Testing the reliability of systems with unstable or low-quality network connectivity

Miroslav Bures, CTU in Prague

Red Hat Research Day Europe, Brno, Sep 16th 2022

Our lab

STILL

System Testing IntelLigent Lab

System **T**esting **I**ntel**L**igent **L**ab
Dept. of Computer Science
Faculty of Electrical Engineering
Czech Technical University in Prague

What we do

Better tools for system testing

- Faster testing process able to detect an increased amount of defects
- **Complex software, electronics, IoT, mission-critical systems**

IoT projects in integrated rescue system, medicine and defence

- Czech Army, NATO ACT Innovation Hub, University of Defence, ...

Selected projects

ŠKODA

Automated model-based generation of test scenarios for integration and end testing of automobiles

POLICIE ČESKÉ REPUBLIKY

ČZU

VSB TECHNICAL UNIVERSITY OF OSTRAVA

Test strategy and test automation for IoT-based rescue mission planning and management system

JOHNS HOPKINS UNIVERSITY

DefSec® INNOVATION HUB

Monitoring of soldier vital functions to allow for more accurate triage and to minimize casualties

Too much dependent on a data network?

Reliability of a system operating with a limited network

IoT and complex software systems

Live example: Our Digital Triage Assistant (DTA) project

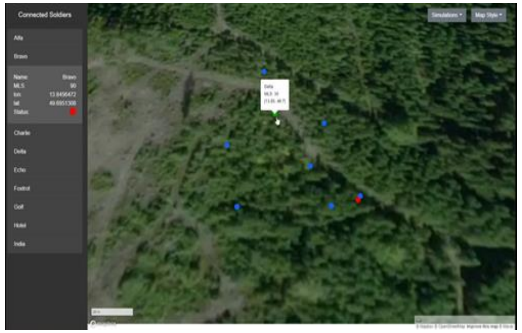
Soldiers' body sensors

Soldier's status estimation

DTA back-end *(can be mobile)*

Integration with other defense systems

Smart glasses



User interface

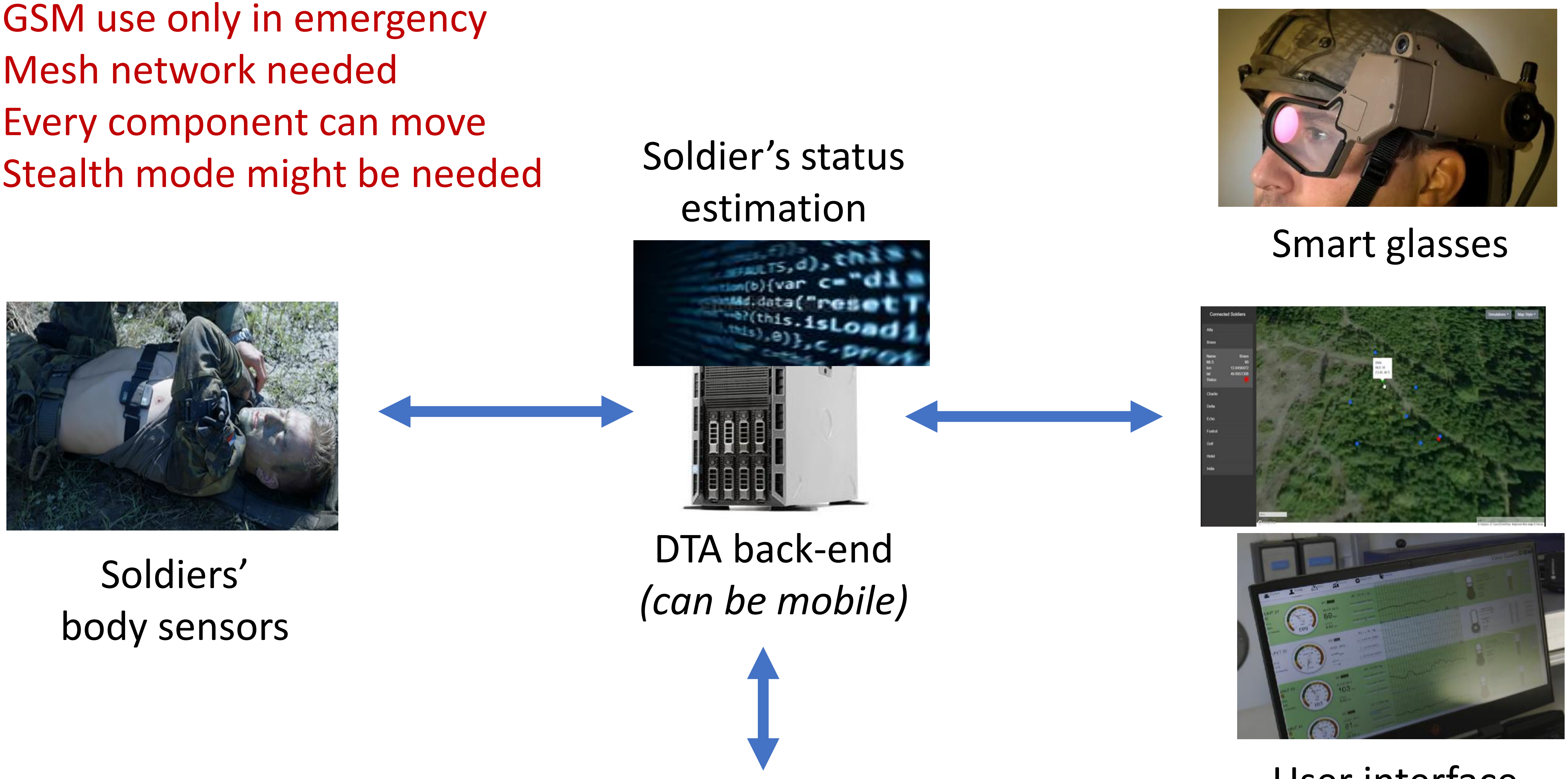
Forget about quality 5G network

GSM use only in emergency

Mesh network needed

Every component can move

Stealth mode might be needed



Soldier’s status estimation

Smart glasses

Soldiers’ body sensors

DTA back-end *(can be mobile)*

User interface

Integration with other defense systems

Weak network situation examples (video)

