Norges Teknisk-Naturvitenskapelige Universitet

TPK4186 - Advanced Tools for Performance Engineering
Spring 2020

Assignment 3: A Production Facility

Prepared by Antoine Rauzy

Contents

1 Introduction

1.1 Presentation of the Problem

Figure 1 shows a production facility of the oil and gas industry. It consists of seven units. The gas separated from the well fluid at upstream side is fed to the facility, treated through separators (`HPS-A`, `B`, `C`) and dehydrators (`DEH-A`, `B`), and led to compressors (`CMP-A`, `B`).

Each of the unit may fail and be repaired independently. It is assumed that failures and repairs obey negative exponential distributions. Failure and repair rates are given Table 1.

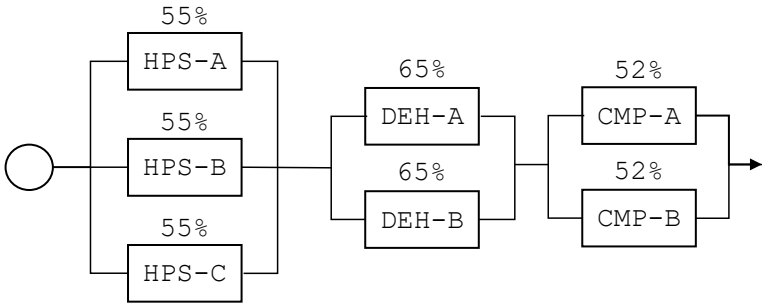

Figure 1: A Production Facility

Table 1: Failure and repair rates of units (in h^{-1}).

Unit	Failure Rate	Repair Rate
`HPS-A, B, C`	8.91×10^{-5}	2.54×10^{-3}
`DEH-A, B`	3.11×10^{-5}	3.95×10^{-3}
`CMP-A, B`	5.50×10^{-5}	5.14×10^{-3}

Each unit has a certain production capacity, here given in percentage of the optimal production. The objective of this assignment is to assess the expected yearly production of the facility.

1.2 Requirements

The objective of this assignment is not only, nor even primarily, to assess the expected production, but rather to show your Pythonic skills.

Here follows a number of requirements.

1. You must provide your program together with a small document explaining how it is organized, what it is doing (which functionalities are implemented) and reporting experiments you have performed with it. You may also provide additional files such as those you use to test your program and to perform experiments.

   The report must be written as a HTML page.

2. Assuming your name is Jack Sparrow, all of the above files must be included in a zip archive named:

   `TPK4186 - 2020 - Assignment3 - Sparrow Jack.zip`

   The deliverable of the assignment is this zip archive.
3. The assignment must be made individually.
4. You must program in an object-oriented style, preferably splitting your program into modules.
5. Recall that the quality of a program can be judge along three criteria: its completeness, its correctness and its maintainability:
   - A program is complete if it provides all functionalities demanded by the client. Some functionalities are however more important than other. You must first concentrate on the main functionalities, then develop the "nice-to-have" ones.
   - A program is correct if it is bug free. To ensure that your program is correct, you must test it extensively. Design tesst before writing the first line of code. There is no such a thing than a program or a functionality that works "most of the time". Either it works, or not. If you are not able to make a functionality work, do not deliver it.
   - A program is maintainable if it is well presented, if the identifiers are significant, and so on. But before all, a program is maintainable if it well organized and as modular as possible. Separate the concerns.

2 Tasks

2.1 First Set of Hypotheses

To assess the expected year production, you shall implement a discrete event simulation, based on what we have done during lectures and tutorial.

Here follows two remarks to help you.

1. The production of the facility can never exceed 100%. It is the minimum of the production of its three successive treatments.
2. The number of states of the Markov chain representing the system is $2^7 = 56$. There is however probably a better way to represent all of the states explicitly.

It would be interesting to assess the expected production month by month and to draw the curve of its evolution.

It would be also interesting to know how each type of units contributes to the production losses.

2.2 Second Set of Hypotheses

Assume now that only one repair crew is available at a time, i.e. that if two components or more are failed simultaneously, they are repaired according to a first entered, first served strategy.

Modify your discrete event simulator to take into account this new characteristics.