11th European Workshop on Multi-Agent Systems

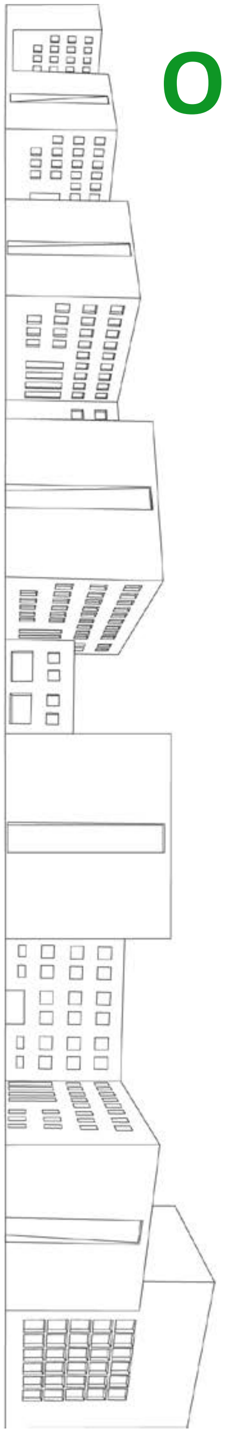
Diagnosis of Unwanted Behaviours in Multi-Agent Systems

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

LIACC artificial intelligence and computer science laboratory

idMEC FEUP engenharia mecânica

Lúcio S. Passos
Rosaldo J. F. Rossetti
Joaquim Gabriel



Outline

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Part I – Introduction

Part II – Relevant Works

Part III – Extended Spectrum-Based Fault Diagnosis for MAS

Part IV – The Cleaning Agents Example

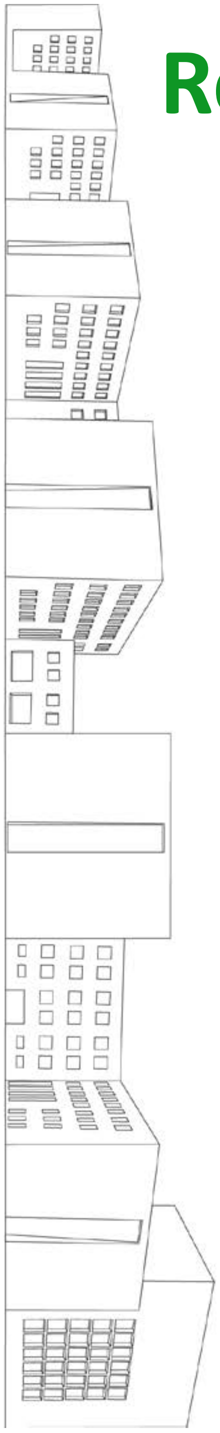
Part V – Conclusion

Part I

Introduction

Research Scope

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO



Multi-agent Systems (MAS) has a set of features to be deployed in real-world applications.

However, agents are under-explored in real deployments (Mckean et al., 2008; Pechoucek and Marík, 2008).

One reason is the lack of reliability of MAS.

To overcome this, a set of works incorporate fault tolerance features and propose better testing techniques.

- Both rely on the correct *diagnosis* of an unforeseen event.

Research Motivation

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

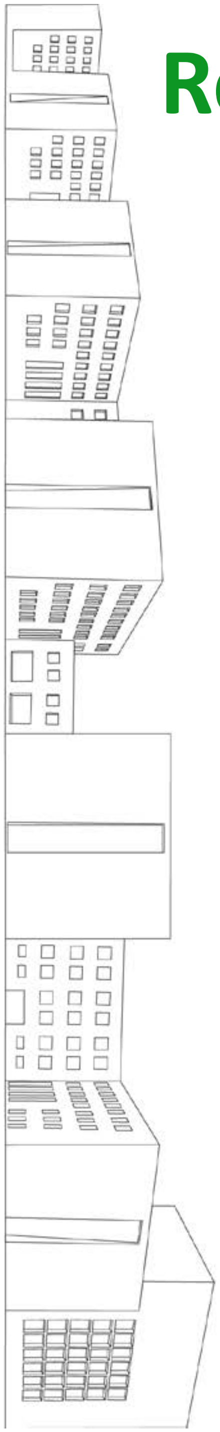
A diagnosis process demands for some knowledge about the system.

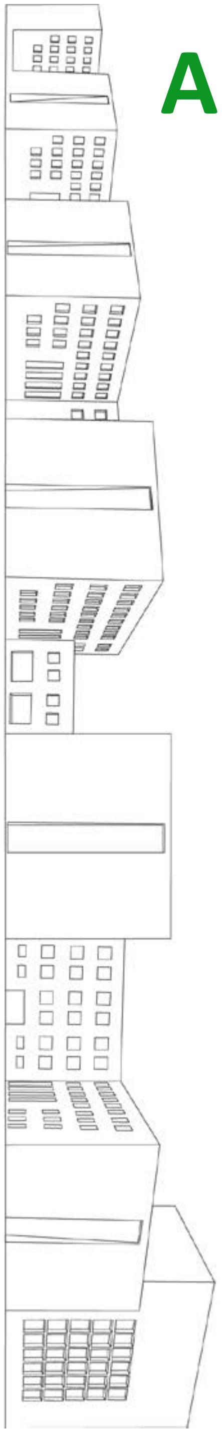
- This approach better suit deterministic systems.

Agents add novel research challenges because of their *non-determinism*.

A requirement to deal with agents is treat them as a *black-box* component.

- Multi-agent diagnosis should not rely on a priori knowledge.





