Background Paper

Samuel Tepoorten & Zhili (Jerry) Pan

Abstract

This paper explores the application of the divide-and-conquer methodology in parallel programming. Traditional algebraic data structures, such as pointer-based lists and unbalanced trees, prove inefficient due to limited middle-division capabilities, constraining list-iterating parallelism. This study proposes a systematic approach for developing balanced trees, using equational reasoning and relational parametricity, to enable purely functional, correct-by-construction parallel programs that can uniformly handle both list and binary tree operations.

Our project objectives are as follows:

- Implement and formally verify the proposed balanced tree structure, Shunt Tree.
- Develop list and binary tree data structures based on the Shunt Tree.
- Implement essential operations, such as `foldl`, `foldr`, `scanr`, and others.

If time allows, we plan to extend our work to explore and re-implement additional data structures, such as sets, maps, and potentially other advanced structures.

References

Akimasa Morihata and Kiminori Matsuzaki. 2011. *Balanced Trees Inhabiting Functional Parallel Programming.* In Proceedings of the 16th ACM SIGPLAN International Conference on Functional Programming (ICFP '11). Association for Computing Machinery, New York, NY, USA, 117–128.
https://doi.org/10.1145/2034773.2034791