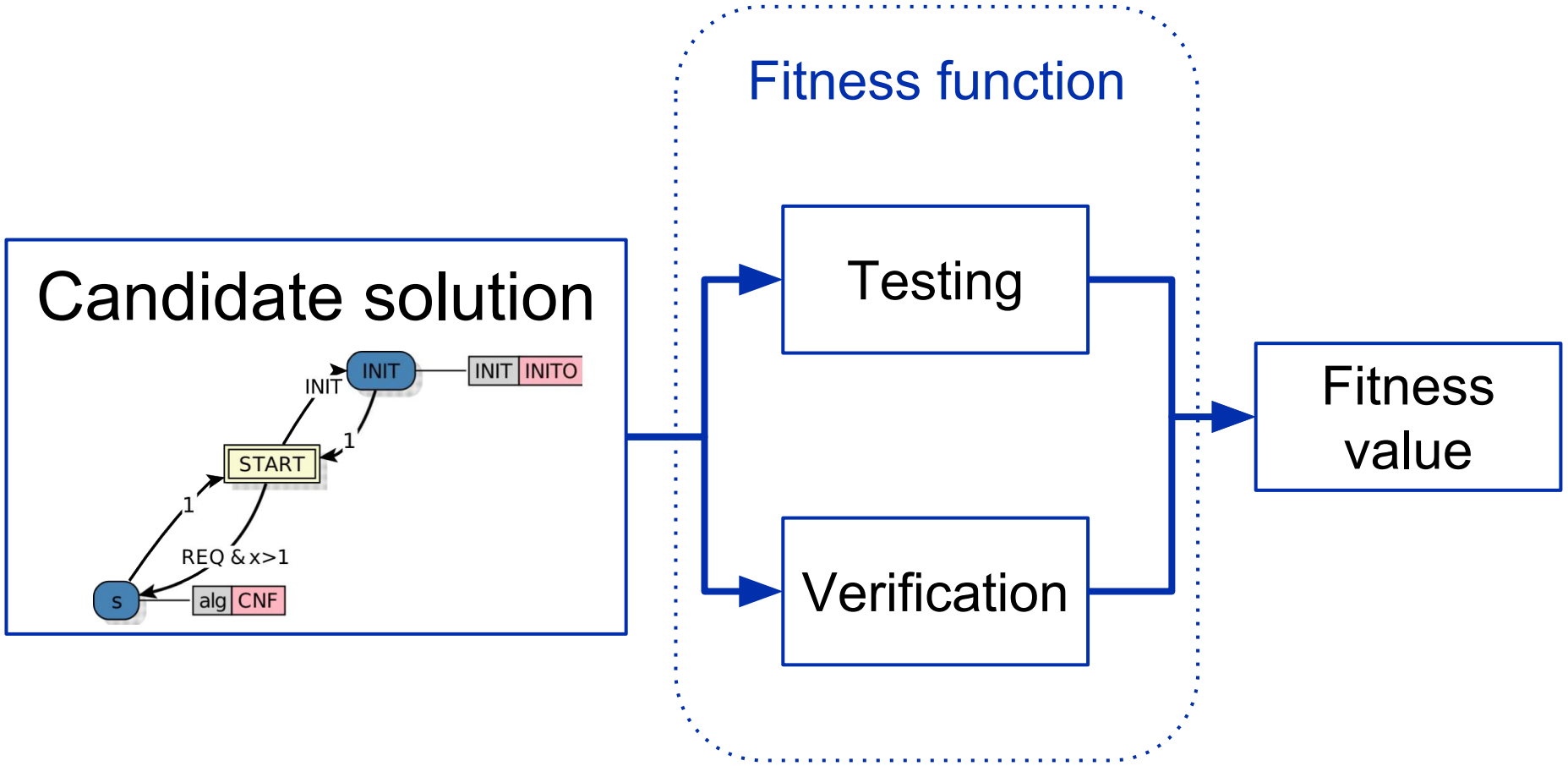
How do we ensure sufficient coverage?

