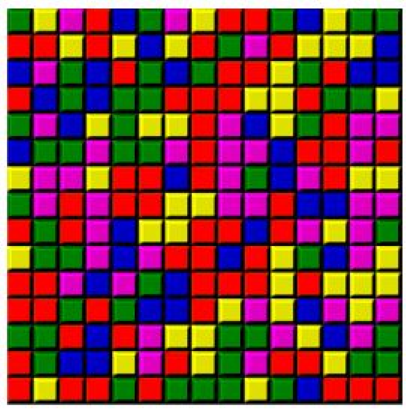
National University of Singapore
School of Computing
CS3243 Introduction to AI

Tutorial 3: Heuristics (Solutions)

1. In the SameGame puzzle, a player is given a two-dimensional, rectangular, $n \times m$ grid of coloured squares. An example of such a grid is depicted in the figure below.



The grid is initially filled with $n \times m$ blocks with c different colours. The position of a grid/tile location is denoted (ROW, COL), where $1 \leq \text{ROW} \leq n$ and $1 \leq \text{COL} \leq m$. Further, let the position (1, 1) reference the top left corner of the grid.

Two tiles are directly adjacent if they are horizontal or vertical direct "neighbors". A tile (i, j) not on the edge of the board has neighbors $(i-1, j), (i+1, j), (i, j-1)$ and $(i, j+1)$, while a tile on the edge has only three of these neighbors, and a corner tile only two.

A group is defined as a set of two or more tiles of the same color, where it is possible to go from any of its tiles to any other by repeatedly visiting adjacent tiles; or briefly: a group is connected. A tile at the board that does not belong to any group is called a singleton.

A move consists of deleting a group (not a singleton), after which, vertical gravity and column shifting are applied: i.e., tiles that are above the deleted tiles fall down, and when any column is completely empty, the columns to the right of the empty column(s) shift to the left.

The goal is to empty the board, leaving no tiles. We call a puzzle solvable if the board can be emptied via a series of moves.

Let the states in this problem correspond to different tile compositions within the $n \times m$ grid, with the initial state corresponding to some initial placement of nm tiles, with each tile corresponding to one of the possible c colours (an example is given in the figure above).

An action is a legal removal of a group of tiles, with the transition model applying vertical gravity and column shifting. The goal state is an empty grid. The transition cost is 1, except for grids where no groups can be removed, in which case the cost of all actions is $\inf$.

Design an admissible heuristic for this puzzle game. Your heuristic may not be $h(s) = 0$ for all states s, the (abstract) optimal heuristic, or a linear combination/simple function thereof. You may assume that the tile layout in the initial grid is solvable i.e. there is some path to a goal state. **You must prove that your heuristic is admissible.**

Solution: $h(s)$ = number of colours remaining.

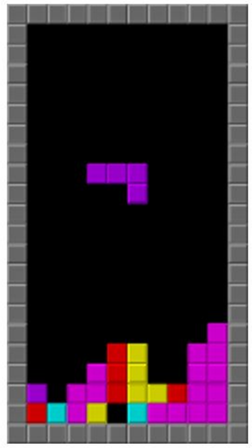
Proof of admissibility.

- Each group contains exactly 1 colour.
- For each remaining colour, 1 or more groups.
- Each move may reduce groups by more than 1 but never reduce remaining groups of 1 colour below 1 (unless that colour is removed by the move).

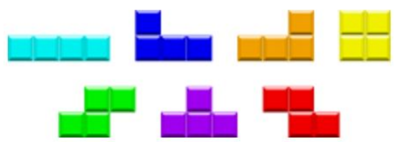
2. Tetris (fill-the-board variant) is a tile-matching game in which pieces of different geometric forms, called *tetriminos*, descend from the top of the field. During this descent, the player can move the pieces laterally (move left, move right) and rotate (rotate right, rotate left) them until they touch the bottom of the field or land on a piece that had been placed before it. The player can neither slow down the falling pieces nor stop them.

   Assume, for our purposes that we are playing a version of Tetris that instead works as follows. Each turn, when a new piece appears to be placed, the player must select the location and orientation before it falls. Once the piece begins to fall, no other adjustments are possible.

   The objective of the game is to configure the pieces to fill a board completely without any surrounded gaps. A gap is defined as an empty cell on the board. A row (or column) is complete if there are no gaps in that row (or column respectively). A gap is called a blocked gap if for the corresponding column where the gap belongs to, there exists an occupied cell somewhere above that gap.



