Implementation of Simulators for Teaching Tasks Using GPT

Godoy Pablo Daniel[1,2,3,4], Marianetti Osvaldo[1,4], Fontana Daniel[1,4], and Manganelli Silvina[1,2]

[1] Facultad de Ingeniería, Universidad Nacional de Cuyo
[2] Facultad de Ciencias Exactas y Naturales, Universidad Nacional de Cuyo
[3] ITIC, Universidad Nacional de Cuyo
[4] Facultad de Ingeniería, Universidad de Mendoza
pablo.godoy@ingenieria.uncuyo.edu.ar

Abstract. The implementation process of a simulator using a GPT is presented. This simulator is a wave simulator for use in teaching tasks of the physics subject in a grade university course. The selected GPT is ChatGPT. Much of the simulator was written using Javascript, HTML, and CSS, but the code was generated by the GPT. The requests made to the GPT to generate the code are presented. This mechanism is proposed as a higher level of programming, in which the programmer writes instructions in a language closer to human language than other programming languages existing until now. An advantage is the reduction of development time to around half compared to traditional programming techniques. However, deep programming knowledge in the mentioned languages is necessary to handle unexpected responses. A disadvantage is that if the code size grows, the GPT begins to fail. As a result, the projects have to be divided into subprojects and integrated when every part works adequately in a separate way. On the other hand, as a second contribution of this paper, the objective of the simulator is described: to assist in teaching tasks. The context in which it will be employed, expected results and partial results achieved with other simulators are presented.

Keywords: Simulators for Teaching Tasks· Wave Simulator· Code Generation through GPT.

1 Introduction

Several remote, relocatable, and virtual laboratories intended for teaching tasks have been developed. A remote laboratory allows performing experiments on real equipment in a remote way through the Internet. A relocatable laboratory, also called a nomadic laboratory, can be moved to the classroom to perform hands-on experiences. A virtual laboratory, also called a simulator, is completely built through software [1]. These tools have been utilized in the Universidad Nacional de Cuyo, Instituto Tecnológico Universitario (ITU), and Universidad de Mendoza for undergraduate, graduate, and postgraduate courses.

Some comparative analyses between courses that utilized these tools and those that did not show significant improvements in the grades obtained by students. Additionally, anonymous surveys filled out by students indicate that they feel comfortable using this type of tools [2–5]. Other authors have also demonstrated the advantages of virtual labs in enhancing student learning. Veza et. al. [6] show that virtual labs are suitable tools to teach engineering concepts and highlight the importance of accessibility and usability. Another benefits of virtual laboratories is that they allow students to practice and experiment without time or space restrictions [7]. Kapilan et. al. [8] explore how virtual labs have helped maintain educational continuity during Covid-19 pandemic by providing students with hands-on learning experiences remotely. Furthermore, through surveys the authors note that virtual laboratories showed good acceptance and satisfaction by faculty members and students. This result suggests that virtual laboratories can be an effective solution in times of crisis.

However, virtual labs have some disadvantages. Santos et al. [9] conducted a meta-analysis to assess the effectiveness of virtual labs. Santos et al. indicate that while virtual labs are effective tools, they do have limitations, such as a lack of interactivity and limited sensory experiences. In addition, students could have technical or access difficulties due to limitations in technological infrastructure [7].

Numerous authors have explored the application of generative artificial intelligence (AI), large language models, and other AI tools in educational contexts, and specifically ChatGPT [10–13]. However, according to Liang et al. [12], there is a scarcity of research focusing on addressing challenges within scientific disciplines, such as physics.

