



CHAOS - Configurations Analysis of Swarms of Cyber-Physical Systems

Valeria Trombetta¹, Maxime Cordy¹, Enrico Tronci² and Axel Legay³

CPSW'22

¹ University of Luxembourg, (SnT)

² Sapienza University of Rome - Italy

³ Catholic University of Louvain - Belgium

Introduction and Motivation (1 / 6)

- Cyber-Physical Systems (CPS)
- Simulations of CPS
- Noises & Disturbances (N&D)
- Variability in CPS

Introduction and Motivation (2 / 6)

Variability in CPS

- Variety of contexts
 - ❖ Environmental conditions
 - ❖ New customer requirements

The main issue

- Develop new variants is expensive

Introduction and Motivation (3 / 6)

The approach

- Injection of variation points
 - ❖ Variability-Intensive Systems (VIS)

VIS as a large class system

- Software Product Lines
 - ❖ Variation points as *Features*
- Configurable Systems
 - ❖ Variation Points as *Configuration Parameters*

Introduction and Motivation (4 / 6)

The concept of Variability

- Each fashion in which the variants are not equal
 - ❖ Different values for a variable

The purpose

- Get a variant that satisfy the requirements
 - ❖ With the highest probability
 - ❖ In the majority of contexts

Introduction and Motivation (5 / 6)

The problems of Variability

- Get the appropriate variant is not trivial
 - ❖ The variability affects the system behavior
 - ❖ It can lead to an exponential computation
 - ❖ Multiple criteria and constraints in case of multi-domain VIS
 - ❖ Uncertainties (N&D) can trigger unpredictable behavior of the system

Introduction and Motivation (6 / 6)

VIS Variability

- Design-Time
 - ❖ Requirements have been a-priori defined
 - ❖ The aim is to discover which variants are more likely to satisfy the requirements
- Run-Time
 - ❖ A variant is already running and it is faced to uncertainties
 - ❖ The goal is to switch to another variant to ensure that requirements are still satisfied

Main purposes of the project (1 / 1)

- help engineers to explore all the possible and appropriate design alternatives wrt:
 - ❖ the scenario
 - ❖ the mission
 - ❖ the budget
- Evaluate which kind of variable had a major impact on the configuration
- Identify which uncertainties affected the system at most
- Explore the state space
- Provide an easy-reusable and extendable tool
- A realistic case-study entirely customizable

Framework (1 / 1)

The image displays the MATLAB environment with a GUI window titled "A Simple GUI" and a 3D surface plot. The GUI window contains a 3D plot of a surface and four buttons: "Surf", "Mesh", "Contour", and "Select Data". Below the buttons is a dropdown menu labeled "peaks". The MATLAB Editor shows the code for the GUI, and the Command Window shows the execution commands.

```

1 function simple_gui2
2 % SIMPLE_GUI2 Select a data set from the pop-up menu, then
3 % click one of the plot-type push buttons. Clicking the button
4 % plots the selected data in the axes.
5
6 % Create and then hide the GUI as it is being constructed.
7 f = figure('Visible','off','Position',[360,500,450,285]);
8
9 % Construct the components.
10 hsurf = uicontrol('Style','pushbutton','String','Surf',...
11 'Position',[315,220,70,25],...
12 'Callback',@surfbutton_Callback);
13 hmesh = uicontrol('Style','pushbutton','String','Mesh',...
14 'Position',[315,245,70,25],...
15 'Callback',@meshbutton_Callback);
16 hcontour = uicontrol('Style','pushbutton','String','Contour',...
17 'Position',[315,270,70,25],...
18 'Callback',@contourbutton_Callback);
19 htext = uicontrol('Style','text','String','Select Data',...
20 'Position',[315,300,70,25],...
21 'Callback',@selectdata_Callback);
22 hpopup = uicontrol('Style','list','String','peaks',...
23 'Position',[315,325,70,25],...
24 'Callback',@selectdata_Callback);
25 ha = axes('Units','normalized','Position',[315,350,70,25]);
26 align([hsurf,hmesh,hcontour,htext,hpopup],ha,'center','top');
27
28

```

Command Window:

```

>> edit
>> Untitled
>>

```

Workspace:

Name	Value
cb	1x1 Cell
r	41x1 double
theta	1x11 double
w	41x41 double
z	41x41 double

Command History:

```

end
if fig.Ch...
display('...
end
if strcmp...
display('...
end
if strcmp...
display('...
end
actreport
> git pull
%-- 02.02...
edit actr...
%-- 10.02...