Source: NATO Multimedia and University of Defence

Typical challenges

Stealth mode ON / OFF

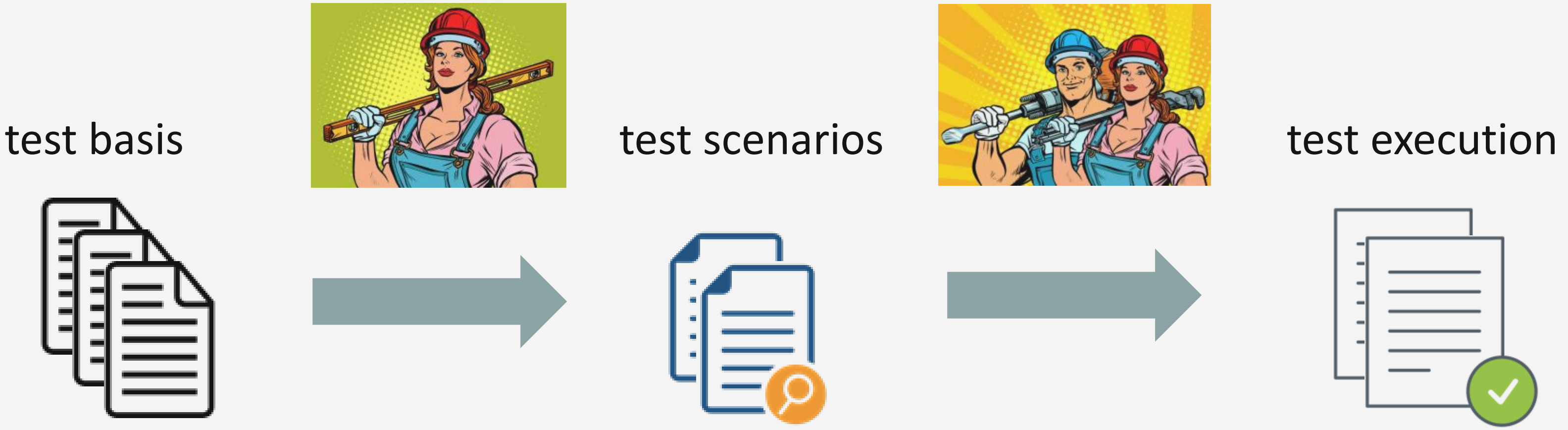
A mesh network + terrain →

- Low bandwidth
- Intermittent connection
- Connectivity disrupted and restored

SYSTEM PROCESSES MUST RUN IN A RELIABLE WAY

Test automation now

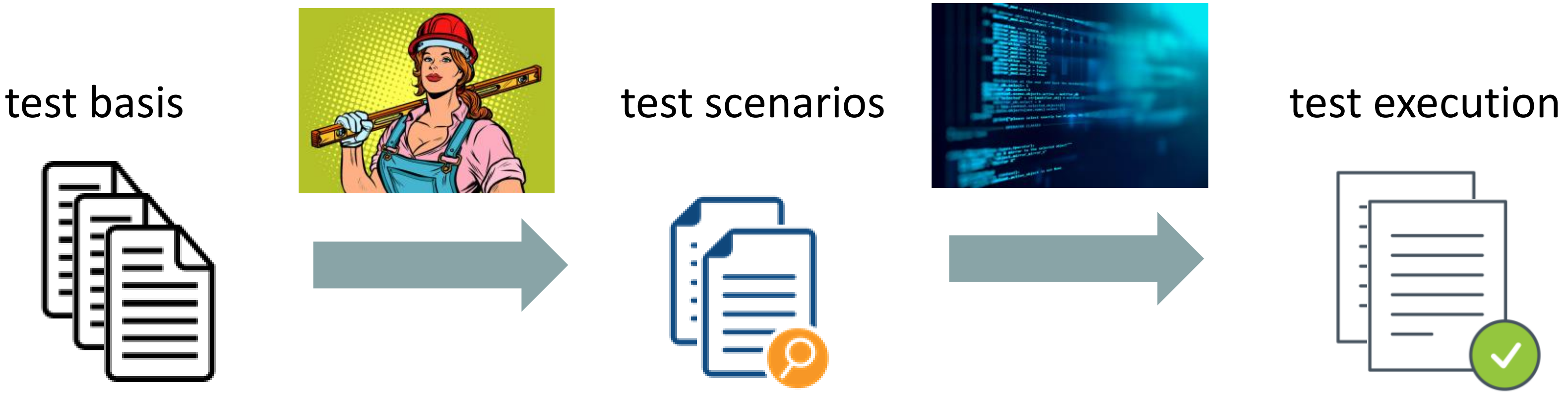
Manual testing



test basis

test scenarios

test execution

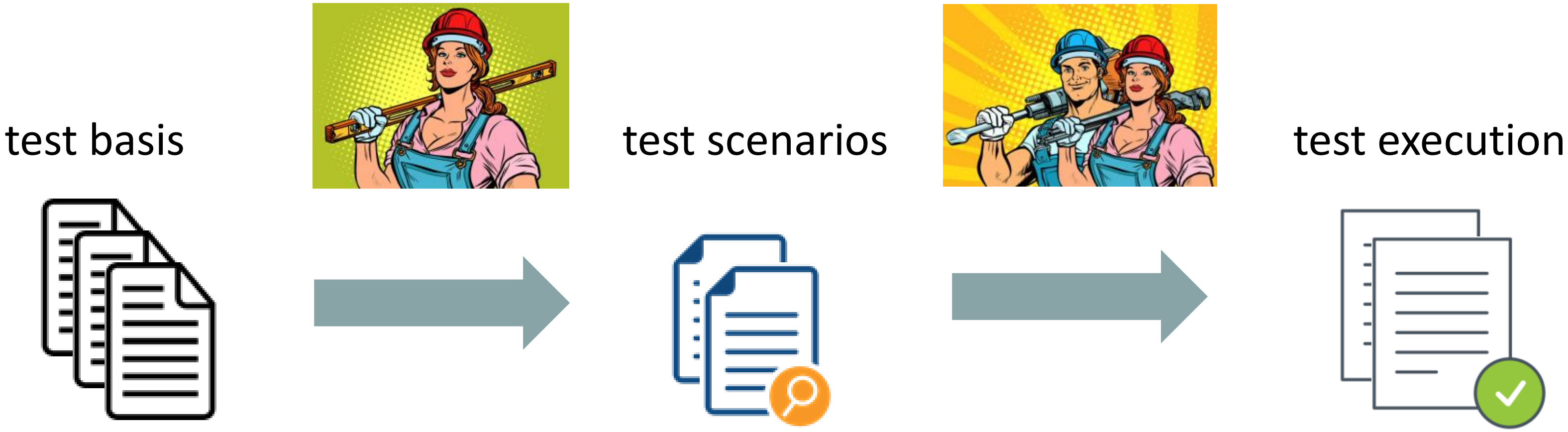


test basis

test scenarios

test execution

Test automation now

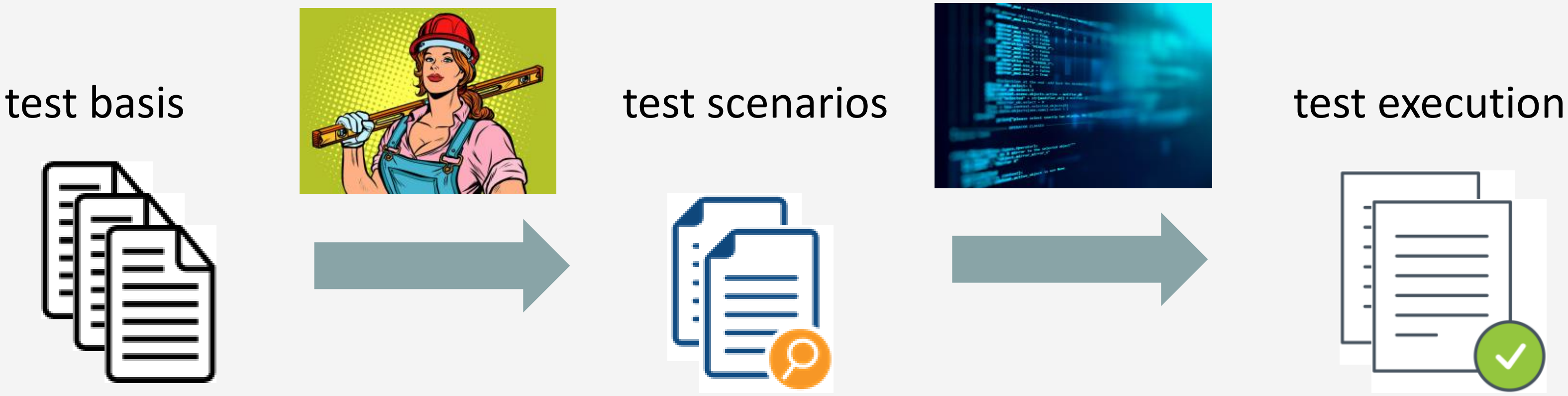


test basis

test scenarios

test execution

Common test automation



test basis

test scenarios

test execution

Test automation now & future

test basis

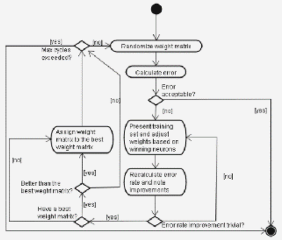
test scenarios

test execution

MBT way

test basis

SUT model


Randomize weight matrix
Calculate error
Error acceptable?
[yes]
[no]
Present training set and adjust weights based on winning neurons
Recalculate error rate and note improvements
Error rate improvement trivial?
[no]
[yes]
Have a best weight matrix?
[yes]
[no]
Better than the best weight matrix?
[yes]
[no]
Assign weight matrix to the best weight matrix
Max cycles exceeded?
[yes]
[no]

test scenarios

test execution

Test automation in the future?

test basis

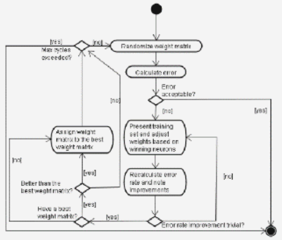
test scenarios

test execution

Future?

test basis

SUT model



Randomize weight matrix
Calculate error
Error acceptable?
[yes]
[no]
Present training set and adjust weights based on winning neurons
Recalculate error rate and note improvements
Error rate improvement trivial?
[no]
[yes]
Have a best weight matrix?
[yes]
[no]
Better than the best weight matrix?
[yes]
[no]
Assign weight matrix to the best weight matrix
Max cycles exceeded?
[yes]
[no]