There are several websites with simulations for teaching Physics. One of the most important are the Phet Interactive Simulations website from the University of Colorado [14]. However, despite the availability of resources, there are some topics lacking virtual laboratories. For example, during the teaching tasks performed by the authors of this article, they encountered the needs for virtual laboratories covering specific areas such as: i) TCP protocol sliding windows, ii) Doppler effect, iii) particles position, velocity and aceleration in a mechanical wave, iv) alternating current phasorial diagram. For these reason, virtual laboratories of the first three topics mentioned were developed by the authors of this article. The first two were constructed using conventional web development tools, including Javascript, HTML and CSS languages. In contrast, the latter simulation was developed using chatGPT to automate a significant portion of the code writing. This paper focuses on the development and implementation of the latter virtual laboratory virtual laboratory mentioned: a mechanical wave virtual laboratory.

2 Virtual Laboratories Description

2.1 TCP Protocol Sliding Windows Virtual Laboratory

Figure 1 shows the TCP protocol sliding windows virtual laboratory user interface. The figure shows the following packages:

- 5 packages correctly acknowledged by the receiver, in green color and numbered from 8 to 12.
- 1 package resent by the transmitter, in blue color with a R letter inside.
- 4 packages sent by the transmitter in first instance, showed in blue color. The timmers of these packages are presented in red color.
- 1 ackowledge package in green color, contains a window advisory inside it, with the w=5 label. The number 5 is the window advisory.

The receiver and transmitter windows are depicted with brown rectangles. The virtual laboratory has several buttons and controls intended for controlling the simulation.

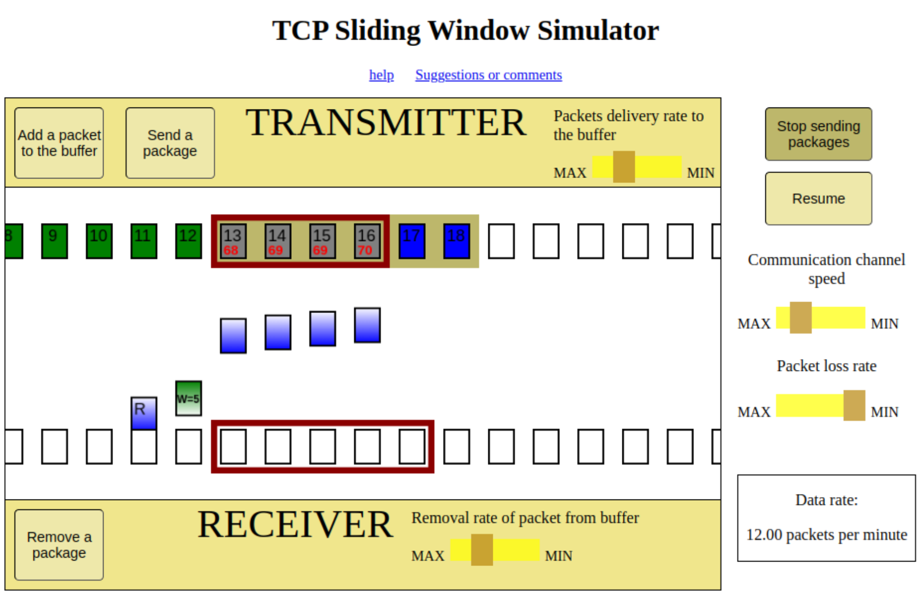

Fig. 1. TCP protocol sliding windows

2.2 Doppler effect virtual laboratory

Figure 2 shows the Doppler effect virtual laboratory user interface. The figure shows the sound wave emitter in blue color. The waves are reflected by the receiver. The reflected waves and the receiver are shown in green color. The reflected waves have not yet reached to the emitter, therefore, the frequency that return to the emitter are labeled with "no data". Several controls are shown. These controls are intended to control simulation parameters like emitter speed, receiver speed, wind speed and emitter frequency. In addition, some buttons have the function of start, stop and restart the simulation, and hide or show the reflected waves.

