Self-Adaptation in Highly Distributed Dynamic Systems

Ilias Gerostathopoulos, Sebastian Götz, Filip Krikava, Adnan Shahzada, Romina Spalazzese

Self-Adaptation in Highly Distributed Dynamic Systems

part of **Report from the GI Dagstuhl Seminar 14433 - Software Engineering for Self-Adaptive Systems**

Ilias Gerostathopoulos[1], Sebastian Götz[2], Filip Křikava[3], Adnan Shahzada[4], and Romina Spalazzese[5]

[1] Charles University in Prague, Czech Republic `iliasg@d3s.mff.cuni.cz`
[2] Technische Universität Dresden, Germany `sebastian.goetz@acm.org`
[3] INRIA Lille / University of Lille 1, France `filip.krikava@inria.fr`
[4] Malmö University, Sweden `romina.spalazzese@mah.se`
[5] Politecnico di Milano, Italy `adnan.shahzada@polimi.it`

This breakout group focused on identifying the challenges of performing self-adaptation in highly distributed dynamic systems. This is a pressing issue in self-adaptive systems research, as proposed "smart" systems are increasingly built out of disparate entities (sensors and actuators) that feature a close connection to the physical world – so-called cyber-physical systems (CPSs). Examples are numerous: intelligent vehicle navigation, fleets of autonomous robots, emergency coordination systems, to mention just a few. CPSs are typically distributed at the physical space and feature no firm boundaries – they are open-ended. They are composed of loosely connected entities, that are often mobile. Grafting such systems with self-adaptive capabilities is a distinct challenge, which projects itself in all phases of the autonomic loop.

After debating on the different problems that can arise, the group focused on the issue of efficient information sharing between the entities of a self-adaptive CPS and how this affects the overall utility of the system. Our main assumption was that there is no central coordination point in our target CPS, but rather each entity follows its own decisions based on its partial view over the rest of the system. We also assumed that entities form ad-hoc collaboration groups to achieve common system-level objectives. Each partial view is then updated according to the information shared between the collaborating groups. In this frame, a separate autonomic loop is instantiated for each entity and consists of (i) monitoring the changes in the environment/nearby entities, (ii) creating a partial runtime model capturing the state of the world as viewed by the entity in question, (iii) deciding based on the partial runtime model and the adaptation logic, (iv) enacting the necessary changes to the runtime behavior of the entity in question. In our discussions, we considered the case of a "smart crossroad", where each car decides whether to cross or wait at a crossroad according to its view over the positions and intentions of the other cars in the crossroad. In this case, the ad-hoc collaboration group between the cars that approach the crossroad has the objective to avoid collisions and increase the safety of the whole system.

Moving into the more concrete questions, the following challenges have been discussed in more detail:

1. **How to disseminate the information necessary to build the individual partial runtime models in an efficient way?** The problem is that communicating the information of each entity with every other entity in the CPS overloads the network and threatens the privacy of individual entities. Hence, there is a need for an efficient dissemination strategy for partial model building. Apart from the obvious yet naive strategy where every entity shares all its information with each and every one (mentioned above), we identified two more elaborate strategies: (i) each entity in a collaboration group shares all its information only with the group members; (ii) each entity in a collaboration group shares only the information that are relevant to its objectives and the objectives of the group only with the group members, i.e. shares only a fraction of its information.
2. **How to design runtime abstractions in order to support explicit information exchange between the entities of a collaboration group?** We have looked into the technical aspects for partial model sharing. Essentially, this is a problem of model querying with role-based access control and model merging. The models of the entity that represent its own view should allow to express what information are accessible to whom and what information are to be exchanged in each collaboration group. Once the information has been gathered, the next step is to find a way to merge it into the entity's existing partial view (at the receiving end). This becomes challenging when we consider that the same information can be provided by multiple entities and with different quality and precision. The merging has to therefore involve some trust schemes, temporal aspects as well as to support versioning and transactions.

Acknowledgment

This abstract is the result of a working group of the GI Dagstuhl Seminar 14433.[6]

[6] `https://www.hpi.uni-potsdam.de/giese/public/selfadapt/2014/12/17/report-from-the-gi-dagstuhl-seminar-14433/`