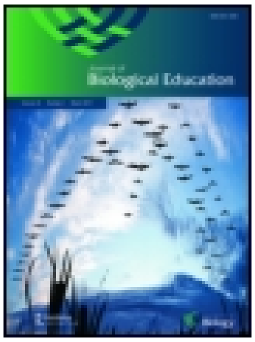


Routledge
Taylor & Francis Group

Journal of Biological Education

ISSN: 0021-9266 (Print) 2157-6009 (Online) Journal homepage: http://www.tandfonline.com/loi/rjbe20

EvoluZion: a computer simulator for teaching genetic and evolutionary concepts

Adolfo R. Zurita

To cite this article: Adolfo R. Zurita (2016): EvoluZion: a computer simulator for teaching genetic and evolutionary concepts, Journal of Biological Education, DOI: 10.1080/00219266.2016.1217907

To link to this article: http://dx.doi.org/10.1080/00219266.2016.1217907

Published online: 31 Aug 2016.

JOURNAL OF BIOLOGICAL EDUCATION, 2016
http://dx.doi.org/10.1080/00219266.2016.1217907

Routledge
Taylor & Francis Group

EvoluZion: a computer simulator for teaching genetic and evolutionary concepts

Adolfo R. Zurita

Área de Biología Molecular, Departamento de Bioquímica y Ciencias Biológicas, UNSL and Instituto Multidisciplinario de Investigaciones Biológicas de San Luis (IMIBIO-SL-CONICET), San Luis, Argentina

ABSTRACT
EvoluZion is a forward-in-time genetic simulator developed in Java and designed to perform real time simulations on the evolutionary history of virtual organisms. These model organisms harbour a set of 13 genes that codify an equal number of phenotypic features. These genes change randomly during replication, and mutant genes can have null, positive or negative effects on the organisms' fitness, allowing to model effects of both selection pressures and drift on gene evolution. There are two versions of this program: version 1.6.x_haploid; focused on macroevolutionary events and depicting prokaryote-like organisms, and version 2.3.x_diploid that simulate diploid, sexually reproducing organisms, and it is more adequate to teach micro-evolution as well as key genetic concepts such as Mendel's laws, epistasis, genetic linkage, genetic mapping among others. Different data sets can be collected periodically during running in order to perform further analyses. In addition, the complete genealogy of extant as well as extinct organisms can be recorded. EvoluZion is well suited for teaching evolutionary biology concepts to students of all levels in a pedagogic way. This is mainly due to three main program features: (i) its intuitive and simple graphical interface (ii) a visualisation similar to videogames (iii) flexible integration of a wide range of biological phenomena into a single simulation.

KEYWORDS
Evolution; genetics; software; simulator; educational

Introduction

Teaching genetics and evolution is not always an easy task in classrooms. In addition, the integrated analysis of biological concepts such as DNA replication, mutation, mitosis and meiosis is always desirable, and quite hard to present to students. These issues are usually studied separately in biology courses but not always well integrated by students.

Another difficulty is that the observing of an evolutionary process, or studying inheritance patterns by crossing species with different alleles requires long and costly experimental observations. Thus, students have no other option but to conform with seeing experimental results already described in textbooks or published in a scientific paper. All of this makes these disciplines somewhat abstract in practice.

Computer simulators are useful tools for bridging this pedagogic gap because it is possible to perform observations that otherwise would take years in minutes or the implementation of complex mathematical models. These give students the opportunity to experience these phenomena firsthand.

EvoluZion is a forward-in-time-simulator designed for educational purposes that creates and follows the evolutionary history of virtual living organisms in a controlled environment.

CONTACT Adolfo R. Zurita arzurita@unsl.edu.ar

The goal of this work is to introduce a computer simulator suitable for all level courses into the scholar community, by illustrating evolutionary events as well as classical genetic inheritance with a simple and attractive interface.

The program

EvoluZion is developed in JAVA language, with the additional use of the libGDX video game framework (https://libgdx.badlogicgames.com/). Therefore, it can run on all platforms harboring Java Runtime Environment. The program is available in two different versions: version 1.6.x_haploid (haploid), that handles haploid organisms, and version 2.3.x_diploid (diploid) for diploid, sexually reproducing organisms.

The haploid version was mainly designed to illustrate macroevolutionary phenomena, such as adaptation to environmental changes, the emergence of new features, as well as the competition for resources and the predator-prey balance. On the other hand – although it can be used in the same way as the haploid version – the diploid version is more suitable for teaching microevolutionary phenomena such as genetic drift, deviations from Hardy–Weinberg equilibrium and especially for the study of classical genetic principles e.g. Mendel's laws, genetic linkage, sex linkage and gene mapping.

The model and its rules

Each simulation takes place in a field of 1024 by 700 pixels, where the simulated organisms (about 18 by 18 pixels) coexist with a customisable number of red and green dots (5 pixels each) representing matter and energy sources, respectively (Figure 1). In order to survive, organisms must incorporate red and green spheres, obtaining resources for their metabolism and reproduction.