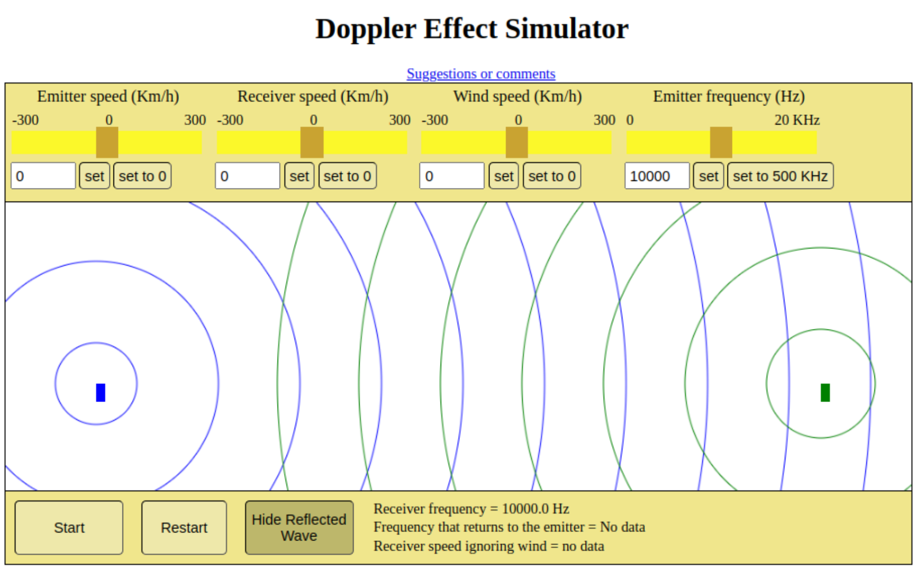

Fig. 2. TCP protocol sliding windows

2.3 Waves Virtual Laboratory

Figures 3 and 4 show the waves virtual laboratory. Figure 3 shows a wave traveling in the +x direction, and figure 4 shows a wave traveling in the -x direction. The virtual laboratory shows:

– Position function in blue color and continuous line.
– Velocity function in green color and dashed line.
– Acceleration function in red color and dashed line.

A black point shows the position of a single particle. The user can select this particle clicking on the graph area or writing the coordinates in the boxes shown on the right superior corner. Over the point, velocity and acceleration vectors are shown. A transverse wave has the same form of the position graph, but it is different for a longitudinal wave. For this reason, on the bottom panel an example of longitudinal wave according to the graph is shown. In addition, the pressure wave is shown. The right panel displays functions of position, velocity and acceleration along with parameter values of the waves. The virtual laboratory has controls to select the wave direction and set the frequency and wavelength. Also, it has controls to:

- Run the simulation in continuous way.
- Stop or restart the simulation.
- Advance the simulation step by step.

This virtual laboratory is hosted in https://simuladoresunc.000webhostapp.com/ondas/index.php. This url could change in the future.

