An Interactive Educational Platform for Introducing Genetic Algorithm Concepts

Gessyca Moreira[1], **Ângelo Magno de Jesus**[1], **Suelen Mapa de Paula**[1]

[1]Department of Information Systems – Federal Institute of Minas Gerais – Ouro Branco – MG – Brazil

0027559@academico.ifmg.edu.br, {angelo.jesus,suelen.mapa}@ifmg.edu.br

Abstract. Introduction: *Teaching Artificial Intelligence (AI) concepts is challenging due to their abstract and interdisciplinary nature.* ***Objective:*** *This work presents an interactive educational platform to support the learning of Genetic Algorithms (GAs), making their processes tangible through simulations and visualizations.* ***Methodology:*** *The platform was developed using web technologies and evaluated with undergraduate students through the MEEGA+ model [Petri et al. 2019].* ***Expected Results:*** *Results indicate that 94.4% of participants considered the tool intuitive and suitable for educational use, highlighting its strong pedagogical potential.*
Keywords *Human-Computer Interaction, Genetic Algorithms, Educational Technology, Interactive Visualization, Simulation-Based Learning.*

1. Introduction

Since *Artificial Intelligence* (AI) is now part of everyday life, teaching AI to students from high school to college can bring several benefits. Furthermore, AI concepts are part of the mental toolkit of *Computational Thinking* (CT) — a problem-solving approach important for everyone [Wing 2006]. However, educators may face challenges in teaching AI techniques since these involve the interdisciplinary use of concepts from *Mathematics*, *Biology*, and *Computer Science* (CS). In this sense, developing educational objects focused on rich user/student interaction can be a promising strategy to address this problem, [Zeman e Holeček 2022] [Clematis et al. 2006]. Therefore, considering human factors in learning systems is essential, as educational experiences can be enhanced by contemporary technology [Crearie 2013] [Hayashi e Baranauskas 2015].

Inspired by natural computation principles, *Genetic Algorithms* (GAs) address search and optimization issues. These algorithms, which mimic biological evolution, search for optimal solutions through selection, crossover, and mutation to refine results over generations [Coppin 2013] [Forrest 1996]. Despite their capabilities, GAs often present a steep learning curve for beginners, especially those without prior exposure to CS or AI. A key difficulty lies in the abstract nature of these processes, which are neither readily observable nor intuitive. Compounding this is the limited availability of user-friendly educational tools for effective academic learning. Although interactive tools exist (Chiao 2021, Wagner et al. 2023, Eck 2020, on Daniel Shiffman's Smart Rockets 2022, and Wilensky 1997), most are not designed for pedagogical purposes and require prior technical knowledge, limiting their classroom effectiveness.

To bridge this gap, this work presents an interactive educational platform[1] that facilitates GA learning through accessible, hands-on experiences. It features a configurable simulator that evolves a fish population toward a target color, and a step-by-step mode that visually demonstrates each GA phase. Usability was evaluated with 18 undergraduate students using the *MEEGA+* model [Petri et al. 2019], with 94.4% finding the tool intuitive and suitable for educational use.

2. The Designed Educational Platform

This section presents an overview of the educational platform designed to support GA learning through affective and interactive design. It was developed using HTML, CSS, and JavaScript after evaluating existing tools and proposing didactic improvements to ensure accessibility and compatibility.

2.1. Design Elements

Elements of *Design for Affectivity* were considered [Hayashi e Baranauskas 2015]: understanding the student's context, providing an abstraction of data structures and algorithms through phenomena observable in daily life; using diverse media such as graphics, texts, and animations; enabling social interaction and teamwork, allowing students to compare and discuss results based on their environment customizations.

2.2. Simulation tool

The platform offers a visual and interactive interface to make GAs more accessible. As shown in Figure 1, users observe fish populations evolving over generations, with configurable parameters via a modal dialog. The first five generations run slowly to highlight selection, where unselected fish are "caught" by a bird, illustrating natural selection. After that, the simulation accelerates automatically.