Since the simulation of actual living organisms could be a very complex -and computer - demanding task, the program uses a simplified approach to the phenomenon of life, by including the following properties on each simulated organism:

- Energy consumption (metabolism)
- Exchange of matter with the environment
- Reproduction with copies of its genetic material (including mutation)
- Environment recognition
- Adaptability
- Death

During a simulation, the energy dots move continuously from top to bottom of the screen and new dots are continuously generated at a regular rate. On the other hand, the matter dots can move from top to bottom or can stay still. In contrast to what happens with energy, there is reciprocal interchange of matter between the environment and the organism, but the total matter of the system remains constant.

Because the energy and the matter are limited resources, the organisms will have to compete for them in order to survive. Eventually, the organisms with better fitness – and luck – will survive and reproduce.

The organisms and its genome

Depending on the software version, each simulated organism carries its own genome with 12 or 13 genes that codify the same number of phenotypic features such as colour, height, width, speed, lifespan, and senses, among others.

A gene is defined as a DNA sequence – actually, as a real sequence of nucleotides – that codifies the computer variables (integers, floats, and booleans) that the program needs to create an organism. In addition, each gene has a promoter that includes a consensus TATA box sequence and an open

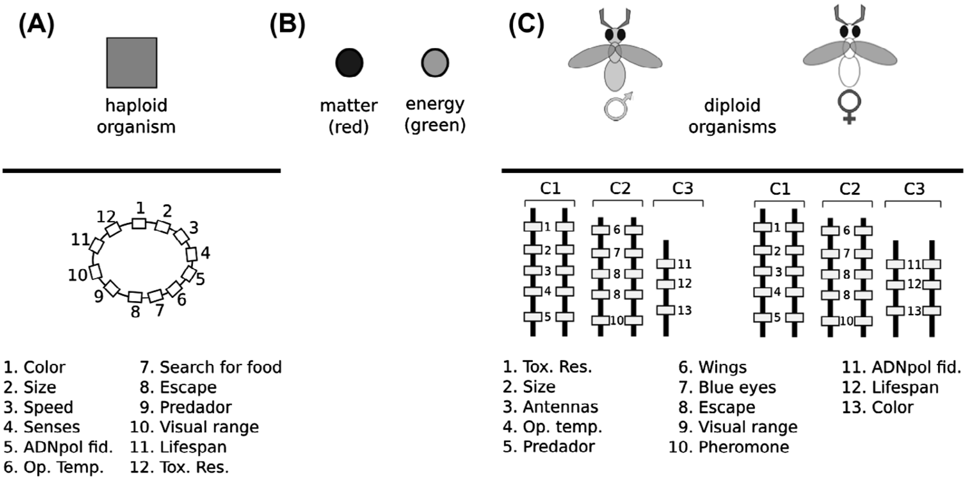


Figure 1. Graphical representation of the elements present in the simulation. (A) Organisms from the haploid version of the program are represented as a rectangle. Their genome is formed for 12 genes which codify the same number of phenotypic features; (B) The sources of energy and matter are represented as spheres and are present in both versions of the program; (C) Male and female organisms from the diploid version of the program. The organisms are represented as flies as a tribute to *Drosophila Melanogaster*. The female genome is formed with 13 pairs of genes, distributed in three pairs of chromosomes. Males are hemizygous for chromosome 3. The shape and size of chromosomes are only illustrative.
Notes: The phenotypic features codified by each gene are described at the bottom of the figure.

reading frame. Thereby, using the genetic code, each gene sequence is translated to an amino acid sequence, which determines a phenotypic feature.

The haploid version handles prokaryotic-like organisms with 12 genes, while the diploid version handles diploid sexually reproducing organisms, where 13 pairs of genes are distributed in three pairs of chromosomes: two somatic (chromosome 1 and 2) and one sexual (chromosome 3). Males have only one copy of chromosome 3 (Figure 1), simulating the X0 sex-determination system that is seen in some insects such as grasshoppers and crickets.

Life cycle

Each organism simulated by the program has a life cycle. This includes energy consumption, matter exchange, displacement, physical interaction, DNA replication, cell division, protein expression, detection of food and/or danger, aging and death. The diploid version adds sexual maturity, chromosomal crossover and gene recombination.

During DNA replication, mutations are probabilistically simulated including events of substitution, deletion, insertion, gene duplication, as well as horizontal gene transfer and gene loss. These mutations will be inherited by the next generation and will be eventually fixed or eliminated by selection pressure and/or genetic drift. Death can be caused by predators, starvation, aging and lethal mutations on essential genes.

Graphic interface and data collection

EvoluZion offers a very simple and intuitive graphical interface, where many parameters can be configured, such as time of simulation, the original number of organisms, energy–matter availability and temperature changes. In addition, some changes in the conditions can be introduced any time during simulation; for instance, the addition of a toxin or massive extinctions, leading to drastic population size reduction (Figure 2). It is also possible to configure the genes that can be susceptible to mutation and/or follow the evolution of specific genes.

During a simulation, some data are shown in real time, such as total number of organism, number of mutants, available matter, system temperature, average speed and the average size of the organisms.