test scenarios

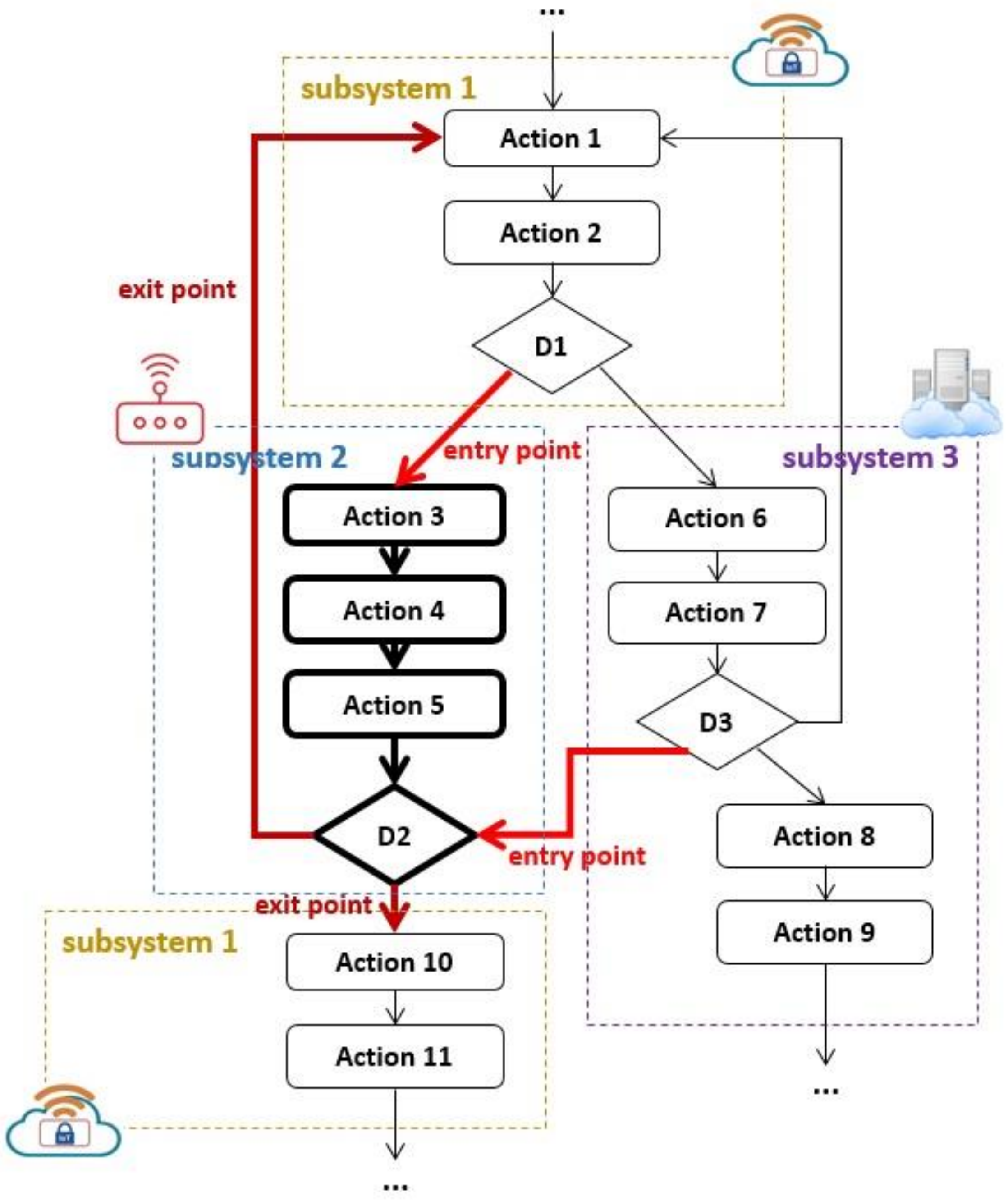
test execution

How to test it?

A process flow model:

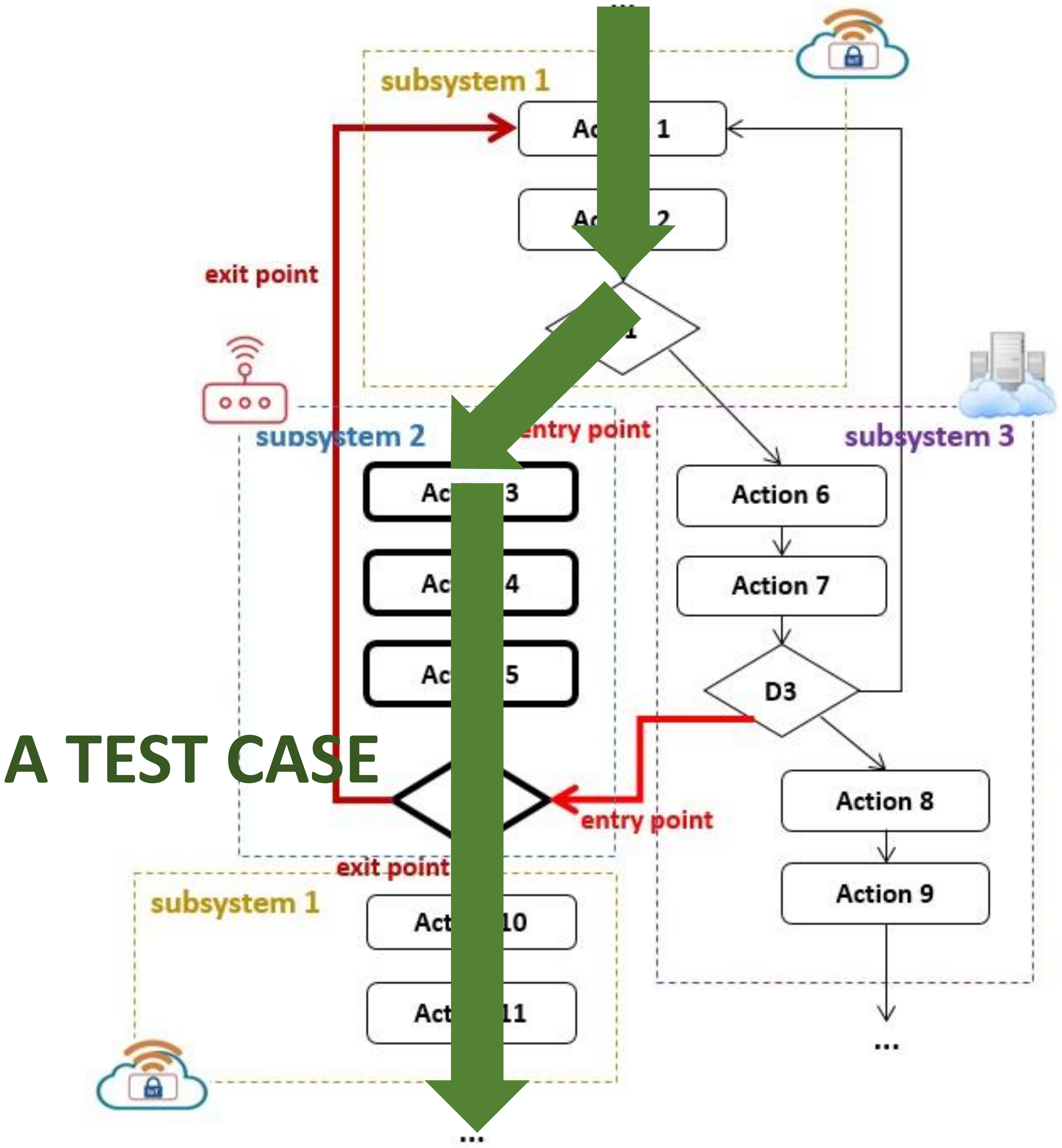
Process parts handled by devices / system modules

