

Inteligência Artificial

Bruno Bastos 93302

Leandro Silva 93446

Structures & Pruning

Structures

- Search Node - node of the search tree.
- Tree Search - data structure responsible for finding the solution by generating nodes for each action.
- Search Problem - initializes the problem.
- Domain - Sokoban class that is responsible for the actions that can be made and evaluates a state using an heuristic.
- State - the Map class.
- Backtrack - dictionary that stores repeated states by storing the hashes of the boxes positions and a list of reachable places by the keeper.

Pruning

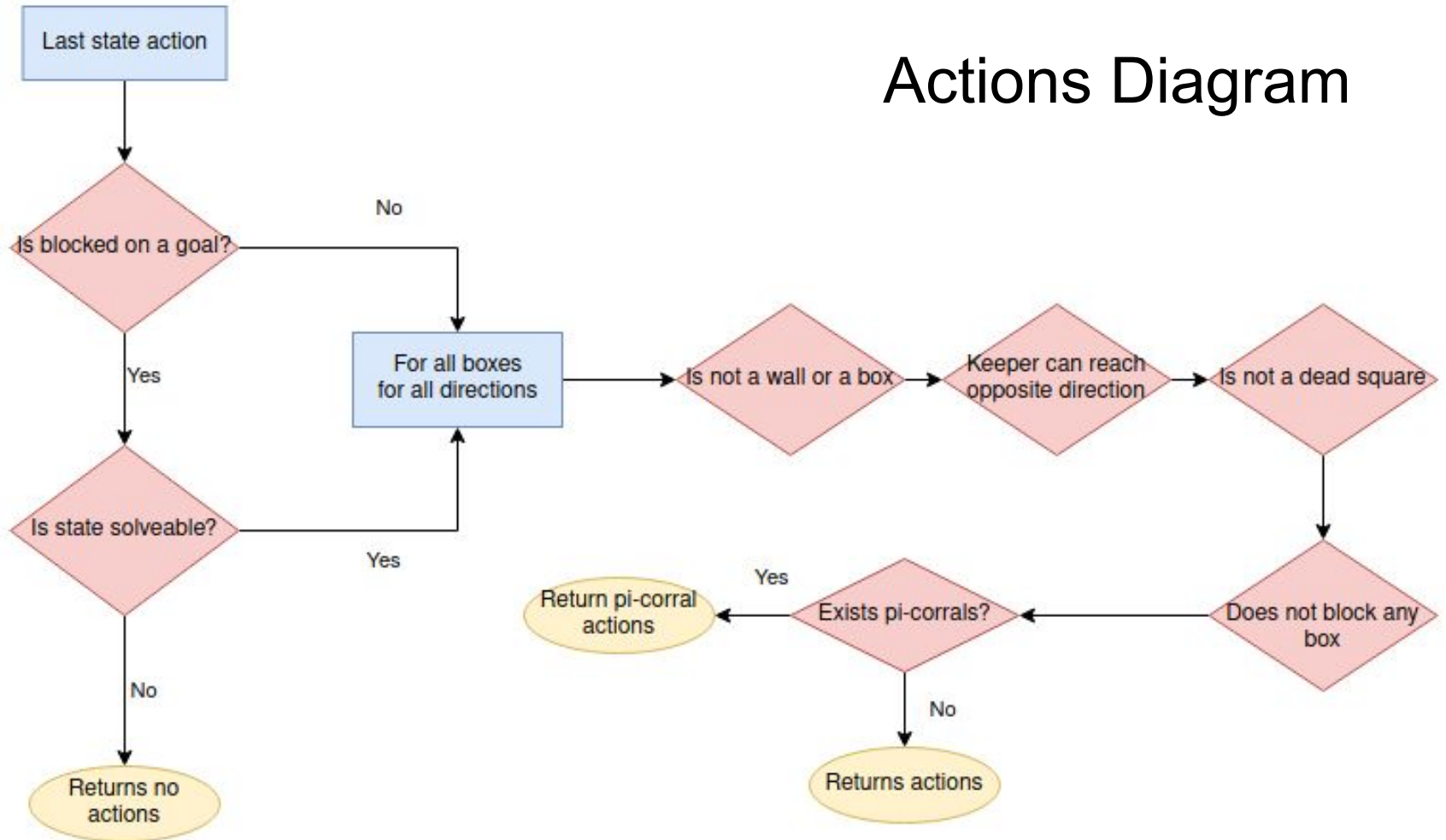
- Backtrack - states already in the backtrack dictionary are discarded if the player can reach one of the stored positions.
- Simple Deadlock- gives us the squares where placing a box makes the level impossible.
- Freeze Deadlock- checks if the placement of a box makes another box impossible to move.
- Pi-Corral Pruning- prioritizes pushes that move boxes to a pi-corral.
- Symmetric Levels - checks if a level is symmetric and if so, stores the states reached and the symmetric equivalents.