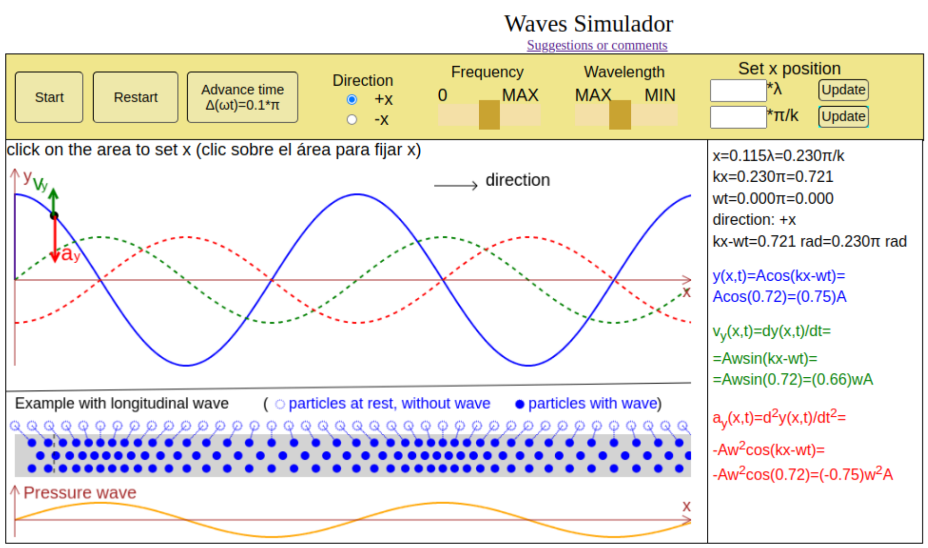

Fig. 3. Wave simulator showing a wave traveling in the +x direction

3 Implementation using ChatGPT

The first and second virtual laboratories presented above were implemented through common web development tools, including HTML, Javascript, CSS and PHP languajes, the Canvas [15] tool of Javascript to make graphs, as well as a popular online integrated development environment (IDE). The third virtual laboratory depicted in this paper, the mechanical waves virtual laboratory, was implemented using a generative AI, específically chatGPT. The purpose of this method was to corroborate if it is possible to reduce the development time of a website using generative AI tools. The method consisted of giving instruction to

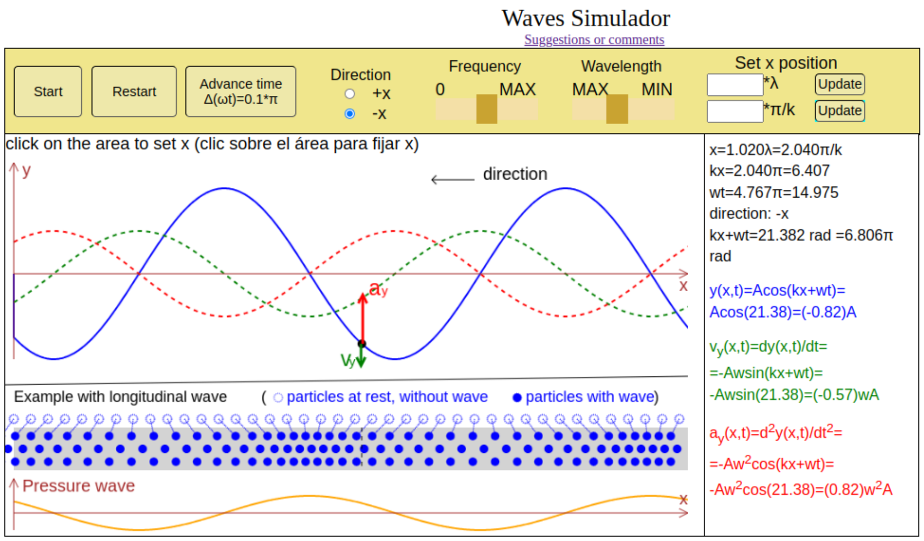

Fig. 4. Wave simulator showing a wave traveling in the -x direction

chatGPT and obtaining the code from chatGPT. Most part of the code was generated by chatGPT. The only part of the code that was not generated througt chatGPT was the control panel, as the same template as the other virtual laboratories was used. The instruction given to chatGPT are list below.

3.1 Instructions to Generate the Main Code

Instructions executed without errors by ChatGPT With the following instructions, ChatGPT generated code without errors.

1. Javascript code that graphs the wave function y(x,t)=Acos(kx-wt), graphing "y" on the vertical axis and "x" on the horizontal axis, assuming "t" growing continuously from 0. Use only Canvas and Javascript.
2. Use lineTo function instead of fillRect.
3. Draw the "x" and "y" axes, draw the axis arrows and the "x" and "y" labels. Axes, arrows and labels in brown color.
4. For the waveform, use amplitude = 100, draw the function in black.
5. Change the code to show only two wavelengths.
6. Change the code to show only two wavelengths that occupy the entire canvas drawing area.
7. Add a circle over the wave at coordinate x=0. Circle radius=5. Circle color: black.
8. Add a circle over the wave at coordinate x=400. Circle radius=5. Circle color: black.
9. Delete the first circle.

