Bayes-Adaptive Multi-Agent Partially Observable Markov Decision Processes for Planning and Learning in Multi-Robot Self-Adaptive Cyber-Physical Systems

Master's thesis

Supervisor: Prof. Dr. Alexander Pretschner
Advisor: Ana Petrovska
Email: {alexander.pretschner, ana.petrovska}@tum.de
Phone: +49 (89) 289 - 17830
Starting date: 15.07.2020

Fakultät für Informatik
Lehrstuhl 4
Software & Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17830
https://www4.in.tum.de

Context

Cyber-physical systems (CPSs) often operate in dynamic, non-deterministic, multi-agent, and only partially observable environments. If the environment is not trivial, the designer of the system has virtually no chance of incorporating adequate handling of arising run-time uncertainty during the system's design-time. The uncertainties might originate internally from the CPSs themselves, or externally from the environment or the context in which the systems operate. To deal with this problem, the CPSs need to be able to observe changes in their context and adapt to them, i.e., the system needs to be self-adaptive. While there is no universally accepted formal definition of self-adaptive systems, a few necessary properties are agreed upon. Self-adaptive cyber-physical systems (SACPSs) often have to execute specific sequences of operations in order to achieve their goals, which is why the ability to plan, as well as the ability to execute the plan are regarded as necessary. In order to adapt, the systems first have to make observations from the context they are operating in. Many problems require the self-adaptive system to be able to learn from its observations, so that it may act accordingly in the future. However, many other problems originate from the fact that the observations themselves are often, besides being partial, also uncertain and ambiguous. In SACPSs, observing and acting often is implemented on the hardware side, through systems' sensors or actuators, while analyzing the observation, planning, and learning form the software part. A widely used architecture implementing the above specifications is MAPE-K (monitor, analyze, plan, execute, knowledge).
In a preceding guided research, the Multi-Agent Partially Observable Markov Decision Process (MPOMDP) mathematical framework has been used [1] in order to implement a reference example for multi-agent CPSs, where multiple robots have to find, navigate to, and solve randomly spawning tasks in a static 2D environment with obstacles while cooperating and learning the probabilities of spawning. The guided research focused on the planning part, while learning was tackled with a naive approach. Thus, the system's ability to learn the environment is limited to the specific environment of the reference problem.
The Bayes-Adaptive Multi-Agent Partially Observable Markov Decision Process (BA-MPOMDP) framework is a generalization of the MPOMDP framework, which, in addition to planning, is able to learn any environment, [2] during its operation. Therefore, BA-MPOMDP will remove the need to construct an environment-dependent learning module during design time and brings us a step towards a *more complete* self-adaptive system. Due to the severe complexity of the added learning capability, solving a BA-MPOMDP demands a dedicated solving algorithm and high computational resources. To deal with the high complexity, the system needs to be parallelized and optimized, so that it is suitable for real-time applications.

Goal

The main goal of the master thesis is to implement a solution to the reference problem by using the BA-MPOMDP framework, so that the system is able to learn any environment and perform planning, both in real-time. Due to BA-MPOMDPs bad scaling in the size of its environment, a suited approximating solver has to be identified, implemented, parallelized and optimized, so that it is suited for real-time application.

The resulting system has to be integrated into the current implementation of two ROS-based in-house built simulators: 1) realistic simulation with Gazebo, and 2) custom simulation in which are omitted the uncertainties from the AMCL and the move-base modules. The new BA-MPOMDPs implementation of the system will be evaluated against the previous (MPOMDP-based) implementation from the guided research, in order to demonstrate the new learning capability. In the evaluation we will use different setups, in which we will variate the structure and the size of the environments (rooms), as well as the task distributions.

Working Plan

1. Study the BA-MPOMDP mathematical framework, identify suitable solver
2. Implement necessary structures for BA-MPOMDP, implement solver (parallelized)
3. Integrate implementation into the ROS simulators: Gazebo and custom one
4. Run system to gather evaluation data, optimize for execution speed if necessary
5. Evaluate on different setups: varying the room structure and the task distributions
6. Compare BA-MPOMDPs implementation with the former MPOMDP implementation to demonstrate the effectiveness of the Bayes-Adaptive learning
7. Write the technical report, git documentation, etc.
8. Write the thesis report.

Fakultät für Informatik
Lehrstuhl 4
Software & Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17830
https://www4.in.tum.de

Deliverables

- Mathematical BA-MPOMDP model of the reference problem
- Source code of the implementation
- Technical report with comprehensive documentation of the implementation, i.e. design decision, architecture description, API description and usage instructions. Usually as part of the gitlab documentation. The technical report should containing instructions how to use the system and brief explanations of the implemented modules.
- Final thesis report written in conformance with TUM guidelines.

Pre-requisite

- Good Python and C/C++ skills
- Previous knowledge and experience with ROS/Gazebo
- Knowledge and very good understanding of Markov Decision Processes
- Ideally be familiar with the technical implementation of the multi-robot use-case

References

[1] C. Amato and F. A. Oliehoek, “Scalable planning and learning for multiagent pomdps,” *CoRR*, vol. abs/1404.1140, 2014.

[2] C. Amato and F. A. Oliehoek, “Bayesian reinforcement learning for multiagent systems with state uncertainty,” pp. 76–83, 2013.