

tEAch Manual

Introduction

tEAch is a Windows program that shows the inner workings of evolutionary algorithms on a simple problem – the counting 1s problem. The aim of the program is twofold:

1. To show all of the steps of an EA in an easy-to-use interface
2. To allow the user to vary the parameters of an EA and see the results

Tutorial

The tEAch program is very simple to operate. Options available to the user are shown by the buttons that are highlighted. Greyed-out buttons are not able to be clicked, for one of the following reasons:

1. The operation e.g. (Selection) has already been conducted.
2. Within an operation, other buttons must be pressed. For instance, in Selection, if the tournament has a size of 6, 6 individuals must be selected before the “Tournament Winner” button can be pressed.

Main Screen

The screenshot shows the tEAch Program interface with the following sections:

- Population:** A table with 10 individuals (Ind1-Ind10) and 10 variables (Var1-Var10) and a fitness value (Fit). The 'Generate Random Population' button is highlighted.
- Selection:** A section for selecting individuals. It includes a 'Tournament Size' input (set to 2), a 'Find Random Individual' button, a 'Random Individuals' list, a 'Find Tournament Winner' button, and a 'Tournament Winners' list (showing 2 winners).
- Parents:** A table showing the selected parents (Ind2 and Ind6) and their variables and fitness values. The 'Crossover' button is highlighted.
- Children:** A table showing the offspring (Ind2 and Ind6) and their variables and fitness values. The 'Mutation' button is highlighted.
- Replacement:** A section for replacing the worst individual in the population. It shows 'Current Worst: 2 Fitness: 5' and 'Current Child: 2 Fitness: 4'. The 'Replace Worst in Population' button is highlighted.
- Performance:** A section showing the current iteration (1), best fitness (7), and average fitness (5.3). It includes a 'Chart Progress' button.

Arrows indicate the flow of the evolutionary process: Population → Selection → Parents → Children → Replacement → Population.

The main screen, with all sections highlighted can be seen above (please note that only some options are shown when starting the software). The sections are as follows:

Population

Shows the individuals in the population (“Indx”), the variables for the problem (“Varx”) and the fitness (“Fit”). Fitness is calculated as the number of 1s in the individual. The button available in this section are as follows:

- **Generate at Random** - generates a new population with a new random seed

Selection

Uses a tournament selection to select individuals from the population. This section requires the user to specify a number of individuals in the tournament. The buttons in this section are as follows:

- **Find random individuals** – Uses the random number generator to select, at random, individuals from the population. Random individuals are shown in the box below.
- **Find tournament winner** – Conducts the tournament to find the individual with the best fitness. *This option is only available if sufficient random members have been found to conduct the tournament.*

This section must be completed twice to select 2 parents.

Parents

The parents section shows the two parents that have been selected.

Crossover

Uses a single point crossover to combine information from the two individuals. There are two buttons in this section:

- **Generate Random Locus** – Generates a random point(s) from which to cross-over the two solutions.
- **Apply Crossover** – Performs crossover on the individuals to form two children which are shown in the 'Children' box. The mutation section is also now available. *This option is only available if sufficient random loci have been generated to run the crossover.*

Children

The children section shows the created children. These children will be the product of crossover and mutation

Mutation

Uses single or multi-point mutation to randomly modify solutions. The number of mutation points is specified by the user. The buttons in this section are as follows:

- **Generate Random Locus** – Generates a position on the chromosome at which to mutate
- **Generate Random Value** – Generates a new random value to place at that position

Once these values have been generated, the mutation will appear in the box below (e.g.

Child 1: 7 >> 0 - which means that position (locus) 7 will be changed to a 0 in child 1. The value might already be 0, but the value is determined at random

- **Apply Mutation** – Once there are enough mutations (e.g. mutation number*2), the apply mutation button will be lit. Press this to apply the mutations to the children.

Replacement

Two new child solutions have been created. The Replacement section replaces the worst in the population with the children in order, starting with child 1, then child 2. If the fitness of the child is the same or better than the fitness of the worst in the population, then replacement occurs. If not, the solution is ignored.

- **Replace Worst in Population** – Once 2 replacements have been made, the GA operations sequence begins again with selection.

Other Sections

The average and best fitness of the population are recorded along with the iteration number and are displayed below the population. The **Chart Progress** button allows the user to see the history of the

best and average fitnesses over the run. **Save Chart** invokes the Save dialog and saves the chart as a bitmap file.