ISSN 2186-7437

NII Shonan Meeting Report

No. 153

The 3rd Controlled Adaptation of Self-adaptive Systems (CASaS2020)

Kenji Tei
Nir Piterman
Javier Camara

January 13–17, 2020

National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-Ku, Tokyo, Japan

The 3rd Controlled Adaptation of Self-adaptive Systems (CASaS2020)

Organizers:
Kenji Tei (Waseda University, Japan)
Nir Piterman (University of Gothenburg, Sweden)
Javier Camara (University of York, United Kingdom)

January 13–17, 2020

Self-adaptive systems are required to adapt their behaviour in the face of changes in their environment and goals. Such a requirement is typically achieved by developing a system as a closed-loop system following a Monitor-Analyse-Plan-Execute (MAPE) scheme. MAPE loops are a mechanism that allows systems to monitor their state and produce changes aiming to guarantee that system goals are met. In practice it is often the case that to achieve their desired goals, self-adaptive systems must combine a number of MAPE loops with different responsibilities and at different abstraction levels.

Higher-level goals require decision-level mechanisms to produce a plan in terms of the high-level system actions to be performed. Various mechanisms have been proposed and developed for automatically generating decision-level plans (e.g., event-based controller synthesis), providing guarantees about the satisfaction of hard goals (e.g., providing a certain level of service), and supporting improvements in soft goals (e.g., achieving goals in an efficient or cost-effective manner). These decisions are often made at time scales that can range from seconds to minutes.

Lower-level goals, on the other hand, typically require control mechanisms that sense the state of the system and environment and react at a fine time granularity of milliseconds. Solutions to this problem are typically based on classical control theory techniques such as discrete-time control.

A successful adaptive system, then, must find ways to integrate these multiple levels of control, leading to important questions such as how to achieve such integration in the best possible way, and which abstractions and mechanisms are appropriate to enable such approach. Additionally, concepts from classical control theory (typically applied at low levels of control) can also be useful in understanding higher-level control.

Recently the software engineering community has begun to study the application of control theory and the formal guarantees it provides in the context of software engineering. For example, the 2014 Dagstuhl Seminar "Control Theory meets Software Engineering", is an example of such recent interest. That seminar discussed a variety of possible applications of control theory to software engineering problems.

Additional examples, and perhaps more relevant, are the 1st CASaS Shonan seminar held in 2016 and the 2nd one held in 2017. The 1st CASaS focused on formal guarantees that can be provided in self-adaptive systems via the use of control theory (e.g., event-based controller synthesis and discrete-time control). The seminar was an active gathering of outstanding researchers in both control theory and software engineering, and provided a forum in which discussions on the connections between control theory and software engineering for self-adaptive systems could be held. Most of the attendees expressed their intention to continue studying and discussing the relation between control theory and software engineering, which was highlighted as key to address systematic provision of guarantees about the satisfaction of requirements in self-adaptive systems.

The 2nd CASaS aimed to suggest foundations of the integration.The seminar focused on sharing basic knowledge of the both communities and discussing two key topics for identifying foundations and challenges: properties and composition. The discussion continued even after the seminar and results of the discussion were presented in two reports: “Bridging the Gap between Control and Self-Adaptive System Properties: Identification, Characterization, and Mapping” and “Composition and Cooperation of Multiple Control Strategies: Automating Control Switch with High-Level Guarantees”.

The success of the two previous CASaS seminars motivated us to continue further exploration of the foundation and challenges of the successful integration of multiple levels of control and start consolidating the product of our discussions by focusing on a concrete domain. The 3rd edition of CASaS aims at involving key researchers in robotics in addition to key areas that were present in the past two seminars such as Self-Adaptive Systems, Control theory, Game theory, Software Engineering, and Requirements Engineering. This will create an ideal environment to discuss possibilities of control theory as a mechanism to provide formal guarantees of not only for low-level goals but also high-level goals of self-adaptive systems (e.g., convergence, safety, stability) in the robotics domain.

Among the research questions that we expect to discuss are: How to coordinate multiple levels of adaptive control in the robotics domain? What kinds of properties from classical control theory can be applied at higher levels to guarantee certain properties? To what extent does integration of classical control theory and self-adaptive systems contribute to challenges in the robotics domain? In what ways can AI techniques of planning and machine learning be applied to adaptive systems? How can one deal with uncertainty in a systematic fashion and at different levels of abstraction?

We envisaged the 5-day meeting to be organised in two main parts. During the first day, participants quickly presented their background and what topics they are interested in. Then, for the remaining four days, we identified and discussed the most relevant topics selected by the participants in working groups. In the end, we decided to discuss about three topics: ”Decentralised/distributed of control”, ”Combining MAPE, Control Theory, and Machine Learning Techniques”, and ”Requirements and Specification”

Group 1 : Decentralised and Distributed Control

Rapid technological advances in recent years have enabled the engineering of autonomous systems and components with sophisticated capabilities. These *autonomous agents* range from mobile robots and IoT devices to cloud-deployed software agents, and the exploitation of their combined capabilities is envisaged to support the execution of highly beneficial *missions*, e.g., in emergency response, space exploration, environmental monitoring, and many other application domains.

