Collective behaviour

Scratch Virtual Traffic Simulation

Duran Can Erdoğan and Nurzhigit Yekibayev

Collective behaviour course final report
GitHub: https://github.com/durancanerdogan/Collective-Behaviour-Group-M

February 9, 2024

Iztok Lebar Bajec | izredni professor | mentor

This report tries to introduce students who are getting mathematics education to the world of computer science by using collective behavior as a link between them. For this reason, in the report, we showed a scratch project that is highly referenced by the book of Braitenberg "Vehicles: Experiments in Synthetic Psychology" and we created a "Virtual Traffic Simulation" in scratch from scratch. For the main idea of the project, we decided to use "Game Theory" [1, 2]for children (high school and middle school level) to make them more engaged in the topic as much as possible. Game theory is based on everyday games that we play and mathematics with the logic behind those games. So, with the game theory, we can analyze "Collective Behaviors" as mentioned. With the help of scratch, we found a way to implement these studies in the world of computer science and try to develop interest in students in wanted areas of the study. When we checked the interest increase of students when we did interviews about scratch environments.

Programming Education | Scratch | Collective Behavior | Real-time Visualization | Educational Technology

Virtual traffic simulation as mentioned above is mainly inspired by the "Vehicles" book. In this experiment, we created a very basic game of cars reacting to different types of stimuli like traffic lights, road work, or even their lights. What we are trying to achieve here is when we teach and show the world of scratch to students, we want to see them trying to create their projects in the environment and try to understand what collective behavior is and why it is important in so many areas, especially mathematics, and programming[3].

Related Work. We rely on different materials to get the best out of the project, but the book "vehicles" is the main source of our experiment and report structure. In this book, the writer tries to explain some of the psychological behaviors with very metaphorical explanations by using real-life cars as metaphors[4, 5]. If we must give examples for these metaphorical explanations, love is associated with the light source for the cars. Like humans also cars react to the light sources as metaphors as some kind of love for the cars. While Braitenberg's work [6] focuses on theoretical constructs and their philosophical implications, our project extends these ideas into an educational context. By implementing these concepts in Scratch, we bridge the gap between theory and application, providing a hands-on experience for children to explore the principles of collective behavior playfully and engagingly.

Methods

Virtual Environment Design.

Virtual Vehicles. The choice of the vehicles is based on creating two different cars in different settings like colors. The cars are programmed to interact with various things on the board. They react to the conditions of traffic lights if there is another car or if there is an obstacle like roadwork. They try to find their routes on the map given.

Traffic Light. On the other hand, designing a traffic light was more automated. The only thing that we must consider is they need to change colors automatically in given periods. The traffic light had two states – red and green – simulating standard traffic signal conditions. They don't react with the cars vice versa cars react with the current light conditions on the traffic light.

Blinkers and Car Lights. For the light sources we must design them in a way that not only will be cosmetic to the whole setup but also has a purpose. So, for that reason, we simply wanted to use them as a kind of interaction link between the vehicle and the environment.

Other elements. Other elements in the equation only serve the purpose of cosmetics to engage the realism of the whole setup.

Programming Logic.

Light Sensor Behavior. In the core of the light sensors, we used a basic "forever" loop that is moving according to the movement set of the car. When the car starts to rotate in a pointed direction the light moves as a part of the car in that direction.

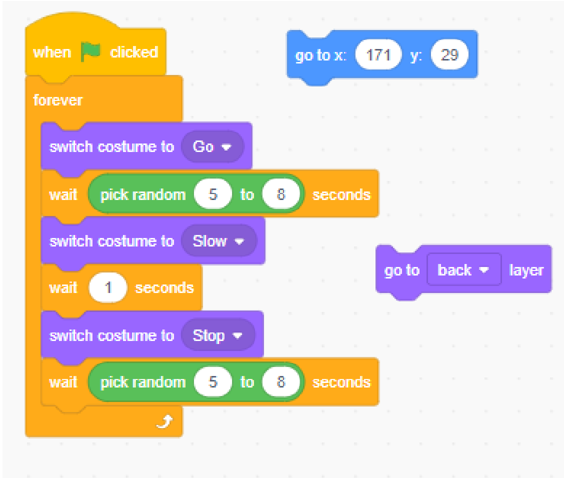

Figure 1. Illustration depicting the coding scheme for the light sensor behavior, demonstrating how it enables virtual vehicles to detect obstacles and shape their movements in response to various stimuli in the Scratch environment.