Data collection can be performed at regular intervals and saved in specific files. These include comma-separated values (CSV) files with DNA and protein sequences (in FASTA format) of mutants from a specific timeframe, and with data for statistical analysis, such as population size, average size, average speed and phenotypic distribution.

The files in CSV format can be read using spreadsheets programs such as LibreOffice-Calc® or Microsoft-Excel® for further analysis.

Finally, selected organisms can be saved in a specific file to be used in future simulations with different conditions, if desired.

Performing a class

In order to facilitate the development of a class, two step-by-step tutorials were included (one for each version of the program), in which a set of experiments that the students have to perform in the computer are described.

In a typical class, the students have to run the desired version of the program in the computer and follow the provided instructions. At the end of each experiment, students – and teachers – will see additional instructions of how to get and interpret the results. Additionally, there is a small set of questions aimed at promoting discussion in the classroom. Teachers can feel free to use and modify these tutorials, to suit their particular needs.

Results

Next, I show the results of some experiments described in the tutorials, which can be performed in the classroom with both versions of the program.

Haploid version

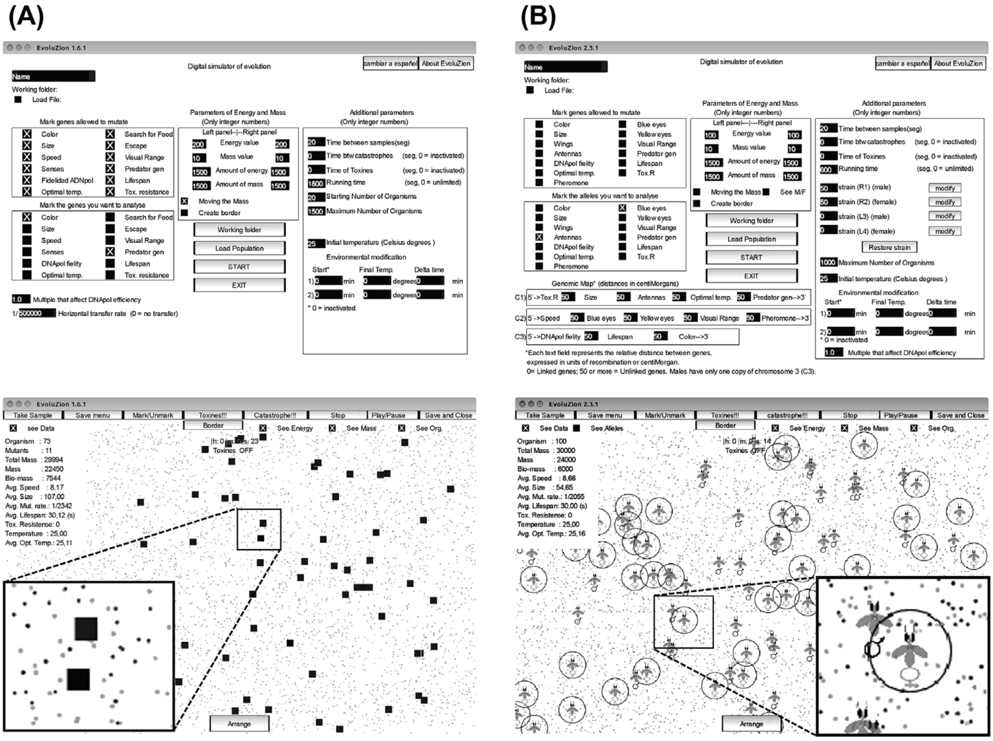
Experiment 1: Adaptation to environmental changes

As an example of gradual environmental changes, figure 3 shows how the organisms generated by the haploid version are capable of adapting to an environment where temperature increases over time. Figure 3 shows a 15-min experiment (starting with 50 identical organisms), for which the system temperature was set to stay at 25 °C (Figure 3(A)) or to be increased (from minute 5) from 25 °C to 35 °C at different rates (Figures 3(B)–(D)). In these experiments, natural selection eliminates organisms unable to live in lower temperatures, favouring mutants that are able to survive in higher temperatures. The average *growth-optimal temperature* (i.e. the temperature at which an organism feels comfortable) tends to get as close as possible to the system temperature. This experiment tells us how the speed of adaptation is limited because if the rate of temperature increase is too high, the existing mutation rate may not be high enough to generate a viable mutant in time, and extinction is inevitable.

Experiment 2: Adaptation to toxins (or antibiotics) and resource competition

EvoluZion can simulate what would happen if some mutants became resistant to certain toxic compounds (or antibiotics) present in the environment. In this experiment, I performed a simulation (starting with 50 organisms) in a toxin-free environment (see Figure 4 for details).

