

Parallel State Space Searching Algorithms

Professor Keyes – APAM 4990
Hart Lambur, Blake Shaw

Overview

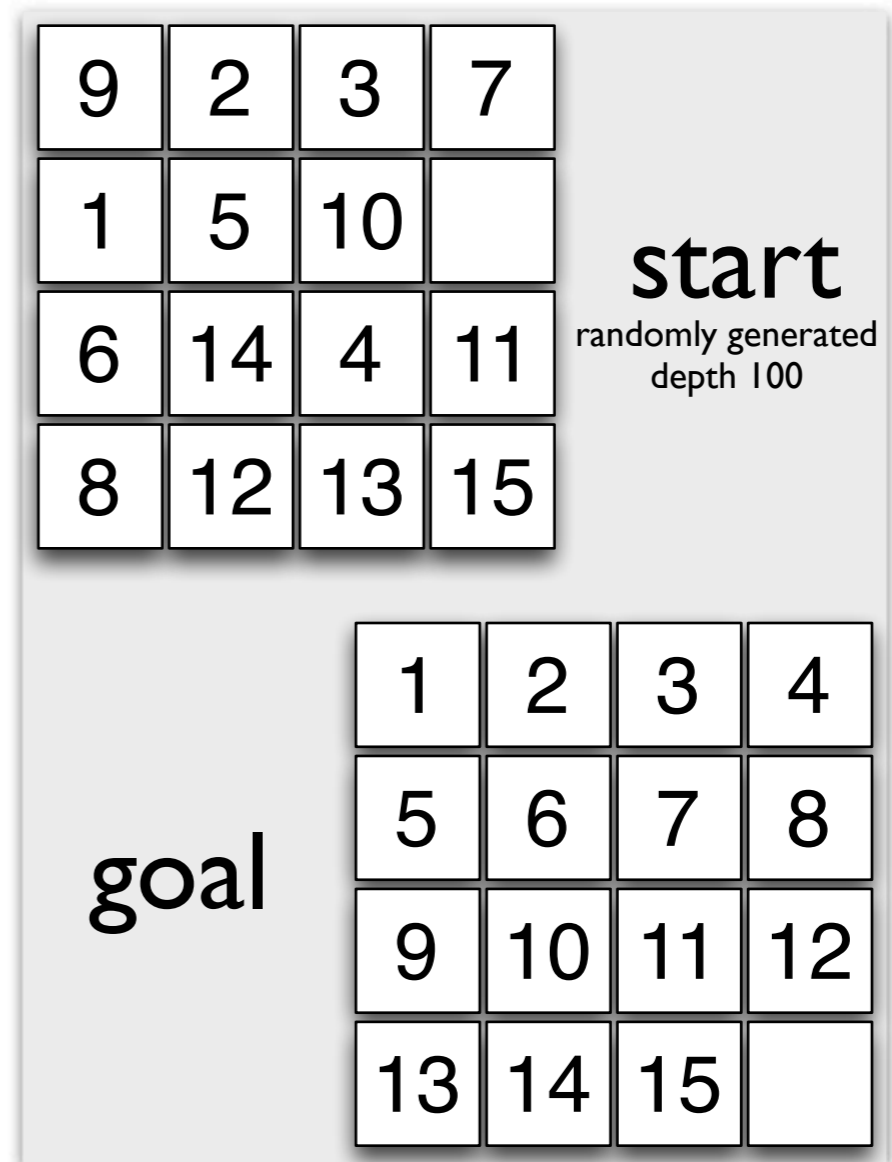
- About us
- The searching problem
- Common search algorithms
- BFS, DFS, and A^* in parallel
- Next time: scaling, profiling with jumpshot, performance, and lessons learned