Aim and Goals

Main goal of the proposed research is

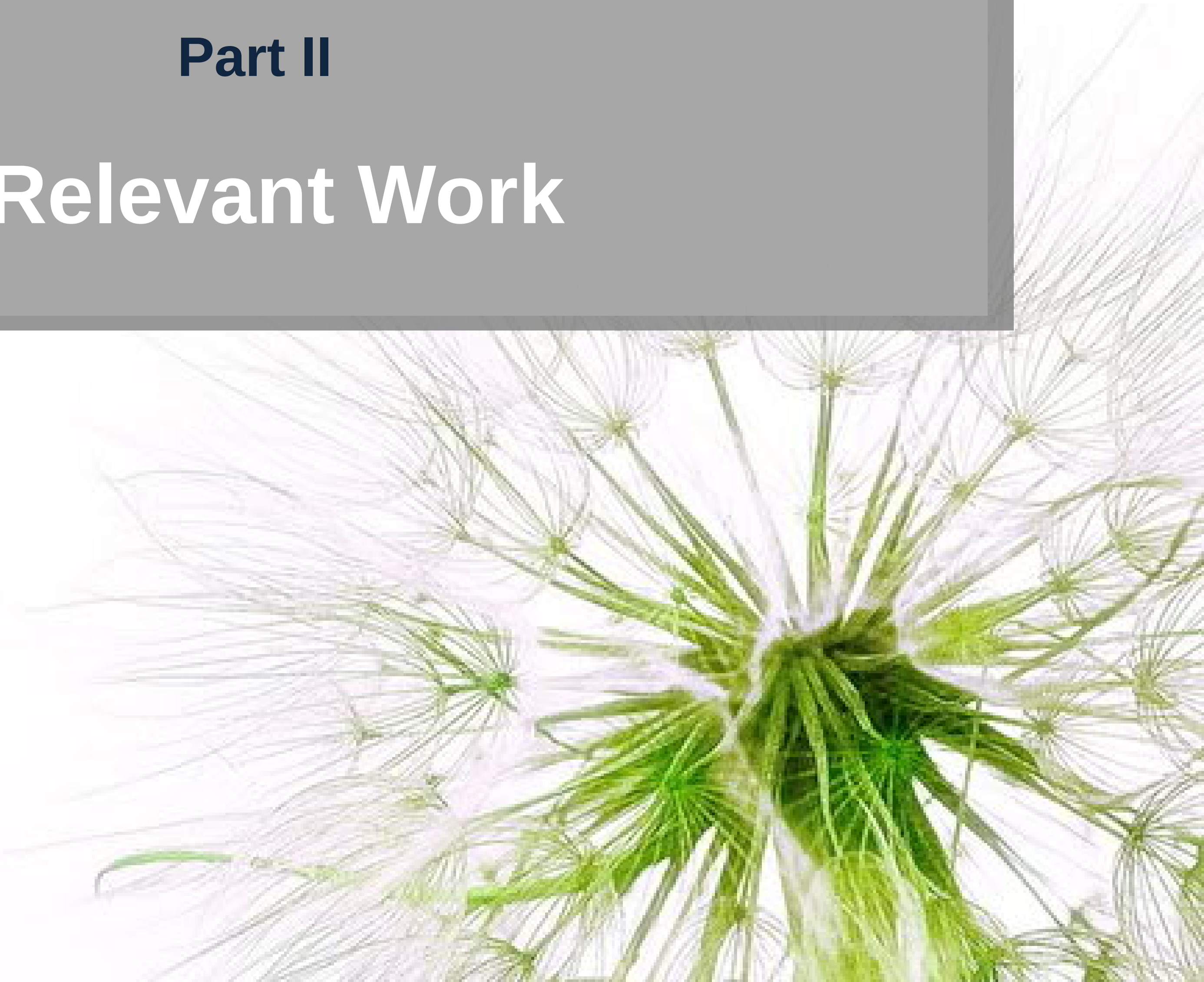
Diagnose agents with unwanted behaviour that affect the system's performance relying on minimal a priori information.

Specific goals of this paper:

- To propose an Extended Spectrum-based Fault Localization approach to Multi-Agent Systems (ESFL-MAS).
- To instantiate the proposal with cleaning agents in a grid-form environment.
- To demonstrate that the ESFL-MAS is able to locate a faulty agent with minimal a priori knowledge.

Part II

Relevant Work



Relevant Work

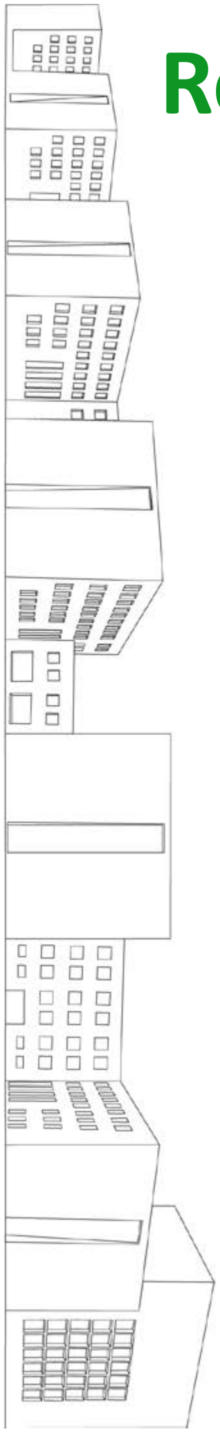
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

***Diagnosis process* is triggered by a detected deviation from the desired behaviour.**

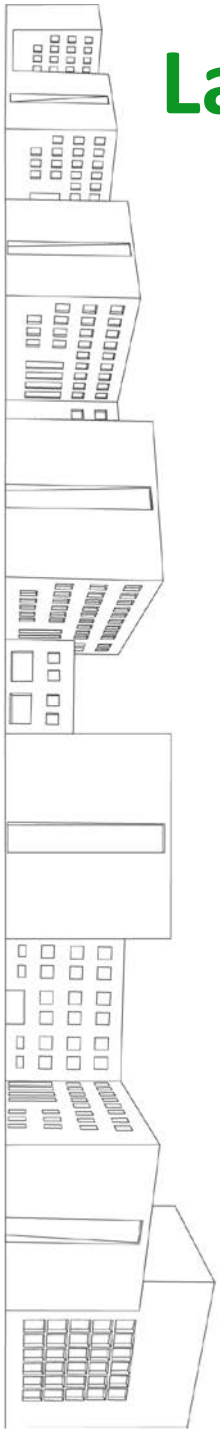
Further, it locates responsible cause(s) for the undesired situation.

The literature focus on two types of information:

- *Fault-based*: rely on all known faults in a given domain.
- *Model-based*: depend on a system's model while any abnormality is classified as a fault.