As shown in figure 4(A), toxin-resistant mutants appear later during simulation by spontaneous mutation of the suitable gene. By then, there are few resources available on the system, and in the absence of selection pressure, these mutants do not have higher and advantageous fitness than wild-type organisms. Consequently, they remain as a minor or marginal population. In contrast, if toxins



Name
Working folder:
■ Load File:

Mark genes allowed to mutate
☒ Color ☒ Search for Food
☒ Size ☒ Escape
☒ Speed ☒ Visual Range
☒ Senses ☒ Predator gen
☒ Fidelidad ADNpol ☒ Lifespan
☒ Optimal temp. ☒ Tox. resistance

Mark the genes you want to analyse
☒ Color ■ Search for Food
■ Size ■ Escape
■ Speed ■ Visual Range
■ Senses ☒ Predator gen
■ DNApol fielity ■ Lifespan
■ Optimal temp. ■ Tox. resistance

1.0 Multiple that affect DNApol efficiency
1/500000 Horizontal transfer rate (0 = no transfer)

Parameters of Energy and Mass
(Only integer numbers)
Left panel--|--Right panel
200 Energy value 200
10 Mass value 10
1500 Amount of energy 1500
1500 Amount of mass 1500
☒ Moving the Mass
■ Create border

Additional parameters
(Only integer numbers)
20 Time between samples(seg)
0 Time btw catastrophes (seg, 0 = inactivated)
0 Time of Toxines (seg, 0 = inactivated)
1800 Running time (seg, 0 = unlimited)
20 Starting Number of Organisms
1500 Maximum Number of Organisms
25 Initial temperature (Celsius degrees)
Environmental modification
Start* Final Temp. Delta time
1) 0 min 0 degrees 0 min
2) 0 min 0 degrees 0 min
* 0 = inactivated

☒ see Data
Organism : 73
Mutants : 11
Total Mass : 29994
Mass : 22450
Bio-mass : 7544
Avg. Speed : 8,17
Avg. Size : 107,00
Avg. Mut. rate.: 1/2342
Avg. Lifespan: 30,12 (s)
Tox. Resistense: 0
Temperature : 25,00
Avg. Opt. Temp.: 25,11
|h: 0 |m: 0 |s: 23
Toxines OFF
☒ See Energy ☒ See Mass ☒ See Org.

Name
Working folder:
■ Load File:

Mark genes allowed to mutate
■ Color ■ Blue eyes
■ Size ■ Yellow eyes
■ Wings ■ Visual Range
■ Antennas ■ Predator gen
■ DNApol fielity ■ Lifespan
■ Optimal temp. ■ Tox.R
■ Pheromone

Mark the alleles you want to analyse
■ Color ☒ Blue eyes
■ Size ■ Yellow eyes
■ Wings ■ Visual Range
☒ Antennas ■ Predator gen
■ DNApol fielity ■ Lifespan
■ Optimal temp. ■ Tox.R
■ Pheromone

Genomic Map* (distances in centiMorgans)
C1) 5`->Tox.R 50 Size 50 Antennas 50 Optimal temp. 50 Predator gen-->3`
C2) 5`->Speed 50 Blue eyes 50 Yellow eyes 50 Visual Range 50 Pheromone-->3`
C3) 5`->DNApol fielity 50 Lifespan 50 Color-->3`
*Each text field represents the relative distance between genes,
expressed in units of recombination or centiMorgan.
0= Linked genes; 50 or more = Unlinked genes. Males have only one copy of chromosome 3 (C3).

Parameters of Energy and Mass
(Only integer numbers)
Left panel---|---Right panel
100 Energy value 100
10 Mass value 10
1500 Amount of energy 1500
1500 Amount of mass 1500
■ Moving the Mass ■ See M/F
■ Create border

Additional parameters
(Only integer numbers)
20 Time between samples(seg)
0 Time btw catastrophes (seg, 0 = inactivated)
0 Time of Toxines (seg, 0 = inactivated)
600 Running time (seg, 0 = unlimited)
50 strain (R1) (male)
50 strain (R2) (female)
0 strain (L3) (male)
0 strain (L4) (female)
1000 Maximum Number of Organisms
25 Initial temperature (Celsius degrees)
Environmental modification
Start* Final Temp. Delta time
1) 0 min 0 degrees 0 min
2) 0 min 0 degrees 0 min
* 0 = inactivated
1.0 Multiple that affect DNApol efficiency

☒ See Data ■ See Alleles
Organism : 100
Total Mass : 30000
Mass : 24000
Bio-mass : 6000
Avg. Speed : 8,66
Avg. Size : 54,65
Avg. Mut. rate.: 1/2055
Avg. Lifespan: 30,00 (s)
Tox. Resistense: 0
Temperature : 25,00
Avg. Opt. Temp.: 25,16
|h: 0 |m: 0 |s: 14
Toxines OFF
☒ See Energy ☒ See Mass ☒ See Org.

Figure 2. Graphic Interface and simulation field*. (A) Haploid version main screen (top): Most of the main configurations are set from here. Simulation field (bottom). Here the organisms (rectangles) interact with the energy and matter (clear dots and dark dots, respectively); (B) Diploid version main screen (top): The configuration is slightly more complex than the haploid version. A representation of the three chromosome maps and the possibility to set distance between genes (left bottom panel) are included. The simulation field is shown at the bottom. *The colours in this paper were inverted from the original to improve the display of the printed version.

are added, wild type organisms die, and the toxin-resistant mutants quickly colonise the system (Figure 4(B)).