However, the adoption of autonomous systems in these contexts is limited due to the fact that multiple missions may share the environment in which the systems are deployed, resources, or even the agents executing the mission. As a result, building such systems is very challenging for several reasons including the heterogeneity in capabilities between agents, possible different levels of mission criticality, and mission interdependencies that might be hard to anticipate, hampering provision of assurances.

In the discussion, we introduced a model that incorporates the main concepts to be considered in scenarios in which multiple (potentially interdependent) missions are executed by teams of autonomous agents. We also analyzed the main challenges associated with these scenarios and identified some general principles and techniques that have been employed to address some of these challenges. Finally, we concluded by identifying open research questions and discuss opportunities for future research in the area.

Group 2 : Better Adaptive Systems by Combining MAPE, Control Theory, and Machine Learning Techniques

Two established techniques to engineer adaptive systems are architecture-based adaptation that relies on a Monitor-Analysis-Planning-Executing (MAPE) loop that reasons over architectural models (aka Knowledge) to make adaptation decisions, and control-based adaptation that relies on principles of control theory (CT) to realize adaptation. Recently, we also observe a rapid growing interest in applying machine learning (ML) to support different adaptation mechanisms. While MAPE and CT have particular characteristics and strengths to be applied independently, in this paper we are concerned with the question of how these techniques are related with one another and whether combining them and supporting them with ML can produce better adaptive systems.

We motivated the combined use of different adaptation techniques using scenarios from two different domains and illustrated the analysis involved in combining different adaptation techniques in the discussion. We also identified suggestions for further research in this interesting area.

Group 3 : Requirements and Properties of Adjustable Autonomy

The concept of adjustable or flexible autonomy, intended as a means to enable a system to operate in different autonomic conditions and to transfers control between the system's operators, has been introduced and largely discussed in the literature. Adjusting autonomy is intended as a way to redistribute the operational control among different parts of the system, as well as human operators. Adjustment can be made autonomously by system, i.e. self-adjustment, as well as by humans or other systems that are external to the system. In order to perform informed distribution and adjustment of autonomy, autonomy measurement is a key operation. In literature the attempts to measure autonomy focus mostly on specific attributes of autonomy, like performance or trust.

In this discussion, we considered a framework to make informed decisions on design of autonomy and trade-off between different autonomy decisions. The framework was motivated by a case study of Mars discovery mission, introduced by NASA participant. We focus on specific parameters regarding the mission feasibility and accomplishment. Specifically, we studied a way to define valid structural assignments of control functions and communication paths in systems with autonomous functions, initially at design time, but envisioned to be extensible to run-time.

List of Participants

- Radu Calinescu, University of York, UK
- John Day, NASA Jet Propulsion Laboratory, USA
- Lukas Esterie, Aarhus University,Denmark
- Simos Gerasimou, University of York, UK
- Holger Giese, Hasso Plattner Institute for Digital Engineering / University of Potsdam, Germany
- Paola Inverardi, University of L'Aquila, Italy
- Masako Kishida, National Institute of Informatics, Japan
- Morteza Lahijanian, University of Colorado Boulder, USA
- Alberto Leva, Politecnico di Milano, Italy
- Marin Litoiu, York University, Canada
- Shahar Maoz, Tel Aviv University, Israel
- Hiroyuki Nakagawa, Osaka University, Japan
- Necmiye Ozay, University of Michigan, USA
- Alessandro Papadopoulos, Malardalen University, Sweden
- Colin Paterson, University of York, UK
- Patrizio Pelliccione, Chalmers University of Technology / University of Gothenburg, Sweden
- Bradley Schmerl, Carnegie Mellon University, USA
- Danny Weyns, KU Leuven, Belgium
- Kristin Yvonne Rozier, Iowa State University, USA

Meeting Schedule

Check-in Day: January 12 (Sun)

- Welcome Reception

Day 1: January 13 (Mon)

- Lightning Self-Introduction
- Mini-tutorial 1: Self-adaptive Systems, Danny Weyns
- Mini-tutorial 2: On the control-friendliness of computing systems, Alberto Leva
- Mini-lecture 3: Towards correct-by-construction controller synthesis for self-driving cars, Necmiye Ozay

Day 2: January 14 (Tue)

- Adaptive System Challenges for Space Missions, John Day
- Topic selection and group building
- Group discussion
- Synchronization

Day 3: January 15 (Wed)

- Group discussion
- Excursion and Main Banquet

Day 4: January 16 (Thu)

- Group discussion
- Synchronization
- Report writing

Day 5: January 17 (Fri)

- Report writing
- Wrap up