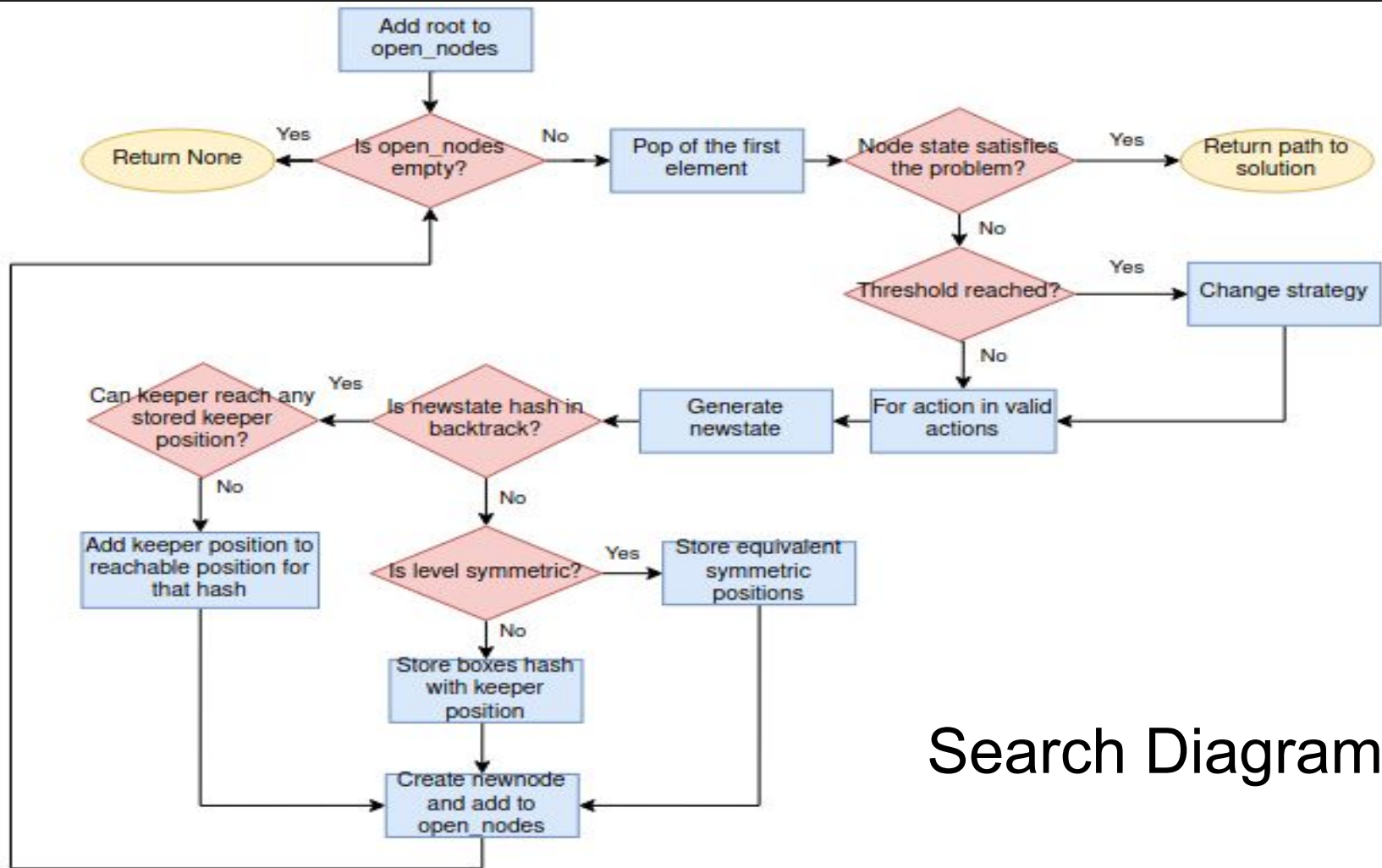
Heuristic and Strategy

At the beginning of a level it is pre-calculated if the distance of the boxes to the goal is the same with walls or without (this is very slow, so if the number of boxes or the map size is high, it is always used greedy). Being the same, it is used a greedy heuristic that doesn't take into account the walls when calculating distances and does not care if the box to storage are perfectly assigned. Else, it is used a more accurate heuristic using the hungarian method and taking into account the walls. The first is faster but less accurate so is best to be used when walls are not in the way.

There are 2 strategies being used. At the beginning of a level it is used A^* to get the a solution that takes the pushes into account. After a Threshold of non-terminal nodes that was chosen by performing many tests, the solver starts using a greedy strategy. Greedy is usually faster at finding the solution but returns worse solutions. Reaching the threshold the level is already taken some time so using greedy helps solving the time restrict problems.

Actions Diagram





Search Diagram

Results & Conclusion

We managed to finish all 155 levels, however some of the new levels added cannot be solved by our solver in the given time. Looking at them, it seems that most of them are symmetric levels and there might some improvements for this type of levels.

We tried lots of strategies and some of them were not finished because the results were not worth it. However, this solver was made with the maximum effort to be able to solve the given levels. This strategies could be important for other type of levels not present in the given ones, for example, levels with lots of tunnels.

During this project, we've shared ideias with Mário Silva (93430) and Daniel Gomes (93015) and with Pedro Tavares (93103) and Gonçalo Pereira (93310).