

Foreword

There are various logic-based approaches to modelling concurrent programming. The use of logic for system development concerns both the specification step and the study of the operational aspects (through proof analysis and construction) of such development. This means that we have to consider different logical systems for different uses of logic. Moreover, we can also consider concurrency from both points of view: we want to reason and to specify systems where some concurrency aspects are involved, but also to have some operational interpretation of concurrency within logic (focusing on the concept of proof). For this purpose, it seems clear that we have to identify and to study, on the one hand, the role and the treatment of objects (representation, inheritance, modularity, communication, and so on) in this context, and, on the other hand, the interaction of work on proofs and concurrency with the Object-Oriented Programming paradigm.

The ECOOP'96 workshop on 'Proof Theory of Concurrent Object-Oriented Programming' took place in Linz, Austria, in July 1996. Its objective was to provide an integrated forum for the presentation of research and the exchange of ideas and experiences in the topics concerned with proofs, concurrency and object-oriented programming (specification, proof development, and so on). The call for papers for this Special Issue of *Mathematical Structures in Computer Science* can be considered to have been a natural scientific continuation of the workshop.

The papers selected for this Special Issue address some of the topics discussed in the workshop. They present different alternative frameworks that are effectively based on a proof-theoretic approach. They illustrate, from different points of view, the interest, potentialities and difficulties of dealing with such an approach in the design of object-based concurrent systems.

Theoretical research on actors has focused on characterizing the operational behaviour of distributed languages in terms of actor components. The paper entitled *Proof-theoretic foundations for the design of actor systems* shows that the actor model can also be used as a basis for the rigorous design of open distributed systems. The proof-theoretic approach is particularly well suited for describing not only how parts of a system perform computations but also properties that the whole system is required to fulfil, which is very important for designers and programmers.

The paper entitled *A specification logic for concurrent object-oriented programming* focuses on the use of linear logic as a language for specifying advanced concepts of programming – in particular the salient aspects of concurrent object-oriented programming. From a refined sequent calculus based on a proof-theoretical characterization of linear logic programming, an accurate encoding of traditional features of concurrent object-oriented programming (with encapsulation as the cornerstone) is proposed and analysed.

Another proposal for such a framework to deal with concurrent object-oriented features for software development can be based on algebra, mainly hidden sorted algebra, which can be seen as an evolution of Algebraic Specification to handle such features. The

paper entitled *Hidden coinduction: behavioural correctness proofs for objects* focuses on semi-mechanical proof techniques for hidden algebra, as a way of proving behavioural properties of concurrent systems, including refinement. The confrontation of this specific algebraic approach with other approaches, such as, for instance, those based on coalgebras or process algebras, could be fruitful for giving semantics for software engineering that support simple and mechanical correctness proofs.

As Guest Editors, we would like to express our thanks to the authors for their high-quality contributions to this Special Issue and to the referees who have performed an important scientific task, essential for the selection of, and improvements to, the final papers. We are very grateful to Giuseppe Longo, Editor-in-Chief, who has given us the opportunity to organise this Special Issue on the topic *Proof theory of concurrent object-oriented programming*.

Guest editors J. P. Bahsoun, J. L. Fiadeiro and D. Galmiche