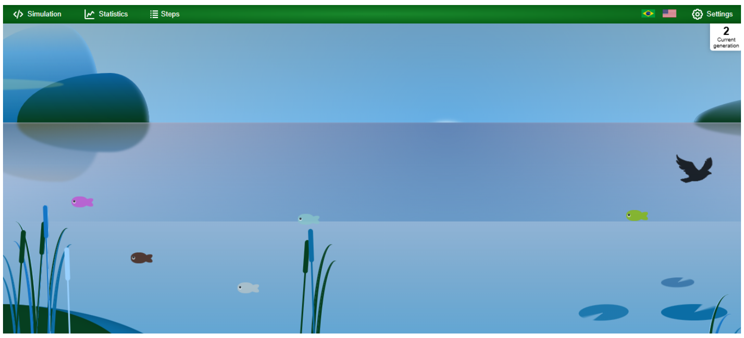

Figure 1. Simulation interface: blue lake with sky and mountain shadows in the background, five colorful fish spread horizontally, aquatic plants on the left, floating leaves on the right, and a black bird on the water; green top bar with navigation options and display of the current generation.

The platform visually represents key GA components. Population visualization uses fish of varying colors to represent individuals and their genetic traits. Core genetic operators: selection, crossover, and mutation, are implemented and customizable via the settings modal (Figure 2). Users can track the evolutionary process by monitoring fitness statistics and viewing a graph of the best fitness history (Figure 2).

[1]Available at: `https://github.com/Geessyca/genetic-algorithm-edu-platform` and `https://agif1.vercel.app/`.

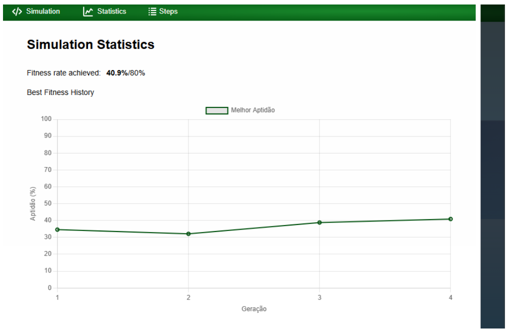

Figure 2. Statistics and settings: on the left, line chart titled Simulation Statistics with label “Best Fitness“ and value below; on the right, settings window with sliders for precision and mutation rates, population size field, selection method options, stopping criteria, and green “Apply Settings“ button.

The platform emphasizes simplicity and education, avoiding jargon and using clear visual metaphors. Configuration options such as mutation rate and selection method are easily accessible through an organized modal. It effectively bridges theory and practice, serving as both a demonstration and educational tool.

2.3. Steps tool

The steps section offers a guided walkthrough of the GA process, ideal for educational use. As shown in Figure 3, the interface breaks the algorithm into phases users can explore sequentially.

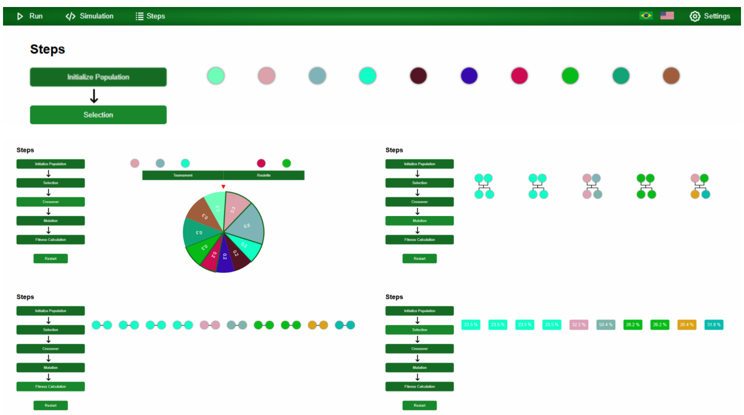

Figure 3. Multi-step algorithm simulation: green top bar with navigation and title “Steps“; vertical step buttons on the left with rows of colored circles representing individuals. Four sections: top left with step buttons, row of circles, and roulette (pie chart); top right with crossover clusters of circles; bottom left with row of circles for mutation; bottom right with green rectangles and percentages for fitness distribution.

Each button corresponds to a fundamental GA step. The initialization function randomly generates individuals, shown as colored circles, to represent genetic diversity. The selection function highlights individuals chosen by tournament or roulette wheel methods. In the crossover step, selected individuals are paired, and their genetic material combined, updating the population. The mutation function introduces random variations, altering colors. Fitness calculation evaluates individuals and identifies the best performers. Finally, the restart button resets the simulation.

This mirrors the classic GA cycle, allowing users to observe each phase. Real-time visualization improves understanding of the algorithm's behavior.

3. Educational Potential

The platform was developed with a strong educational focus to support the teaching of GAs in introductory courses on AI or *Evolutionary Computation*. It bridges theory and practice by offering hands-on interaction with key GAs components. Visualizing each algorithm step helps students grasp concepts like population evolution and natural selection.