Model of network outage probability



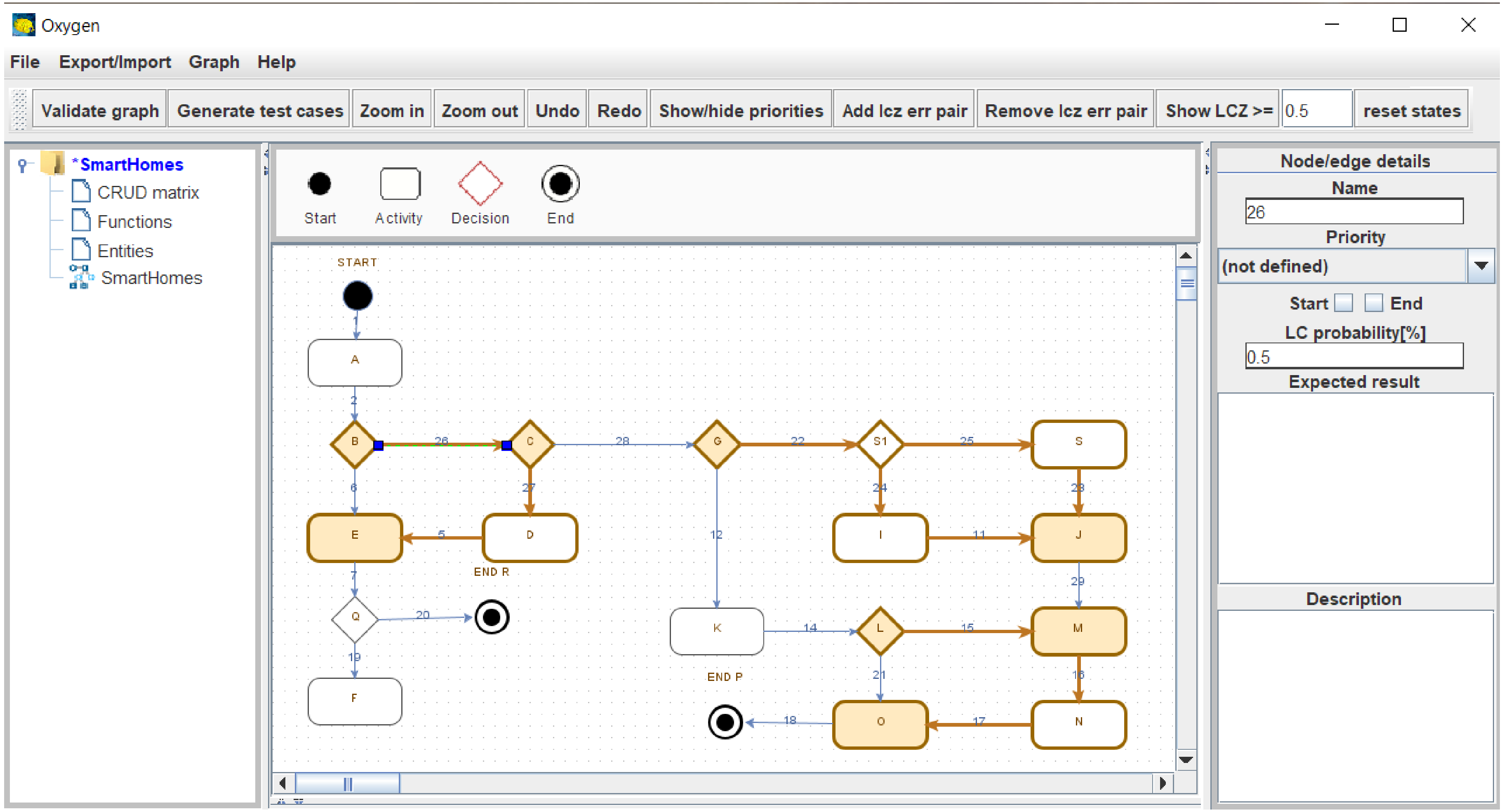
How to test it?

