

# Semantic Models and Formal Analysis of Global Algorithms

Uwe.Nestmann@EPFL.ch

School of Computer and Communication Sciences, EPFL, Switzerland

February 26, 2004

## Ongoing work

We are currently running two funded research projects situated in between the two fields of Concurrency Theory and Distributed Algorithms. One project aims at a semantics-guided design and implementation of distributed algorithms that support group communication middleware. Another project aims at formalizing peer-to-peer algorithms that establish and maintain network overlay structures implementing distributed hash tables; the challenge arises from the unpredictability of the regular usage traffic as well as from the unpredictability of the joins and leaves of network nodes and their failures. In both projects, we use extensions of calculi and techniques that have been studied over the last two decades in Concurrency Theory. The chosen Distributed Algorithm applications appear to be rather non-trivial and challenging. Both can be interpreted as steps towards the study of *global algorithms* that we have to master in the context of global ubiquitous computing.

## Experience I

Globally distributed algorithms can be regarded both at a local level, where individual nodes run (possibly replicated) programs, as well as at a global level where the overall impact of the interacting nodes can be seen at a larger scale. Up to now, programming language technology (e.g., using type systems) has allowed us to study local properties of the code that is run at individual nodes and, maybe, to conclude globally that each individual node of a multi-node system satisfies its local properties. However, in both above-mentioned projects, we rather quickly felt the need to reason about global properties that cannot simply be seen as a conjunction of local ones. For example, the correctness of a consensus algorithm consists of an interwoven set of temporal properties always involving all (non-crashed) participating nodes. Or, for the other example, the correctness of a peer-to-peer algorithm running on a system with usually millions of nodes must at any time take into account the “proper treatment” of

any number of incoming request at any of its active nodes, which depends not only on the local states of nodes, but also on the current traffic in the network.

In order to cope with this need to reason about global states, we were forced to include information supporting a global view within our formal model. To this aim, we extend the “traditional” system descriptions of program code as parallel terms in a process algebra: in essence, we equip process structures with explicit informative environment components that capture assumptions about and the state of the underlying network. For the formal analysis of global properties, we seek to construct corresponding data-oriented views that, at a higher level, try to abstract from the underlying node and code structure.

Comparing our two projects, the global peer-to-peer algorithm is far more complex than the more traditional consensus algorithm in that:

- the number of nodes is far higher;
- the involved local algorithms quite more complicated;
- there are many more dependencies (routing tables) between nodes;
- the global properties require richer formal description techniques;
- nodes operate in a completely desynchronized manner.

## Experience II

Researchers who build distributed systems or algorithms and researchers who are more inclined towards the mathematical foundations and design of calculi and programming languages for distributed systems usually speak rather different “dialects”: they use different terminologies for very similar concepts, and they use the same terminology for rather different concepts. Nevertheless, getting these two groups work together is needed and can be fruitful.

In the context of GUC, the collaboration between the two parties could and should be extended to also include database researchers (c.f. transactions and serializability theory will likely be required for the treatment of global information systems) and their terminology.

## Required Future Work on Algorithms for GUC

We have not yet formally studied the impact of mobility in global algorithms. However, it appears to be natural, for example, to use mobile code for the implementation of peer-to-peer algorithms in dynamically changing networks.

The move from statically distributed systems to dynamic global systems also strongly suggests the move from deterministic correctness properties to only probabilistic and stochastic guarantees about the correctness of algorithms.

Computer-aided analysis and verification seems, for now, out of reach due to the non-finite character of the involved data domains; much more research needs to be invested to this topic.

Case studies, more than toy examples, are important. However, the formal analysis of global algorithms is hard work; we must think very carefully about which algorithms are worthwhile spending the effort. Can SGUC provide them?