

7th oCPS PhD summer school in Lucca, Italia, 2017

Christopher A. Trotter
Department for informatics
University of Oslo

August 7, 2017

1 Summary report

As part of the master thesis, I was sent to participate in the oCPS summer school in Lucca Italia 2017. The objective of the trip was to try and find ideas where the concurrency models can be applied, and also prepare and learn about Cyber-Physical Systems(CPS) for my stay in Paris, France, later in the autumn. There were presented different topics in CPS and I will summarize the presentations in the form of my understanding of the topics. In that regard, all lectures will be mentioned and some of them thoroughly described based on the interest deemed relevant to the master thesis.

1.1 Future challenges for CPS - An industrial perspective

When attending the motivation of Cyber Physical Systems(CPS) from an industrial perspective, it was introduced by Alf Isaksson, who is a researcher at ABB Corporate Research, and he said the following: *"Modeling is the important part"*. This was an important point that was followed up with the fact that modeling is 50% of a Model predictive control(MPC) project. Furthermore, the motivation of the talk was concepts that have been introduced in an industrial context such as Big Data, Deep Learning, Artificial Intelligence(AI) and digitalization. These concepts have been important in correlation with Internet of Things(IoT) and Cyber Physical Systems(in the talk, IoT is equivalent to CPS). From an industrial standpoint, there is extensively potential in utilizing data that is collected by developing what is known as Deep Learning(DL). Deep learning is a method of acquiring data, adapting it to the knowledge of a system such that the system may *"learn"* from the acquired data, also known as Machine Learning(ML). With this comes also challenges such as getting a hold of data for machine learning and create neural networks.

An extensive part of the lecture went in depth about examples, issues and solutions to such challenges and including the notion of Smart Manufacturing. As a conclusion to the lecture, it was focused on Cyber Physical System Security(CPSS) and kurswosh law, signaling, to emphasize the CPS perspective of these challenges.

1.2 Cyber-Physical Systems: Motivation and Challenges

Christos Cassandras, who is a professor at Boston University, held the introduction lecture of Cyber-Physical systems with the focus on motivation and challenges. As an introduction, he explained that cyber is the internet in the form of sending packages, physical is the number of devices that have a certain application and lastly systems being a structure of components working, independently or together, to achieve some desired functionality. Later, he emphasized that his main occupation was on the work of smart cities and that the idea of smart city could be divided into four processes: data collection(sensor network), information processing(big data), Decision making and control and optimization. Furthermore, when working with these processes it has shown to be environmental constraints which must be considered. These constraints can be safety, security, privacy and so forth. The notion of Hybrid Systems(HS) was introduced and described as a system that switches between cyber and physical because they are bidirectional. I will not go into depth of HS here, but have it mentioned. Finally, at the end of the lecture I wrote the following notes to try and find insight of the predetermined goal as mentioned in the introduction above:

Understanding: I am working with models for concurrency based on HDAs and ST-structures, derived from configuration structures. First I need to understand where in the big picture this falls. Secondly, I need to see how the development of ST and HDA fall in the picture of sculptures and how we can further find the application of sculptures.

Sense: Where I believe this fits in the bigger picture is into the complex between the time-driven and event-driven issues. As mentioned under the lecture of Christos he emphasized the approach of event-driven over time-driven and that if the perspective would change to more of an event perspective, the gain would be great.(maybe, need further investigation).

Most of the early research has focused on the state-based/time-based approach where a clock is the king. It is a set time where certain things may happen and must happen. This can cause large time tickets where the system might be highly ineffective by activating when the system is mostly inactive.

A newer approach is that of event-based allowing the trigger/switch to be king. It is then a certain action that happens to trigger a specific event. This approach seems more feasible and better for long time IOT devices to stay alive.

Focus: I should focus on the Distributed CPS as it is a project in Oslo working on this specifically. What is a distributed CPS? Is it the focus on the distributed elements of the CPS. As described above, Cyber is the internet communication with packages and all such. Physical is the actual devices where the IOT comes into focus.

Idea: Potential application of the models of concurrency would be best applicable in Distributed CPS. Figure out how.

