# Problems in sequential and parallel game tree search

Bernhard Balkenhol[*]
Fakultät für Mathematik,
Universität Bielefeld,
Posfach 100131,
33501 Bielefeld,
Germany

October 29, 1996

**Abstract**

The $\alpha$—$\beta$– algorithm is an efficient technique for searching game trees. As parallel computers become more available, it is important to have good parallel game tree search algorithms. Until now it is an open problem whether a linear speedup can be achieved with respect to sequential alpha–beta. This note presents a sample of game trees with distinct leaf values, which are easy for sequential alpha–beta – independent of the move ordering in the trees. We conjecture that these trees are difficult test cases for parallel algorithms.

[*]email:bernhard@mathematik.uni-bielefeld.de

# Contents



# 1 Introduction

In the last years the development of parallel computers has made impressive progress. From the algorithmic point of view, game tree searching and especially alpha–beta seems to be hard to parallelise. Normally the running time of the alpha–beta algorithm depends heavily on the ordering of the tree [11]. Karp and Zhang [8, 16] were the first to design a parallel algorithm for searching *bivalued* game trees, which has a linear speedup (even in the worst case). For multi-valued game trees it is still a widely open problem whether a linear speedup can be achieved [5].

More oriented towards practice is the approach of Hsu [7], who belongs to the team of the current computer chess world champion “Deep Thought”. He designed parallel alpha–beta versions with a linear speedup (and good speedup constants), if deep cutoffs in sequential alpha–beta are neglected.

However, in [9] (p. 310) there is an example of a uniform binary game tree of depth 4 with distinct leaf values in which alpha–beta will always find two cutoffs no matter how the branches of the tree are ordered. In [4] the example of Knuth and Moore [9] is generalised. It is shown that a whole class of trees exists which are “alpha–beta easy”.

These “alpha–beta easy” trees may be interesting candidates for testing parallel game tree search algorithms. Since sequential alpha–beta works