Traffic Light Interaction. Building upon the light sensor behavior, additional programming instructions were implemented to simulate traffic light interactions. When the traffic light turned red, the virtual vehicles were programmed to come to a stop. Conversely, when the light turned green, the vehicles resumed their movement. This is created by writing infinite loop set of light condition.

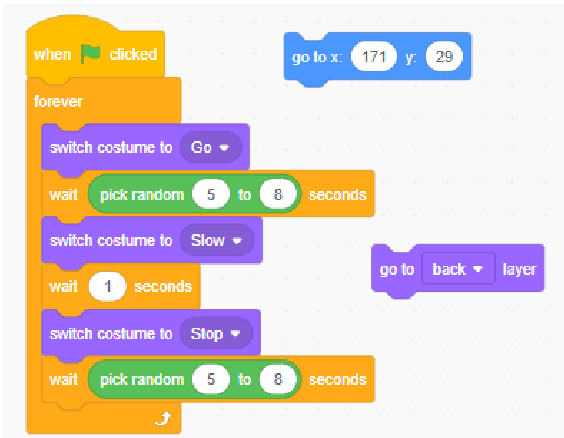

Figure 2. Visual representation of the coding scheme for the traffic light interaction, showcasing how the virtual traffic signal changes colors automatically within a preset sequence. The illustration illustrates the logic behind the interaction with virtual vehicles, controlling their movement in response to dynamic traffic conditions in the Scratch environment.

Virtual vehicles. Vehicles of the equation are most connected to other pieces of the puzzle. For the vehicles in the given code, we put a part that reacts to the condition of the traffic lights. It is automated as they will stop when it is red, they will move when it is not. For the blinker, it is connected to turn movements in the movement pattern of the vehicle. Finally, the pathfinding of the cars is done by pre-assembled code telling vehicles when to return and when to return.

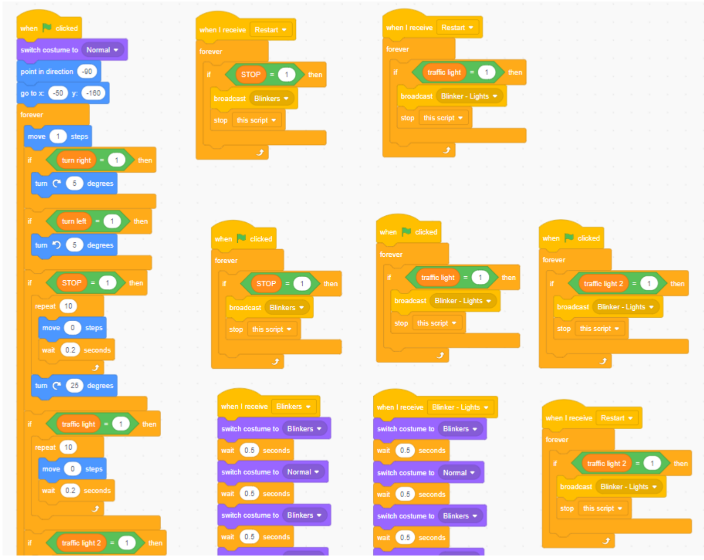

Figure 3. Visualization detailing the coding scheme governing virtual vehicles in the Scratch environment. The illustration demonstrates how these virtual entities react to various stimuli, including traffic lights, roadwork, and other vehicles, showcasing the programmed behaviors that mimic real-world interactions.

Blinkers and Car Lights. They serve their real-life purpose also here they detect the obstacles for the cars. In programming hand on scratch, we wrote a color detection code for car lights and for the blinkers, they are working according to the car's turn set.

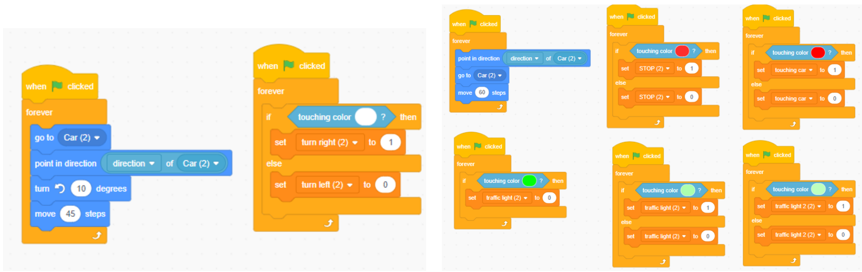

Figure 4. Detailed depiction of the coding scheme governing blinkers and car lights in the virtual environment. This illustration highlights how these elements are intricately connected to the turn movements of virtual vehicles, contributing to the overall realism and functionality of the Scratch project.

