

Maximizing the Speedup of Parallel Search Using HyPS *

Diane J. Cook and Shubha Nerur

University of Texas at Arlington
cook@centauri.uta.edu

Abstract

Many artificial intelligence applications rely on performing heuristic search through large, complex spaces. Iterative-Deepening-A* (IDA*) search has been shown to be useful for large search spaces, because it requires no intermediate state storage and is guaranteed to find optimal solutions. However, the time taken to perform IDA* search on real-world tasks often prevents the everyday usage of AI techniques.

Parallel processing can considerably reduce the time spent in search, and can thereby speed up AI applications. This paper describes HyPS, a hybrid parallel window / distributed tree algorithm. Using this algorithm, the set of available processors is divided into clusters. Each cluster searches simultaneously through the same search space, but to a unique cost threshold. Within each cluster, the search space is divided so that an individual processor will search a smaller portion of the overall search space. Load balancing and operator ordering techniques are used to further improve the performance of the HyPS system. Results show a substantial performance improvement over serial search algorithms, and indicate an improvement over existing parallel search approaches.

1 Introduction

Heuristic search provides the driving force for many applications of artificial intelligence including problem solving, robot motion planning, concept learning, theorem proving and natural language understanding. Computational complexity is one of the limitations of search. The research community is continually trying to develop efficient search algorithms.

Parallel search algorithms significantly increase the number of nodes evaluated in a given amount of time. Parallel search techniques have been implemented on MIMD and SIMD architectures. Because of the overwhelming size of real-world AI applications, and because of the increasing power and accessibility of parallel computers, there exists a constant need for improvement of parallel search algorithms.

This paper introduces a hybrid parallel search technique (HyPS) that improves the performance of search using a MIMD architecture. The idea behind the approach is to blend the strengths of existing parallel search algorithms. In particular, HyPS merges the power of a parallel window search with that of a distributed tree search. The resulting algorithm offers improvements over either approach used by itself. The addition of load balancing and operator ordering further improves the performance of the HyPS system. The remainder of this paper describes the HyPS system and demonstrates its performance on two well-known AI application areas.

*This work was supported by National Science Foundation grant IRI-9308308 and by grant TRA93029N from the National Center for Supercomputing Applications.