Proposal

- Use **Temporal Logic formulas** as input
- We assume that these temporal properties cover the **most important** functionality of the FB



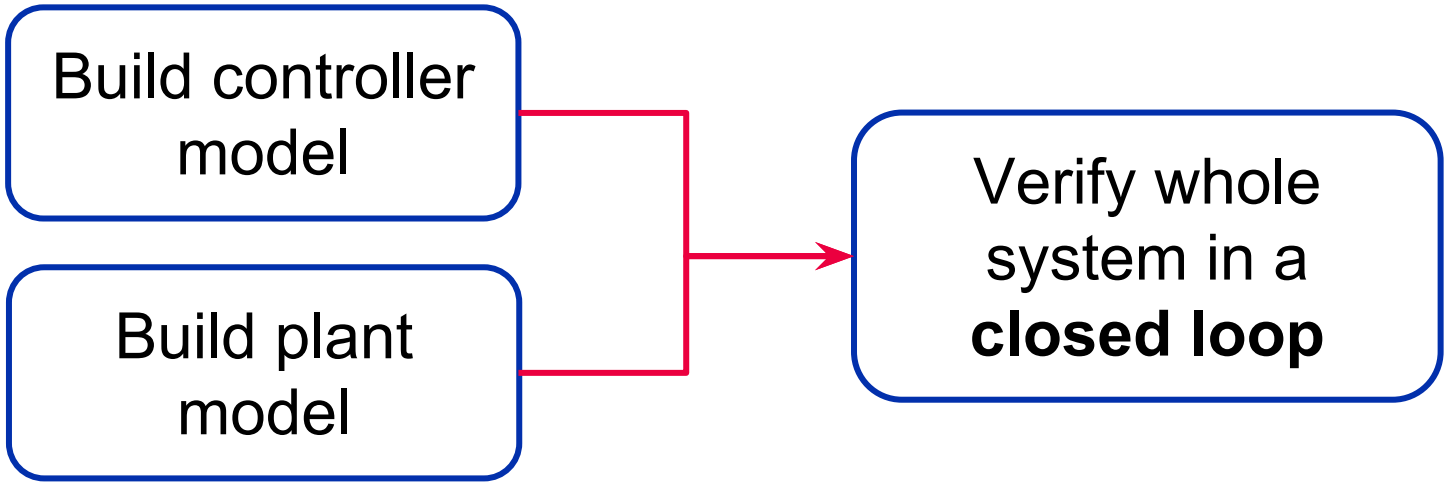
Essence of the approach



Which temporal logic to use?

