Abstraction and cooperation

Daniel Tang

June 20, 2024

1 Abstraction in evolving agents

Suppose there exists an agent and an environment. The agent receives a stream of sense experiences from the environment and emits a stream of control signals to the environment. The next control signal is a, possibly stochastic, function of the past senses and past control signals, call this the agent's *policy*. Signals can be turns-based, Poisson or per timestep. In the Poisson case, the policy must specify a rate for all possible control signals. In the remainder of this document we assume that signals are turns-based. Clearly, there is a symmetry between agent and environment, so we can model both as a policy over an output alphabet.

In this view, the agent's body is considered part of the environment. The agent's life begins in an environment drawn from a prior distribution of environments. The environment also contains the agent's brain, which is the physical substrate upon which the behavioural function operates. However, the environment abstracts over the calculation of the agent's policy, but has certain physical requirements in order to maintain its validity as a substrate for the calculation of the policy. If this validity fails (e.g. through not eating or falling head-first off a cliff) then the agent's life ends.

Every agent also has a *genome* which identifies the policy from a family of functions spanned by the genome space. Certain behaviours result in the (possibly mutated) reproduction of the geneome. For a fixed environment, every policy has an associated probability of reproduction, averaged over all possible lives (if agents reproduce sexually this should be a probability of reproduction with each other agent in the environment, but here we'll consider asexual reproduction for simplicity).

While the genome defines the policy, it also affects the physical requirements of the agent's brain. E.g. more computation requires more size/energy. So, a genome that computes the same policy in a more computationally efficient way may live longer under scarce resources. So, we should expect that a more computationally efficient computation of the same policy should have a larger expected probability of reproduction.

We happen to live in a world where the consequences of behaviour on the probability of reproduction are insensitive to detailed behaviour within certain categories. We either succeed or fail to kill an animal or mate. This means that there is ample opportunity for an agent to use abstraction in order to reduce computational requirements.

1.1 Abstraction in agents

Abstraction for an agent can happen in two domains: first, an agent can abstract over the situation it finds iteslf in. i.e. over the history of observation and behaviour. Secondly, an agent can abstract over its output alphabet.

A policy can be thought of as a partition of the space of all possible histories, such that each partition is associated with a member of the output alphabet.

We assume that the environment is in some state, ψ_e, and its next output is a, possibly stochastic, function only of its state $f_e(\psi_e)$, and that its state changes as a function of the output signal $\psi_e^{t+1} = g(\psi_e^t, o^t)$ and subsequent input signal $\psi_e^{t+2} = h(\psi_e^{t+1}, i^{t+1})$. We also assume that if the agent knows the state of the environment, then its output is well defined as $f_a(\psi_a, \psi_e)$