Evidence 2. Final Activity

Modelación de Multiagentes con gráficas

Full name	Student License	Career initials
Antonio Rafael Cedillo Rodríguez	A01656823	ITC
Jesus Alexander Meister Careaga	A01656699	ITC
Samuel Sandoval Delgado	A01656191	ITC
Luis Roberto Martinez Ramirez	A01662619	ITC

Proposal

The aim of this proposal is to create a Mesa model represented in Unity, which looks to optimize traffic in a city with the objective of improving people movement and showing the contamination in the environment. With this in mind, various agents in mesa are used:

First off, we have agents called *people*, who have the objective of arriving at a set destiny, and have the options to take their own car or ride the bus. In this manner, diverse quantities of carbon dioxide are emitted to the atmosphere.

The tools we used are the Python library mesa with the aim of the creation of agents, data collection and the simulation, while we used Unity as a graphic motor to be used as a 3D modeler and medium for representation of the simulation.

This proposal has evidence to be based upon the solution of Mexico City's governor, Claudia Sheinbaum, of investing various millions in improving and making more prominent public transportation, according to *El economista (2022).*

Environments - Mesa

Agents:

- House: Can have from 1 and up to 3 people living there.

- Person: This agent is spawned with the purpose of going somewhere. To get there, there is a chance they choose either their own car, or public transportation. When they select their method of transportation, they are going to trace a route using Dijkstra's algorithm to find their way. If they choose a car, they are going to find the nearest parking spot to their final destination. If they choose a bus, they are going to get down at the bus stop nearest to their final destination.

- Car: Starts at a personal parking spot, and can be driven by a person, following a certain route the person needs to take to get to their destination. If a person is inside it, it follows the route established by the person who got in. It reacts to buildings, people, traffic lights and other vehicles, and stops when needed to. When it gets to the destination, if the parking spot is already taken, it searches for another until no more are left, in which case, goes back to its original location, which makes it a hybrid agent. As well, if the car does not move for a certain amount of steps, it can take a random decision to look for a new path, which makes it stochastic.

- Bus: Follows a list of stops and picks people up. It can hold up to 20 people. Always follows the same route that is calculated using Dijkstra's Algorithm, but stops if needed to avoid crashing with other vehicles or people, as well, it stops in case it sees a red light. This is a

deterministic agent, as well as hybrid one, since it does not make any mistakes and follows a planned route, but it can react to events that happen around it.

- Parking spot: Can be either a personal parking spot or a public parking spot. Cars only spawn in the first ones, and always search for the second ones.

- Driveway: Tells the automobile where it is allowed to go. It is used to determine the traffic rules everyone must follow, works in the same way a directed graph does, to facilitate the use of search algorithms.

- Sidewalk: It tells where a person is allowed to walk, and can be made of concrete or grass.

- Obstacles: This agent is made to determine places that neither cars or people can walk through. This includes mainly buildings.

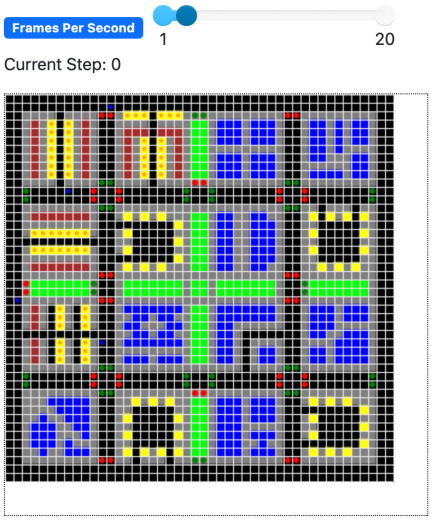
Model:

Our model is the simulation of a city divided into blocks. Each block can be either a neighborhood, a parking lot or a building area. Each neighborhood has a bus stop and the bus has a route that gets to every building area block. The main metrics of the simulation are the number that get to their destination in a certain amount of time and the total contamination the city receives.

