GPU Teaching Kit: Accelerated Computing

Game Tree Search Demo Project

How to Build and Run the Different Agents

CPU Depth-first Search

To build

```
cd src/sequential
make
```

To run

```
./main      -- default game length and search depth
./main x y  -- length of x plies, search depth of y
```

GPU Depth-first Search

```
cd src/cuda
make
./checkers
```

This will cause the agent to play a game of checkers against its self, for at most 100 plies. This code will evaluate each board, to 6 plies deep with the CPU breadth-first search, then it will continue searching an additional 5 plies deeper with the GPU depth-first search, effectively 11 plies. With this depth of search, the game should conclude on the 36th turn.

CPU Breadth-first Search

```
cd src/bfs
make
./main
```

To run the game agent, run the executable ‘main’ located in the `bfs/CPU` directory. This will cause the agent to play a game against itself until there is a winner, or the maximum number of plies is reached. The agent uses a fixed search-depth that is defined in `main.cpp` file. To change the search depth, edit the `bfs::search()` call in the `main.cpp`. The search-depth is the second parameter.

Hybrid Breadth-first Search

```
cd src/bfs
make cuda
./main
```

To run the game agent, run the executable ‘main’ located in the bfs/GPU directory. This will cause the agent to play a game against itself until there is a winner, or the maximum number of plies is reached. The agent uses the same fixed search-depth as the CPU version.