```

Case- Study (1 / 1)

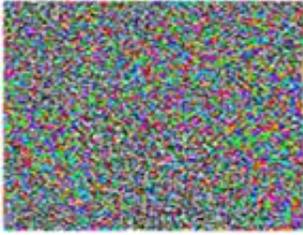
Swarm of Drones

- Drones Features
 - ❖ Battery (H-M-L)
 - ❖ Radio (H-M-L)
 - TX
 - RX
- Drones Characteristics
 - ❖ Anti-collisions
 - ❖ Same velocity
 - ❖ Constraints on spawn point and locations
 - Wrt the target and obstacles
 - Wrt the swarm

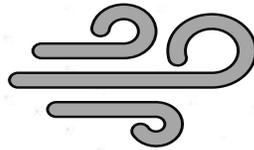


Uncertainties (1 / 1)

- Noises

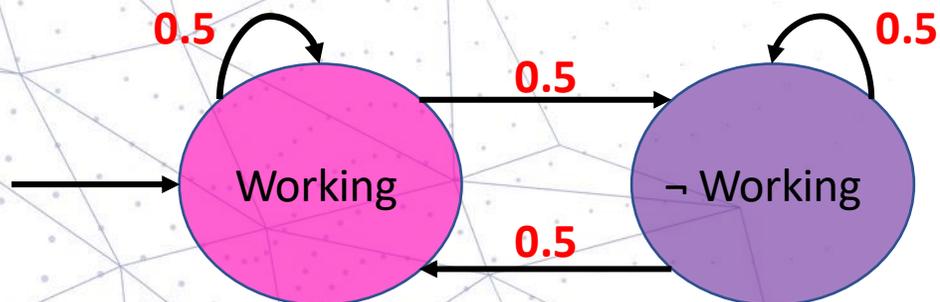
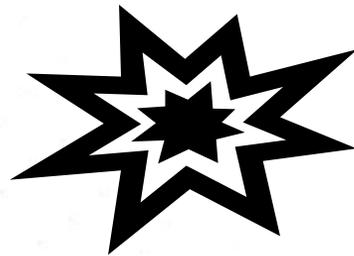