About Us

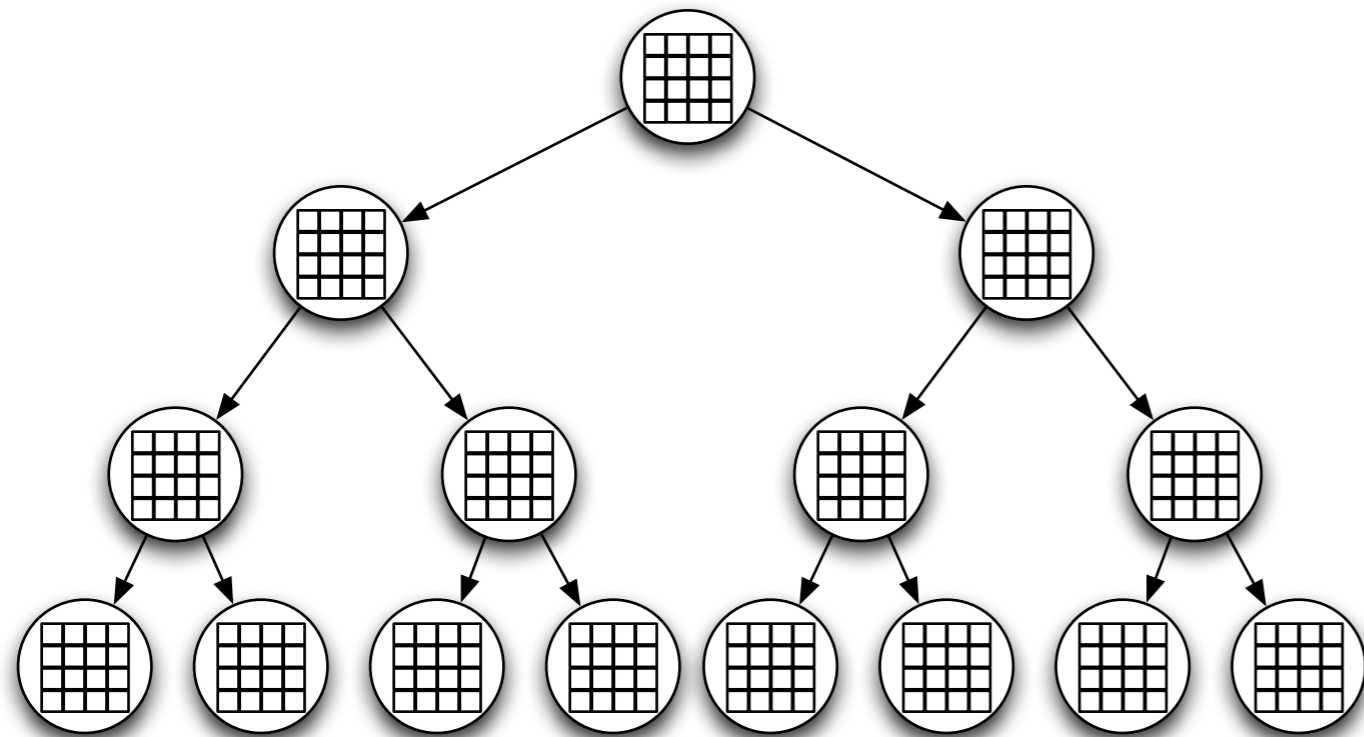
- Undergraduate computer science majors
 - Interest in Artificial Intelligence
- Appeal of APAM 4990
 - Parallelizing common CS problems
 - Importance/availability of clusters

The Problem

- State space searching
- 15 puzzle
- Complexity
- Relevance
- Discrete optimization problems

