Experiment

To test their Theory, they implemented ALAN Framework in **C++**and tested its performance across a variety of simulation scenarios using pure greedy, e-greedy, wUCB and their proposed context-aware algorithm as action selection techniques. ​

In all experiments they followed these rules: ​​

* The maximum speed of vMax, for each agent was set to 1.5 milli-seconds ​​
* Maximum Radius was set to 0.5m​​
* Obstacles within were 1 meters apart
* Fixed value of 0.1 for **B** value.
* The values of**y** and **T** for the context aware algorithm were determined by observing the performance of context aware agents with different values of**y.**​
* For e-greedy and wUCB, the best **e** and **t** values used respectively, typically between 0.1 or 0.2 for **e**and 50 for **T**. ​

NEXT SLIDE

Simulated Scenarios

They simulated 9 scenarios with the context-aware algorithm.

The first picture is 1-Exit and in this, 48 agents must travel through the room, exit the room through a small doorway, and then traverse through an open area to reach their goal​.

----------------------------------

And then 3-Exit, similar to 1-Exit, the room has 3 exits.

----------------------------------

Second picture is congested in which 32 agents are placed very close to the narrow exit of an open hallway and must escape the hallway through this exit

----------------------------------

NEXT SLIDE

Simulated Scenarios

Then there is Crossing, where a single agent interacts with a group of 10 agents moving in the opposite direction

Then 2-Sides which is Similar to Congested, but there are two groups of 20 agents, each on the opposite side of the narrow exit

NEXT SLIDE

Simulated Scenarios

Perpendicular Crossing where two agents have ​ intersecting paths with a crowd of 24 agents

----------------------------------

Intersection where 3 agents cross paths while moving​ towards their goals

NEXT SLIDE

Simulated Scenarios

Deadlock where ten agents start at opposite sides of a long, narrow corridor and only one agent can fit in the narrow​ space at a time.

----------------------------------

AND Final simulated scenario called Blocks where five agents need to avoid 3 static obstacles in​ front of them to reach their goals​

NEXT SLIDE

Comparison Metrics​

To measure the performance of ALAN in each scenario, they evaluated**R1 of the trajectories**generated by each method.

----------------------------------

This is a graph showing the performance of ALAN measured in 1-Exit and 3 Exit and r1 trajectories of each method.

NEXT SLIDE

Comparison Metrics​

To analyze the effect that exploration has on different action selection techniques within the ALAN framework, they evaluated each simulation in terms of the**average agent acceleration of per agent**

----------------------------------

This is a table showing Average acceleration per agent in ms-2, generated by each method

NEXT SLIDE

Comparison Metrics​

For individual parameters, T and y, the **arrival time of the last agent** in a simulation was evaluated, ​ denoted as **maxTravelTime.**

----------------------------------

The first graph shows the travel time of context-aware agents, with different values of y.

----------------------------------

The second graph shows the travel time for context-aware agents with different values T.

NEXT SLIDE

Results – Average of 100 Simulations

They ran 100 simulations, and these are the results that were found.

---------------------------------

The simulations provided a set of 5 velocity choices which gave the best performance. ​

​

This provided agents enough variety of behaviors while avoiding spending too much time in exploration. Specifically, the actions defined correspond to: ​

1. Moving straight towards the goal​

2. Left 45°​

3. Right 45°​

4. Backwards, all at a speed of vMax​

5. A complete stop​

----------------------------------

From congested scenario they found out that context-aware significantly faster, because agents started in a constrained configuration, hence they couldn't take advantage of the goal-oriented action until the very last part of the simulation. Therefore, in congested scenario, implicit coordination is achieved by using politeness component of the reward.

----------------------------------

In Blocks scenario, pure greedy motion caused some agents to get stuck behind static obstacles and they could never reach their goals without using a roadmap. Agents using ALAN learned that sideways motions were better and reached their goals avoiding the obstacles.

----------------------------------

In Deadlock scenario, agents faced a deadlock situation inside the corridor, which ORCA couldn’t solve. Instead, context aware agents were able to backtrack and quickly reach their goals.

---------------------------------

For scalability, the number of agents were increased in 3-Exit and 1-Exit scenarios to analyze the performance. The context aware approach maintained a near-linear growth in travel time, while greedy agents showed a super-linear growth in travel time. Greedy approach also added extra constraints to the system as the agents were increased, which significantly slowed down the global motion of the crowd.

NEXT SLIDE

Results – Average of 100 Simulations ​

This graph represents performance comparison between the 4 approaches. In all scenarios, the context-aware agents have the lowest R1.

---------------------------------

E-greedy and wUCB have higher R1 than ORCA because of unnecessary exploration making the agents choose different actions often between learning cycles which results in a repetitive behavior.

---------------------------------

ORCA agents have the lowest R1, because of the selection of the static-goal orientated Vpref which leads to locally efficient motions at the expense of large travel times. ​

---------------------------------

The travel time produced by the context-aware approach is close to the optimal value (R1 = 0) in crossing, perpendicular crossing, intersection and blocks. This means that the generated paths are as efficient as those could have been computed by an optimal centralized planner. ​

​

​ NEXT SLIDE

Conclusion

​

​

In conclusion, ALAN framework for agent navigation formulates the problem of time-efficient decentralized navigation of multiple agents as a multi-armed bandit problem, where the action space corresponds to a set of preferred velocities.

ALAN evaluates their actions through a reward function that encourages goal-orientated motion and polite behavior. To balance the exploration versus exploitation and to improved navigation efficiency, context aware selection technique was used. Ii has incorporated 'win-stay lose-shift' rule, making its agents exhibit low accelerations and fast travel times. It exabits motions which are diverse using ALAN framework allowing the agents to reach their goals by learning sideways motions are better, which helps to avoid obstacles, when ORCA and wUCB motion fails to. Context-aware agents can backtrack and quickly reach their goals, introducing a fixed burden to the travel time as the number of agents increases, it does not slow down the motion of all other agents. This makes it ideal for time-efficient navigation.   ​

​

​

FINISH