Experiment 3: Predator-prey balance

Among all the genes of each simulated organism, there is one (called *predator* gene) that changes its nutritional preferences. Mutants of this gene become predators that get the energy and matter from smaller non-predator organisms. Figure 5 shows the results of a simulation in which, at the beginning, predators are present in small amounts and the characteristic predator–prey equilibrium starts to emerge.

Experiment 4: Phylogenetic analysis (for advanced courses)

As mentioned above, each simulated organism carries its own genome harbouring genes that are susceptible to mutation after each cell division. In each simulation, the sequences of the mutated genes are collected and saved in a CSV file, which can be used in sequence alignment and/or the generation of phylogenetic trees. Figure 6 shows a phylogenetic tree performed with the data of experiment 1 (Figure 3(B)). In this case, we collected the sequences of all mutations for a specific gene, present at the end of the simulation. The phylogenetic analysis was performed using the multiple alignment program MAFFT 7 (Katoh and Standley 2013) with default options, and the phylogenetic tree was visualised with the program FigTree 1.4.2 tree.bio.ed.ac.uk/software/figtree/(see Figure 6 for details).

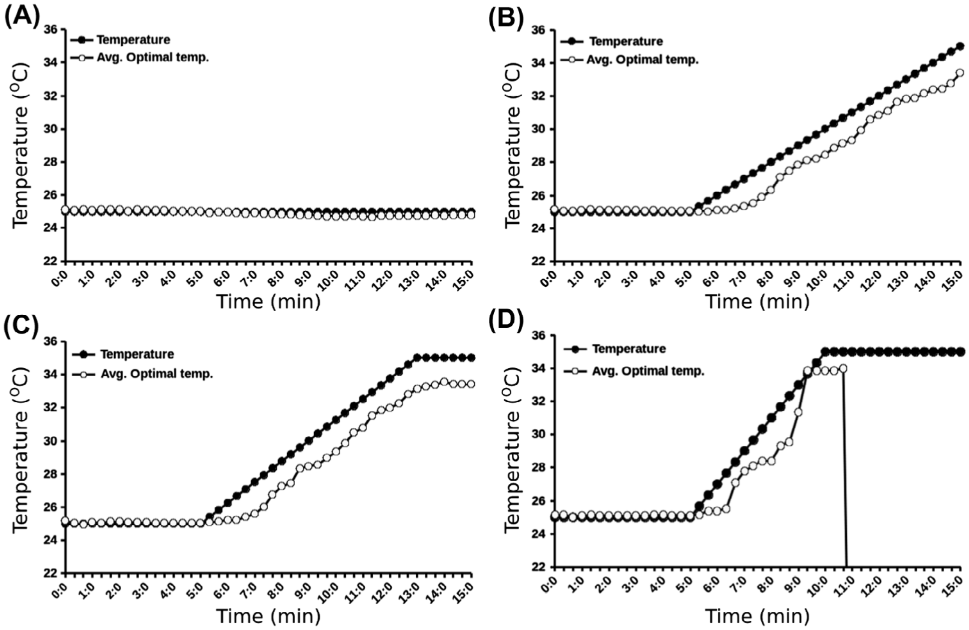


Figure 3. Adaptation to temperature changes. Four simulations of 15 min were performed with the haploid version as follows: (A) The system temperature (black circles) was kept at 25 °C . The average *optimal-growth temperature* (white circles) of all organisms remains around 25 °C; (B) From minute 5, the system temperature is increased from 25 to 35 °C during the next 10 minutes. Mutants that are able to live in higher temperatures are selected, and the average *optimal-growth temperature* increases with the system temperature; (C) The system temperature is increased to 35 °C during the next eight minutes; (D) The system temperature is increased to 35 °C during the next five minutes. The temperature increase rate was too high, the organisms could not adapt fast enough and became extinct.

Diploid version

Experiment 5: Mendel's laws

The simulated organisms of the diploid version are ideal for illustrating principles of Mendel's laws in genetic courses. Figure 7 shows an experiment starting with 50 males with yellow eyes and antennas, both determined by two dominant alleles, and an equal amount of females with white eyes and no antennas, both determined by two recessive alleles. The simulation is run selecting the genes of interest (yellow eyes and antennas) and banning gene mutations. This is because the interest of the procedure is on the distribution of existing alleles and not the spontaneous appearance of new ones.

At the end of the simulation, the phenotypes resulting from the F1 and F2 generations are the expected from theory. The deviations observed with respect to the theoretical values are the result of stochastic factors, including spatial location over time, age and matter availability.

Experiment 6: Persistence of lethal genes

An important concept in genetic and evolution courses is the unbalanced distribution of alleles differentially affecting fitness, which are partially responsible for the deviations from the Hardy–Weinberg equilibrium. One of the most extreme fitness effects driven by different alleles is the presence of lethal variants. As an example in our model, I compared a functional allele vs. a non-functional allele of the *toxin-resistance* gene. Thus, the non-functional allele would be lethal for homozygous organisms living in an environment with toxins.

