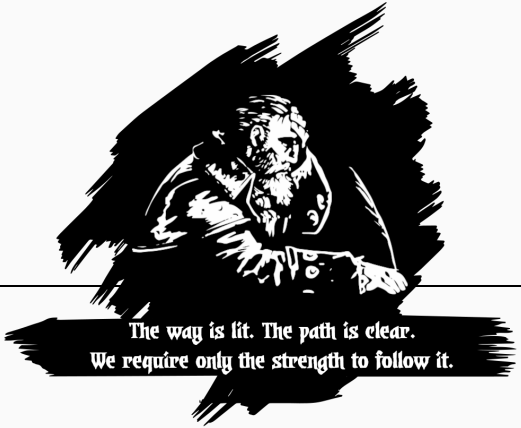
SYMUA - a simple survival guide

ACU 2023 Team



Copyright

Rules

- You must have downloaded your copy from the Assistants' Intranet <https://intra.forge.epita.fr>.
- This document is strictly personal and must **not** be passed on to someone else.
- Non-compliance with these rules can lead to severe sanctions.

Introduction

Subject

Crowd movement

- When agent are too packed disasters can happen.
- We want to simulate crowded environment to analyze the place where problem occured.
- We choose to reproduce a famous street called Shibuya in Japan.

Crowd simulation

- Therefore for the project purpose we needed proper engine to simulate a crowd.

Base Project

We found an engine Crowddynamics to emulate a crowd.

This engine was written in python so it was possible for us to work on it.

Therefore the project is mostly about modifying UI and creating comportment to agent.

Setup simulation

Create Simple simulation

We started to launch a simple example simulation to understand how and where agents were interacting.

We search how the map was instanciated and how we can exploit it for our simulation.

Adapt code base to our project

Firstly, we needed to fix multiple problesm in the project:

- All agents were linked to their neighbours and not to their direction.
- Multpile missing information for debugging (such as hitbox).
- Agents were still alive after reaching their last goal.
- UI was not perfect and was plotting some useless representation.

Create our maps

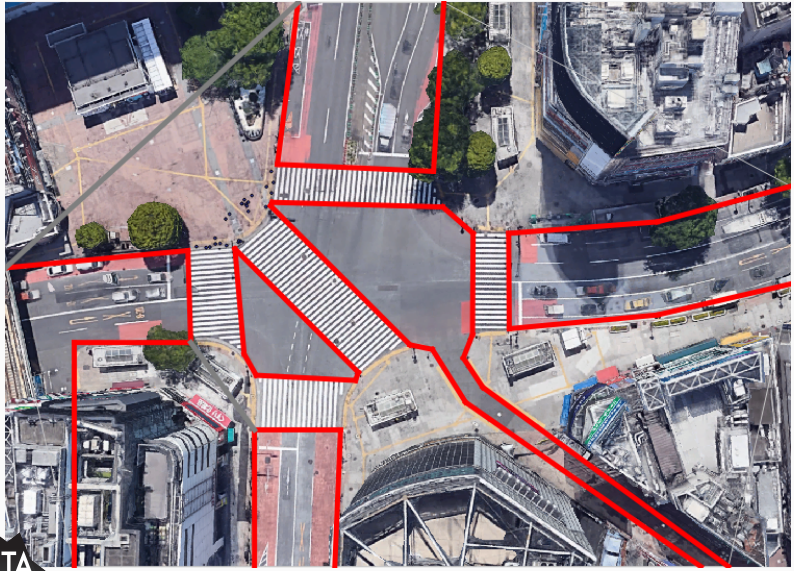
We first tried to represents the Shibuya street, as it is well known for his high pedestrian traffic.

We took a picture of the street for more explicit representation, put it in background to delemitate our map.

Later on, we will create another map representing another street.

