**Dynamic Crowd Simulation using Unity’s NavMesh**

HUANG Maxton, TEO Jeryl, and YEO Xavier

# Introduction

## Overview of the Research Topic

Crowd simulation in video games is a critical aspect of creating immersive and realistic game worlds. This research examines how entities navigate and avoid other pathfinding entities while carrying out their behaviors by utilizing Unity’s NavMesh and implementing a custom Behavior Tree architecture. This will enhance our understanding of current systems in place and the behavior of pathfinding entities, allowing us to create more realistic and immersive games.

## Aims and Goals of the Research

The primary aim of this research is to explore how dynamic pathfinding entities interact with each other within a simulated environment. To achieve our aim, we will be seeking to observe the following.

Firstly, we want to understand Dynamic Pathfinding Interactions. In CS380 we have only learnt Pathfinding on a dynamic-to-static level. We will investigate how pathfinding entities navigate and interact when both are in motion, expanding beyond static pathfinding scenarios.

Following which, we will analyze the real-time interaction dynamics between pathfinding entities, including how they adjust their paths to avoid collisions and respond to each other’s movements.

We will then develop realistic behavior trees that will drive our entities to have behaviors that require interaction between one another in a high-density crowd, to enhance the simulation observation.

After we have analyzed the behavior of the pathfinding entities with one another, we want to explore modern techniques and investigate how contemporary games approach crowd simulation and control.

# Background and Literature Review

## Behavior Between Multiple Pathfinding Agents

Research in Artificial Intelligence (AI) for crowd simulation delves into the complex interactions among multiple pathfinding agents within shared environments. These agents must navigate, avoid collisions, coordinate their movements and cooperate to achieve shared goals. For example, this could be achieved by Unity’s NavMesh.

## Unity’s NavMesh

Unity's NavMesh is a crucial tool in game development for implementing pathfinding. It allows developers to define walkable areas and automatically generates efficient paths for agents. Key features include the use of algorithms like A\* to compute optimal paths, simplifying development and enhancing simulation fidelity. Real-time updates of the NavMesh accommodate environmental changes, such as moving agents, ensuring adaptability and robustness in dynamic game simulations. Its effectiveness spans diverse gaming scenarios, from small-scale indie games to large-scale AAA titles.

## Behavior Trees

Behavior trees provide a structured and modular approach to defining complex agent behaviors, widely used in game AI and crowd simulations due to their flexibility and scalability. They break down behaviors into smaller tasks, allowing for reusable and adaptable modules, thus enhancing development flexibility and efficiency. Behavior trees effectively manage decision-making processes, enabling agents to react appropriately to environmental changes and the actions of other agents. Their ability to handle many agents with diverse behaviors makes them ideal for scalable crowd simulations in games.

## Integration of NavMesh, and Behavior Trees

Combining Unity's NavMesh and behavior trees creates a robust framework for advanced crowd simulation in games. The synergy of these tools allows simulations to achieve high levels of realism in agent movement and behavior. Agents can navigate intricate environments, avoid collisions, and exhibit sophisticated behaviors, thereby enhancing the overall gaming experience. Integrating Unity’s NavMesh with behavior trees optimizes simulation performance, ensuring efficient operation even with numerous agents. This is critical for maintaining smooth gameplay in high-density crowd scenarios.

# Methodology

This section details the methods and tools used to implement and evaluate the crowd simulation system.

## Overview of Techniques and Tools

This research employs Unity's NavMesh pathfinding system due to its efficiency and flexibility, making it an industry standard for AI navigation in complex environments. The primary tool for this research is Unity, with custom scripts in C# for implementing behavior trees. Visual Studio was used for scripting, and no additional tools or libraries were involved.

## Handling Crowds and Avoidance

In crowded environments, local avoidance algorithms are crucial to prevent collisions. Unity's NavMeshAgents are equipped with built-in local avoidance, constantly checking their surroundings and adjusting paths to avoid collisions. Agent priorities were configured to manage scenarios where specific agents need to push through crowds, ensuring these key interactions are not hindered by congestion.

## Environment and Agent Behavior Design

The Demo's virtual park environment includes five key locations: a water fountain, a café, a restroom, a playground, and a flower garden. This setup provides diverse interaction points for Agents, allowing the simulation of various behaviors and interactions.