10. Add buttons that allow user to pause and resume the animation. Place the buttons on top of the canvas.
11. Add an input of type range that allows changing the angular frequency of the wave. Use minimum value equal to current value divided by 10 and maximum value equal to current value. Place this input on the canvas.
12. Change the position of the black circle on the waveform, so that the x coordinate is taken based on the part of the drawing area where the user left clicks the mouse.
13. Superimpose the following waveform: y=50*sin(kx-wt). Draw it in green with a dotted line.
14. Superimpose the following waveform: y=-50*cos(kx-wt). Draw it in red with a dotted line.
15. Add a button that allows user to advance it one step when the animation is paused, increasing the time variable one step.

Instructions in which ChatGPT Made Mistakes and Failures From this point, chatGPT began to fail at every step. Code generation was interrupted several times and had to be resumed. Consequently, some sentences had to be rewritten, other sentences had to be reformulated multiple times until the desired result was achieved. The final sentences are shown below.

16. Superimpose the following waveform: y=50*sin(kx-wt). Draw it in green with a dotted line.
17. Superimpose the following waveform: y=-50*cos(kx-wt). Draw it in red with a dotted line.
18. On the black circle of the black wave, draw a vector whose origin is the center of the black circle. The other end of the vector must have the same x coordinate, and its y coordinate must be yPos+50*sin(kx-wt), with yPos being the y coordinate of the black circle. Do not delete any of the above.
19. Go back to the code before adding the vector.
20. On the code presented below, without deleting any element, superimpose two vectors. The first vector with the following characteristics: its origin must be the center of the black circle. The other end of the vector must have the same x coordinate, and its y coordinate must be the y coordinate of the black circle - 100*sin(kx-wt). The second end of the vector must end in an arrow. The width of the vector line must be 3. Its color must be green. The second vector with the following characteristics: its origin must be the center of the black circle. The other end of the vector must have the same x coordinate, and its y coordinate must be the y coordinate of the black circle + 100*cos(kx-wt). The second end of the vector must end in an arrow. The width of the vector line must be 3. Its color must be red. The rest of the drawing should have line width 1.
21. Change the minimum value of the input range "angular frequency" to 0. Set the initial value to 0.

3.2 Instructions to Generate the Bottom Panel

The instructions given to chatGPT to graph the bottom panel are presented below:

1. Javascript code using canvas, where a horizontal tube of length 800 px and height 60 px is drawn, inside which some molecules are drawn, similar to a Kund tube. The horizontal distance of each molecule with respect to the equilibrium position must be given by a function y=f(x) to be defined by the user, where x is the x coordinate of each molecule.

From this point on the GPT began to fail at every step. Code generation was stopped several times and had to be resumed. Therefore, the last statement was implemented in a different file, because there were too many interruptions and unwanted results in the code generation.

4 Conclusions and Future works

The instructions list given to chatGPT includes specific instructions like to i) use Javascript, ii) use Canvas, iii) use specific Canvas and HTML instructions like lineTo instead of fillRect, iv) add an input of type range. Therefore, deep knowledge of programming is needed to employ this method.
In addition, the instructions include specific concepts of physical science, like i) number of wavelength to show, ii) Kund tube, iii) velocity and acceleration of a particle and velocity of the wave, iv) wave functions. Therefore, deep knowledge of physical science was needed in order to develop this virtual laboratory.
Future works include to build other virtual laboratories, to investigate the needs not covered by the current virtual laboratories, to use on classes with students the virtual laboratories presented in this article and to gather usage statistics and student opinions.
The three virtual laboratories presented in this paper will be used for teaching tasks during the first half of the year 2024. The mechanical waves and the Doppler effect virtual laboratories will be used in the Physic 2 subject, and the slidding windows virtual laboratory will be used in the Computer Networks subject. The first subject is taught at the Faculty of Exact and Natural Sciences, and the second at the Faculty of Engineering, both of the Universidad Nacional de Cuyo. The authors of this article are teachers on both subject. Therefore, the authors hope to rely on these usage statistics and student opinions when the first semester of classes ends. These expected results will allow to write other articles with the analyses of the same.

