National University of Singapore
School of Computing
CS3243 Introduction to AI

Tutorial 3: Heuristics

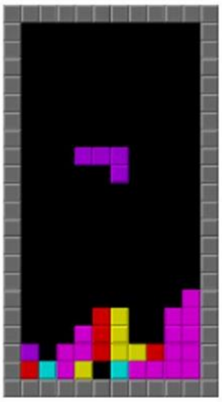
Issued: Week 4 Discussion in: Week 5

Important Instructions:

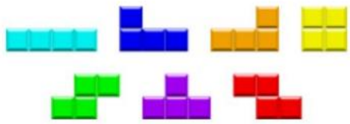
- **Tutorial Assignment 3** consists of **Question 2** from this tutorial.
- Your solution(s) must be TYPE-WRITTEN, though diagrams may be hand-drawn.
- You must submit your solution(s) via **Canvas** > **Assignments**, satisfying the deadlines:
  - Pre-tutorial 3 submission by **Week 4 Sunday, 2359 hrs**.
  - Post-tutorial 3 submission by **Week 5 Friday, 2359 hrs**.
- You must make both submissions for your assignment score to be counted.

---

1. 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 (refer to the image above). 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 the heuristics that are admissible from. If you feel that none **are admissible**, select only the option "*None of these 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 these options are admissible.

(b) With reference to the heuristics defined in Part **(a)**, select all the following statements that **are True**. If you feel that none of the statements are True, select the option "*None of these options are True*". For each option, briefly, but clearly, explain why it is True/False.

- ***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.
- None of these options are True.

(c) With reference to the heuristics defined in Part **(a)**, select all the following statements that **are True**. If you feel that none of the statements are True, select the option "*None of these options are True*". For each option, briefly, but clearly, explain why it is True/False.

- h_1 dominates h_2.
- h_2 dominates h_4.
- h_3 does not dominate h_2.
- h_4 does not dominate $h_2/2$.
- None of these options are True.

2. 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 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 the dots (i.e., pellets) are eaten. We model the Pac-Man game as such:

   - **State representation**: The position of Pac-Man within the grid at any point in time and the positions of the remaining (i.e., uneaten) pellets.
   - **Initial State**: A grid that is entirely filled with pellets except at Pac-Man's starting position.
   - **Goal State**: A grid with no (uneaten) pellets remaining.
   - **Action**: Moving up, down, left, or 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.

   Consider the following heuristics.

   - h_1: The number of pellets left on the grid.
   - h_2: The number of pellets left on the grid + the minimum among all the Manhattan distances between each remaining pellet and the current position of Pac-Man.
   - h_3: The Maximum among all Manhattan distances between each remaining pellet and the current position of Pac-Man.
   - h_4: The average over all Euclidean distances between each remaining pellet and the current position of Pac-Man.

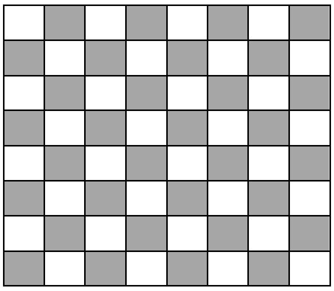
   Determine the admissibility of each of the above heuristics. Provide justifications.

3. Extending from the previous question (i.e., Question **2**), 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.

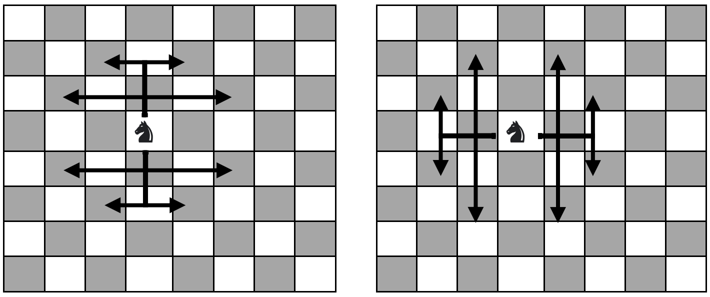
4. Consider the following puzzle that is played out on a Chess board.

   Chess is typically played on an 8-by-8 board, which is depicted below.



However, assume that the current puzzle is played on a Chess board of infinite size.

A Knight (♞) is a Chess piece that moves in an "L" shape. More precisely, the Knight piece can move two cells vertically followed by one square horizontally, or two cells horizontally followed by one cell vertically. The Knight piece's movement is summarised in the figures below.



In this puzzle, a Knight starts at a given Chess board cell, (x_s, y_s), and must find the ***shortest path*** to a specified *goal* cell at (0, 0) using the **A* Search** algorithm.

Do note that the Chess board in this puzzle will contain several obstacles. The Knight may **NOT** occupy any square that is occupied by an obstacle, although it may jump over the obstacles. For example, a Knight is allowed to move from cell (x, y) to cell $(x+2, y+1)$ even if square $(x+1, y)$ contains an obstacle.

Assume that the ***cost*** of any action taken by the Knight is equal to 1.

Further, assume that **two heuristics**, h_1 and h_2, have been defined.

The heuristic, h_1, is defined as follows.

- $h_1(n) = 0$ if $n = goal$, where n and $goal$ are coordinates on the Chess board
- $h_1(n) = 1$ if $(n \neq goal) \land (\textbf{\textit{ManhattanDistance}}(n, goal) \% 2 = 1)$
- $h_1(n) = 2$ if $(n \neq goal) \land (\textbf{\textit{ManhattanDistance}}(n, goal) \% 2 = 0)$

The heuristic, h_2, is defined as follows.

- $h_2(n) = \lceil \textbf{\textit{ManhattanDistance}}(n, goal) / 3 \rceil$

Note that the **Manhattan distance** (i.e., the output of the ***ManhattanDistance*** function) between two cells, (x_1, y_1) and (x_2, y_2), is given by $|x_2 - x_1| + |y_2 - y_1|$.

Consequently, it should be noted that h_1 is admissible since every Knight's move will change the Manhattan distance to the goal from odd to even, or from even to odd. The heuristic h_2 is also admissible since every Knight's move covers a Manhattan distance of 3, but not every cell that has a Manhattan distance of 3 away may be reached via a Knight's move.

(a) Prove or disprove the **admissibility** and **consistency** of the following heuristic.

$$h_3(n) = h_1(n) + h_2(n)$$

(b) Let the **Chebyshev distance** between two cells, (x_1, y_1) and (x_2, y_2), be defined by $\boldsymbol{max}(|x_2 - x_1|, |y_2 - y_1|)$. Assuming that the function ***ChebyshevDistance***(*a*, *b*) determines the Chebyshev distance between cells *a* and *b*, prove or disprove the **admissibility** and **consistency** of the following heuristic.

$$h_4(n) = \lceil \boldsymbol{ChebyshevDistance}(n, goal) \ / \ 2 \rceil$$

(c) Define an admissible heuristic h_5 that dominates h_4. The heuristic must be defined mathematically, and given an infinite board, must differ from h_4 in an infinite number of cells. Provide the justification that (1) h_5 is admissible, (2) h_5 is dominant over h_4, and (3) h_5 differs from h_4 in an infinite number of cells (i.e., you should show that h_5 does not, trivially, only improve on h_4 for a finite number of points). *Note that your heuristic must not be the (abstract) optimal heuristic,* h* *or any function over it.*