Run simulation

Run simulation



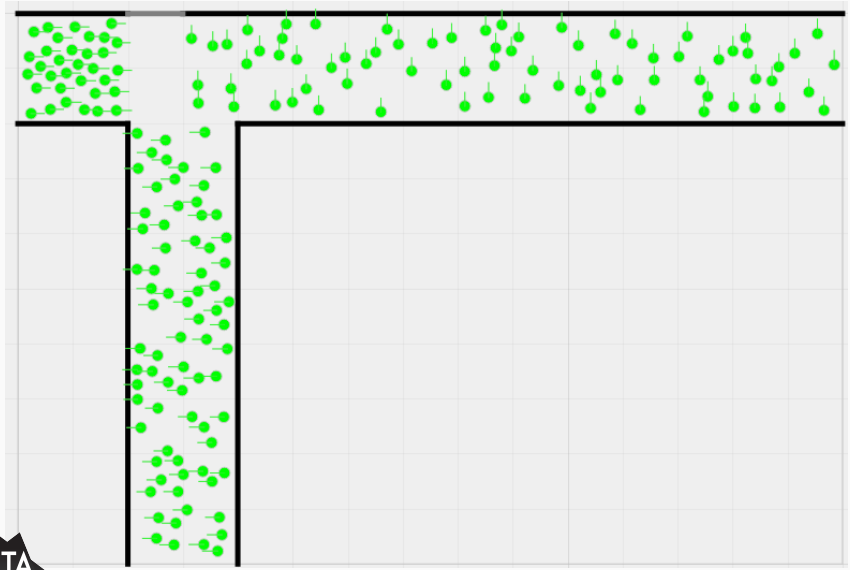
Run simulation

After launching the simulation, we can see that people push each other outside the pedestiran crossing, which is what happen in real life when there is many too many people.

As a result we where not able to use this map as simulatinf too many agent was making the whole really slow and not usable.

After seeing this, we understood we needed to represents a smaller street with one exit only to compress more the people and find a good map to represent the panic.

Run simulation



Adding comportement

Oxygen

We needed to allow the agent to represent people being hurt:

- Add oxygen var from 4096 to 0.
- Once to 0 they died.

Then, we computed the oxygen variation:

- +1 for each other agent too close
- +3 if too cloose from wall

If they got a variation more than 5 they start to losse oxygen else they regain it.

Panic

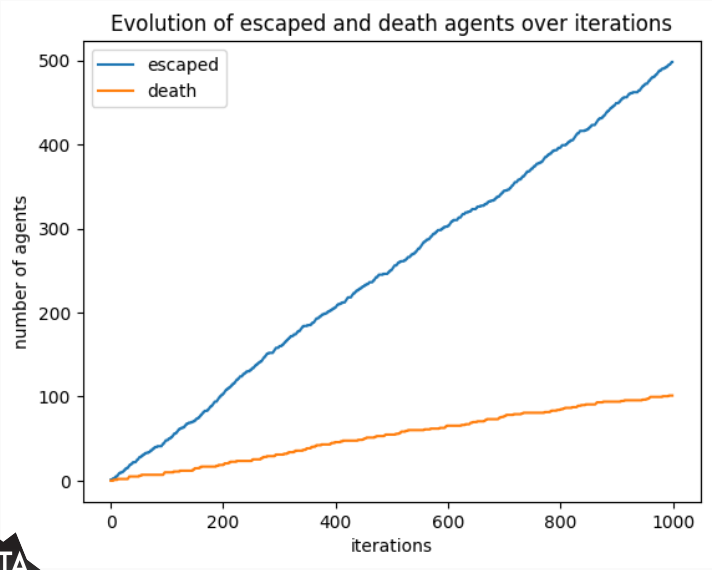
We add panic to the agent to simulate chaos.

We take the variation of oxygen to add change to their behaviour after one point.

People will try to run away, to escape the crowd.

Demonstration

Results


Evolution of escaped and death agents over iterations
escaped
death
number of agents
iterations
0
100
200
300
400
500
0
200
400
600
800
1000

The end

Do you have any questions?