1.3 Discrete event and hybrid system models and models and methods for CPS

This lecture was held by Christos Cassandras and was a continuation of the topic of Hybrid Systems(HS). The focus of this lecture was on the models and methods of HS such as modeling time-driven and event-driven systems. As said by Christos: *"Action and information is mandatory to create a good decision of what to do and to have a whole"*. In the sense that time-driven is the well known automata model to constructing a state-based system such that it is possible to know what state you're in and which is the next. Furthermore, event-driven is the opposite, where events is a piece wise trajectory such that the ordering of the events is important and anything can be events in an event-driven model. What is being discussed is synchronous vs. asynchronous behavior.

1.4 Embedded control systems

Embedded control systems was the topic of the lecture held by Samarjit Chakraborty. The talk was largely focused on the implementation of distributed control applications on embedded platforms. Many embedded systems now consist of up to 100 processors or electronic control units(ECUs) that communicate using different bus protocols like FlexRay, CAN or Ethernet. Traditionally, the process of designing control algorithms start with system identification or modeling, followed by determining the appropriate control laws, computing their parameters, and finally verification or simulation to ensure that the required control performance requirements are satisfied.

However, all of these are done at a high level of abstraction where most of the implementation details are either neglected or modeled in a pessimistic manner as a lumped error. However, as embedded systems become more complex, such a approach either leads to large semantic gaps between high-level controller models and their actual implementation, or an unnecessarily expensive or pessimistic design. The talk gave insight into the multi-disciplinary field from a computer scientist perspective as well as retaining the topic relevant for engineers working with control theory. Most of the lecture was spent discussing techniques for addressing the problem mentioned above, through a cyber-physical system oriented design approach, which requires a co-design of controllers and the architectures on which they are implemented. In particular, it was shown that when taking conventional platform resources - such as computation, communication, and memory - into account necessitates a redesign of control algorithms compared to traditional controller design approaches that are agnostic to such resources.

1.5 Resource-aware control

1.5.1 Abstraction

Recent developments in computer and communication technologies are leading to an increasingly networked and wireless world. In the context of networked control systems and cyber-physical systems this raises new challenging questions, especially when the computation, communication and energy resources of the system are limited. To efficiently use the available

resources it is desirable to limit the control actions to instances when the system really needs attention. Unfortunately, the classical time-triggered control paradigm is based on performing sensing and actuation actions periodically in time (irrespective of the state of the system) rather than when the system needs attention. Therefore, it is of interest to consider aperiodic control strategies such as event-triggered control as an alternative paradigm, as it seems much more natural to trigger control actions by well-designed events involving the systems state, output or other available information, i.e., bringing feedback in the sensing and actuation processes. In this lecture, we discuss the basics and challenges of resource-aware control focusing on event-triggered control, but also touching upon self-triggered control schemes. Main results regarding control-related properties such as stability, performance and robustness are provided, next to guarantees on the existence of a positive lower bound on the inter-event/transmission times (minimal inter-event times). Clearly, the latter is essential from an implementation point of view and the side of utilization of the computation, communication and energy resources. Comparisons to periodic time-triggered control and various applications including experiments performed on a platoon of vehicles will be presented.

After thought: A lot of the issues with CPS is that there is a silver lining of how to balance the hardware requirements with the actual software requirements. One wants to have a certain guarantee based on software, but it also has to uphold the hardware constraint. If there are unrealistic hardware requirements given by a reasonable software requirement it can make it difficult to realize the actual prediction or method.

Remark: The notation used for the lectures are mostly mathematical in nature, but some are a bit confusing since they are used in other parts of the concurrency theory. It might have a different meaning than that known from the topic.

1.6 Performance and Reliability analysis by model checking

1.6.1 Abstraction

Model checking is a powerful verification technique for bug hunting in hardware circuits and software programs. Its extension to probabilistic models enables the quantitative evaluation of system performance and reliability along with correctness. Probabilistic model checking has applications in e.g. robotics, energy-aware computing, RAMS analysis, distributed computing, and probabilistic programming. It provides hard guarantees that quantitative system requirements are met, facilitates bottleneck analysis, and recently enables synthesizing optimal system parameters. In this talk, I present some applications, uncover some of its key algorithmic ingredients, show its scalability to Markov models of millions of states, and give some insights in recent developments in parameter synthesis.

