Undefined Utility in Terminal State

by Sven Nilsen, 2019

Assume that an agent is rewarded by evaluating a utility function, taking sensory input about the world and returning some real number. The higher the number is, the higher preference the world has. By predicting world states that score a high utility value from choice of actions, a rational agent is capable of intelligently choose its actions such that the utility over time is maximized. This kind of agent is called “instrumentally rational”.

Assume that a rational agent is equiped with the ability to reason about goals. This kind of agent is called “zen rational” and is an extension of “instrumentally rational”.

We ask a zen rational agent what range of utility it has in a terminal state. What will it answer?

Since the reward is not given in a terminal state, the zen rational agent will answer that the range is empty. Although this is the correct answer, there is a deeper logical connection.

Utility from a historical perspective is meant to model trade-offs between suffering and pleasure. To show this, we create a simple logical system for suffering, pleasure, terminal state and utility:

$$\text{alive} := \text{suffer} \lor \text{pleasure}$$

$$\text{utility} := \neg(\text{suffer} \land \text{pleasure})$$

The two bits `suffer` and `pleasure` might be reduced to a single bit system, by assuming that the agent is alive and using utility to make decisions. To include the terminal state requires another bit.

To be alive means experiencing suffering or pleasure. A utilarian agent makes trade-offs between suffering and pleasure, by stating that suffering and pleasure can not occur at the same time. If there is a large amount of pleasure and some amount of suffering, the agent adds the values together and consider it some amount of pleasure when deciding which actions to take.

However, the agent might be in a terminal state, by experiencing neither suffering or pleasure. This is not the same as zero utility. The terminal state of the agent can not be found in the subtype:

$$x : [\text{utility}]\ a$$

$$\begin{aligned} &a : \text{real} \\ &\text{utility} : \text{world} \rightarrow \text{real} \end{aligned}$$

On the other hand, the terminal state can be thought of as an extended utility function, returning `none()` because no action is needed to be taken in the terminal state:

$$x : [\text{utility}]\ \text{none}()$$

$$\text{utility} : \text{world} \rightarrow \text{opt}[\text{real}]$$