Lack in the Literature



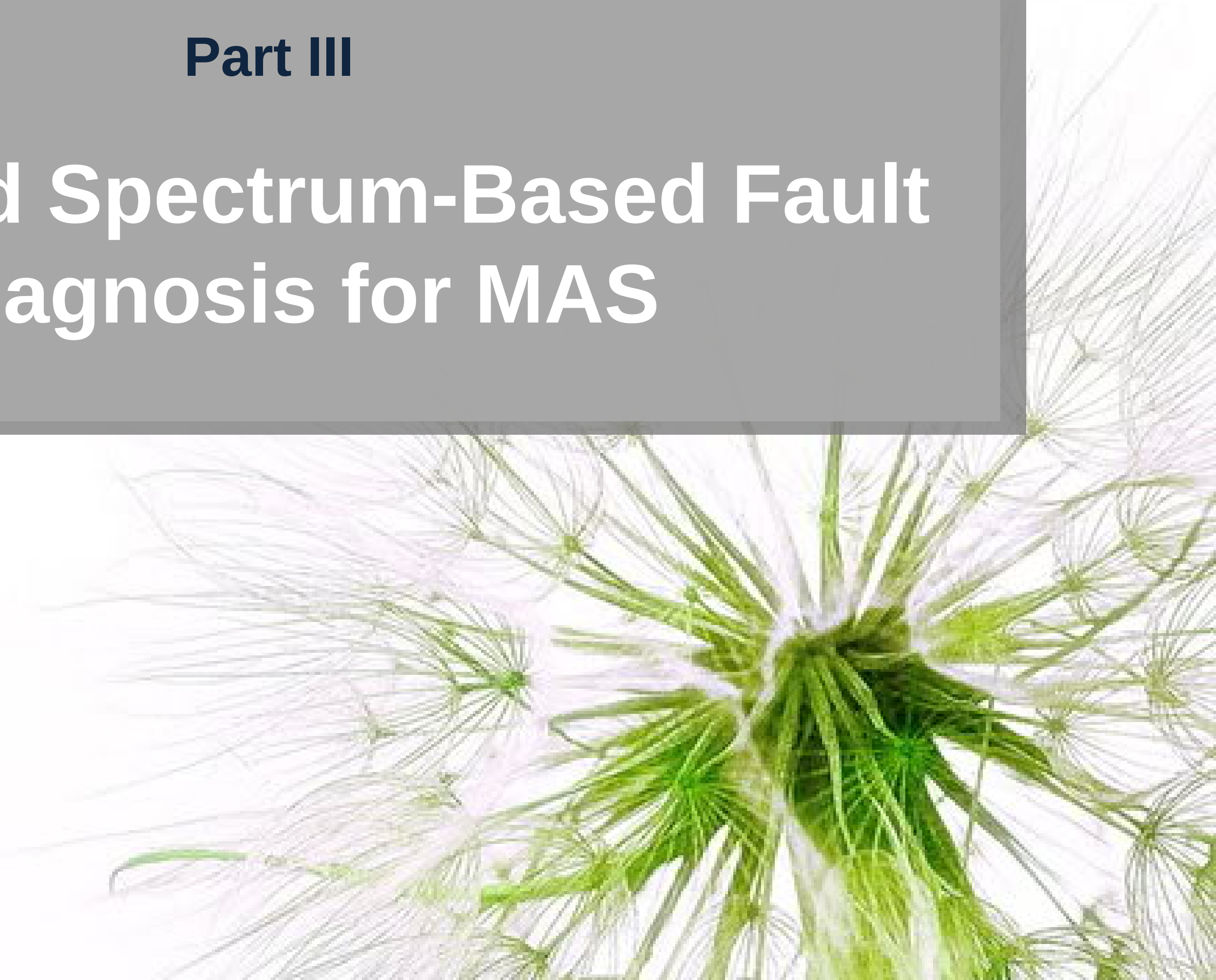
Summary:

The assumption of a priori (either correct or faulty) knowledge is a constant in agent diagnosis endeavours.

Our technique aims to *decouple* the diagnosis process from given information.

Part III

Extended Spectrum-Based Fault Diagnosis for MAS

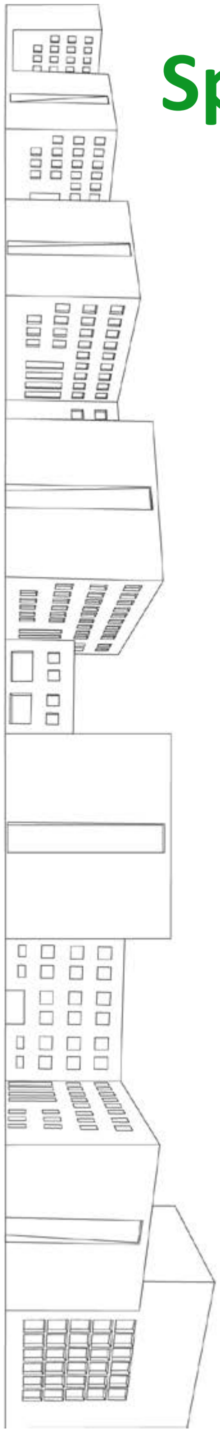


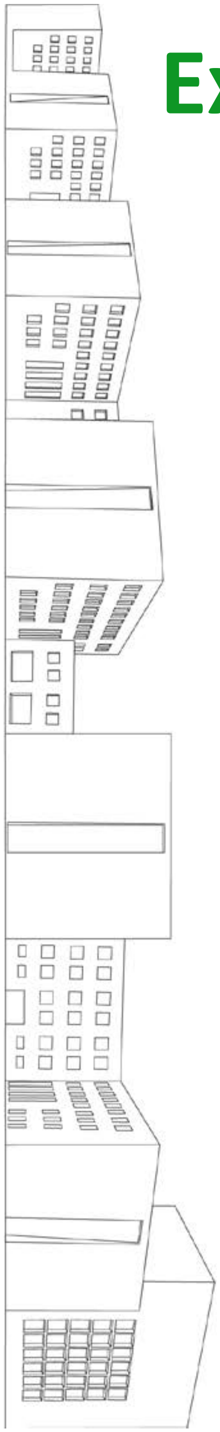
Spectrum-based Fault Localization

It is a statistical technique which analyses the software behaviour over multiples runs to find the faulty component.

- *Program spectrum* is a set of runtime profiles that gives a specific view of a software behaviour.
- *Testing results* inform the SFL whether the program behaves correctly (*passed*) or not (*failed*)

Our proposed approach aims to extend some of its characteristics to cover MAS applications.