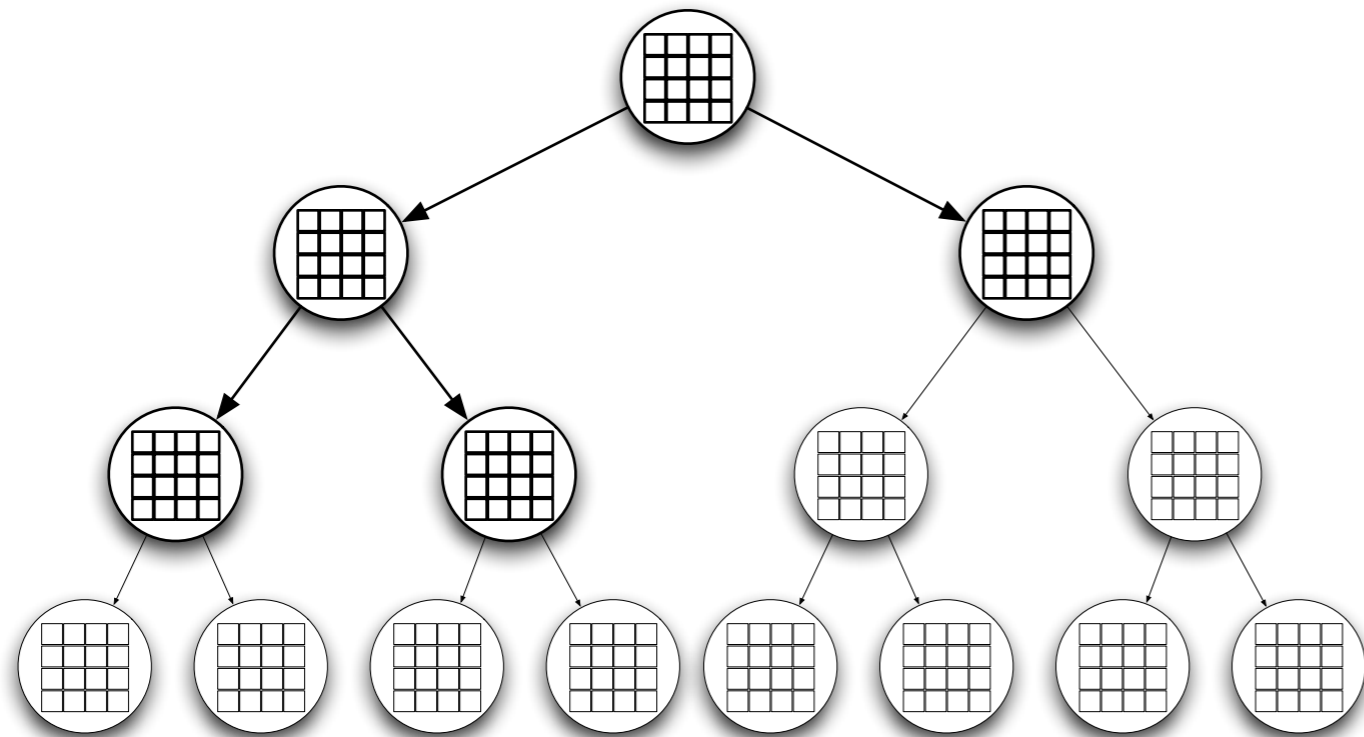


Searching Techniques



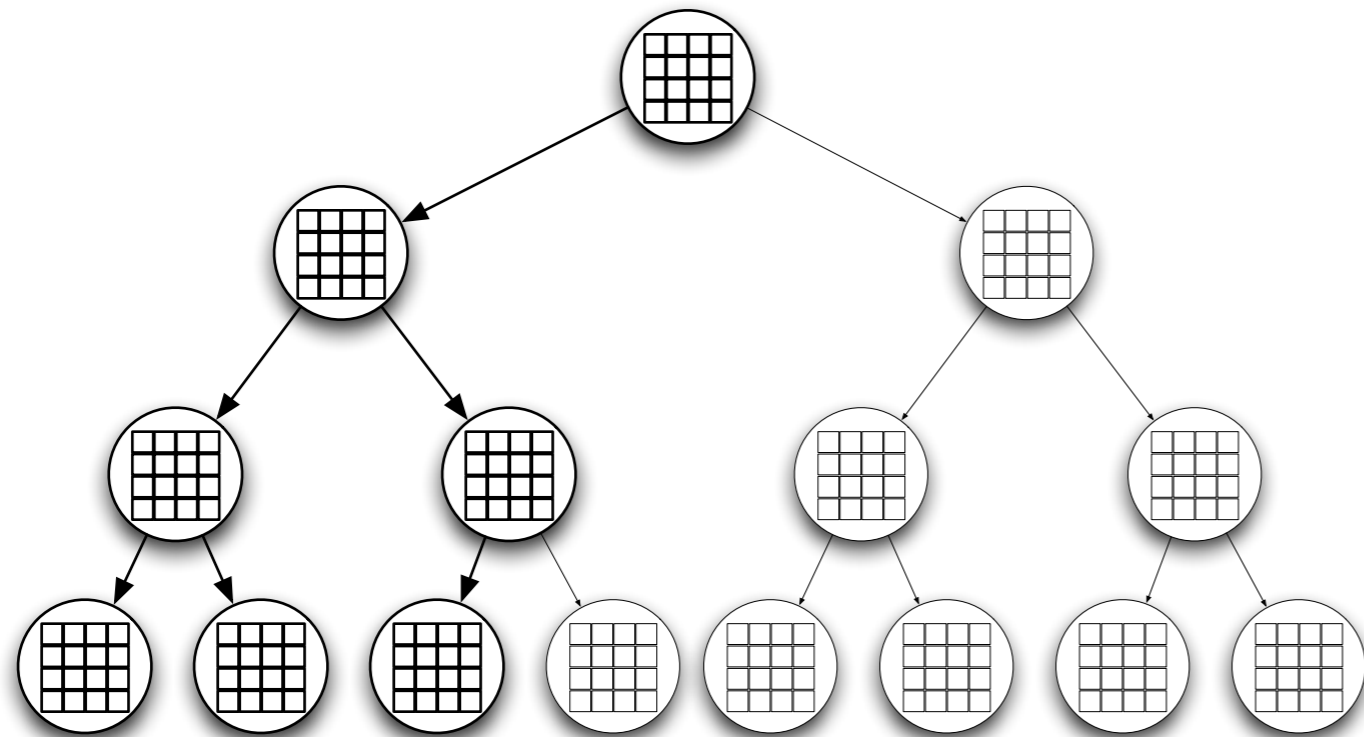
- Breadth First Search (BFS)
- Depth First Search (DFS)
- Best First Search (A*)