A screenshot of a video game

Description automatically generated

Figure 1: Demo Environment – Virtual Park

Custom behavior trees were designed for two different Agent roles: Tourists and the Ice Cream Man (ICM). The Tourist Agent's behavior tree includes sequences for changing emotions, moving to various locations within the park, and pausing for a random duration. When Tourists come into close proximity to the ICM, they are attracted to him and will move towards his location. The ICM follows a simpler behavior tree focused on moving to specific locations and staying there for some time. This dynamic interaction causes Tourists to frequently adjust their paths, creating a realistic and engaging simulation of crowd behavior.

A screenshot of a computer program

Description automatically generated

Figure 2: Behavior Tree for Tourist

A group of black text with orange circles

Description automatically generated

Figure 3: Behavior Tree for Ice Cream Man

## Dynamic Interaction and Pathfinding

Unity's NavMesh system enables Agents to navigate around other dynamic entities in real-time, ensuring smooth and efficient path adjustments. In the dynamic environment, Agents constantly recalibrate their routes in response to moving entities.

For Collision and Avoidance, this involves Agents pausing or steering to avoid collisions with other dynamic entities, maintaining fluid movement and interaction within the environment.

Additionally, behavior trees dictate how Agents resolve competing goals through parallel sequencing, allowing them to adapt their behaviors dynamically based on real-time conditions.

This approach ensures that Agents can adjust their paths seamlessly, preventing bottlenecks and maintaining a realistic simulation of crowd behavior.

## Demo Implementation and Evaluation

The demo showcases a park populated by hundreds of Agents with various goals, such as visiting the water fountain, relaxing at the café, using the restroom, playing at the playground, or strolling through the flower garden. The environment includes dynamic entities to simulate a real-world setting.

To simulate real-life scenarios, Agents change their mood based on their location. While navigating the park, they display an "Idle" mood with a blue face. Upon reaching any location other than the restroom, they change to a "Happy" mood with a yellow smiley face. At the restroom, they show a "Disgusted" mood with a green face. If they are unable to reach any location after a prolonged period due to overcrowding, such as when too many agents are attracted to the ICM, they change to an "Angry" mood with an orange face.

The system was tested with varying densities, maintaining smooth navigation and effective collision avoidance even in high-density scenarios. A limit of 100 Agents was chosen to create a sufficiently large crowd while avoiding excessive overcrowding that could obscure the demo's clarity.

Evaluation criteria included Agent interaction, avoidance, and their respective mood changes. The demo highlighted the system's ability to handle high-density crowds while maintaining realistic and believable Agent behaviors. The use of Agent priority settings was particularly effective in managing crowd congestion and ensuring key Agents, like the ICM, could move through crowds as intended.

In the following section, we will present the results and key takeaways from our simulation, discussing performance metrics, observed behaviors, and the overall effectiveness of the implemented systems.

# Results and Key Takeaways

## Overview of the Simulation

Our demo simulation utilizes Unity’s NavMesh and behavior trees to model the behavior of pathfinding entities in a dynamic virtual park and aimed to explore how pathfinding entities interact with each other. By observing the real-time navigation and interactions amongst these entities, we sought to gain insights into their adaptive behaviors and the emergent dynamics of their collective movement.

## Crowd Dynamics and Adaptability

A key observation from the simulation was the fluid movement of the crowd. Tourists continually adjusted their paths to avoid collisions with one another and with the ICM. These real-time adjustments demonstrated the adaptability of the pathfinding entities. They were able to navigate smoothly around other moving agents, maintaining a natural flow despite the dynamic and crowded environment. This adaptability was crucial in preventing bottlenecks and ensuring a realistic simulation of crowd behavior.

## Collision Avoidance Strategies

The entities employed various strategies to avoid collisions. These included pausing momentarily, altering their speed, and steering to create space between themselves and other agents. The real-time recalculations of paths allowed the agents to dynamically adjust their movements, displaying the collision avoidance techniques used my Unity’s NavMesh that contributed to the overall fluidity and realism of the crowd movement.

## Role of Interest-Driven Behavior

The introduction of the ICM provided a focal point for interest-driven behavior among the Tourists. As the ICM appeared, Tourists altered their predefined paths to approach him, demonstrating how environmental stimuli can influence agent actions. This interaction illustrated the dynamic nature of pathfinding entities and their ability to prioritize new goals based on changing environmental conditions. The ICM attracted Tourists, causing localized increases in density around him, which in turn required the agents to adjust their paths more frequently to avoid collisions with one another.

## Emotional and Adaptive Responses

The simulation also highlighted emotional responses among the agents. Tourists who were unable to reach the ICM exhibited frustration, adding a layer of emotional realism to the simulation. This emotional response influenced their subsequent behavior, as some Tourists would abandon their attempts to reach the ICM and resume their original paths, while others persisted in trying to find a way through the crowd. This diversity in responses enriched the simulation, providing a more nuanced understanding of how pathfinding entities react to obstacles and dynamic changes in their environment.