Test the process by paths through it



System modeling

Oxygen platform by STILL



Show LCZ >= 0.5

Name: 26

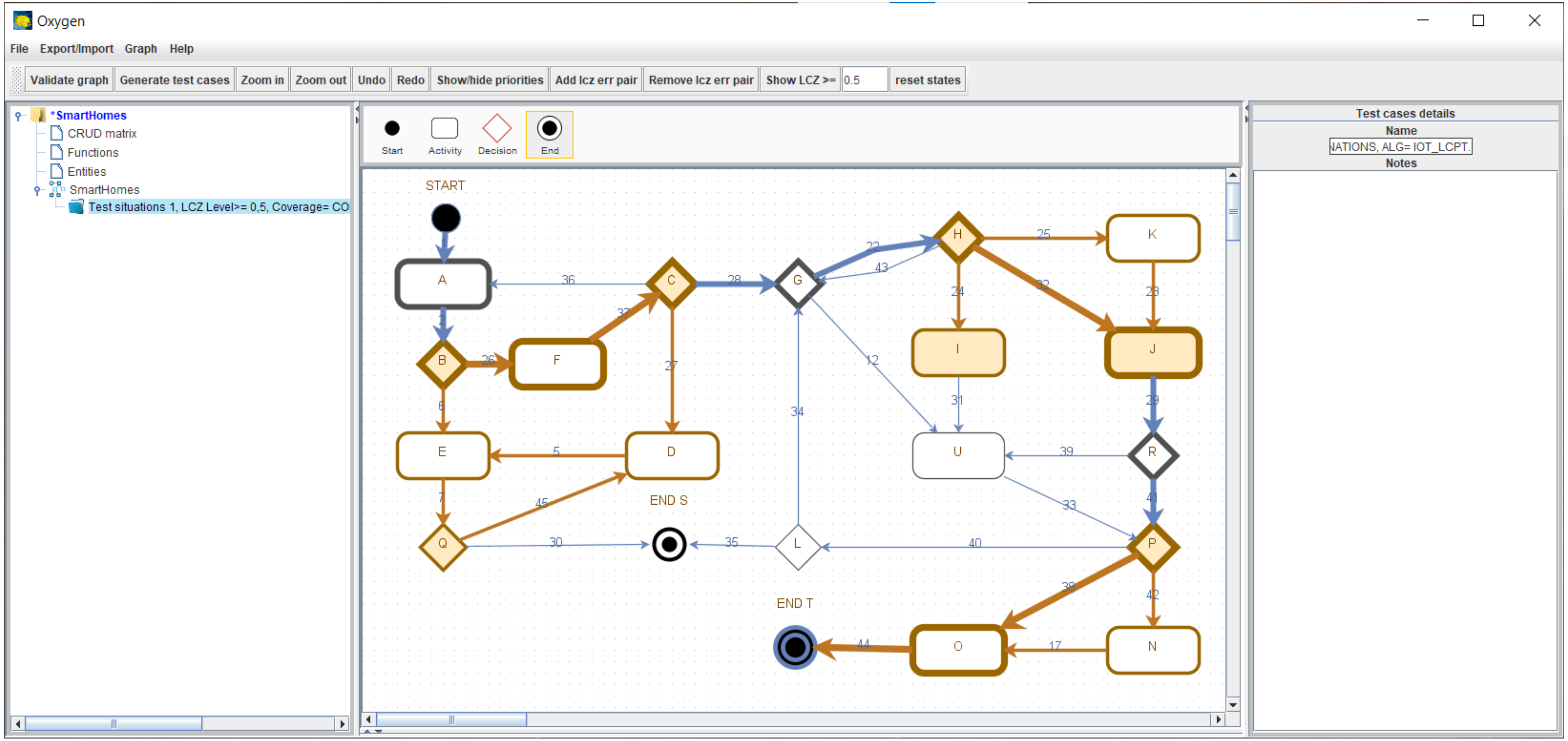
Priority: (not defined)

Start □ □ End

LC probability[%]: 0.5

Expected result

Test case visualization



Show LCZ >= 0.5

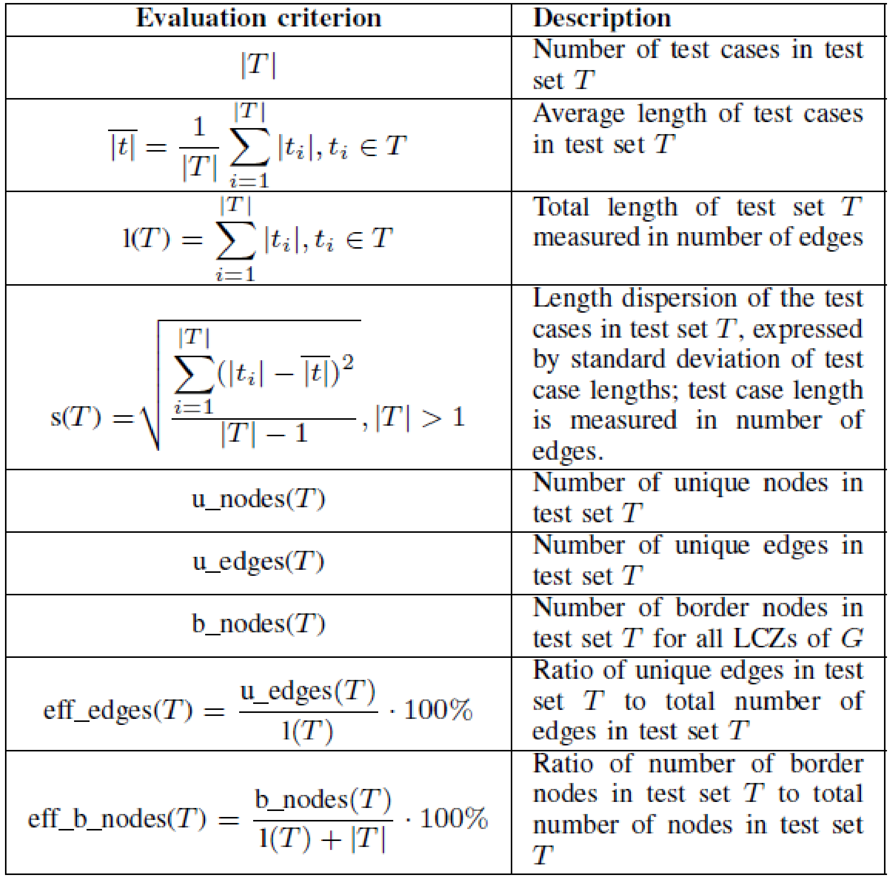
Test situations 1, LCZ Level>= 0,5, Coverage= CO

Name: NATIONS, ALG= IOT_LCPT.

What is a “good test set” ?

4 test coverage criteria

Evaluation criteria



Evaluation criterion	Description
$\lvert T \rvert$	Number of test cases in test set T
$\overline{\lvert t \rvert} = \frac{1}{\lvert T \rvert} \sum_{i=1}^{\lvert T \rvert} \lvert t_i \rvert, t_i \in T$	Average length of test cases in test set T
$\mathrm{l}(T) = \sum_{i=1}^{\lvert T \rvert} \lvert t_i \rvert, t_i \in T$	Total length of test set T measured in number of edges
$\mathrm{s}(T) = \sqrt{\frac{\sum_{i=1}^{\lvert T \rvert} (\lvert t_i \rvert - \overline{\lvert t \rvert})^2}{\lvert T \rvert - 1}}, \lvert T \rvert > 1$	Length dispersion of the test cases in test set T, expressed by standard deviation of test case lengths; test case length is measured in number of edges.
u_nodes(T)	Number of unique nodes in test set T
u_edges(T)	Number of unique edges in test set T
b_nodes(T)	Number of border nodes in test set T for all LCZs of G
$\mathrm{eff\_edges}(T) = \frac{\mathrm{u\_edges}(T)}{\mathrm{l}(T)} \cdot 100\%$	Ratio of unique edges in test set T to total number of edges in test set T
$\mathrm{eff\_b\_nodes}(T) = \frac{\mathrm{b\_nodes}(T)}{\mathrm{l}(T) + \lvert T \rvert} \cdot 100\%$	Ratio of number of border nodes in test set T to total number of nodes in test set T

How to compute it?

