**Crowd Simulation Application**

MS Project Report

Zhicheng Chen

Understanding complicated crowd behaviors is essential to urban designers and architects. However, grouping a large amount of people to do experiment is dangerous and unrealistic. Among these, design and create an application which could correctly represent crowd behavior is crucial. This project report describes an implementation called *Crowd Simulation* that aim at creating realistic, unique and dynamic crowd by takes agents’ data as input and outputs the result in animation form.

**Introduction**

The increasing proportion of people living in urban areas brings new challenges to urban planning and architecture. Crowd simulation plays an important role in addressing these challenges. With the help of crowd simulation techniques, urban designers and architects could determine the evacuation time of a massive crowd, predict the behavior of a crowd flow inside of a building or prevent overcrowding during certain events.

A crowd forms when a large amount of people gathers in a limited space. Simulating the whole crowd as a single unit could help understand the behavior of the moving crowd. However, if we divide the crowd into groups that contains 2 to 3 people or individuals, the behavior of the crowd can be more realistic. In a group, people know each might walk together. Previous researcher Reynolds [1] proposed a steering approach known as Leader Following (LF). This approach involves pair agents where the “follower” agent follows the leader and stays on its side. This disadvantage of this approach is that in this basic steering approach, the leader agent does not wait for its follower agent if the distance between these two agents is too large, which is not realistic.

More recent simulations of crowds of people use more complicated calculation. For example, previous approach [2] designs agent as ellipses that have a sense of the environment and plan their own path ahead of time to avoid agent collisions. Unfortunately, the output of this kind of simulation lacks realism and flexibility. Since it does not involve dynamic behaviors such as allowing agents to move in and out of different group or queues based on agent’s desire, agents who have planned a path ahead of time might end up in the longest waiting line without being able to switch. In reality, people do not just stay in their waiting line once they choose it, they might need to change waiting lines if there is a better option.

Related Work

Report Outline

**Resources**

Java Port of Recast & Detour navigation mesh toolset

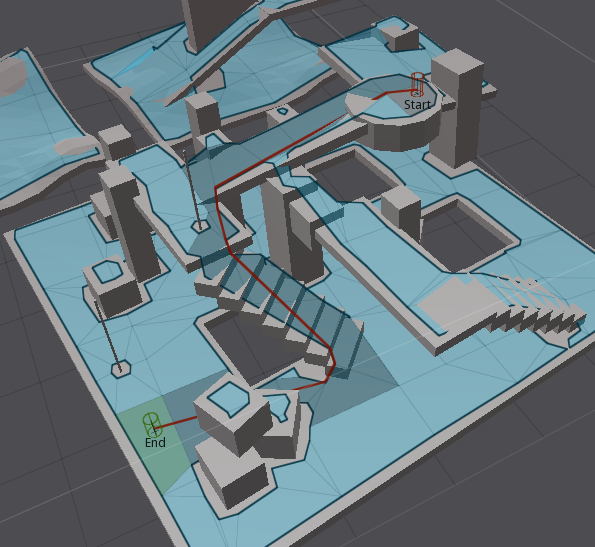
(ref: <http://masagroup.github.io/recastdetour/index.html>)

**Recast**

Recast is a state-of-the-art navigation mesh construction toolset for games. Recast is an open source which could automatically provide you a mesh at any level geometry in instant time; Recast could also be customized to achieve user’s specific purpose.

**Detour**

Detour is a spatial reasoning toolkit which accompanies with Recast to offer a simple static navigation mesh. DetourCrowd is a crowd management module offers features for agents handling and behavior customization. Detour allows user to create lots of agents and move agents in navigation mesh. What’s more, Detour allows user to create customized behaviors that determines agents how to move and react.



Implementation, Challenges, and Output

Detail

Result

Evaluation

Future Work

Acknowledgements???

References

Test Cases???