Project Checkpoint

Simulation of Virus Interaction with an Environment

Rikky Roy Koganti (rkoganti)
Elena Feldman (efeldman)

Work Completed

- Completed research on which viruses we are planning to simulate, the interactions between viruses, people, vaccines and cures, and also how a virus could evolve
- Came up with a complete outline of how the simulation will work, to refer to while coding the program (Overview of simulation given at the end)
- Finishing implementing several classes to be used in the simulation (various virus classes, people classes, land classes, and medicine classes among others)
- We are about half a week behind as compared to our original schedule, as we should have finished the serial implementation of our simulation by this time, so we came up with a revised schedule below
- All tasks will be done by both teammates together
-

Revised Schedule

Week	**What We Plan To Do**
Apr 19-23	Finish implementation of simulation (all classes, models, ai etc)
Apr 24-29	Test, optimize and parallelize implementation via CUDA
Apr 30-May 2	Implement lower-end graphics to the simulation
May 3-9	work on demo and final report, implement higher-end graphics, and try to do openMP parallelization

Goals and Deliverables

- We still plan to come up with a working simulation showing the activity of 3 RNA viruses in 6 different environments that try to replicate the attributes of several real world environments (such as population density, amount of wealth etc)
- We will also implement the viruses such that they have their own AI and even when countered by a vaccine or cure, have a chance to use all the previous samples (people they infected) to use as data to possible evolve

- Finally, we will parallelize the implementation via CUDA, as there will be thousands, and possibly millions of people in the simulation being infected, and we will also implement some form of graphics to show the actual interactions between viruses, people and the medicine

- As for our previous high-end goals, we might not have time to fulfill both of them (implementing high-end graphics via openGL and parallelize via OpenMP as well), so we will focus more on just implementing openGL first.

Concerns

At this point, we have a clear view of what it is we wish to achieve, and all the work looks reasonable and achievable, so we believe all that is left is to actually sit down and finish the coding. If there is any concern, it will be the implementation of the graphics portion. We haven't decided yet how the simulation will look like, and how we will be able to fit thousands to millions of points into it. At this point, our plan is to have a grid based graphics interface, with each grid representing a different area for the virus to attack. Each grid would contain its own data points for people, which would have different colors depending on whether they are healthy or infected. This could very well change though, if we find a more efficient or natural way to represent the simulation.

Virus Simulation Overview

- Initialize the simulation with 6 different zones of different sizes/locations and densities and give them some initial population. The 6 different zones represent the different kinds of living environments we might have in the real world. Our aim would be to use world's either most dense, populated, polluted, wealthiest, etc. cities.

- Each zone has several different attributes. The main attributes would be land size and population size. Other possible attributes include distributions of the population. For instance, wealth, age, living conditions (hygiene, etc). All these attributes play a part in deciding how fast the virus might propagate and whether the zone population survives the virus or not.

- We then interactively choose to introduce one or more RNA viruses into any of the zones. The virus has the capacity to evolve, and it has its own AI. This is where machine learning comes in.

- Once a virus enters a zone, it starts infecting people. People can either be infected or be carriers (people who are immune to the virus but can still pass it on to others). The virus continues propagating through the population of the zone. As mentioned before, whether the virus wipes out the population of that zone or not depends on several factors. The zones attributes for one.

- Then, we introduce either a cure or a vaccine, depending on the virus, to any one of the zones.[The other zones will have access to this medicine depending on their distance relation with that zone or how medically advanced the zone is itself and whether it can develop the cure on its own]

- When the medicine reaches each zone, the cure will be transmitted according to the wealth distributions.

- At this point, the virus goes into a stage of evolution, becoming resistant to the vaccine and the cure and starts affecting people in the zone where it is again.
- Some points to note: the same virus in different zones can evolve in different ways (genetic shift or drift), becoming resistant virus with different traits. Hence, new vaccines must be reinvented and the virus becomes highly active again.
- Both the viruses and the vaccines have several attributes. A strength level for one. The virus will have an ai. If a virus evolves, it uses all the people it infected before as samples and uses all those samples to make a new class of virus, whose attributes values depends on the samples.
- Once a virus has evolved, we repeat the same process as before. It starts infecting people again. A vaccine is found once enough time or people have died. ... and so on
- The simulation ends, either when there are no more viruses left (all viruses were completely wiped out by the vaccine and no virus evolved) or when all the people are dead. A zone can have its population wiped out if the vaccine takes too long to be found. The time it takes to find the vaccine depends on the strength level or threat level of the virus (another attribute).
- By our implementation, we could have multiple viruses in a single zone. If all viruses in other zones died out before this time elapsed, then the simulation itself ends.
-

Parts we intend to parallelize

- Initializing the zones (independent of each other)
- propagating the virus aka updating each person (Transmitting the virus from person to person can be seen as a task. we add all tasks to a scheduler and threads just pick them up when ready. Propagation is also independent of each other. Each infected person just would infect all non-infected people near him with a certain probability)
- transmitting the vaccine/cure throughout a zone
- When a virus evolves, it goes through thousands, maybe millions of samples in order to improve. This can be done in parallel, as we are just going through data to come up with some final values for the new virus's attributes.