The experiment illustrated in figure 8 shows the distribution of these two alleles along a 20-min simulation. In this experiment, 50 males resistant to toxins (dominant homozygous) are crossed with 50 non-resistant females (recessive homozygous). The simulation was run without toxins in the system until the appearance of the F1 generation (toxin-resistant heterozygous). Then, toxins were added to

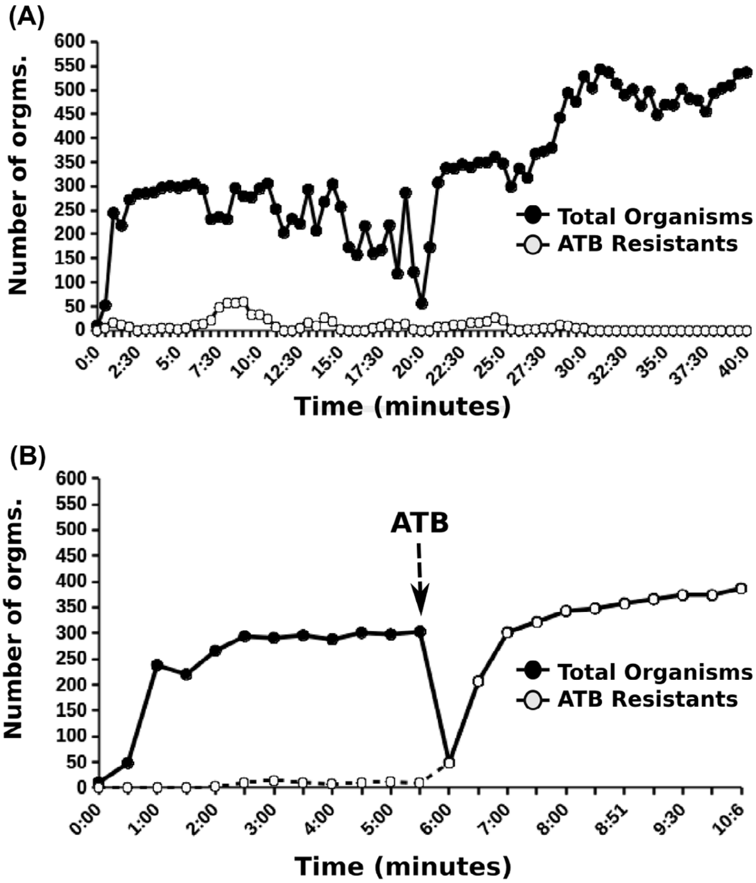


Figure 4. Adaptation to antibiotics. Simulations (haploid version) showing the number of total organisms (black circles) along time and the fraction of organisms that are resistant to an antibiotic (white circles), in two different experiments. (A) No antibiotic is added. ATB-resistant organisms remain as a minority during all the simulation; (B) Antibiotic is added at minute 5:30 of the simulation. ATB-resistant organisms quickly became the dominant species of the simulation.

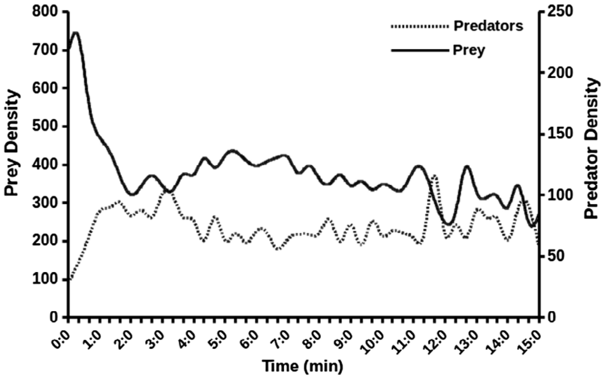


Figure 5. Predator/prey equilibrium. Results of a 15-min simulation (haploid version) showing the equilibrium between prey population (black line) and the predator population (dashed line). This simulation starts with a fixed number of predators and preys that came from a previous simulation. Saved simulations can be reloaded and run with different conditions as long as desired.

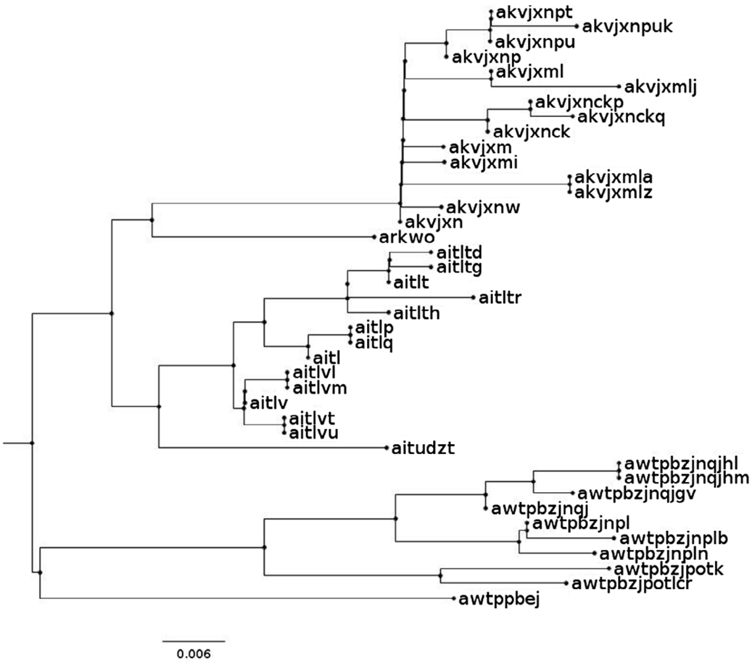


Figure 6. Phylogenetic analysis of the *optimal-growth temperature* gene of mutants collected at the end of simulation of figure 3(B). The name of each mutant was generated by the program. This analysis involved 62 nucleotide sequences saved as a FASTA format during the simulation.

