CSE305 Project: Parallel expression evaluation

Gleb Pogudin, gleb.pogudin@polytechnique.edu

Assume that you have a large arithmetic expression (presented as an expression tree) which you would like to evaluate in parallel. One natural approach is to use divide-and-conquer paradigm: take the outer-most operation, run the algorithm recursively in parallel on subexpressions, and combine. The natural drawback of this approach is, if the expression tree is unbalanced, then the split of work would be not very optimal.

An orthogonal "bottom-up" approach has been proposed by Miller and Raif in the paper "Parallel tree contraction and its applications" [2]. The idea to use two parallelizable procedures "rake" and "compress" in order to reduce a tree to a point in a logarithmic number of steps. This approach can be used for a variety of tasks, not only for expression evaluation, see [3].

The goal of the project would be to implement the tree contraction algorithm for expression evaluation. Note that, while the algorithm not so hard to describe on the diagrams of trees, there are several design choices to make regarding the tree representation in memory and efficiently preforming the main contraction operations. The resulting algorithm should be thoroughly tested on a set of benchmark expressions of varying size, balancedness, etc.

For a group of three, some additional work would be expected, for example, a randomized version of the algorithm (following the original paper) or exploring an alternative approach via reduction to list ranking [1].

References

[1] K. Abrahamson, N. Dadoun, D. Kirkpatrick, and T. Przytycka. A simple parallel tree contraction algorithm. *Journal of Algorithms*, 10(2):287–302, June 1989. ISSN 0196-6774. URL http://dx.doi.org/10.1016/0196-6774(89)90017-5.

[2] G. L. Miller and J. H. Reif. Parallel tree contraction and its application. In *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, page 478–489. IEEE, 1985. URL https://www.cs.cmu.edu/~guyb/paralg/papers/MillerReif89.pdf.

[3] G. L. Miller and J. H. Reif. Parallel tree contraction part 2: Further applications. *SIAM Journal on Computing*, 20(6):1128–1147, Dec. 1991. ISSN 1095-7111. URL http://dx.doi.org/10.1137/0220070.