Acknowledgements The authors acknowledge the funding received from the "Secretaría de Investigación, Internacionales y Posgrado (SIIP), Universidad Nacional de Cuyo", through the projects "SIIP B008: Desarrollo y aplicación de laboratorios virtuales destinados a educación de grado y pregrado" y "SIIP B036-T1: La seguridad de los dispositivos utilizados en IoT mediante la utilización de Blockchain, NFT y/o VPN".

References

1. Olof Rensfelt, Frederik Hermans, Lars Åke Larzon, and Per Gunningberg. Senseiuu: a relocatable sensor network testbed. In *Proceedings of the fifth ACM international workshop on Wireless network testbeds, experimental evaluation and characterization (WiNTECH '10)*, pages 63–70, New York, NY, USA, 2010. Association for Computing Machinery.
2. P. Godoy, O. Marianetti, D. Fontana, and S. Manganelli. Desarrollo y aplicación de laboratorios virtuales destinados a educación de grado y pregrado. In *XXV Workshop de Investigadores en Ciencias de la Computación (WICC)*, pages 826–835, Buenos Aires, 2023. RedUNCI.
3. Pablo Godoy, Ricardo Cayssials, and Carlos García Garino. Laboratorio remoto para la formación de usuarios basado en el cloud. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, (18):p. 7–18, dic. 2016.
4. P. D. Godoy, R. Cayssials, and C. Garcia Garino. A WSN testbed for teaching purposes. *IEEE Latin America Transactions*, 14(7):3351–3357, July 2016.
5. P.D. Godoy, C.G.G. Garino, and R.L. Cayssials. A nomadic testbed for teaching computer architecture. In P. Pesado and C. Aciti, editors, *Computer Science – CACIC 2018*, volume 995 of *Communications in Computer and Information Science*, pages 201–216. Springer, Cham, 2019.
6. Ibham Veza, Ahmed Sule, Nicky Putra, Muhammad Idris, Ihwan Ghazali, Irianto Irianto, Gipin Mosliano, and Arasmatusy. Virtual laboratory for engineering education: Review of virtual laboratory for students learning. *Engineering Science Letter*, 1:41–46, 2022.
7. S.M. Reeves and K.J. Crippen. Virtual laboratories in undergraduate science and engineering courses: a systematic review, 2009–2019. *Journal of Science Education and Technology*, 30:16–30, 2021.
8. N. Kapilan, P. Vidhya, and X.-Z. Gao. Virtual laboratory: A boon to the mechanical engineering education during covid-19 pandemic. *Higher Education for the Future*, 8(1):31–46, 2021.
9. Marc Lancer Santos and Maricar Prudente. Effectiveness of virtual laboratories in science education: A meta-analysis. *International Journal of Information and Education Technology*, 12:150–156, 2022.
10. OpenAI. Chatgpt website, 2024. Retrieved in 2024.
11. Yi Wu. Integrating generative AI in education: How Chatgpt brings challenges for future learning and teaching. *Journal of Advanced Research in Education*, 2:6–10, 2023.
12. Y. Liang, D. Zou, H. Xie, and et al. Exploring the potential of using Chatgpt in physics education. *Smart Learning Environments*, 10:52, 2023.
13. Mohammad Alafnan, Samira Dishari, Marina Jovic, and Koba Lomidze. Chatgpt as an educational tool: Opportunities, challenges, and recommendations for communication, business writing, and composition courses. 3:60–68, 03 2023.
14. University of Colorado. Phet interactive simulations website, 2024. Retrieved in 2024.
15. W3schools. Html canvas graphics website, 2024. Retrieved in 2024.