- Gusts



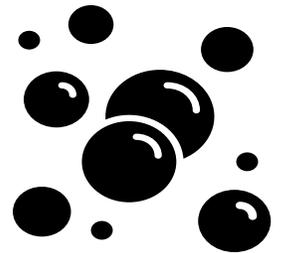
- Faults

- Markov Chain



- Obsctacles

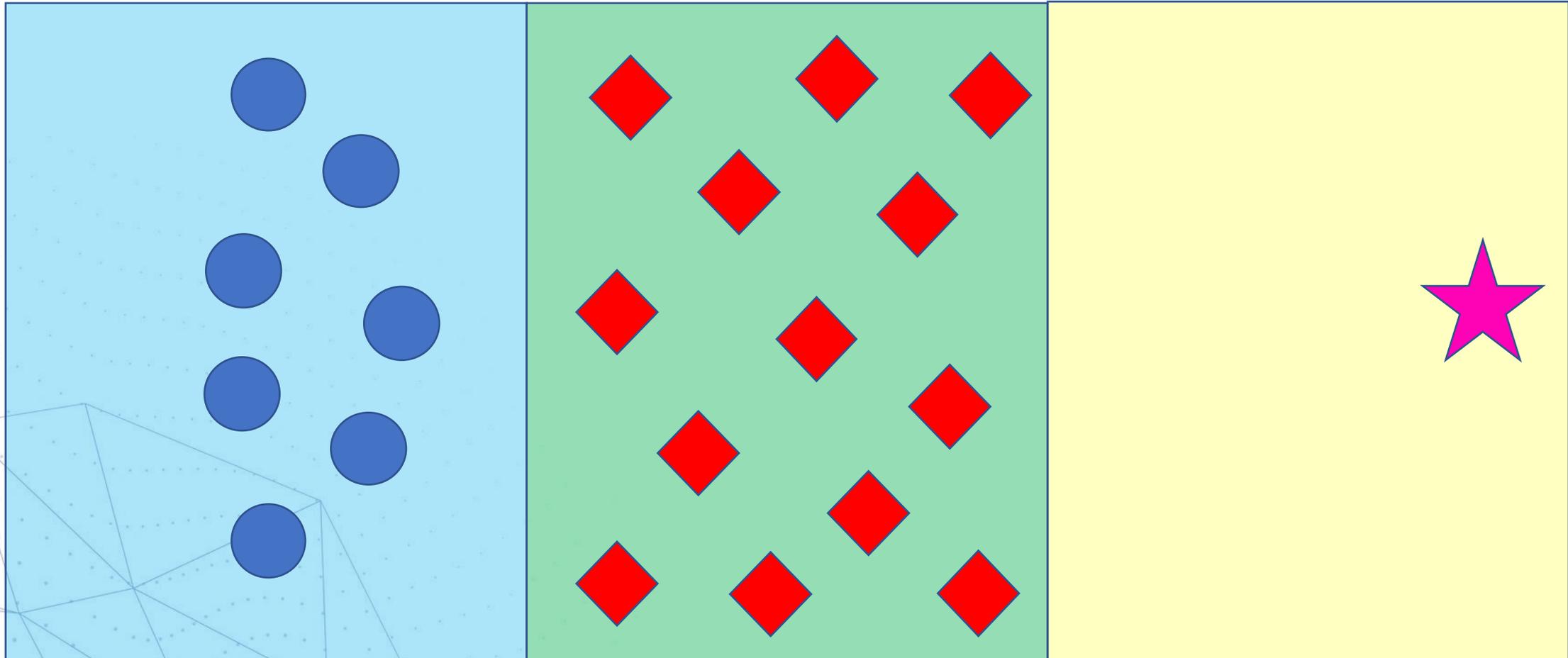
- ❖ Constraints on spawn point (like drones)
- ❖ Anti-collision (like drones)
- ❖ Absent
- ❖ Present
 - Fixed
 - Moving (Random Motion)



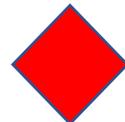
- Target

- ❖ Unique & always present
- ❖ Fixed or Moving (Random Motion)
- ❖ Constraints on spawn point

A general example of scenario (1 / 1)



 = DRONE

 = OBSTACLE

 = TARGET

Research Questions (1 / 3)

RQ1: Given a scenario, is it fundamental to monitor the effect of variability even in small systems?

ARQ1: Yes, even in small systems like drones which in our case study have only 2 features, especially when in the swarm there are few drones.

RQ2: What is the total size of configuration space composed by multiple small configuration systems and their scenarios?

ARQ2: The total size of configuration space is 12264

RQ3: Can we reduce this complex configuration space to a smaller step of solutions?

ARQ3: Yes, but the only if the drones are all identical (e.g. top quality ones)

Research Questions (2 / 3)

RQ4: Do the identified best configurations remain valid in the presence of uncertainties?

ARQ4: Yes, they change only in case of gusts (better quality drones) or mobile target (extra drones).

RQ5: Given a set of scenarios, can we identify the optimal configuration wrt the budget and the desired assurance level?

ARQ5: We identified a set of configuration that over the 12264 analyzed, assures a probability of success of 96% minimizing the expenditure of the budget.

RQ6: Do the identified configurations remain valid taking into account both the design-time and run-time variability?

ARQ6: [In a development phase]

Research Questions (3 / 3)

RQ7: Given a set of scenarios, an assurance level, a budget whose expenditure must be minimised, is the identified configuration the same for both design and run-time?

ARQ7: [In a development phase]

RQ8: Is it feasible to develop an approach to reduce the computational time related to the simulations?

ARQ8: [In a development phase]

Future Directions (1 / 1)

- Validate the approach with further models also belonging to further domains, multi-target setting, and different scenarios (e.g. military ones with attacking drones/obstacles/target).
- Adopt different kind of uncertainties.
- Insert a leader in the Swarm for reconfiguration purposes.
- **ARQ6** Preliminary experiments have shown that given a budget and a level of assurance it is possible to find the optimal configuration wrt all of these.
- **ARQ8** It is under development an approach based on simulation snapshots.
- Answer **RQ1**, **RQ2**, **RQ3** and **RQ7** by performing experiments considering various specific safety industrial standards, additional missions and scenarios, by testing the validity to handle variability at design-time and run-time.

Thanks for your attention.