the system and the simulation continued until it finished. This experiment shows two outcomes: (1) the quick change in the alleles distribution along the generations, with a clear advantage for the functional allele; and (2) the persistence of the lethal allele, which, despite its condition, remains stable at low frequencies in heterozygous organisms that act as healthy carriers.

Feedback

After using the program in a class of genetics by performing the experiments of the tutorial_v2.0 (included with the program), we conducted a poll with the following results: 100% of the students reported no problems running the program on their computers, that it was easy to use and provides clear information. Also, 100% of the students manifest that the program helped them to reinforce the topics of the class. With respect to the manuals and tutorials, 71% of the students answered that they were clear and understandable while the remaining 29% did not answer this question.

As is expected for any new specific software, some students reported the need of the professor's guidance at the beginning, in order to start to work and understand the output information.

Discussion

Many genetic simulators have been developed in order to explore the evolution of natural processes, including mutation, natural selection, migration and allelic distribution (Hoban, Bertorelle, and Gaggiotti 2012). These genetic simulators fall into two categories: backward-in-time-simulators (Buendia and Narasimhan 2006; Spencer and Coop 2004), used to infer an evolutionarily history based upon gene samples (not individuals), and forward-in-time-simulators (O'Fallon 2010; Coombs, Letcher, and Nislow 2010) that follow the development of individuals along multiple scenarios.

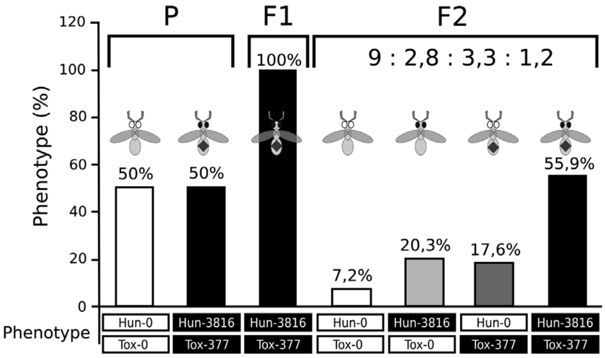


Figure 7. Gene recombination (diploid version). The result of a simulation in which 50 homozygous males for the dominant alleles Hum-3816 (dark eyes) and Tox-377 (grey abdominal diamond) was crossed with 50 homozygous females for the recessive alleles Hum-0 (white eyes) and Tox-0 (no grey abdominal diamond). All organisms of F1 generation are heterozygous with the dominant phenotype (dark eyes/grey diamond). At the end of the simulation, the resulting distribution of phenotypes in the F2 generation obeys Mendel's third law and it is close to 9:3:3:1.

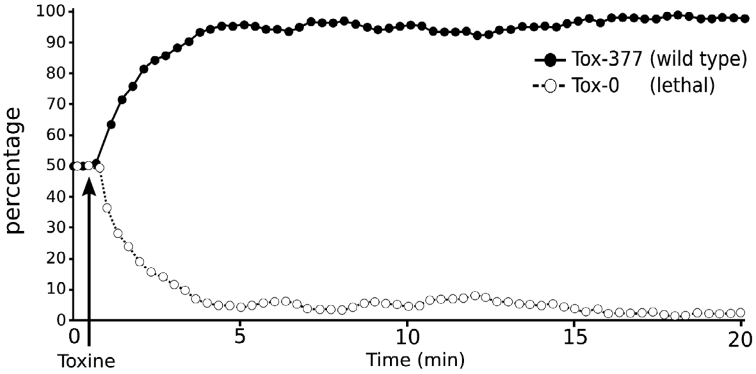


Figure 8. Persistence of lethal alleles. By using the diploid version of the program, 50 homozygous males for the dominant allele Tox-377 (resistant to toxin) were crossed with 50 homozygous females for the recessive allele Tox-0 (non-resistant to a toxin). At 30 s, a toxin is then added and the system becomes lethal to any Tox-0/Tox-0 homozygotes (The allele Tox-0 is lethal under this condition). As expected, after the addition of a toxin, the percentage of allele Tox-0 reduces quickly. Nevertheless, while the simulation continues, this lethal allele remains stable in small quantities without disappearing completely.

Some of these programs are mainly used for research goals, such as the prediction of population changes, the validation of statistical methods and checking the performance of multiple sequence alignment and phylogenetic inference methods (Hoban, Bertorelle, and Gaggiotti 2012), while other simulators are meant mainly for educational purposes (Jones and Laughlin 2010).