Searching Techniques



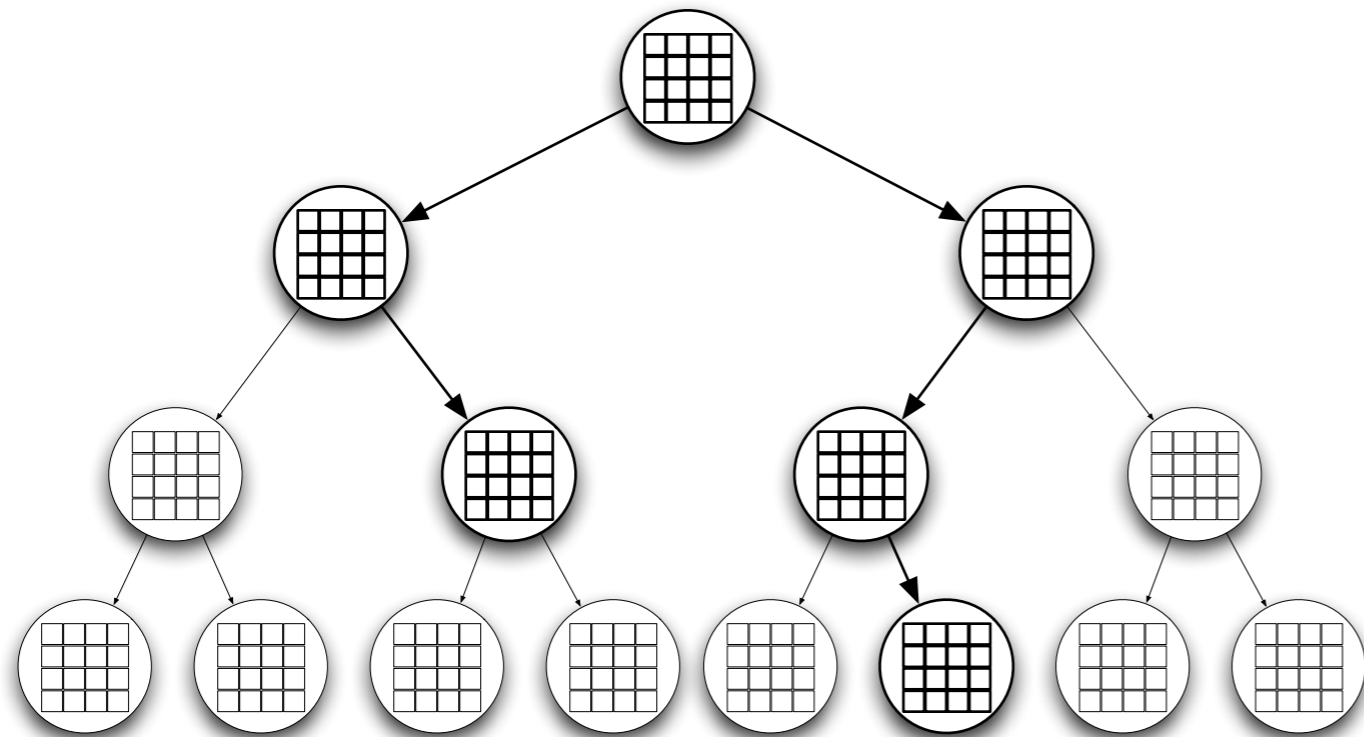
- **Breadth First Search (BFS)**
- Depth First Search (DFS)
- Best First Search (A^*)

Searching Techniques



- Breadth First Search (BFS)
- **Depth First Search (DFS)**
- Best First Search (A*)