Extended SFL for MAS

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Test Suite:

- A multi-agent system must be executed throughout several time steps to observe its behaviour.
- We must execute several rounds of the same test case to cover as many variations as possible to enhance coverage.

Spectra:

- *Metric-based spectra:* agents are assessed using a certain performance threshold respectively.

Performance Assessment:

- Measure how effectively the multi-agent system performs.

ESFL-MAS Components

Information of a multi-agent system *P* analysed by the ESFL-MAS.

$$\begin{array}{c} \mathcal{T} \\ \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_m \end{bmatrix} \end{array} \times C \qquad \begin{array}{c} X\left[T_\alpha, c_\beta\right] \\ \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{n1} \\ x_{12} & x_{22} & \cdots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1K} & x_{2K} & \cdots & x_{nK} \end{bmatrix} \\ \begin{matrix} Ag_1 & Ag_2 & \cdots & Ag_n \end{matrix} \end{array} \qquad \begin{array}{c} E\left[T_\alpha, c_\beta\right] \\ \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_K \end{bmatrix} \\ \end{array}$$

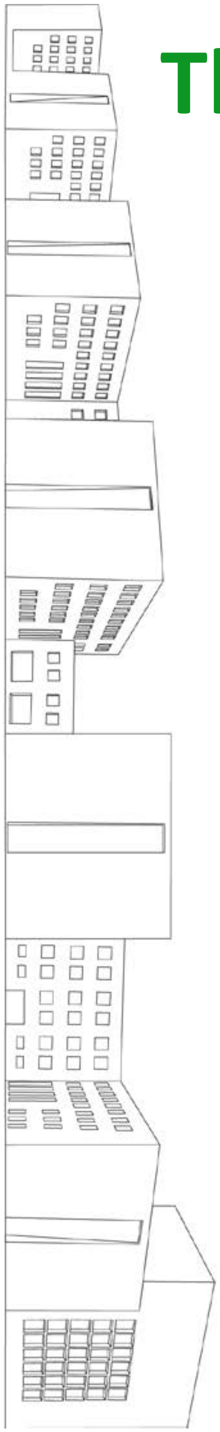
- $\mathcal{T}$ is a set of test cases;
- Test case T_i considers a set of environmental variables;
- $C \in \mathbb{N}$ is the number of executions per test case;
- K is the total number of time steps in a given execution;
- $X\left[T_\alpha, c_\beta\right], 0 \leqslant \alpha < m, 0 \leqslant j < C$ is the spectrum matrix;
- E is a vector of Boolean variables, where

$$e_i = \begin{cases} 0 & \text{if } p_{MAS} \geq thr \\ 1 & \text{if } p_{MAS} < thr \end{cases}$$

The Algorithm

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

```
Input: Multi-agent system, set of test cases, number of executions, and similarity coefficient
Output: Diagnosis report
begin
    Get the number of Agents
    Initialize counter
    Run MAS and keep spectra
    for each test case and execution
        Get the number of time steps
        for each agent
            Feed the respective counter
        end
    end
    for each agent
        Calculate similarity
    end
    Sort agents by crescent value of similarity
    Return diagnosis report
end
```



The Algorithm

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

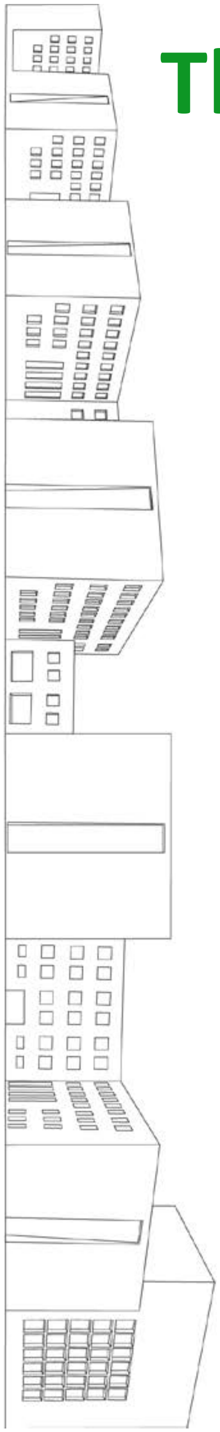
***Input*: Multi-agent system, set of test cases, number of executions, and similarity coefficient**

***Output*: Diagnosis report**

```
begin
        Get the number of Agents
        Initialize counter
        Run MAS and keep spectra
        for each test case and execution
                Get the number of time steps
                for each agent
                        Feed the respective counter
                end
        end
        for each agent
                Calculate similarity
        end
        Sort agents by crescent value of similarity
        Return diagnosis report
end
```

The Algorithm

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO



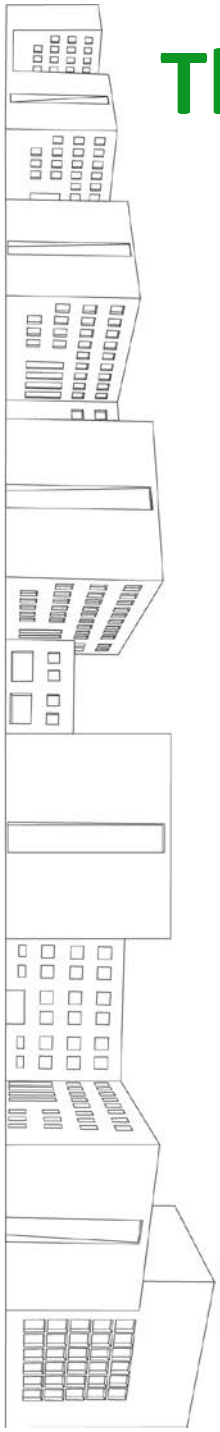
Input: Multi-agent system, set of test cases, number of executions, and similarity coefficient

