

PACMAN CONTEST ABSTRACT (ANJA SKRLJ, GUILLEM MASDEMONT)

First, we attempted to implement a standard minimax-alpha-betta algorithm, but we soon realised that not only the final state at depth 4 matters, but also rewards accumulated along the way. For example, if Pacman is eaten at depth 2, that intermediate event must be accounted for.

Based on this observation, we implemented a modified version of minimax with alpha–beta pruning that seeks the path of maximum cumulative reward, taking into account the reward obtained when transitioning through states, and we used Bellman’s equation to compute the optimal action $Q^*(x,a)$ as the sum of the rewards over a fixed horizon plus the value function at the resulting state.

We believe our algorithm can work effectively, however, we would have need more time to fine-tune the parameters of the value function. We were also concerned about the computational cost of deeper search, so we limited the minimax depth to 4. Both of our agents are offensive, since tuning rewards for offensive behavior was simpler, and attacking is generally easier to optimize than defending.