Searching Techniques

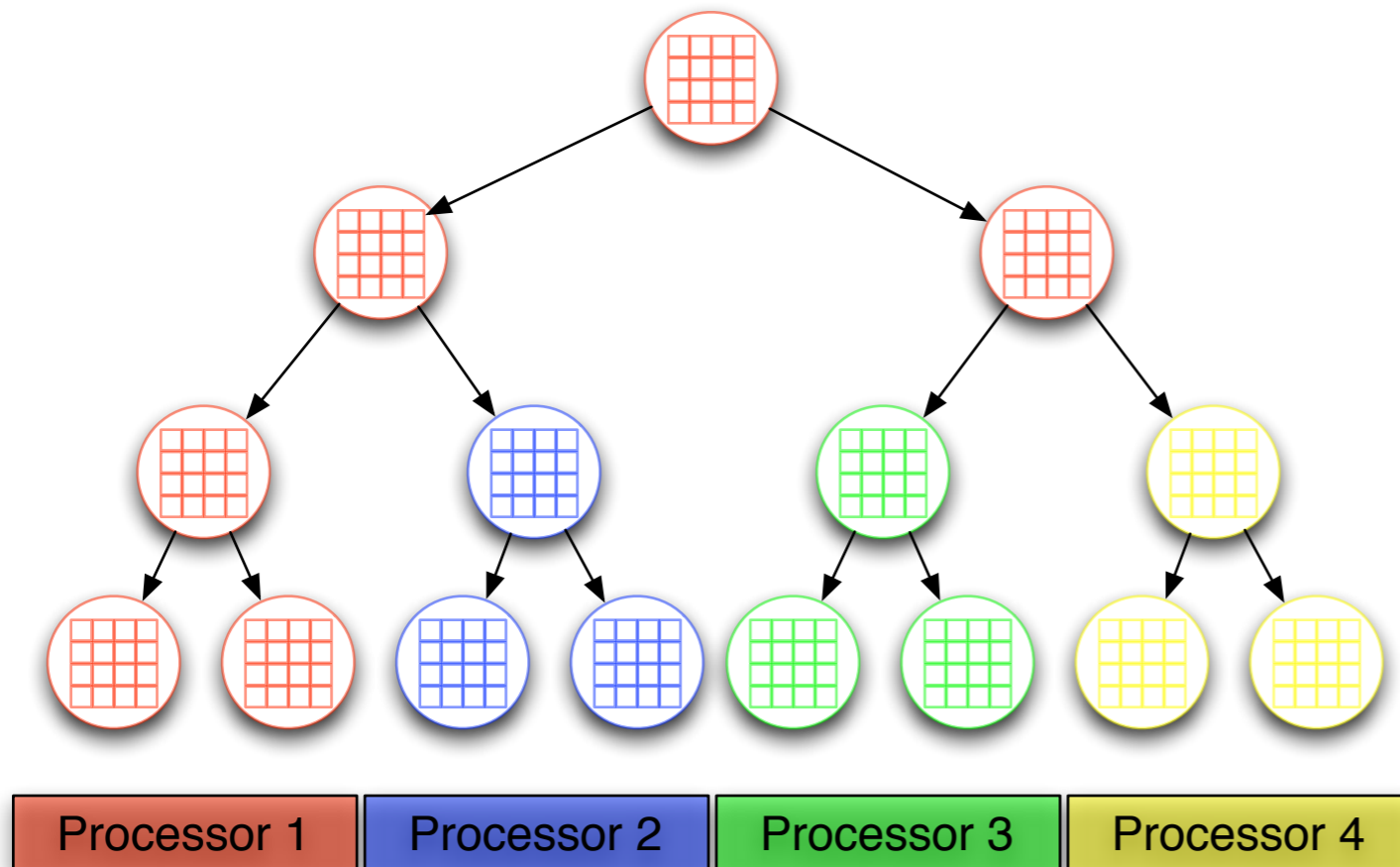


- Breadth First Search (BFS)
- Depth First Search (DFS)
- **Best First Search (A*)**
 - Manhattan Distance Heuristic

Parallel Implementations

- Search overhead factor
- Dynamic load balancing
- Reduce redundancy in each processor's search space

Parallel DFS



- Work splitting strategy
 - Head processor distributes start nodes
 - Each processor maintains own open / closed list
- Load balancing
 - Round robin
 - Polling

Parallel Best First (A^*)

- Work splitting strategy:
 - Head processor distributes nodes
 - Each processor maintains own 'best' list (on open list) and closed list
 - Synchronization strategy to ensure 'best' part of tree is spread to each processor
- General strategies:
 - Random communication
 - Ring communication
 - Blackboard

Implementation

- Existing packages / software:
 - APPSPACK – Sandia National Laboratories
- Our implementation
 - Written from scratch in C++ using MPI-2
 - Object-oriented code structure with separate puzzle and search function classes