Model. The computational model underlying the Scratch project involves a set of rules governing the behavior of the virtual vehicles in response to environmental stimuli. The model can be described as follows

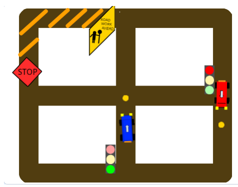

Figure 5. Visual representation illustrating the overall layout and arrangement of components within the experimental framework. This figure provides an overview of how the virtual traffic simulation, light sensors, traffic lights, and other elements are structured in the Scratch environment to create an engaging and educational experience for students.

Light Sensor Response Model. The main use of this light sensor model is to indicate when to stop or when to move. As mentioned before virtual vehicles react to lights, roadwork, and another vehicle on the map. So, to navigate them we placed a light in front of them as a car light, but it is not only for decoration it has an important purpose in detecting obstacles in front of the car to shape its movements. The logic behind the light is that it is purposely put in front of the car to determine these obstacles before the car even reaches them so the car can act more naturally.

Traffic light interaction Model. This model is pre-set and works automatically without directly reacting to any vehicle on the map. It is programmed to pick a random number at given position and change its colors one by one starting from red, yellow, and green.

Emergent Collective Behavior Model. This model serves as a good showcase of simplistic traffic simulation within the context of collective behavior x game theory. In the model students will engage with the game theory side of the project by designing different conditions for the vehicles, on the other hand, he/she will create an organism that is moving on its own according to the rules given to them.

Implementation

The Scratch environment served as the canvas for implementing the virtual traffic simulation. Using the Scratch programming interface, we created sprites to represent virtual vehicles and a traffic light. The visual drag-and-drop interface provided an intuitive platform for coding, making it accessible for the target audience—children.

Results

Without a proper set of environments, it is hard to determine exact results for the experiment but on the other hand we know with our small group of students that we did the interview with 3 different students we acquired the information that implementing scratch in their normal education program we monitored increase in their interest for both subjects.

Discussion

Educational Significance.

Programming Concepts. The projects want to help students at an early age engage with the basics of programming concepts. It is obvious that to achieve great interest for students, we must make different projects and let them engage and create new things with those but in the end, it is undeniable that it is going to help students develop interest in the area and work.

Collective Behavior. The simulation provided a tangible illustration of collective behavior through the interplay of individual vehicle rules. This concept laid the groundwork for discussions on emergent properties in systems and the influence of individual actions on collective outcomes.

Real-World Analogies. Best way to teach a child something is show them the thing they are learning has a purpose and real-life existence[4]. In our virtual traffic simulation students will try to create what they see in their daily lives on a virtual setting. This will help them to stay interested in the whole project[5].

Limitations and Future Directions. It is brilliant way to start something bigger in the future of course but with time pressure of most study plans in most countries sometimes it is much harder to focus on side project like this but as educators we hope for better systems where students can develop interests for their future careers. On the other hand, this, and project like this helps students feel much more comfortable in university settings.

Conclusion. To sum up all the things that mentioned this project gave us a vision to search for better ways to teach students. For the students we can say that they will be having time to create critical thinking and problem solving. This study will help to understand big complex structures little by little. For further education there are platforms like "Code Academy" and many others [7, 8].

CONTRIBUTIONS. Reading of the base material is done by every group member. Creation of the game and this final paper is done by each group member doing half of the requirements

Bibliography

1. Naldi G, Pareschi L, Toscani G (2010) *Mathematical modeling of collective behavior in socio-economic and life sciences.* (Springer Science & Business Media).
2. Baltzell JE (1982) The application of game theory in collective behavior: A critique of berk's "gaming approach". *Sociological Focus* 15(3):269–278.
3. Sanne F, Risheim I, Impelluso TJ (2019) Inspiring engineering in the k12: Biomimicry as a bridge between math and biology in *ASME International Mechanical Engineering Congress and Exposition*. (American Society of Mechanical Engineers), Vol. 59421, p. V005T07A015.
4. Öztürk AF (2015) Master's thesis (Maltepe Üniversitesi, Fen Bilimleri Enstitüsü).
5. Fatma A (2019) Doğa ve inovasyon: okullarda biyomimikri. *Anadolu Öğretmen Dergisi* 3(2):214–233.
6. Braitenberg V (1986) *Vehicles: Experiments in synthetic psychology.* (MIT press).
7. Koenig G (2007) Orchard software and the nctm principles and standards for school mathematics. *Reston: Siboney Learning Group.*
8. Stein MK, Remillard J, Smith MS (2007) How curriculum influences student learning. *Second handbook of research on mathematics teaching and learning* 1(1):319–370.