## Collective Movement and Interaction Patterns

Observing the crowd as a whole, we noticed emergent patterns of collective movement. The Tourists exhibited a natural flow as they navigated the park, constantly recalculating their routes to avoid collisions. The fluidity of the crowd was maintained through a combination of individual pathfinding decisions and collective behavior, resulting in a dynamic yet harmonious movement throughout the environment.

## Key Takeaway

The results from our virtual park simulation provide valuable insights into how pathfinding entities interact with each other in a dynamic environment. The adaptive behaviors, collision avoidance strategies, and interest-driven actions observed in the simulation highlight the complexity and realism that can be achieved in modeling crowd dynamics. These findings emphasize the importance of understanding the interactions among pathfinding entities, which is crucial for advancing the field of crowd simulation and developing more sophisticated and realistic models of agent behavior.

# Future Directions in Crowd Simulation

## Enhanced Realism and Efficiency

Unity’s NavMesh excels at efficient pathfinding, offering flexibility to adapt to various scenarios. However, its limitations in handling nuanced interactions between agents necessitate the integration of behavior trees. Behavior trees provide the capability to model complex decision-making processes, resulting in more realistic and varied agent behaviors. By combining NavMesh with behavior trees, we achieve simulations that are not only efficient but also believable. Agents navigate effectively, avoid collisions intelligently, and interact meaningfully, creating a dynamic and engaging virtual environment.

## Deepening Interaction Complexity

The next step is to build on what we've learned and dive deeper into how pathfinding entities interact. While the demo showed that agents could avoid each other and navigate around obstacles, there's room for more sophisticated interaction modeling. Future work should focus on developing more advanced behavior trees that simulate a wider range of social interactions and decision-making processes. For example, we could explore how agents cooperate or compete within groups, which would lead to more nuanced and realistic simulations.

## Exploring New Techniques

Building on the strengths of Unity's NavMesh and behavior trees, we can integrate new techniques such as advanced steering behaviors and emergent behavior modeling to address some of the limitations observed in the demo. Implementing systems that simulate collective behavior or adapt dynamically to changes in the environment could add more depth to how agents interact. Experimenting with these new approaches will help create more immersive and responsive crowd simulations.

## Advanced Behavioral Models

As we look ahead, developing more advanced behavioral models will be crucial. Games like "Left 4 Dead" showcase realistic panic reactions that enhance gameplay immersion. By incorporating similar advanced behaviors, we can create simulations where agents respond authentically to various stimuli. This would significantly boost the realism and depth of virtual environments, making them more engaging and lifelike.

# Conclusion

## Summary of the Research and Key Findings

In this study, we explored the interactions between pathfinding entities within a simulated environment using Unity's NavMesh and behavior trees. The primary goal was to observe how these entities navigate and interact with one another, highlighting the nuances of their behavior in a dynamic virtual setting. By integrating NavMesh for efficient pathfinding and behavior trees for complex decision-making, we aimed to create a simulation that reflects real-world crowd dynamics.

## Key Findings

Our observations revealed several key insights into how pathfinding entities react to each other. The combination of NavMesh and behavior trees enabled agents to navigate around obstacles and other entities with impressive fluidity. Agents used a mix of pausing, steering, and real-time decision-making to avoid collisions, demonstrating an advanced level of interaction and adaptability.

The integration of behavior trees allowed us to model a range of behaviors, from cooperative interactions to conflict resolution. This capability provided a deeper understanding of how entities can engage with one another in a simulated environment, leading to more realistic and believable crowd behaviors.

The demo showcased how even simple AI, when combined with sophisticated tools, can generate complex and lifelike interactions. The ability to simulate nuanced behaviors, such as emotional responses and group dynamics, highlights the effectiveness of using NavMesh and behavior trees together.

## Concluding Thoughts

The research has shown the value of combining Unity’s NavMesh with behaviour trees for creating realistic and engaging simulations of pathfinding entities. By observing how agents interact with one another, we have gained insights into the dynamics of crowd behaviour and the effectiveness of these tools in modeling complex interactions.

Moving forward, there is significant potential to build on these findings by exploring more advanced interaction models and integrating new techniques. The insights gained from this research not only enhance our understanding of virtual crowd dynamics but also pave the way for future advancements in our understanding of AI and pathfinding.