Introduction

Formalizing Intelligence

Introduction to Artificial Intelligence

Chandra Gummaluru
University of Toronto

Version W22.1

- The following is based on material developed by many individuals, including (but not limited to):
  - Sheila McIlraith
  - Bahar Aameri
  - Fahiem Bacchus
  - Sonya Allin

- Our goal is to develop agents that exhibit “intelligent” behaviour through computational means.
- Some common definitions of “intelligence” include:
  - the ability to optimally deal with new situations
  - the ability to acquire and apply knowledge and skills
  - the ability to act like a human
- All of these definitions are too imprecise to build computational algorithms around.
- We seek a formal definition that still captures the core ideas of the colloquial ones.

- Our definition of "intelligence" will be:

  the ability to optimally play games,

  where by "game" we mean any situation in which:

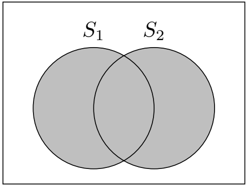
  - the environment can be in a number of different states
  - there are multiple players capable of manipulating the state
  - each player seeks to alter the state to its own benefit

  and by "optimally play", we mean to alter the state so as to maximize our benefit.

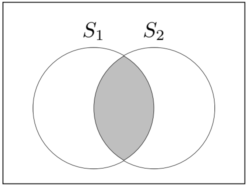
- To define games more formally, we will need to have some understanding of sets.

- A **set**, S, can be thought of as a list of objects, called its **elements**; we write $s \in S$ to denote that s is an element of S.
- A set, S', is a **subset** of S, denoted $S' \subseteq S$ if S contains every element in S'; it is a **proper subset**, denoted $S' \subset S$, if S contains at least one element not in S'.
- The **power-set** of a set, S denoted $\mathcal{P}(S)$ is the set of all of S's subsets.

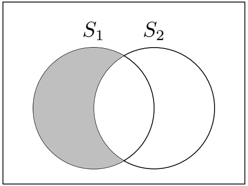
- The **union** of two sets, S_1 and S_2, denoted $S_1 \cup S_2$, is the set of elements in either S_1 or S_2.



- The **intersection** of two sets, S_1 and S_2, denoted $S_1 \cap S_2$, is the set of elements in both S_1 and S_2; if $S_1 \cap S_2 = \emptyset$, then S_1 and S_2 are **disjoint**.



- The **difference** of two sets, S_1 and S_2, denoted $S_1 \setminus S_2$, is the set of elements in S_1 but not in S_2.



- Formally, a **game** consists of:
  - N players, indexed 1 through N
  - a set of states, S, in which the game could be in
  - a set of terminal states, $T \subseteq S$ in which the game ends
  - a utility function, u_i for each player i, so that $u_i(s)$ is the benefit of $s \in T$ to i
  - a set of actions, $A(s)$ from each state $s \in S$; each action $a \in A(s)$ is an n-vector where a_i denotes i's action with $a_i = \emptyset$ to denote no action.
  - optimally, a cost function, c, so that $c(a)_i$ is i's cost of playing a_i.
- Given an $s_0 \in S$, the outcome of a game is called a **path** from s_0; it is a sequence of actions, $\langle a^{(1)}, \ldots, a^{(n)} \rangle$, such that $a^{(i)} \in A(s_i)$, $s_i = a^{(i)}(s_{i-1})$ and $s_n \in T$:
  - the utility for i is $u_i(s_n)$.
  - the cost to i is $\sum_j c\left(a^{(j)}\right)_i I\left(a_i^{(j)} \neq \emptyset\right)$, where I is the indicator function.

- Given some $s_0 \in S$, we want to find the cheapest path to some $s \in T$ that also maximizes our utility.
- This is difficult to do in general because we do not complete control over the other players' actions, and consequently, the outcome of the game.
- As such, we will start by making several simplifying assumptions.
- Over time, we will relax some of these assumptions.