Project Proposal for CS261A

Vaibhav Shah

[vaibhav289@cs.ucla.edu](mailto:vaibhav289@cs.ucla.edu)

Mulled: A Puzzle Game

Problem Statement:-

Mulled: A Puzzle Game is a game developed by Hako Games where the objective of game is to move the white ball to the position marked with a star. Black balls are placed at random positions which cannot be moved directly but can be pushed via the white ball. The problem statement is to solve the game optimally by reaching the goal state in a minimum number of steps.

Problem Space Model:-

1. Set of States :- A physical configuration of the puzzle.
2. Operators:- 4 of them namely Left, Right, Up and Down.
3. Initial State:- White and black balls placed at random positions such that white ball is not at the position marked with a star.

Eg. B-Black Ball; W- White Ball; S- Star position

B

B

B

S

B

W

Initial State

1. Goal State:- White ball must be at the position marked with a star.

Eg.

B

B

B

B

W

Goal State

1. Objective Function:- To make the white ball reach to the position marked with a star in minimum number of steps.

Basic Algorithm:-

1. Add the initial state (the level begin) to the queue.
2. Generate all the successor states of the state we got in the queue.
3. Delete the duplicate successor states.
4. Delete all successor states that are deadlock. Deadlock is when black ball comes on star position and two or more of the four directions are not available for this black ball to move. Make this deadlock detection better so that probability of solving the puzzle increases.
5. Add all left states to the Queue
6. Distance to goal state will also be calculated and taken into consideration.
7. Repeat steps 2-6 until you reach the goal state.

I would also try to implement all the search algorithms like BFS, DFS, Iterative Deepening Depth First Search etc to compare the timings required in different scenarios.

The problem is extremely interesting as in this game the grid can be of any shape and size. As the level of difficulty increases; the presence of a large number of black balls increases the probability of reaching a deadlock state. The logic needs to be understood and analyzed to find the optimal solution. I wish to learn in detail about the basic intelligence of a machine and to inspect the decisions taken by it in order to achieve the goal optimally.

Hypothesis: - For mulled; the foremost obstacle is avoiding the deadlock state as it definitely leads to a losing the game. So I feel if the deadlock states can be avoided in a detailed manner; the chances of winning will increase drastically. The different types of deadlock states are as follows:-

1. Simple Deadlock – Black ball in star position with no way to move.
2. Freeze Deadlock – No way to move in any of the four directions.
3. Psuedo Deadlock – There are ways to move but all of them finally will reach the deadlock state.

My main focus is to improve the deadlock detection algorithm so that you never reach a position from where you cannot win at all. Secondly, I will take into consideration the number of steps needed to reach the destination.

The objective which I will definitely accomplish is to avoid deadlocks in a better manner for a single white ball taking it to the desired destination. I also hope to extend the algorithm for multiple white balls where the decisions taken will be dependent on all of them ultimately taking all of them to destination.