EvoluZion offers an energetically based – and biologically realistic – computer model that emulates many properties of living beings, such as dependence on matter and energy, DNA replication, protein translation, mutation and – additionally in the diploid version – meiosis, genetic recombination and chromosomal crossover. All these properties are integrated in real time and shown in a videogame-style output.

The set of experiments presented above is an example of the capabilities of the program. The first two experiments show how mutation and natural selection are responsible for the emergence of new species (or mutants) with increased fitness, which are eventually completely different from their predecessors. In these experiments, the environment variables such as temperature, energy and matter can be customised by students and each new experiment can be seen as completely different evolutionary history.

These experiments can also be useful as examples for illustrating what would happen with some species when changes in the ecosystem are too fast and the consequences for our health that would lead us to the abuse of antibiotics.

The third experiment shows how the simple addition of an organism with predatory behaviour exhibits a predator–prey balance. Despite the fact that my program was not originally designed to simulate this equilibrium, it is similar to those predicted by mathematical models (Taylor 1984; Krebs 1994) and specific computer simulators like the Lutterschmidt–Schaefer simulator (Lutterschmidt and Schaefer 1997), Biota from BioQuest (bioquest.org/) and Rabbits and Wolves from Shodor (www.shodor.org/), and whose result may be a proof of the robustness of the basic program model.

The fourth experiment shows how the DNA and protein sequences exported by the program can be used in advanced courses of molecular phylogeny, as material to perform multiple sequence alignments and phylogenetic trees. Additionally, because each simulated organism keeps a record of its own genealogy, we can use this information to assess the robustness of algorithms for multiple alignment methods, phylogenetic inference and ancestral sequence reconstruction.

Finally, the last two experiments show how the program can also be used for the study of micro-evolutionary processes as well as classical genetic concepts by banning mutations and observing the distribution of the existing alleles.

I believe that EvoluZion can be a valuable tool for teaching evolution and genetics to students of all levels, as it provides a simple and ludic interface,without sacrificing robustness. Finally, it is important to mention that an unlimited number of experiments can be carried out with both versions of the program. Several examples of these are present in the previously mentioned tutorials (see Performing a class), while others should be created by the teacher according to particular motivations and students' curiosity.

Availability

- EvoluZion can be downloaded from http://jevoluzion.jimdo.com/.
- The downloaded file includes the binary files ready to be used in all platforms, user manuals, tutorials and a description of the technical details of the program. At the moment, new versions of the program are incorporated on a regular basis.
- The web page also has some instructional videos that show an overall use of the program and a poll for a continuous evaluation of the program.
- EvoluZion is under GNU general public license; thus, it is free to use, distribute and modify. For the last option, the source code of the program is available on the web page.

Acknowledgements

I want to thank Maximiliano Ayub, Laura Mascotti, Walter Lapadula and Osvaldo Martin for their support and advise during the development of this project. I would also like to thank GAECI for their help and support.

Funding

This work was supported by the National Agency of Scientific and Technological Promotion (FONCyT) [grant number PICT 2011-0672]; and the National Scientific and Technical Research Council (CONICET) [grant number PIP-2011].

References

Buendia, Patricia, and Giri Narasimhan. 2006. "Serial NetEvolve: A Flexible Utility for Generating Serially-sampled Sequences along a Tree or Recombinant Network." *Bioinformatics* 22 (18): 2313–2314.

Coombs, Jason A., B. H., Letcher, and K. H. Nislow. 2010. "Pedagog: Software for Simulating Eco-evolutionary Population Dynamics." *Molecular Ecology Resources* 10 (3): 558–563.

Hoban, Sean, Giorgio Bertorelle, and Oscar E. Gaggiotti. 2012. "Computer Simulations: Tools for Population and Evolutionary Genetics." *Nature Reviews Genetics* 13 (2): 110–122.Nature Publishing Group.

Jones, Thomas C., and Thomas F. Laughlin. 2010. "PopGen Fishbowl: A Free Online Simulation Model of Microevolutionary Processes." *The American Biology Teacher* 72 (2): 100–103.

Katoh, K., and D. M. Standley. 2013. "MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability." *Molecular Biology and Evolution* 30 (4): 772–780.

Krebs, C. J. 1994. *Ecology: The Experimental Analysis of Distribution and Abundance*. 4 th ed. New York: Harper Collins Collegue Pub.

Lutterschmidt, William I., and Jacob F. Schaefer. 1997. "A Computer Simulotion for Demonstrating and Modelling Predatorprey Oscillations." *Journal of Biological Education* 31 (3): 221–227.

O'Fallon, Brendan. 2010. "TreesimJ: A Flexible, Forward Time Population Genetic Simulator." *Bioinformatics* 26 (17): 2200–2201.

Spencer, C. C. A., and Graham Coop. 2004. "SelSim: A Program to Simulate Population Genetic Data with Natural Selection and Recombination." *Bioinformatics* 20 (18): 3673–3675.

Taylor, Robert J. 1984. *Predation*. New York: Chapman and Hall.