It supports many teaching styles and adapts to different settings like high school, undergraduate, and continuing education courses. Running smoothly on any browser without needing extra software, it allows instructors to demonstrate evolutionary dynamics live during lectures, showing how changing parameters affects the process. Students can work alone or in groups to try out different settings and discuss the results together. The tool is also useful for lab sessions or assignments, helping students explore and think more deeply about how GAs work.

A usability study with 18 undergraduate students familiar with GAs assessed the platform's potential by administering five questions adapted from Petri et al. 2019. The results were positive: 94.4% considered the tool useful and appropriate, 83.3% rated the design as excellent, and 88.9% found the interface readable, demonstrating its pedagogical potential for complex computational topics.

4. Ethical Considerations

The study followed Brazilian standards for research with human subjects. Participation was voluntary, with informed consent, and data were collected anonymously and in aggregate form, ensuring confidentiality. As the research involved only a tool evaluation, no risks to participants were identified, and results were disseminated while fully protecting identities.

5. Conclusion

The platform effectively addresses a central challenge in teaching GAs: the difficulty of presenting abstract, dynamic processes tangibly and intuitively. Combining interactive simulations with step-by-step visualizations makes core GAs concepts more accessible to learners from diverse backgrounds. With configurable parameters, real-time feedback, and visual metaphors, the platform reinforces theoretical knowledge and encourages experimentation and discovery-based learning. Its design supports various educational settings, from in-class demonstrations to hands-on assignments, making it a versatile tool for instructors and students. A usability study conducted with undergraduate students who had prior exposure to GAs showed that 94.4% of participants rated the tool as both useful and appropriate for educational purposes, further supporting its potential as an effective learning resource. Future enhancements include the integration of additional evolutionary mechanisms, compatibility with learning management systems, and adaptive difficulty levels to personalize the learning experience further. Ultimately, the platform represents a significant step toward more inclusive and effective teaching of GAs principles.

References

Chiao, A. (2021). Genetic algorithm visualization (tsp solver). `https://abelchiao.github.io/genetic-algorithm-visualization/`. Accessed: 2025-05-19.

Clematis, A., Forcheri, P., e Quarati, A. (2006). Interacting with Learning Objects in a Distributed Environment . In *16th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008)*, pages 322–329, Los Alamitos, CA, USA. IEEE Computer Society.

Coppin, B. (2013). *Artificial Intelligence (in Portuguese)*. LTC.

Crearie, L. (2013). Human computer interaction (hci) factors in technology enhanced learning. In *Proceedings of the International Conference on Information Communication Technologies in Education*, pages 99–108.

Eck, D. (2020). Genetic algorithms demo. `https://math.hws.edu/eck/js/genetic-algorithm/GA.html`. Accessed: 2025-05-19.

Forrest, S. (1996). Genetic algorithms. *ACM computing surveys (CSUR)*, 28(1):77–80.

Hayashi, E. C. e Baranauskas, M. C. C. (2015). Designing for affectibility: principles and guidelines. In *HCI International 2015-Posters' Extended Abstracts: International Conference, HCI International 2015, Los Angeles, CA, USA, August 2–7, 2015. Proceedings, Part I*, pages 25–31. Springer.

on Daniel Shiffman's Smart Rockets, B. (2022). Smart rockets - genetic algorithm demo. `https://genetic-smart-rockets.vercel.app/`. Accessed: 2025-05-19.

Petri, G., von Wangenheim, C., e Borgatto, A. (2019). Meega+: a model for the evaluation of educational games for computing education. *Brazilian Journal of Computers in Education (Revista Brasileira de Informática na Educação-RBIE)*, 27(3):52–81.

Wagner, M., Kohlros, E., Quantmeyer, G., e Kötzing, T. (2023). Elea – build your own evolutionary algorithm in your browser.

Wilensky, U. (1997). Simple genetic algorithm (netlogo model). `https://ccl.northwestern.edu/netlogo/models/SimpleGeneticAlgorithm`. Center for Connected Learning and Computer-Based Modeling, Accessed: 2025-05-19.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.

Zeman, T. e Holeček, J. (2022). Effective creation of interactive educational objects for on-line teaching. In *2022 31st Annual Conference of the European Association for Education in Electrical and Information Engineering (EAEEIE)*, pages 1–5.