Exercise sheet 4

Issuance: 20.11.2024 Submission: 26.11.2024, 23:59 Uhr

In this exercise sheet, due to the cancellations of the last lectures we need to refer to a concept which will be introduced in a now future lecture. Note that the following is not a replacement for the lecture!

In energy systems it is often necessary to react on environmental or internal changes and realize adaptive behavior to ensure robustness and flexibility. For this reason the generic Observer/Controller architecture can be used.

The observer observes the *system under observation* (i.e. a multi-agent system) and reports its observations to the controller. The controller uses the observations of the observer and applies some control action (i.e. changing MAS parameters). The controller incorporates user-defined goals to find a feasible control action. The controller can also adapt the observation model of the observer.

There are many variants of the observer/controller pattern: i.e., central, distributed, multi-level. The figure 1 shows the difference between those.

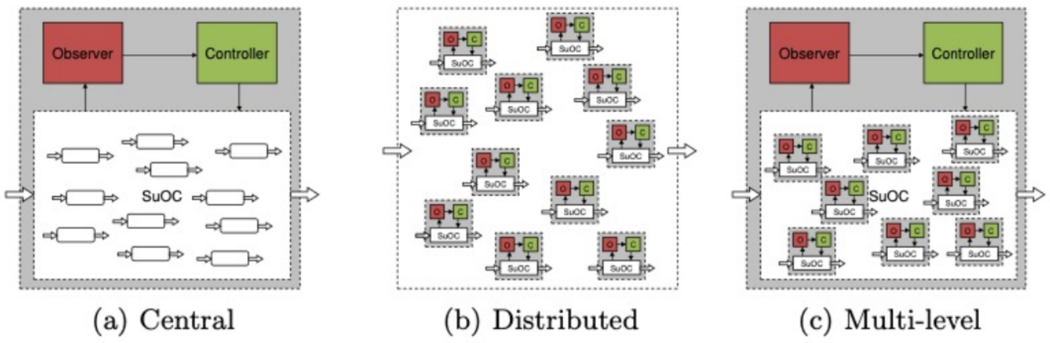

Figure 1: Variants from [Tomforde]

More information on this topic can be found in the following publication:

- [Tomforde] “Observation and control of organic systems“. In: Organic Computing – A Paradigm Shift for Complex Systems (Sven Tomforde, Holger Prothmann, Jürgen Branke, Jörg Hähner, Moez Mnif, Christian Müller-Schloer, Urban Richter, Hartmut Schmeck.)

Exercise 1: In this exercise, we will focus on an Observer/Controller Architecture. Here, the agent system represents the system under observation and control for which you will implement an observer and a controller.

1. Consider your agent system modeled for the constraint satisfaction problem on the last exercise sheet. Create another agent, an observer. This agent observes all the exchanged messages of you system. All agents must therefore be able to use the observer and send every message sent to it. ⇒**Code**

2. Discuss possible parameters regarding your system which the observer could monitor. ⇒**Answer**

3. The observer should also be informed when the constraint satisfaction problem is fully solved. There are several ways of doing this. Implement one of these options and make sure that the observer is informed that the problem has been solved. ⇒**Code**

Exercise 2: In this task, we will focus on the controller agent.

1. Now implement another agent, the controller agent. The controller agent is connected to the observer and also knows all the addresses of the other agents, as it needs to be able to take actions on the system. ⇒**Code**

2. Discuss which things in the system the controller might react to and how. Also consider the things you listed that the observer could monitor. ⇒**Answer**

Exercise 3: In the final task, we will focus on the interaction between observer, controller and the system under observation and control.

1. You have already implemented that the observer is somehow informed about the termination of the constraint satisfaction problem. As soon as the problem is solved, the observer should update the controller. This update should include the fact that the termination has occurred, the total number of messages exchanged, and the time it took to solve the problem. ⇒**Code**

2. When the controller receives the update, it will change a setting regarding the constraint satisfaction problem. This part depends on your implementation on the previous exercise, therefore, discuss, which parameters you chose to change and why. ⇒**Answer**

3. Once the controller has chosen a setting, it updates the agents and restarts the optimization of the constraint satisfaction problem. ⇒**Code**

4. Make sure that the observer starts the monitoring again (continues the monitoring). When the agent system terminated the second time, the observer informs the controller again, using the same information as in the run before. ⇒**Code**

5. When the controller receives the information about the second termination, it compares the two solutions (in terms of number of messages exchanged and runtime). ⇒**Code**

6. Hand in your results and discuss the differences. ⇒**Answer**