There are 7 kinds of tetriminos. Assume that we start with a fixed number of tetriminos, N (comprising some of each kind), and all are required to be used to fill the board (i.e. there exists a way to place all these tetriminos such that the board is filled).



Thus, in this problem, the **states** are different partially filled tetris fields with a tetrimino that is about to be placed in the field next (but not placed yet); the **initial state** is an empty field with a starting tetrimino; an **action** is the choice of orientation and column for the give piece (assume the player selects an intended configuration before descent); the **transition model** takes in a state, applies the given action on the tetrimino that enters the field, and outputs a state where the tetrimino of that specified configuration descended onto the field; the **goal state** is a completely filled board where there are no gaps (and every tetrimino fits perfectly); the **transition cost** is 1. You may assume there exists such a goal state.

(a) Select all of the heuristics that are admissible. If you feel that none are admissible, select only the option "None of the options are admissible". For each option, briefly, but clearly explain, why it is admissible/inadmissible.

- $h_1(n)$ = number of unfielded tetriminos
- $h_2(n)$ = number of gaps
- $h_3(n)$ = number of incomplete rows
- $h_4(n)$ = number of blocked gaps
- None of the options are admissible.

(b) With reference to the heuristics in the first question of this section, select all of the following that are True. Upon making your choice, pick any **one** of the options, clearly indicate which option you picked and justify your answer for that option.

- $max(h_1, h_2)$ is admissible
- $min(h_2, h_3)$ is admissible
- $max(h_3, h_4)$ is inadmissible
- $min(h_1, h_4)$ is admissible

(c) With reference to the heuristics in the first question of this section, select all of the following that are True. Upon making your choice, pick any **one** of the options, clearly indicate which option you picked and justify your answer for that option.

- h_1 dominates h_2
- h_2 dominates h_4
- h_3 does not dominate h_2
- h_4 does not dominate $h_2/2$

Solution:

(a) $h_1(n)$ and $h_4(n)$ are admissible.

- h_1: Admissible. The number of unfielded tetriminos is the minimum number of steps required to get to the goal, so it is admissible.
- h_2: Inadmissible. Consider a field where a there is a horizontal gap line of size 4. Then the cyan tetrimino rotated can fill it in 1 move. The cost to the goal is 1, but h_2 returns 4 - i.e., it overestimates.
- h_3: Inadmissible. Consider a field where there is a vertical gap line of size 4. Then the cyan tetrimino can fill it in 1 move. The cost to the goal is 1, but h_3 returns 4 - i.e., it overestimates.
- h_4: Admissible (under assumption that blocked gaps may not be filled). The actual cost is infinity as long as there is 1 blocked gap, and the number of gaps on the field must be finite, so underestimates. On the optimal path, this heuristic returns 0, which always underestimates. However, if one assumes that blocked gaps may still be filled, then h_4 is inadmissible since a single piece may fill more than 1 blocked gap.

(b) "$max(h_3, h_4)$ is inadmissible", and "$min(h_1, h_4)$ is admissible", are both True.

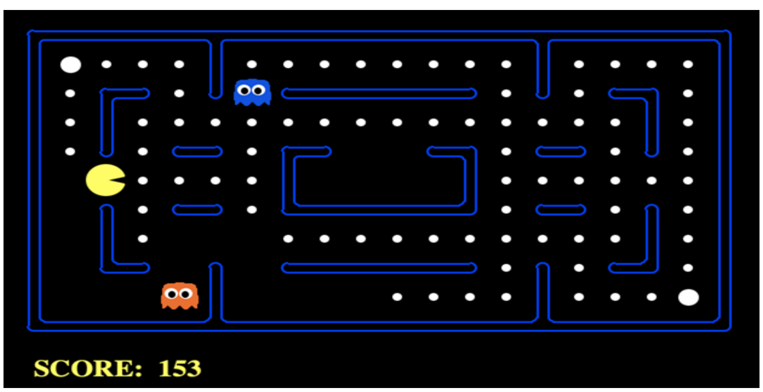
- $max(h_1, h_2)$ is admissible: **False** - the maximum of an admissible heuristic and an inadmissible heuristic is inadmissible since we would pick the higher inadmissible value for some states

- $min(h_2, h_3)$ is admissible: **False** - while is it possible that the minimum of two inadmissible heuristics may still be admissible, this is not generally true since is it trivial to construct a counterexample where both heuristics are inadmissible for the same state, which would cause the minimum of both heursitics at such a state to also be inadmissible
- $max(h_3, h_4)$ is inadmissible: **True** - the maximum of an inadmissible heuristic and some another heuristic is inadmissible since we would pick the higher inadmissible value for some states
- $min(h_1, h_4)$ is admissible: **True** - the minimum of two heuristics when either one of them is admissible, is admissible since we would always choose the smaller (admissible) value

(c) "h_2 dominates h_4", "h_3 does not dominate h_2", and "h_4 does not dominate $h_2/2$", are all True.

