Graph-Transformation-Driven Correct-by-Construction Development of Communication System Topology Adaptation Algorithms

Andy Schürr[1]

Extended Abstract: How will the Internet of the future look like? Which communication mechanisms - as we know them today - will prevail, which novel forms of communication will emerge, and what are the challenges faced regarding the constantly increasing mobile use of communication networks? The Collaborative Research Center 1053 MAKI (Multi-Mechanism-Adaptation for the Future Internet) addresses these questions with a specific focus on the development of offline/online adaptation and optimization concepts for all kinds of communication mechanisms. Twelve subprojects, clustered in three project areas, study for this purpose communication system construction and adaptation mechanisms on different network layers for application domains such as Complex Event Processing, Wireless Sensor Networks, … .

In the Wireless Sensor Network domain (as well as in many other application domains), a large research area focuses on Topology Control (TC), which aims at optimizing the network topology (i.e., the graph-based structure of a communication network) to achieve certain optimization goals (e.g., reducing energy consumption and increasing lifetime of battery-powered nodes) while preserving crucial integrity constraints (e.g., connectivity of the topology or coverage of an observed area). A TC algorithm takes a "raw" topology as input and returns a topology as output that consists of the same nodes but only a subset of the edges (connections) of the input topology. The output topology should perform "better" w.r.t. the specified optimization goals while still fulfilling its integrity constraints.

A typical development workflow for TC algorithms (i) starts with a more or less formal specification (e.g., based on atomic graph operations or constraints that the output topology takes into account) that is then employed to prove its desired and required properties on a high level of abstraction. Afterwards, (ii) the TC algorithm is implemented (often in some general-purpose language such as C/C++ or Java) and tested inside a network simulator or - less often, unfortunately - in a hardware testbed. While numerous approaches exist that facilitate the way from simulation to testbed (e.g., using platform-independent APIs or UML-based code generation), only few attempts have been made to ensure that the implemented Topology Control algorithm is still correct w.r.t. the specified set of integrity constraints!

We propose a model-based approach to overcome this reliability problem of TC algorithms by (i) specifying their desired/required properties using graph constraints, (ii) turning these graph constraints into correct-by construction programmed graph transformations, (iii) adjusting the graph-transformation-based specification to the particularities of the target platform (e.g., limited local knowledge of nodes), and (iv) generating finally platform-specific Java code for simulation and C code for testbed evaluation purposes.

---

[1] Technische Universität Darmstadt, Fachbereich Elektrotechnik und Informationstechnik, Fachgebiet Echtzeitsysteme, Merckstr. 25, 64283 Darmstadt, andy.schuerr@es.tu-darmstadt.de