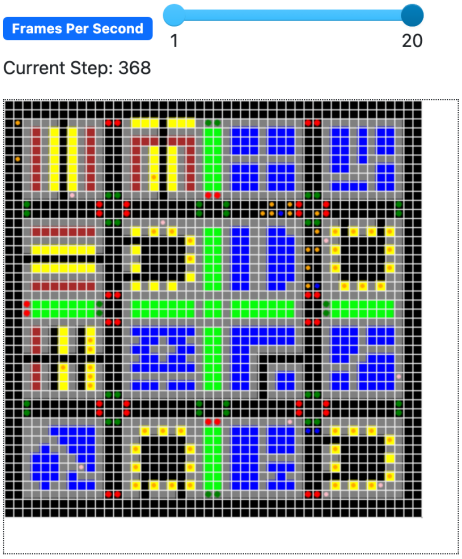
The model can take as a parameter the chances of people taking public transportation so that metrics can be tested.

The model ends once every person has gotten to their destination or once it reaches a time limit.

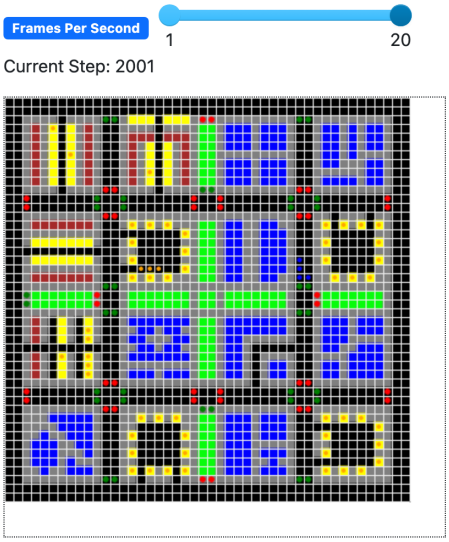
Start:



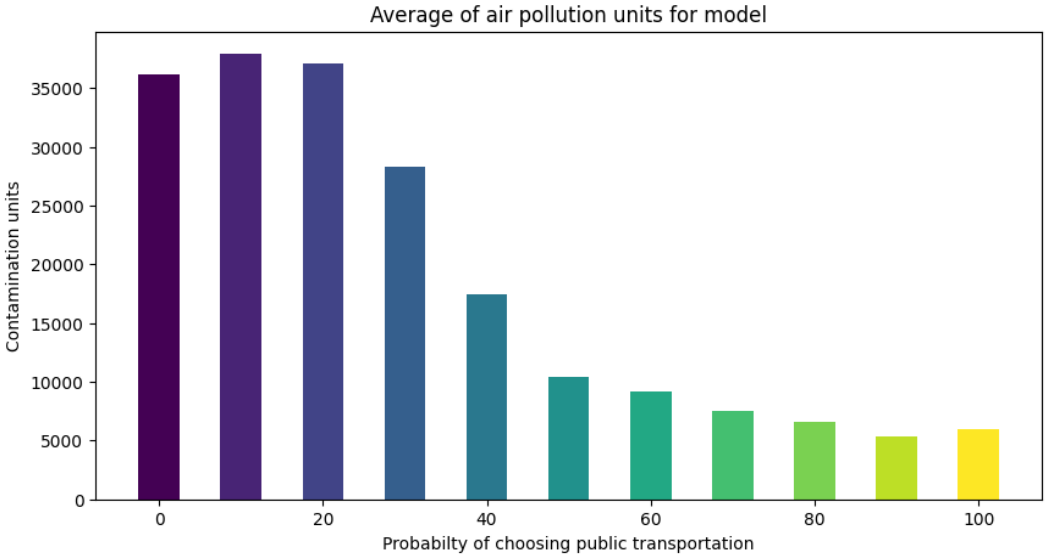
Middle:



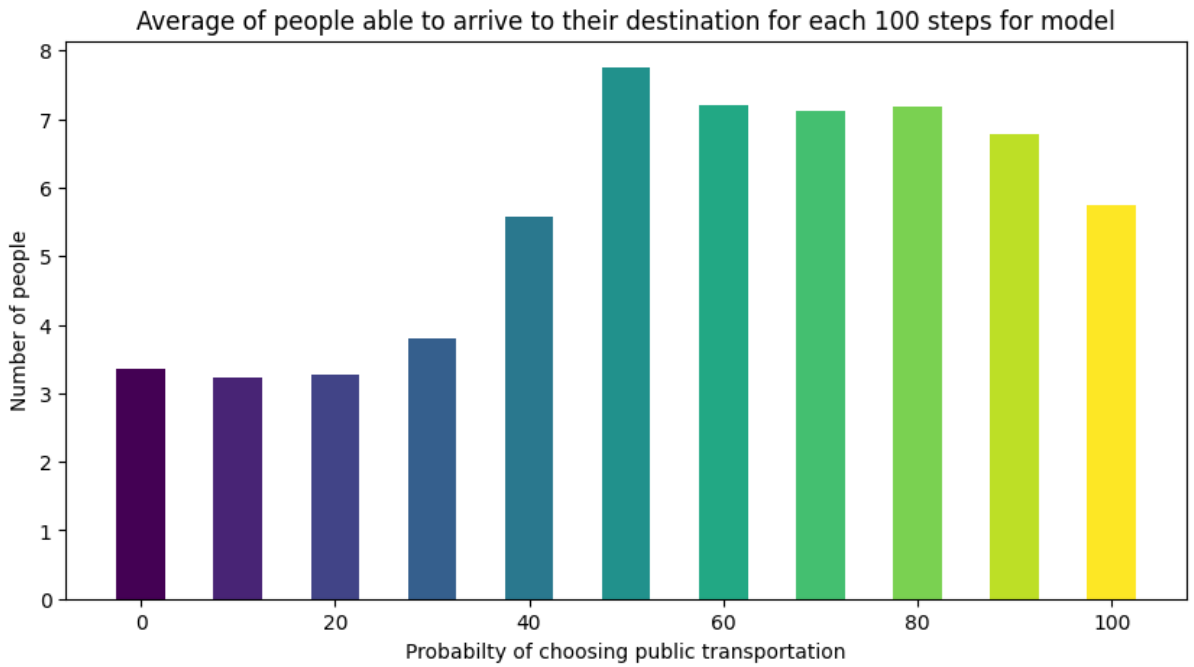
End:



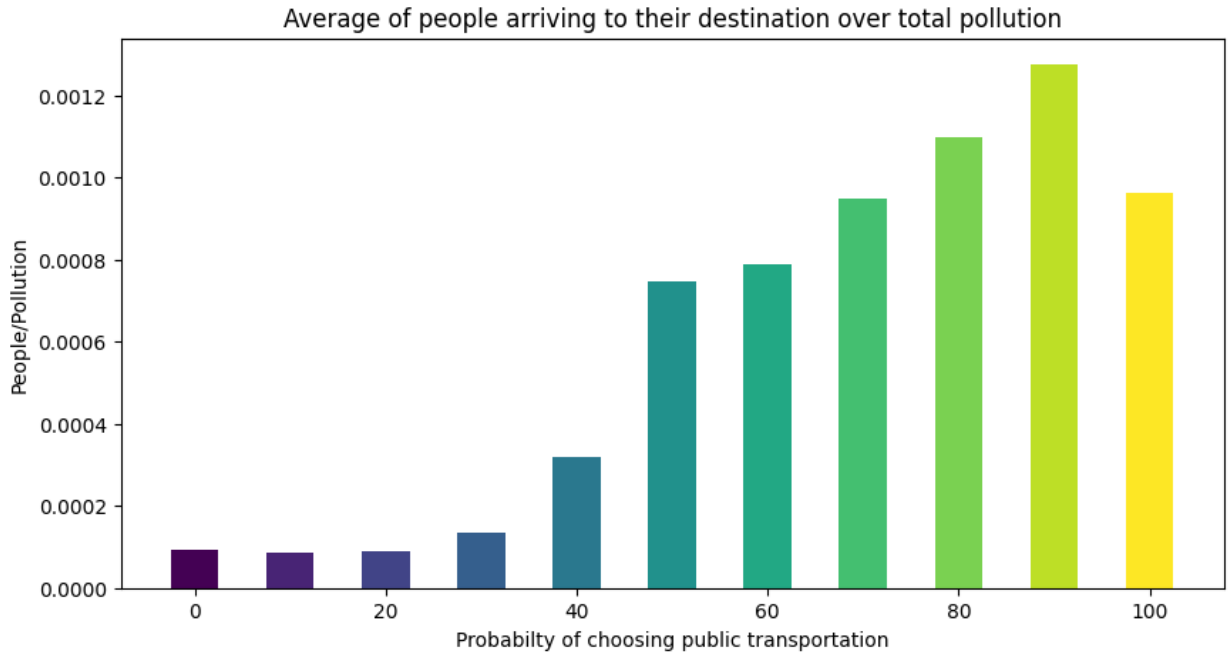
Results and Graphs



On this first graph, we can observe that the average air pollution tends to go down as the use of public transportation increases, however it is not completely linear and has variations. This happens due to the fact that sometimes the capacity of public transportation can be overflowed. This causes counterproductive effects on the total distance traveled, which increases the total pollution.



On this second graph, we can observe that the average number of people that are able to get to their destination is seemingly normally distributed, as the maximum number happens when half of the population uses public transportation.



In the final graph, we integrated both graphs in one, dividing the average number of people able to get to their destination in 100 steps over the average pollution. Analyzing this graph we can conclude that

the best distribution which minimizes pollution and maximizes people getting to their destination is 90% public transportation and 10% private transportation.

Graph conclusions:

Our main result defines that our main proposal is for the government to strive for a 90 - 10 distribution between pubic and private transportation. However, the results are heavily influenced by the design of our model. To get more accurate data, we should increase the size of our simulation and perfect the variables that our agents use to get more realistic results.

Environments - Unity

The use of Unity is to represent in a more graphical and appealing way the simulation in the python Mesa code. This is the resources and methods it uses to “transfer” the Python/Mesa code to the Unity/C# environment:

- JSON communication: The Mesa code makes sure of creating and sending Jsons with the positions or states of each of the agents, which are sent by the use of a flask server, that the Unity C# code receives (which is placed in an empty game object to interact with Unity).

- Prefabs: Various prefabs (models created by other people and offered in the Unity Asset Store) are used in the Unity project, these are used in the creation of the map and for the person, bus and car agents.

- Skyboxes: These are used as a background sky for the model. it can be changed between night and day whenever the user wishes.

Performance Report

There were various elements taken into consideration to reduce the performance issues and make more possible a higher scalability of the simulation is to remove the use of animations and particles to reduce the usage of the GPU.

To show the performance, this is the results of the Mesa and Unity simulations running on a Macbook with a M2 pro:

Nombre del...	% CPU	Tiempo...	Subpro...	Activaciones d...	Clase	% GPU	Tiempo d...	PID	Usuario
Unity	128.6	12:36.64	89	434	Apple	66.6	3:16.42	2694	luismtz

Conclusions

Team/Group

As the simulation and the graphs show us, a more dominant use of public transport such as buses, reduces the amount of traffic in the city, it also might solve some of the parking space issues, thanks to the reduced use of personal cars. Another thing to consider is that the simulation itself can be beneficial for urban planning and policy formulation. Urban planners and policymakers can use this data to develop more sustainable transportation strategies. According to a study in the Journal of Transport Geography, integrating simulation data into planning can lead to more informed and effective decisions to reduce air pollution and the mobility in the city.