- Linear temporal logic
- NuSMV is used for formula verification

Closed-loop verification



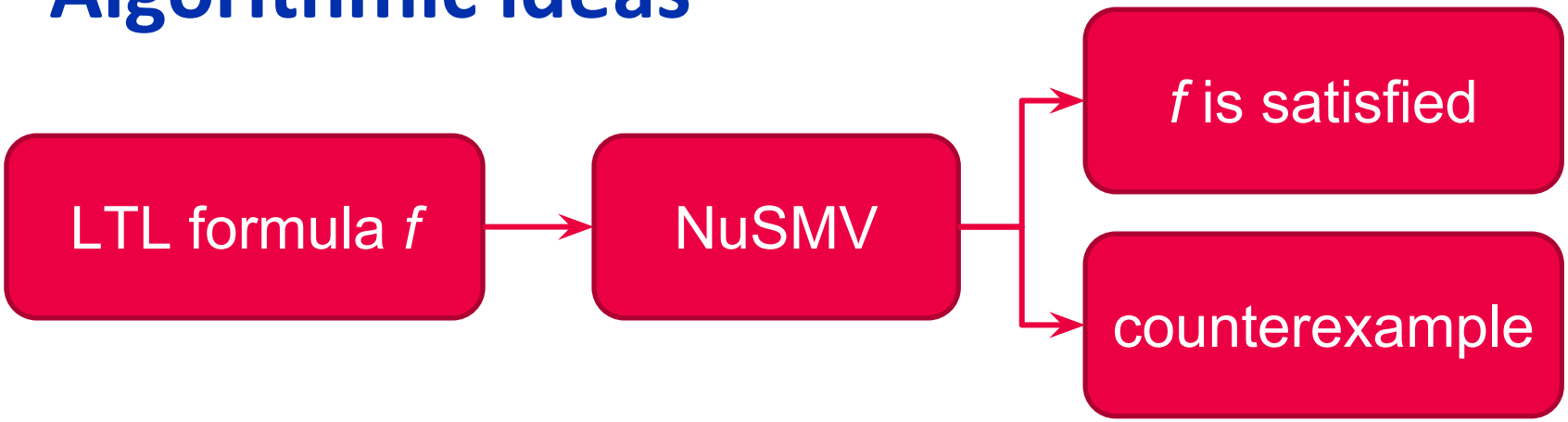
Issues

- We need the model of the plant
- Verification will take a lot of time
  - e.g., verification of PnP properties takes several hundred seconds

Closed-loop verification with surrogate plant model

- Solution – create small **surrogate** model
- Use the model for FB synthesis
  - “+”: fast verification
  - “-”: it may be nontrivial to create the model

Algorithmic ideas



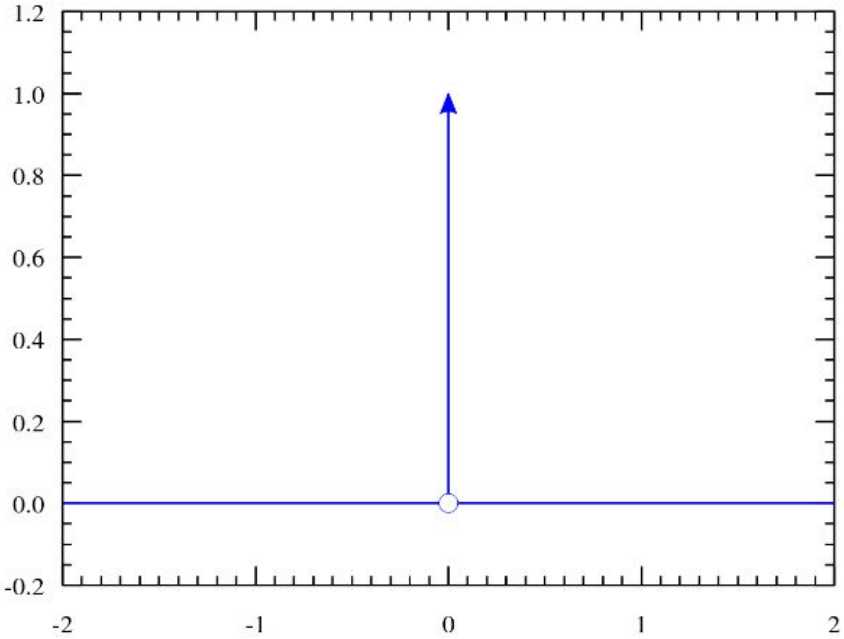
How can we use this information for FB synthesis?

1. Ratio of satisfied formulas
2. Longest counterexample length
3. Verification-aware mutation operator

1. Ratio of satisfied formulas

$$\frac{\#\text{ of satisfied formulas}}{\#\text{ of formulas}}$$

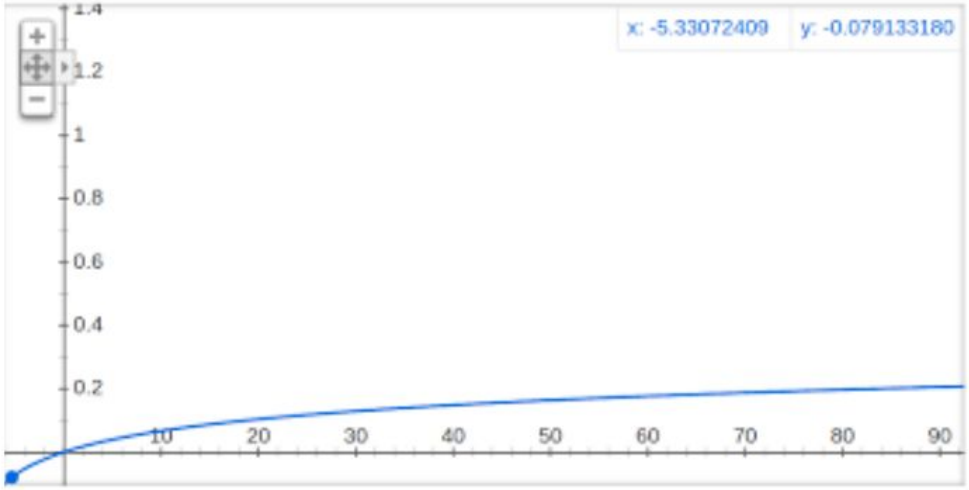
Issue – this variable has too few possible values!