Output: Diagnosis report

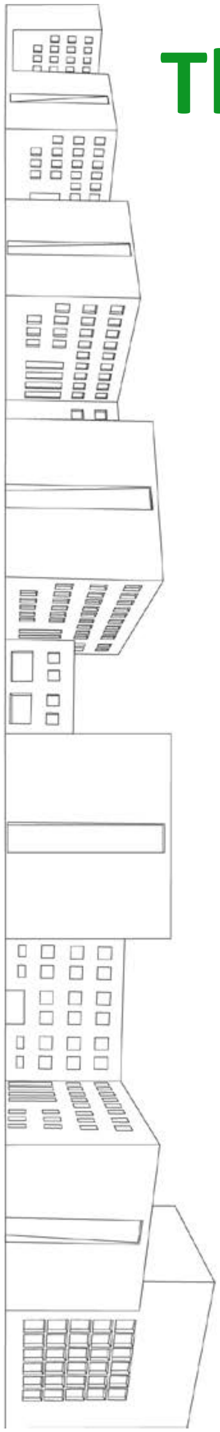
```
begin
    Get the number of Agents
    Initialize counter
    Run MAS and keep spectra
    for each test case and execution
        Get the number of time steps
        for each agent
            Feed the respective counter
        end
    end
    for each agent
        Calculate similarity
    end
    Sort agents by crescent value of similarity
    Return diagnosis report
end
```

The Algorithm

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

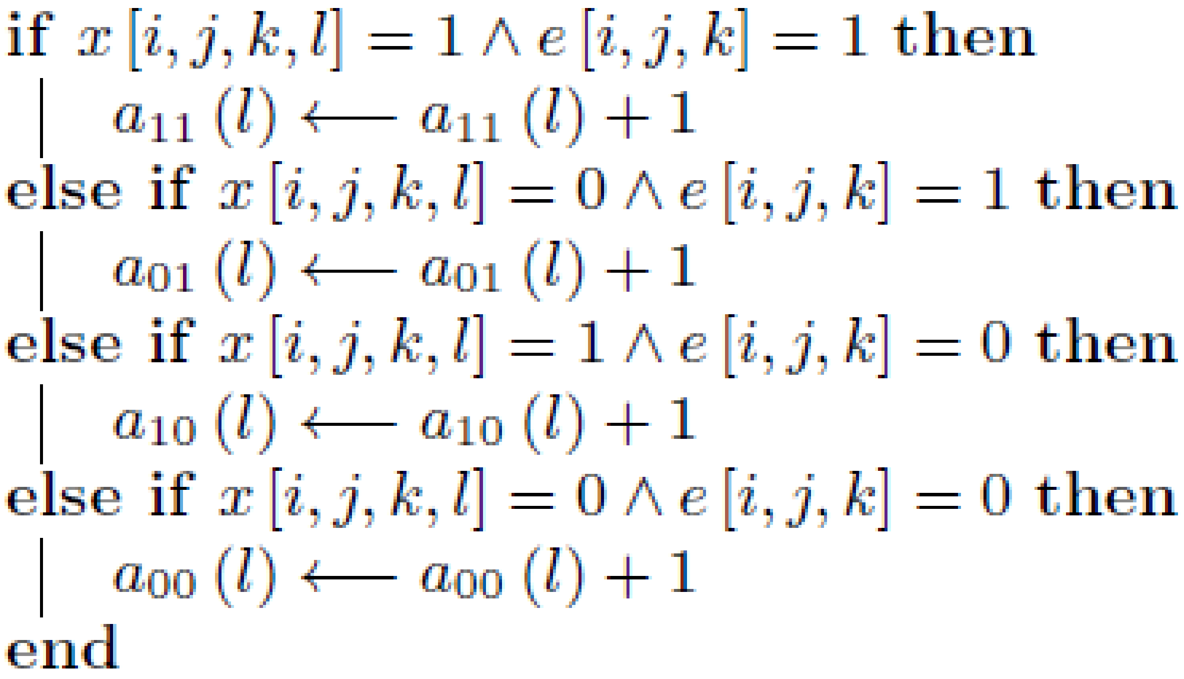


```
Input: Multi-agent system, set of test cases, number of executions, and similarity coefficient
Output: Diagnosis report
begin
    Get the number of Agents
    Initialize counter
    Run MAS and keep spectra
    for each test case and execution
        Get the number of time steps
        for each agent
            Feed the respective counter
        end
    end
    for each agent
        Calculate similarity
    end
    Sort agents by crescent value of similarity
    Return diagnosis report
end
```

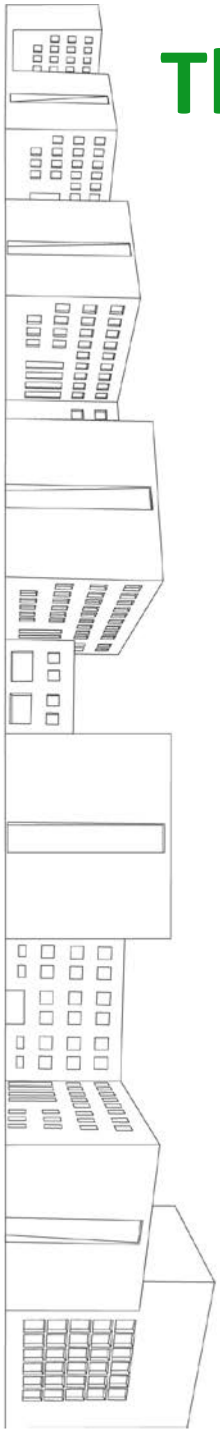


The Algorithm

```
Input: Multi-agent system, set of test cases, number of executions, and similarity coefficient
Output: Diagnosis report
begin
```



```
        if $x[i,j,k,l] = 1 \wedge e[i,j,k] = 1$ then
            $a_{11}(l) \longleftarrow a_{11}(l) + 1$
        else if $x[i,j,k,l] = 0 \wedge e[i,j,k] = 1$ then
            $a_{01}(l) \longleftarrow a_{01}(l) + 1$
        else if $x[i,j,k,l] = 1 \wedge e[i,j,k] = 0$ then
            $a_{10}(l) \longleftarrow a_{10}(l) + 1$
        else if $x[i,j,k,l] = 0 \wedge e[i,j,k] = 0$ then
            $a_{00}(l) \longleftarrow a_{00}(l) + 1$
        end
                Calculate similarity
        end
        Sort agents by crescent value of similarity
        Return diagnosis report
end
```



The Algorithm

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

```
Input: Multi-agent system, set of test cases, number of executions, and similarity coefficient
Output: Diagnosis report
begin
    Get the number of Agents
    Initialize counter
    Run MAS and keep spectra
    for each test case and execution
        Get the number of time steps
        for each agent
            Feed the respective counter
        end
    end
    for each agent
        Calculate similarity
    end
    Sort agents by crescent value of similarity
    Return diagnosis report
end
```

Remarks about ESFL-MAS

The technique essentially consists in identifying the agent whose column vector resembles the underperformance vector the most.