But, one element to consider, is that the cost to implement this might be pretty high, points for these arguments are that it not only does have to include the costs to only create said buses, it also has to include the costs to maintain them, create the stops or other extra resources such as the cards and machines to recharge bus passes or currency (in case there is a need for it).

Individual

Antonio Rafael Cedillo Rodríguez

I focused on the area of Unity and documentation of the resulting simulations, while on the line of work, i found again the spark that made me like game development and the tools that are used, such as Unity, while also learning new things such as how to use blender for 3d modeling and assigning various graphic elements to the game, such as materials or skyboxes.

As for the project itself, it's possible to be expanded even further, thanks to some of the prefabs being optimized for mobile devices and the lack of animations for the persons and the cars. With our results I found out that a solution like our proposal may be possible, since it reduces the emission of contaminating gasses and traffic, but the implementation might be expensive depending on the number of buses we want to implement.

Jesús Alexander Meister Careaga

Through the work I made with my team, we were able to generate a Mesa model that successfully represented the proposal we developed. Contrary to the past evidence, we managed to do a model with diverse types of agents, such as cars, buses or persons, each with their own gimmicks and quirks. I also supported the Unity team with the implementation of the Mesa code and logic into the graphical representation that is Unity.

With the Mesa simulation I found out that the solution may be possible to implement, since our results may positively affect the way traffic is handled and reduce the emission of contaminating gasses. However, it is important to find a balance between private and public transportation according to what we are wanting to achieve.

As well, we found that running mesa simulations mainly uses the CPU, so if we were to expand our main model to a bigger one, to find even more accurate results, it would be advisable to expand the CPU power of the computer running the simulations.

Luis Roberto Martinez Ramirez

In this challenge I focused especially on video game development in Unity, where I was able to develop and improve my skills both in design and in some aspects of programming, as well as learning to use how to implement a skybox in the video game as well as the different combinations of materials, prefabs, etc. that could be made

Samuel Sandoval Delgado

In this challenge I worked on Unity with a 3d model. This time i was improve my skills on the 3d models and I learned some of C# of spawning objects. Other skills that I learned in this subject was animation on the prefabs with mixamo and the application with code if the character moves. Almost, i learned the materials in Unity and the lights for traffic lights.

Resources

Drive:
https://drive.google.com/file/d/1BUBTK7lur2fTIktbNf9aaIa-korQ9JVm/view?usp=sharing

References and Sources

El Economista. (2022, May 31). CDMX alista plan de acción enfocado en desahogar el tráfico vial. El Economista. https://www.eleconomista.com.mx/estados/CDMX-alista-plan-de-accion-enfocado-en-desahogar-el-trafico-vial-20220531-0088.html

Union of Concerned Scientists. (n.d.). Car Emissions & Global Warming. Retrieved from https://www.ucsusa.org/resources/car-emissions-global-warming

Center for Neighborhood Technology. (n.d.). Public Transportation's Impacts on Greenhouse Gas Emissions. Retrieved from https://www.cnt.org/

npj Urban Sustainability. (n.d.). On the impact of urbanisation on CO2 emissions. Retrieved from https://www.nature.com/npjurbansustain/

Prefabs used

Realistic Tree 9 [Rainbow Tree] | 3D Trees | Unity Asset Store. (2016, 2 febrero). Unity Asset Store.

https://assetstore.unity.com/packages/3d/vegetation/trees/realistic-tree-9-rainbow-tree-54622

Low Poly Cars: Toy Cars | 3D land | Unity Asset Store. (s. f.). Unity Asset Store. https://assetstore.unity.com/packages/3d/vehicles/land/low-poly-cars-toy-cars-258470

Street Lamps 2 | 3D Exterior | Unity Asset Store. (s. f.). Unity Asset Store. https://assetstore.unity.com/packages/3d/props/exterior/street-lamps-2-260395