

Un contrôleur de types pour WAVE

Eric Violard (Eric.Violard@inria.fr)

Sujet de TER



WAVE [3] est un langage de programmation parallèle qui inclut certaines idées relatives à l'expression du parallélisme [2]. A l'instar d'autres langages comme par exemple LISP, il est faiblement typé, c.-à-d. que le type des expressions est déduit dynamiquement (au cours de l'exécution du programme). Au contraire, d'autres langages comme OCaml sont fortement typés, c.-à-d. qu'une expression a un seul type et que ce type est déduit automatiquement avant l'exécution du programme, ce qui permet d'éviter toute erreur à l'exécution due à une incohérence de type.

Ce TER vise à munir WAVE d'un typage fort : il s'agit de définir un système de typage qui trouve automatiquement et statiquement le type de toute expression du langage. On pourra largement s'inspirer du typage de OCaml qui s'appuie sur l'algorithme W de Damas et Milner [1]. Il faudra donc définir des règles de typage (qui sont en fait des règles d'inférence), puis concevoir et implémenter l'algorithme qui applique ces règles.

Références

- [1] Luis Damas and Robin Milner. Principal type-schemes for functional programs. In *Proceedings of the 9th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, POPL '82, pages 207–212, New York, NY, USA, 1982. ACM.
- [2] E. Violard. A Semantic Framework To Adress Data Locality in Data Parallel Languages. *Parallel Computing*, 30, 2004.
- [3] WAVE : A Topological Parallel Language, 2013. <http://wave.gforge.inria.fr>.

Référence clé (pour l'UE Initiation à la recherche) : [1]