2. Length of the longest counterexample

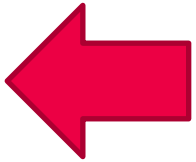
- Solutions with long counterexamples are probably better than solutions with short ones

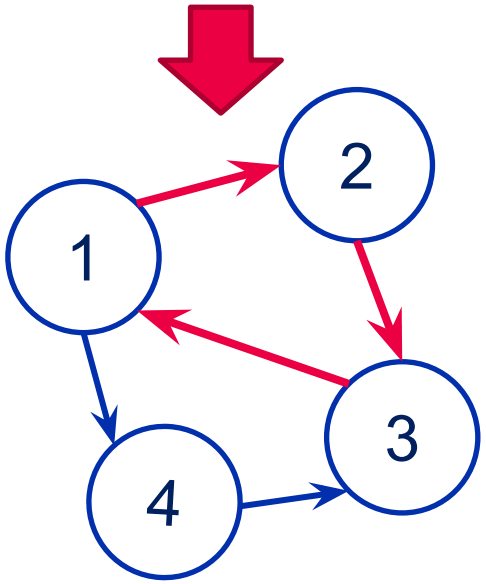
$$F^{\text{ce}}_{\text{smv}} = \begin{cases} 1, & \text{if } l_{\max} = 0; \\ 1 - \dfrac{1}{(1+\frac{1}{10} l_{\max})^{\frac{1}{10}}}, & \text{otherwise.} \end{cases}$$



