

## Purpose

Dawkins Weasel is a NetLogo model that illustrates the principle of evolution by natural selection. It is based on the thought experiment and computer model proposed by Richard Dawkins in *The Blind Watchmaker* (1996).

## Entities, state variables, and scales

Entities are STRINGS that are comprised of a list of characters. For example, a STRING could be 'ABCDEFGH' or 'METHINKS IT IS LIKE A WEASEL.' These characters are the same length as the *target-phrase*, a phrase provided upon initialization. The list of possible characters include any uppercase letter in the English alphabet or a blank space. These STRINGS exist non-spatially and do not interact with one another.

## Process overview and scheduling

Each timestep ends with the selection of the "parent" to produce the next generation. Upon initialization, a STRING is randomly generated to be the first parent. At the start of each timestep, the previously selected parent STRING produces new offspring equal to the *number-of-offspring* input parameter. Each new offspring is a copy of the original parent except each copied character is subject to mutate with a chance equal to the *mutation-rate*. Mutated characters randomly change to one of the letters of the English alphabet or a blank space. Once all offspring have been created, Dawkins Weasel selects the parent for the next generation based on which offspring STRING has most characters in common with the initial *target-phrase*. If multiple STRINGS are equally similar to the *target-phrase*, one is selected at random to be the parent. This process continues until the selected parent exactly matches the *target-phrase*.

## Basic principles

Dawkins Weasel is inspired by a thought experiment and computer model presented by Richard Dawkins in his book *The Blind Watchmaker* (1996). He presents the idea that a monkey randomly bashing on a typewriter is unlikely to produce the works of Shakespeare, or even a phrase like 'METHINKS IT IS LIKE A WEASEL,' in a reasonable amount of time. However, with selection, the problem becomes quite manageable. Instead of a monkey typing randomly, it reviews its typed phrases and selects out the phrase that best matches the goal phrase 'METHINKS IT IS LIKE A WEASEL.' The monkey then proceeds to type out this new phrase multiple times. In this process, there is a small chance that the monkey could mistype and

produce a slightly different phrase. Through this iterative process, the monkey will eventually type out the original phrase.

## Emergence

If *with-selection?* is set to true, then populations of STRINGS are expected to evolutionarily converge on the *target-phrase* in relatively few generations. However, other factors impact this outcome. If the *mutation-rate* is set to 0 then no variation in STRINGS can be produced and thus there is nothing upon which selection could operate. Conversely, if *mutation-rate* is set too high, then too much variation is produced at each generation and any progress in evolving towards the *target-phrase* is lost. Likewise, the *number-of-offspring* must also be high enough to produce variation, and certainly greater than one, which would produce no variation. If *with-selection?* is set to false, then populations of STRINGS are not expected to converge on the *target-phrase* unless it only comprises a few characters. If the *target-phrase* is too long, then mutation alone is likely insufficient to enable populations to evolve to match it through random chance.

## Adaptation

STRINGS inherit the same phrase as their parent, with some chance of error when each character is copied to the offspring. Error is determined by the input parameter *mutation-rate*. A mutation causes one of the characters inherited from the parent to change to either a randomly selected letter or blank space.

## Objectives

Every offspring STRING produced at each generation is checked against the *target-phrase*. The offspring with the closest matching phrase to the *target-phrase* is selected as the parent for the next generation.

## Prediction

Simulations with *selection-on?* set to true, and reasonable settings for *mutation-rate* (non-zero and relatively low), *number-of-offspring* (greater than one), and *target-phrase* (relatively short in character length) are expected to produce STRING populations that evolutionarily converge on the *target-phrase* in relatively few generations.

## Collectives

Each generation comprises a collection of offspring STRINGS generated from the selected parent from the previous generation.

## Observation

A simulation will print out the results from each generation, which includes the current generation number and the STRING from that generation that most closely matches the *target-phrase*. For example, the selected parent for an intermediate generation might look like 'ZETHINKS IT AS LIKW A WEABEL.' If the simulation produces a STRING that completely matches the *target-phrase*, then the simulation will stop and print out how many generations it took in the NetLogo Command Center.

## Initialization

Upon initialization, an initial STRING is created with a randomly generated list of characters with the following criteria: (1) the list of characters is the same length as the list of characters in the *target-phrase*, and (2) the only characters allowed are either a blank space or an uppercase English alphabet letter. The *target-phrase* determines the goal phrase upon which STRING populations are selected. The *mutation-rate* determines the rate at which each character in a string is imperfectly copied during reproduction. The higher the *mutation-rate*, the more likely that each character present in the offspring is not a match to the parent. The *number-of-offspring* determines how many offspring each selected parent will produce at every generation. The *with-selection?* boolean input parameter (true or false) determines whether selection occurs during reproduction. Without selection, Dawkins Weasel simulates a monkey typing on a keyboard, producing random phrases of characters. With selection, Dawkins Weasel simulates cumulative selection, and it is possible for the population of STRINGS to evolve to match the *target-phrase*.