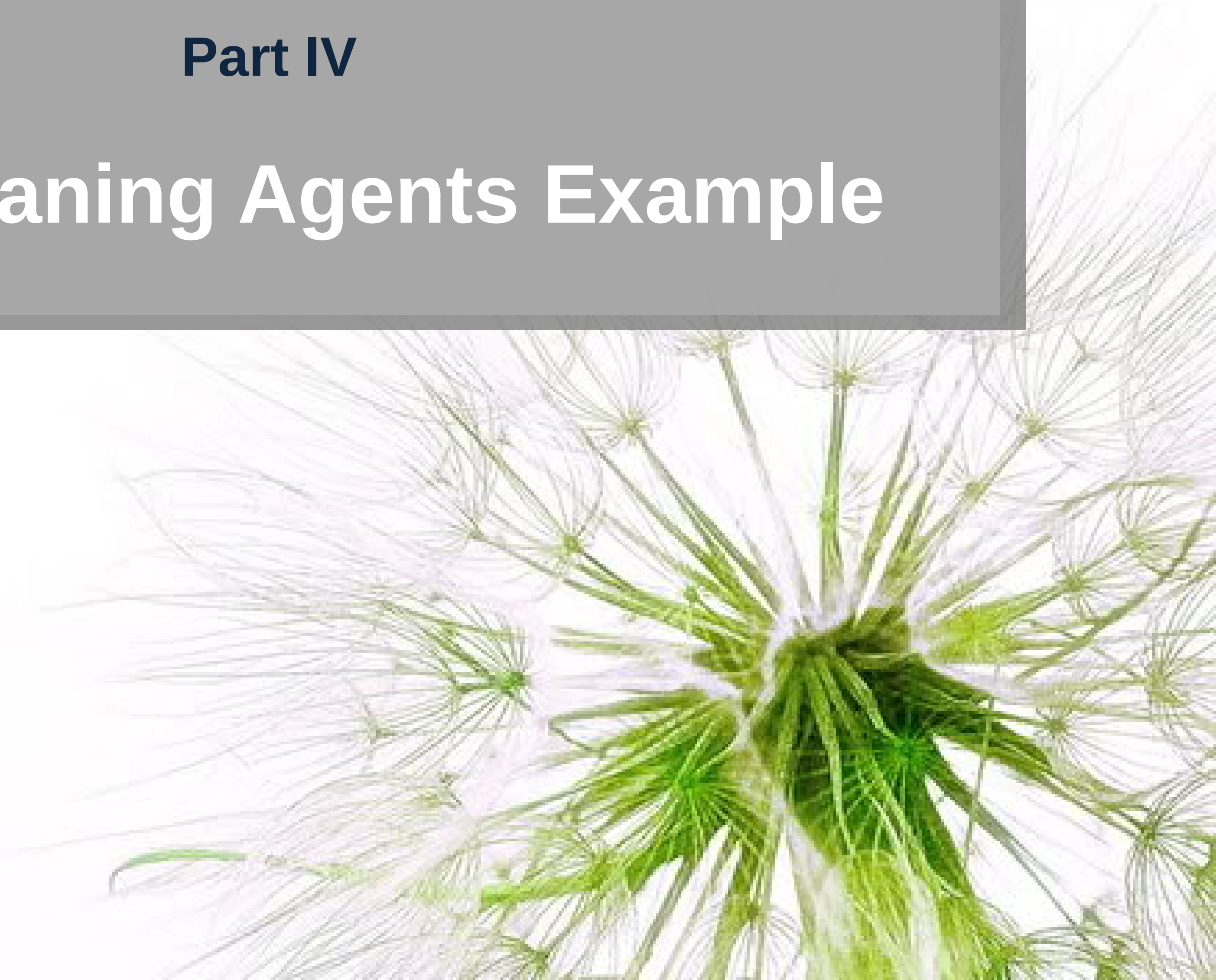
- Similarity coefficient quantifies these resemblances.

Must highlight:

- It not the “silver bullet” for fault diagnosis in MAS.
- Only applicable to close multi-agent systems that work within a static environment.

Part IV

The Cleaning Agents Example



Cleaning Agents Example

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Example characteristics:

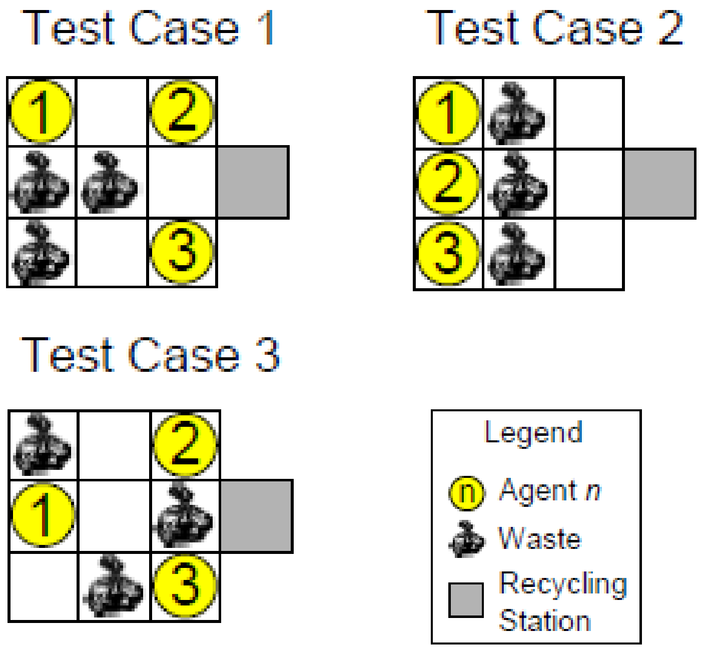
- *Close* multi-agent system upon a *static* environment and at *discrete* time.
- 3 cleaner agents, 3 wastes, and 1 recycling station. One agent with a injected fault.
- Allowed actions:
  Move: *UP*, *DOWN*, *LEFT*, and *RIGHT*.
  With waste: *PICK* and *DROP*.
- Agent have a priori knowledge about the recycling station's location.
- The environment is represented as a grid.
- No communication between agents.