3. Verification-aware mutation operator

Increase probability of changing visited transitions





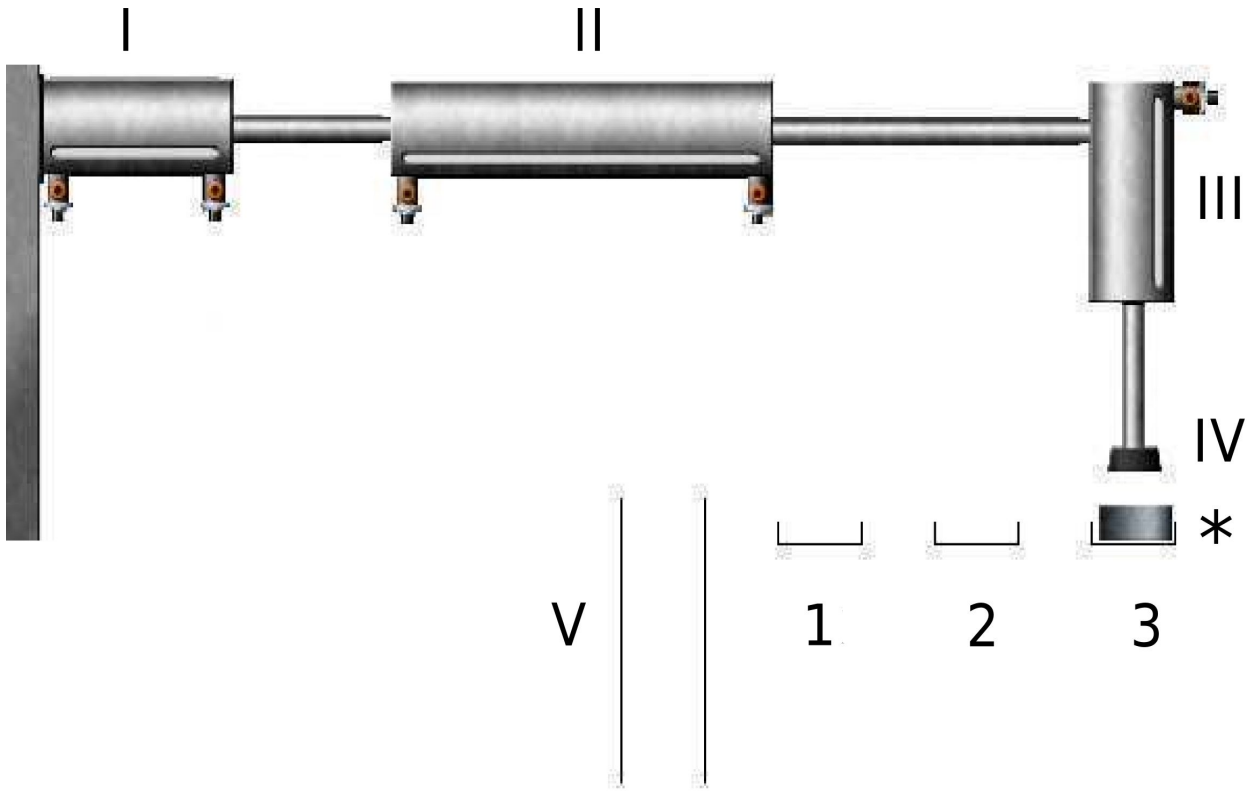
Efficiency issues

- Closed-loop with surrogate model is fast, but not fast enough
  - Verification takes ~ 0.5-1 seconds

Solution

- Calculate verification-based fitness only for *p* % of solutions
- Definitely calculate for good solutions

Example: Pick-and-Place manipulator



Considered LTL properties

Property	Description
G(not (c_1Extend & c_1Retract))	Cylinder I must never be given commands to extend and retract simultaneously
G(not (c_2Extend & c_2Retract))	Analogous safety property about cylinder II
G($pp_1 \rightarrow$ not F(vp_1))	If a work piece appears on the first input track, it will eventually be picked up by the manipulator

Experiments

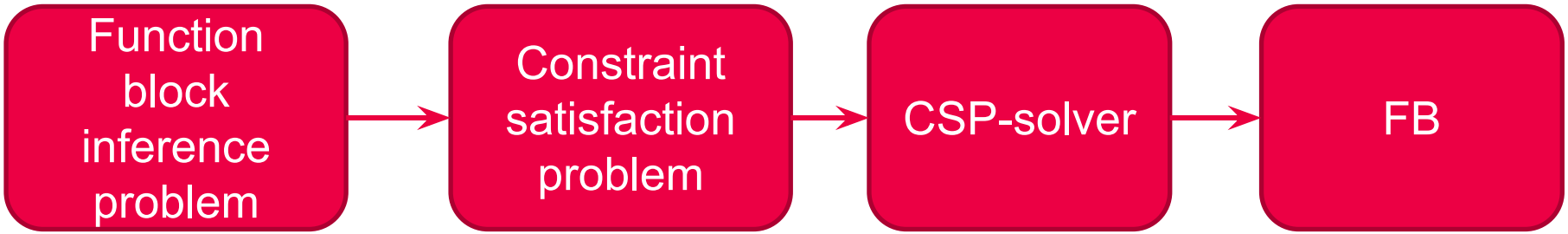
- Machine with 64-core AMD OpteronTM6378 @ 2.4 GHz processor, 32 Gb of RAM
- Used 16 cores

#	Configuration	Time, s				Satisfied all LTL
		min	mean	median	max	
1	Scenario	32	85	81	280	0%
2	Scenario + LTL (no F^{ce}_{smv})	222	752	656	1689	100%
3	Scenario + LTL	164	563	561	1575	100%

Conclusion

- Developed method of FB inference from tests and LTL properties
- Demonstrated viability on the PnP example
- Still a long way to go…

Ongoing work: CSP-based inference

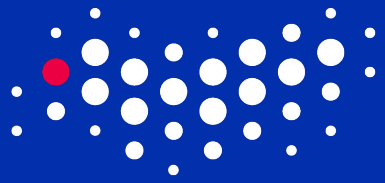


Advantages

- Typically – very fast
- Possibility to find all solutions
- Symmetry breaking
- Indirect solution of “tests + LTL” synthesis problem

Acknowledgements

- Financially supported by Government of Russian Federation, Grant 074-U01.
- Partially funded by RFBR according to the research project No. 16-37-00205 mol_a.



ITMO UNIVERSITY

Thank you for your attention!

rain.ifmo.ru/~chivdan