System Dependability Lab
Exercises on Safety Assessment of Dynamic Systems

Wednesday 18th January, 2023

Report must be named **SURNAME1_SURNAME2.pdf**
and uploaded on moodle before
Wednesday 1st February, 2023

1 Preliminaries

1.1 Installation of Open Altarica

1. If **Open AltaRica** is not already installed, download the Open AltaRica Platform from `https://www.openaltarica.fr/docs-downloads/`.
2. Unzip the file.
3. Launch **AltaricaWizard.exe**.
4. Download the **initialSystem.zip** project from moodle `https://moodle.insa-toulouse.fr/course/view.php?id=264` and open it.

1.2 Lab reporting instructions

Each question clarifies what are the expected report inputs. Concise answers are welcome.

2 Introduction

You will study alternative architectures of the absolute localisation system. This study will develop the initial assessment by considering new failure modes for the component of the localisation system. In this lab, a component can be:

erroneous the component provides an incorrect data

lost the component does not provide any data

The first step of your analysis is to model the initial architecture of the localisation system knowing that every component owns the erroneous and lost failure modes.

3 Basic component modelling

The first step of the modelling consists in providing the generic model used for the components of the architecture. As said previously, a component can either be erroneous, lost or functional. Once a component is in a given failure mode it cannot evolve to another one. This behaviour is encoded by the automaton of the figure 1.

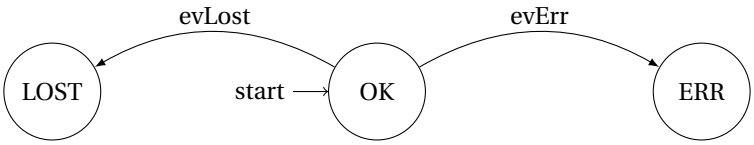

Figure 1: Basic component behaviour

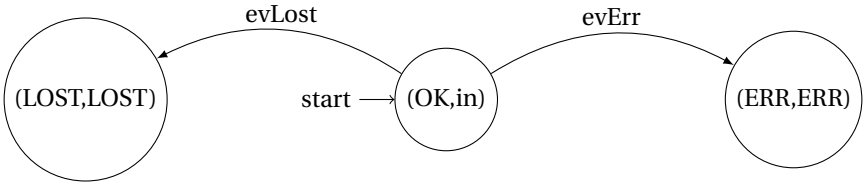

Figure 2: In out component behaviour

Question 1 Open the **Exercice2.ar3w** project file and open the **Components/BasicComponents.alt** file. Complete the altarica model of the **OutComponent** knowing that the erroneous (resp. lost) failure event occurence time is modelled by an exponential distribution where $\lambda = 10^{-6}.h^{-1}$ (resp. $10^{-5}.h^{-1}$). Put the commented model in you report.

Some components may depend on external data to provide their own data. Thus the output data can be erroneous (resp. lost) either if the component itself is erroneous (resp. lost) or if the input data is already erroneous (resp. lost). This behaviour is encoded by the automaton of the figure 2 where each node is labelled by a pair indicating the actual failure mode (first member of the pair) and the quality of the output data (second member of the pair).

Question 2 In the **Components/BasicComponents.alt** file, complete the altarica model of **InOutComponent** knowing that the erroneous (resp. lost) failure event occurence time is modelled by an exponential distribution where $\lambda = 10^{-6}.h^{-1}$ (resp. $10^{-5}.h^{-1}$). Put the commented model in you report.

To express mutliple dependencies, some basic operators are defined in the **Components/Needs.alt** file. These components are modelling artifacts hence do not own failure modes. There are two types of multiple dependencies:

NeedOne The component can rely on redundant sources of information, thus when one of the data is correct then the dependancy is fulfilled. If none of the data is available then the resulting failure mode is lost. Eventually if none of the data is correct but one is erroneous, then the resulting failure mode is erroneous.

NeedAll The component needs all of the incoming sources, thus when one is missing the output signal is lost. Otherwise if one is erroneous then the output signal is erroneous. Otherwise the output is correct.

Question 3 In the **Components/Needs.alt** file, complete the altarica models of **NeedOne2**, **NeedAll2** and **NeedAll3**. Put the commented models in you report.

Question 4 In the **Components/Localisation.alt** file, complete the altarica model of **Localisation**. Put the commented models in you report.
⚠We remind you that an accurate localisation depends the redundant visual information (VO and SI), the GPS and the GIS.

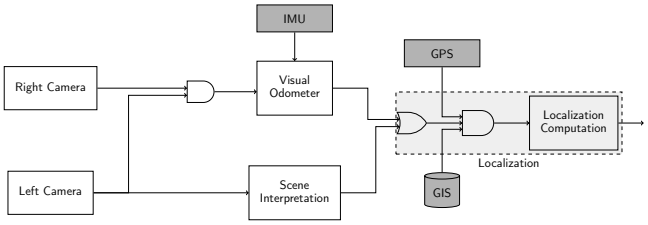

Figure 3: Localisation system architecture

4 Initial architecture modelling

Question 5 Open the **Exercice2/System.alt** file and complete the altarica model of **MainSystem** accordingly to the architecture depicted in the figure 3.