Test Conditions – Part 1

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

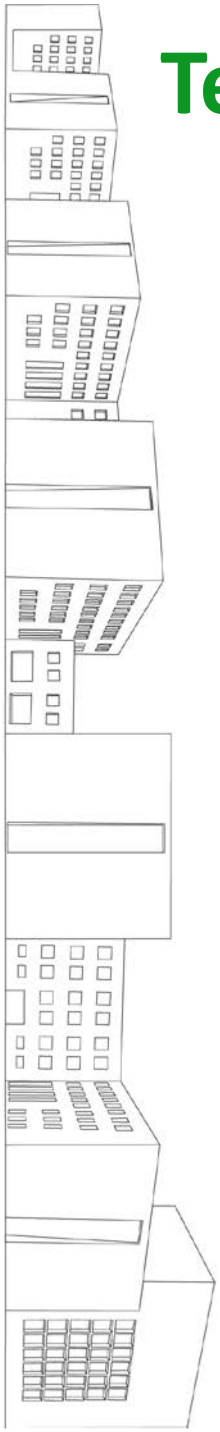
Agent metric-based spectra:

- an agent must pick up a waste in four time steps
- an agent must drop a waste in four time steps



Both are simple and domain-dependent.

We assume that external entities assess each agent



Test Conditions – Part 2

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

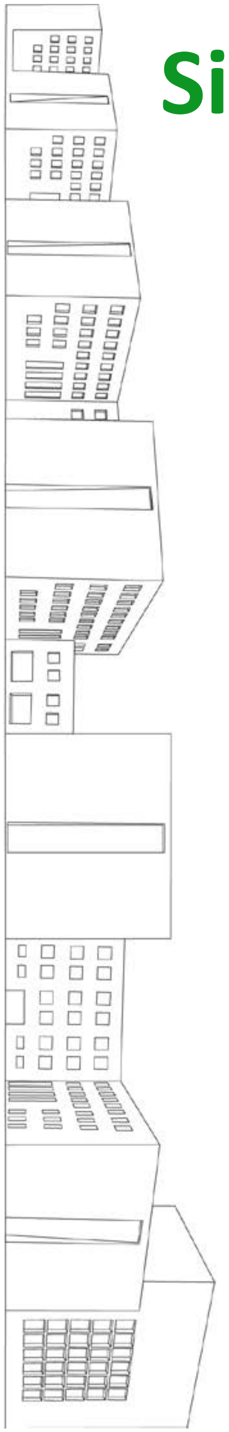
We adopted a simple metric to assess the overall performance:

- The recycling station must receive one waste each 4 time steps.

$$X[2,2] \qquad\qquad\qquad\qquad E[2,2]$$

$$\begin{array}{c} \left[\begin{array}{ccc} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{array}\right] \\ \begin{array}{ccc} Ag_1 & Ag_2 & Ag_3 \end{array} \end{array} \qquad \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array}\right]$$

Similarity Coefficient



The *similarity coefficient* indicates the probability of a certain agent to be the faulty one.

For the example we use the Ochiai coefficient, given by:

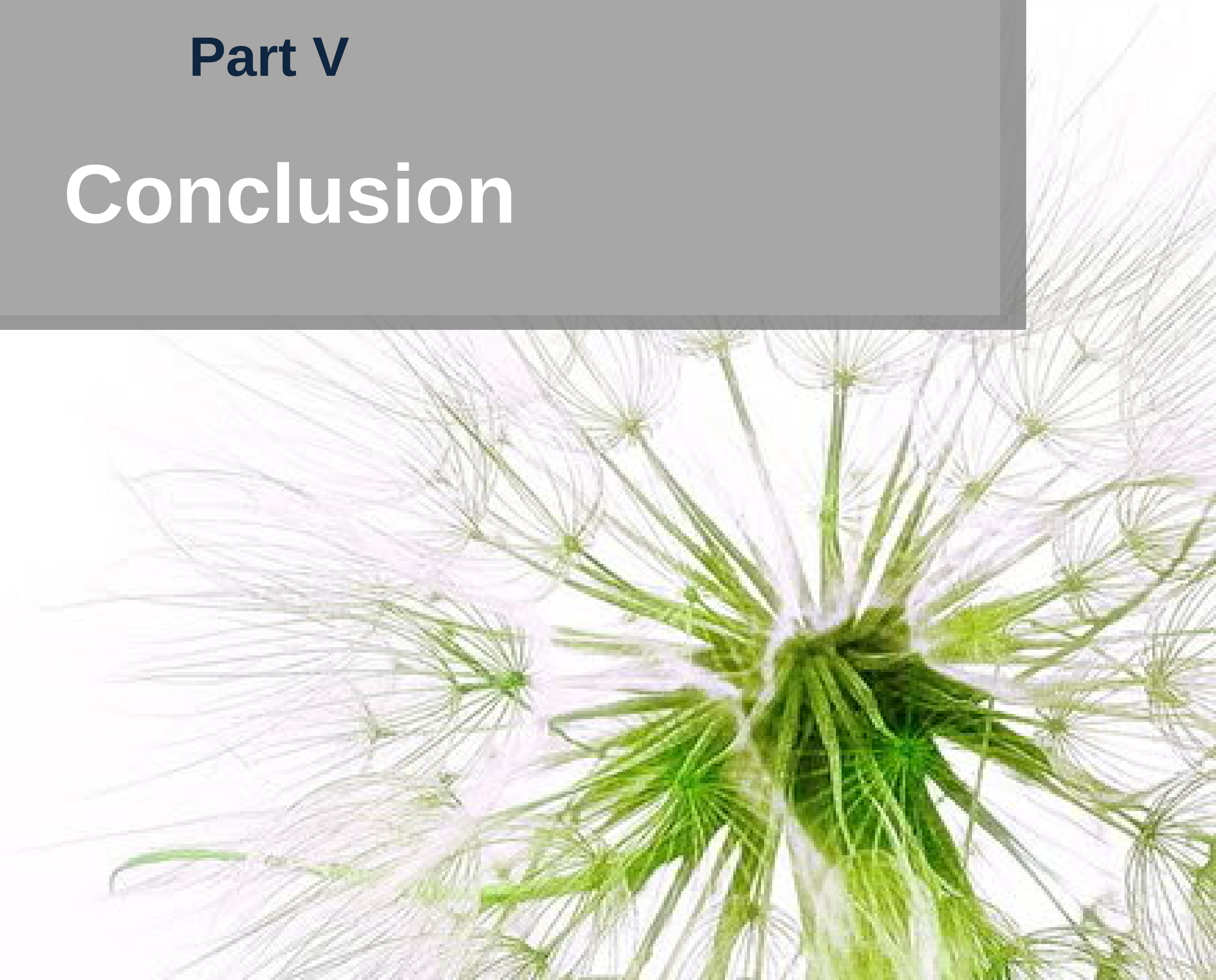
$$s_O = \frac{a_{11}}{\sqrt{(a_{11} + a_{01}) * (a_{11} + a_{10})}}$$

Obtained Results:

- It was able to diagnose agent 2 as the faulty one.
- The D = {agent 2, agent 1, agent 3} with values of similarity equal to 0.304, 0.225, and 0.091 respectively.