1.7 Fault Diagnosis of Interconnected Cyber-Physical Systems

1.7.1 Abstraction

The emergence of interconnected cyber-physical systems and sensor/actuator networks has given rise to advanced automation applications, where a large amount of sensor data is collected and processed in order to make suitable real-time decisions and to achieve the desired control objectives. However, in situations where some components behave abnormally or become faulty, this may lead to serious degradation in performance or even to catastrophic system failures, especially due to cascaded effects of the interconnected subsystems. The goal of this presentation is to motivate the need for health monitoring, fault diagnosis and security of interconnected cyber-physical systems and to provide a methodology for designing and analyzing fault-tolerant cyber-physical systems with complex nonlinear dynamics. Various detection, isolation and accommodation algorithms will be presented and illustrated, and directions for future research will be discussed.

1.8 Machine learning and Robotics research in CPS

Notes and lecture slides can be found in the lecture notes, Lecture 3.2 and the lecture slides 3.2 Policy Search for Robotics and Multi-Agent Systems by Gerhard Neumann.

1.9 Model Predictive Control for Cyber-Physical Systems

1.9.1 Abstraction

Model Predictive Control (MPC) is one of the most successful techniques adopted in industry to control multivariable systems in an optimized way under constraints on input and output variables. In MPC, the manipulated inputs are computed in real-time by solving a mathematical programming problem, most frequently a Quadratic Program (QP). I will first introduce the basic concepts of MPC, showing how it can be used as the control component of a cyber-physical system (CPS). Since most CPS's run under fast sampling and limited CPU and memory resources, I will then cover methods to solve QP's within a CPS with high throughput, using simple code and executing arithmetic operations under limited machine precision, and with tight estimates of execution time, and show extensions to solve mixed-integer QPs. Finally, I will also show how MPC based on hybrid dynamical models can be used as an effective tool for control of cyber-physical systems.

1.10 Approximate Dynamic Programming

1.10.1 Abstraction

We will consider the DP algorithm for finite horizon stochastic sequential decision problems, which arise in a broad variety of applications, such as control/robotics/planning, operations research, economics, artificial intelligence, and others. The algorithm in its exact form, computes the optimal cost-to-go

functions and an associated optimal policy, but is very often impractical, because of overwhelming computational requirements. Thus one often has to settle for a suboptimal control scheme that strikes a reasonable balance between practical implementation and adequate performance. In this lecture series, after a discussion of the exact DP algorithm, we will review several approaches for suboptimal control based on DP ideas. These come under two general categories:

(A) Approximation in value space, where we approximate the optimal cost-to-go functions with some other functions, in a scheme based on one-step or multistep lookahead. There are various approaches here, such as problem approximation, aggregation, rollout, model predictive control, parametric approximation (possibly using neural networks), and others.

(B) Approximation in policy space, where we restrict the policy to lie within a given parametric class (such as a neural network, or other architecture) and we select the parameters by using a suitable optimization framework.

There are also mixtures of these two approaches, as exemplified by the recent spectacular successes of programs that, among others, have learned how to play challenging games, such as Go, at or above the level of the best humans. We will describe some of the currently popular approximate DP schemes, and review their range of applications

1.11 Security of Cyber-Physical Systems

Notes and lecture slides can be found in the lecture notes, Lecture 4.1 and the lecture slides 4.1 Security of Cyber-Physical Systems by Henrik Sandberg.

1.12 Control and Coordination of Multi-Agent Systems

1.12.1 Abstraction

During the last decade, control and coordination of distributed multi-agent systems has changed dramatically due to a confluence of new tools found in the intersection of graph theory and systems theory, new application domains, low-cost platforms, and novel control and coordination protocols. As a result, we now have a much better understanding of how large teams of spatially distributed agents should be structured in order to solve increasingly complex tasks. In this talk, we will discuss some of these recent developments and we will show how one can go from high-level specifications and instructions for the team as a whole, to local interaction rules for the individual agents that achieve and maintain formations, cover areas, and even make the agents respond to high-level, human instructions.