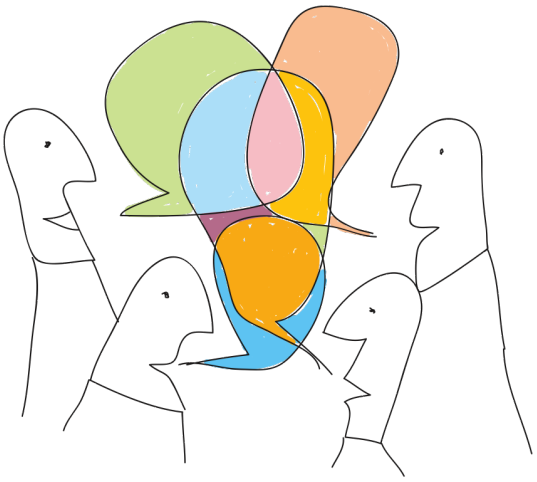
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

Distributed Graph Languages



Remember the seventies, when there were major advances in the study of Turing Machines and Formal Languages? For a fixed language, the task was to design an algorithm (a Turing Machine) that accepts this language, i.e., given *any* word w, the algorithm determines whether w is contained in the language or not.

Similar notions are currently studied in the setting where computers are connected in some network. This is typically modeled using a graph. The nodes in the graph correspond to computers that communicate among each other via the graph's edges. In this thesis, we wish to study what kind of languages a network can accept.

Consider, for example, the set of "all 3-colored graphs". This corresponds to a language in the original setting, and the words of that language are exactly the graphs that are 3-colored. In other words, given any colored graph, the task is to come up with a *distributed algorithm* that determines whether the graph is properly 3-colored.

Of course, there are many more examples of "graph languages". Just like in the seventies, some graph languages can be accepted by a distributed algorithm, while others cannot. In this thesis, you task is to investigate how graph languages can be designed so that they work in faulty networks, and for which languages this is impossible.

Goals

- Develop a deeper understanding of "distributed graph languages".
- Study the impact of faults on existing algorithms for distributed graph languages.
- Design new algorithms that can cope with faulty networks.

Requirements

- You should not be afraid of graphs and networks.
- The ability to work independently on the topic.

Interested? For more details please contact

- Klaus Tycho Förster: klaus.foerster@tik.ee.ethz.ch, ETZ G95
- Jochen Seidel: jochen.seidel@tik.ee.ethz.ch, ETZ G60.1