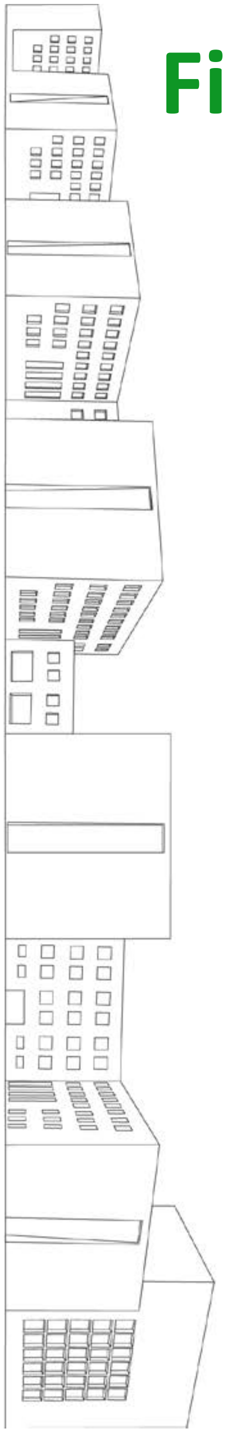
Part V

Conclusion



Final Remarks

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO



Diagnosing faults in MAS is a very challenging task.

We propose a diagnosis technique for MAS relying on minimal given information.

ESFL-MAS collects run-time spectra and identify the underlying causes of a failure.

- However this first appraisal addresses close MAS and static environments

By the illustrate example, we conclude that ESFL-MAS was able to identify the faulty agent.

- ESFL-MAS must be validated in more complex scenarios

Future Research Directions

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

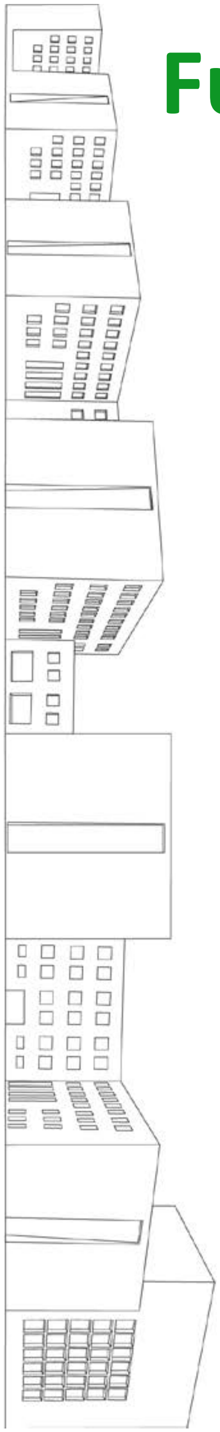
Study the influence of similarity coefficients and metrics on the accuracy of the ESFL-MAS.

Domain-independent metric which would be based on individual and global utilities.

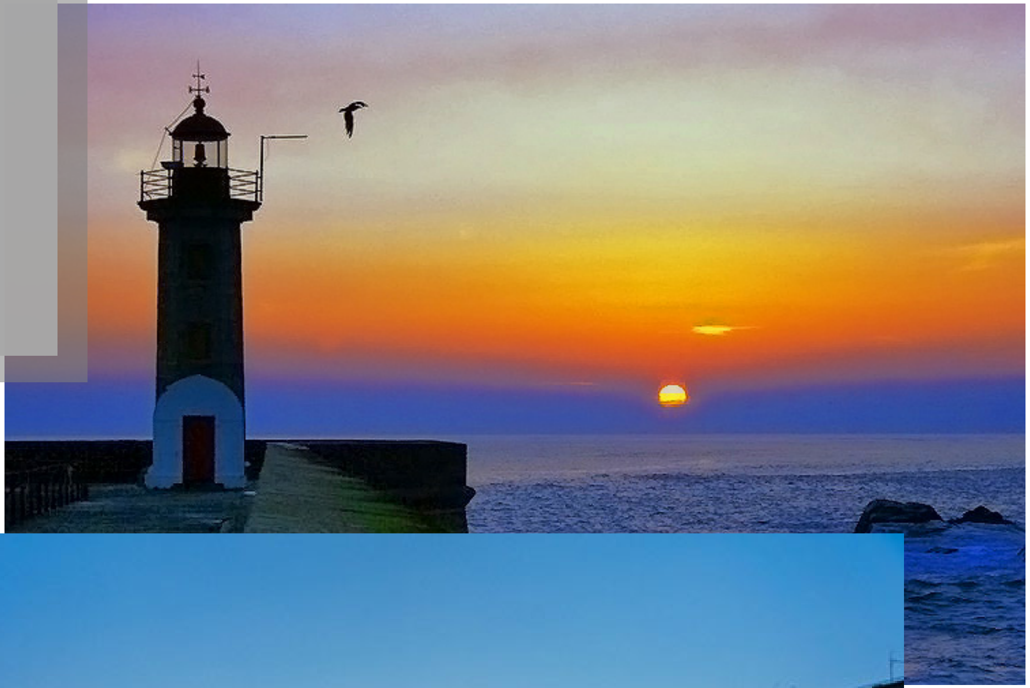
Take into account the social interactions when diagnosing the potential faulty agents.

Investigate the cleaning scenario in a more realistic condition.

- Larger environment, more agents, different test cases and executions.



Thank You!



Lúcio Sanchez Passos
lucio.san.passos@gmail.com
pro09026@fe.up.pt

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

LIACC artificial intelligence and computer science laboratory

idMEC FEUP engenharia mecânica