Skeleton Algorithms in Concurrent Clean

Viktória Zsók, Zoltán Horváth
Faculty of Sciences, Eötvös Loránd University, 1117 Budapest, Hungary
zsv@inf.elte.hu, hz@inf.elte.hu

Skeletons can express very well the paradigms of the parallel programming. Functional skeletons are polymorphic higher order functions. The static part of a skeletal program is the description of the algorithm. Combining the functional skeletons with evaluation strategies we can obtain optimal efficiency in the dynamics of skeletal programming.

The implementations of the skeletons are not always self-evident, because of the use of the low level language constructs or because of nesting some parts in the compiler.

In the present paper parallel programming skeletal algorithms are investigated regarding the implementation, efficiency, data processing, abstraction levels. The solution will be also examined from the formal reasoning point of view: the semantic and temporal properties of parallel programs.

References

[1] Kesseler, M.H.G.: *The Implementation of Functional Languages on Parallel Machines with Distributed Memory*, PhD Thesis, Catholic University of Nijmegen, 1996.

[2] Chandy, K.M., Misra, J.: *Parallel program design: a foundation*, Addison-Wesley, 1989.

[3] Plasmeijer, R., van Eekelen, M., Pil, M., Serrarens, P.: Parallel and Distributed Programming in Concurrent Clean, In: Hammond, K., Michaelson, G., eds., *Research Directions in Parallel Functional Programming*, pp. 223-338, Springer-Verlag, 1999.