Number of algorithms possible

AI gives good results, e.g.

Artificial ANT colony

Genetic algorithm

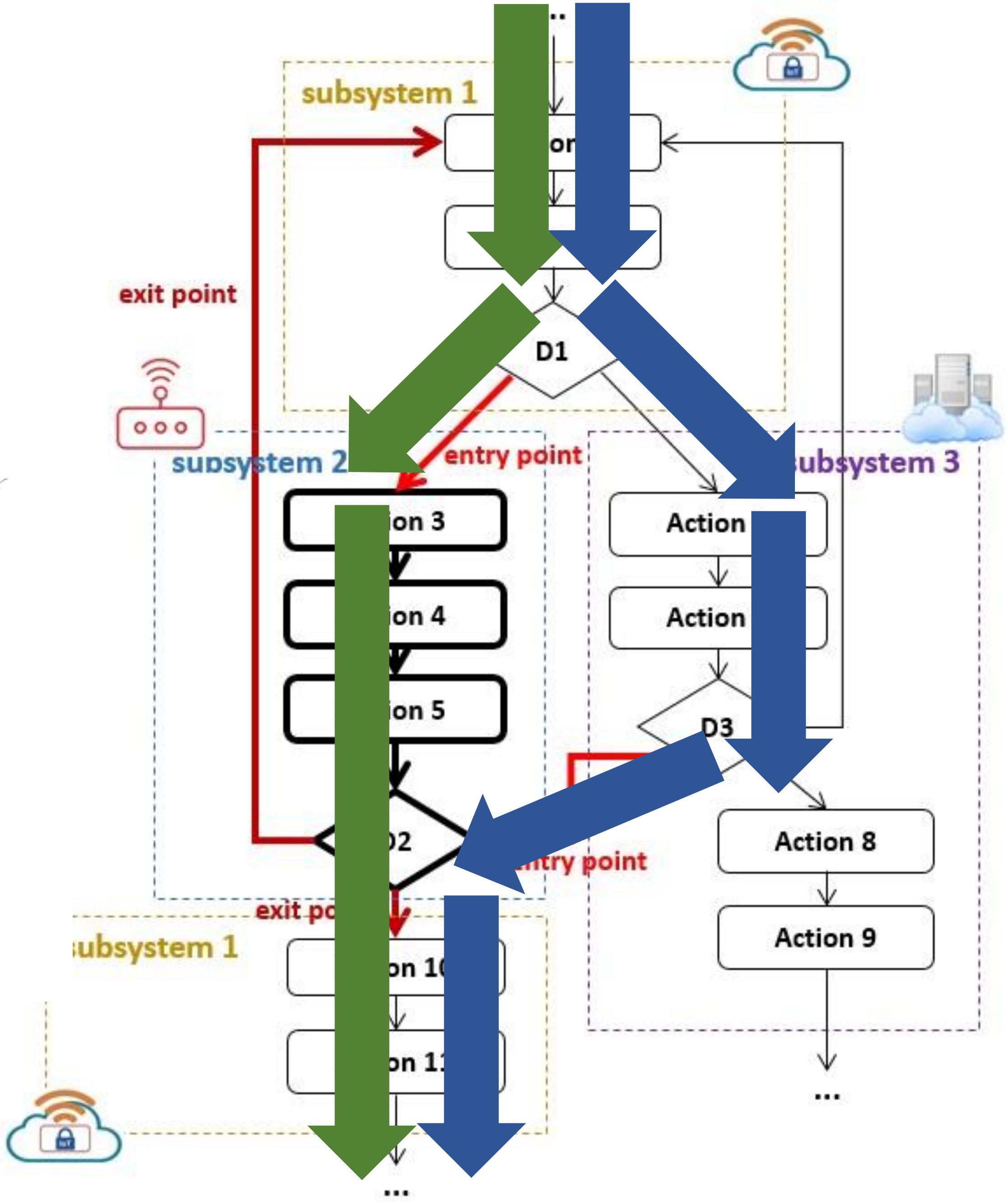
…

AI example: Artificial Ant colony

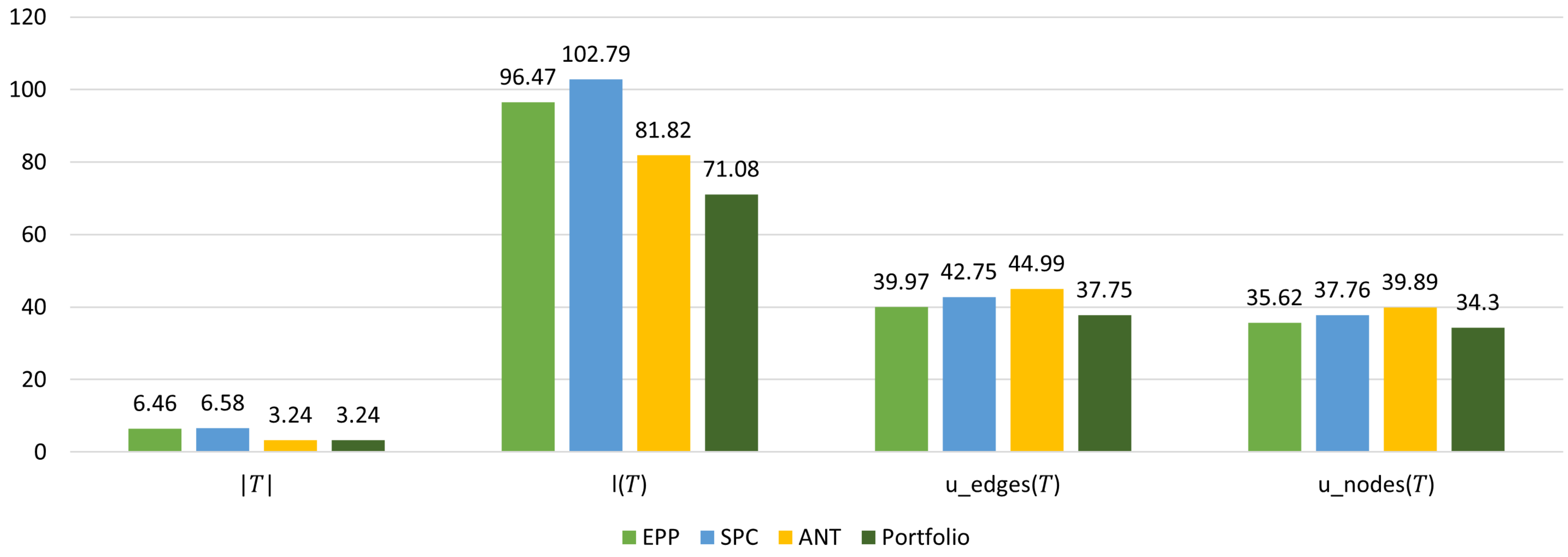
Nature-inspired algorithm

Ants depositing
their pheromone path

“Together” they compute
the best test set



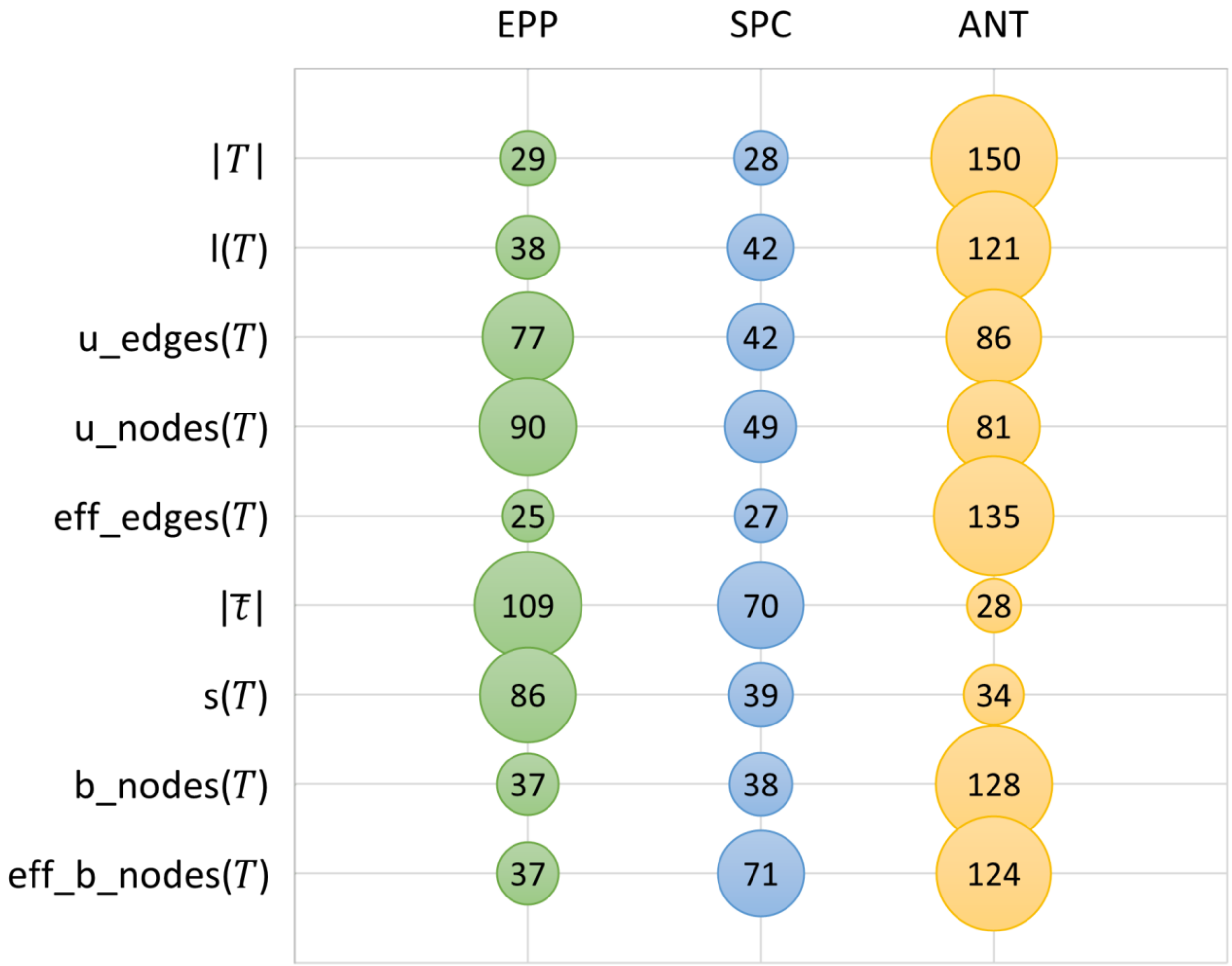
An example of results



Much better than common techniques currently in place

To get the best solution for sure

Algorithm adapts to particular system model



Algorithms combined together

- Portflolio strategy
- Composition of algorithm from blocks

Machine Learning used

Solving similar issues in your software testing? Get in touch!

Miroslav Bures
Software **T**esting **I**ntel**L**igent **L**ab (STILL)
Dept. of Computer Science
Faculty of Electrical Engineering
Czech Technical University in Prague

miroslav.bures@fel.cvut.cz
http://still.felk.cvut.cz

🐦 @intelligent_lab

CTU
CZECH TECHNICAL
UNIVERSITY
IN PRAGUE

STILL
System Testing IntelLigent Lab