The initial architecture contains now an alarm that is able to warn the pilot in case of the **loss** of the absolute localisation (the erroneous behaviour cannot be detected by the alarm). This alarm also owns the following failure modes:

false negative (FN) the alarm does not send any alert;

false positive (FP) the alarm always sends an alert.

The failure condition called **detectionLoss** is **unnoticed loss or an erroneous localisation**.

Question 6 Complete the observer **dectectionLoss** is the **Exercice2/System.alt** file. Put the complete and commented model of the **MainSystem** in your report.

Question 7 Compute the minimal cutsets leading to the **dectectionLoss**. Put the result in your report. Among these cutsets find

- a cutset that has already been identified in the previous lab and explain why the old one is still a cutset despite the introduction of the alarm;
- and one involving the failure of the alarm and explain why this alarm failure is necessary to trigger the **detectionLoss** failure condition.

Question 8 Compute the unreliability of the initial architecture out of the cutsets. Provide the computation and the numerical application for a mission time of $500h$ and $10^4 h$.

5 Redundant localisation processing

To enhance the reliability of the localisation system, a designer proposes to use analytical redundancy *i.e.* use various sources to compute the same data. This technique is used to avoid localisation loss when a data is unavailable and to detect inconsistent data. In this lab, we consider that this redundancy is used to perform a redundancy between a nominal and backup system as depicted by the figure 4. If the nominal is lost, then this loss is detected and a selector switches to the backup system.
In addition, a breaker monitors the integrity of the two data, if both data are available and an inconsistency is detected then the breaker opens the circuit resulting in the loss of the computation. Note that the breaker must remain open after its first activation.

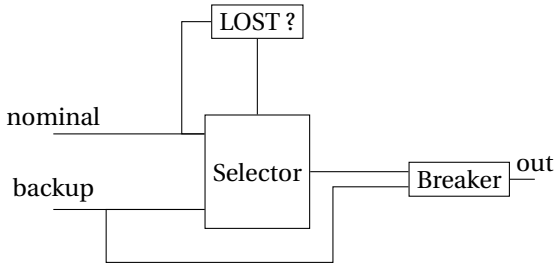

Figure 4: Redundancy pattern

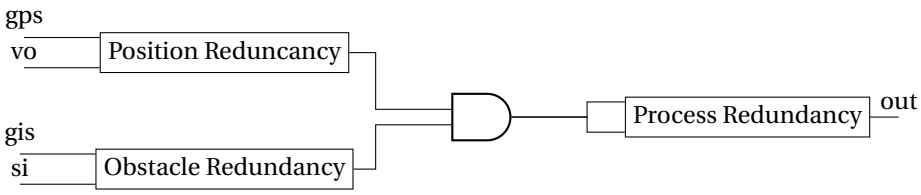

Figure 5: Redundant localisation

Question 9 Open the **Exercice3.ar3w** project file and complete the altarica model of the **Selector** in **Components/Selector.alt** and **Breaker** in **Components/Breaker.alt**. Put the commented models in your report.

Question 10 Complete the altarica model of the **Redundancy** in **Components/Redundancy.alt**. Put the commented model in your report.

As shown by the figure 5, this redundancy is used at three different stages in the localisaton component:

1. The position can be obtained either by GPS or VO
2. The obstacle detection can be obtained either by GIS or SI
3. The process of the position and obstacle data is based on two redundant processes

Question 11 Complete the altarica model of the **RedundantLocalisation** in **Components/RedundantLocalisation.alt** and the system **MainSystemRed** in **Exercice3/System.alt**. Put the commented models in your report.

Question 12 By using the stepwise simulator, can you find a sequence of failures triggering **detectionLoss** for which a different order of the failures does not lead to **detectionLoss**? What can you conclude about this system?

6 Bonus: Redundant sensors

To limit the impact of the erroneous behaviour of the GPS and IMU sensors, a designer suggests to use a triplication of these sensors with the following strategy:

- if no majority can be found among the inputs data (dismissing lost data) or if all data are lost then the voter does not transmit any data
- if the only remaining signal is erroneous (others dismissed) then the voter transmit the remaining data

Question 13 Complete the altarica model of the **Voter** in **Components/Voter.alt**. Put the commented model in your report.

Question 14 Complete the altarica model of the **MainSystemSensorRed** in **Exercice4/System.alt** by triplicating the IMU and GPS and adding the voters on the triplicated sensors. Put the commented model in your report.