- h_1 dominates h_2: **False** - admissible heuristics cannot dominate inadmissible heuristics
- h_2 dominates h_4: **True** - the number of blocked gaps has to be $\leq$ the number of gaps since blocked gaps is a subset of gaps
- h_3 does not dominate h_2: **True** - consider the state n, where a horizontal (cyan) tetrimino piece is required to complete a row; we have: $h_3(n) = 1 < h_2(n) = 4$
- h_4 does not dominate $h_2/2$: **True** - consider the initial state where there are 0 blocked gaps but many gaps

3. Pac-Man is a maze chase video game where the player controls the eponymous character through an enclosed maze. The objective of the game is to eat all of the dots placed in the maze while avoiding the four ghosts that pursue him. (Source: Wikipedia)



Consider a simplified model of the game, where we exclude the ghosts from the game. Also, the player wins when all of the dots (i.e., pellets) are eaten. We model the Pac-Man game as such:

- State representation: Position of Pac-Man in the grid at any point in time and the positions of the remaining (i.e., uneaten) pellets
- Initial State: Grid entirely filled with pellets except at Pac-Man's starting position
- Goal State: No (uneaten) pellets left in the grid
- Action: Moving up/down/left/right
- Transition Model: Updating the position of Pac-Man and eating (i.e., removing) the pellet at this new position (if a pellet exists there)
- Cost function: 1 for each action taken

Consider the following heuristics.

- h_1: Number of pellets left at any point in time
- h_2: Number of pellets left + the minimum among all Manhattan distances from each remaining pellet to the current position of Pac-Man
- h_3: The Maximum among all Manhattan distances from each remaining pellet to the current position of Pac-Man
- h_4: The average over all Euclidean distances from each remaining pellet to the current position of Pac-Man

Determine the admissibility of each of the above heuristics. Provide justifications.

Solution:

- h_1 is admissible. At any point of time, the number of pellets will always be less than or equals to the number of moves that the pacman take. Best case/optimal scenario occurs when at every move pacman makes, he eats a pellet. Hence, actual cost = total number of pellets.
- h_2 is inadmissible. Consider the case when there's 1 pellet left and pacman is 1 step away from the pellet. h_2 = 1 + 1 = 2. Actual path cost = 1. Overestimation.
- h_3 is admissible. Manhattan distance is the x,y distance from the pacman position to a particular pellet. The max value of Manhattan distance ensures that there is no overestimation as this is the maximum distance that the pacman is required to move to consume a pellet.

- h_4 is admissible. Euclidean distance is similar to Manhattan distance, but with a further relaxation on the constraint of pacman movement. We consider the straight line distance when computing Euclidean distance, where it is an underestimation of the (x,y) movement that pacman can make.

4. Now compare the dominance of the admissible heuristics that you have selected from the previous question. Justify your answer.

   For this question, assume that dominance **does not** assume admissibility. Further, note that there will be no error carried forward if you have selected any inadmissible heuristic.

   Solution:

   - h_3 dominates h_4. Euclidean distance is a looser constraint as compared to manhanttan distance. At all nodes, the value of h_3 will be $\geq$ the value of h_4.
   - h_1 does not dominate h_3 or h_4. We can simply construct two counter examples.
     - Case 1: There is one pellet remaining, and pacman is 10 units away from it. Here, we have $h_1 = 1$, $h_3 = 10$, $h_4 = 10$. Consequently, h_3 and h_4 both dominate h_1.
     - Case 2: There are four pellets remaining, with each being 1 unit away from pacman - i.e., 1 unit above, 1 unit beside (left and right), and 1 unit below. Here, we have $h_1 = 4$, $h_3 = 1$, $h_4 = 4/4 = 1$ h_1 dominates both h_3 and h_4.