A Hitchhiker's Guide To Robot Navigation: People-Powered Path Planning

Anandi Arora[1], Sebastian Helgeson[1], Akshunn Trivedi[1]
[1]Department of Robotics, University of Michigan, Ann Arbor, MI, USA
{anandiar, sebhelge, akshunn}@umich.edu

***Abstract*—One of the biggest challenges for widespread use of robots is difficulty in navigating unstructured and dynamic environments such as crowds. Interestingly, humans are expert agents in this setting, being able to effortlessly navigate through highly crowded and unstructured environments regularly. Liao et. al. [1] developed a strategy of utilizing human's innate ability to traverse crowds by having robots follow select humans in a crowd as a form of navigation. This Leader-Follower algorithm offloads robot navigation computing to humans and potentially produces robots that better follow social navigation rules in crowds. Our work aims to integrate the Leader-Follower algorithm proposed by Liao et. al. into the Social Force Model (SFM) based crowd simulator developed in our past assignments to produce a system with the pedestrian planning from SFM but with added robot navigation capabilities from Leader-Follower. We plan to test our method on the scenarios developed by Liao et. al. in their paper (i.e. promenade, crossing, roundabout), but also plan to further extend the algorithm onto three less-ideal pedestrian-centric scenarios (i.e., convention, museum, Black Friday sale). Finally, we compare the performance of this approach against a tuned Model Predictive Control (MPC) based planner to identify the strengths and weaknesses of each algorithm. This comparison uses four standard quantitative measurements, two newly developed quantitative measurements and qualitative observations. Code is available at People-Powered-Path-Planning**

I. INTRODUCTION

One of the major challenges in deploying robots for navigation in unstructured environments is the difficulty of effectively managing crowd interactions. With numerous methods available for crowd navigation, selecting the most suitable approach can be challenging. Interestingly, humans excel at navigating crowds successfully, which inspired our approach to delegate the complex decision-making processes involved in crowd navigation to the humans within the crowd, allowing the robot to follow a person rather than make difficult navigational decisions itself.

Our inspiration was drawn from the work of Liao et al. [1], who employed a human-as-planner approach. They selected humans as guides and applied a subgoal selection process, utilizing a simple path planner based on the Social Force Model to reach each designated subgoal. The claimed advantages of this approach include its simplicity given the use of the Social Force Model (SFM), minimal training requirements, and lower computational demands both offline and online. Their methodology resulted in trajectories that are socially compliant, collision-free, faster, and more efficient.

Liao et. al. tested their approach across three scenarios: promenade, crossings, and roundabouts. Their experiments, conducted both in simulation and in real-world environments, demonstrated that their framework generated safe and efficient robot navigation plans, outperforming existing planners even without predictive or data-driven modules. However, their evaluation of the Leader-Follower algorithm in simulations had limitations, such as reliance on pre-recorded human trajectories (thus, non-reactive humans), a focus solely on road-centric environments with static obstacles, and no testing with aggressive road users. In addition, their real-world experiment lacked of quantitative performance evaluation in real-world scenarios.

In our work, we address these limitations by recreating their three environments using the Social Force Model to simulate reactive humans, and we include static obstacles such as walls and shelves. Additionally, we introduce an aggressive environment with one hundred intermoving pedestrians and employed four existing performance metrics, along with two newly developed metrics to compare the Leader Follower algorithm to an MPC-based planner.

To evaluate the leader-follower approach comprehensively, we modify the SFM-based class simulation assignments to accommodate multiple goals. We also develop a workflow capable of generating complex simulations involving up to 100 individuals. We reconstruct Liao et. al.'s three original simulations with the SFM and develop three additional indoor environments, integrating the leader-follower algorithm into the simulation framework. We calibrate the Model Predictive Control (MPC) parameters, and compare the performance of MPC-based planner against the leader-follower approach using four established metrics, and our two new quantitative metrics.

In terms of simulation development, pedestrian agents were endowed with multiple goals, capable of waiting at these goals for specified durations. For creating obstacle and human path layouts, we utilize Blender to design the environments, then convert Blender meshes and curves to simulation-compatible code via Python scripts, which we subsequently execute within the simulation framework. We designed scenarios to evaluate how the Leader-Follower method adapts to group changes and potential navigation around clusters. Additional environment complexities included the addition of walls and obstacles, with pedestrians programmed to stop and start at specific points, such as in the museum scenario.

The Leader-Follower algorithm integrates into our simulation by selecting a leader based on movement direction, speed, and position, and then designates a nearby subgoal for the leader to reach using SFM. The follower then tracks this leader, with the process repeating each frame of the simulation. We compare this algorithm with a fine-tuned MPC using two new metrics of path roughness and personal space invasion. Our analysis identifies some limitations, which we have outlined below.

Our contributions are threefold. First, we recreated Liao et. al.'s three benchmark crowd-navigation environments with our addition of reactive humans using Social Force Model (SFM) in a simulator, and then we introduced three new pedestrian-centric environments (convention, museum, and Black Friday). Secondly, we designed two additional evaluation metrics (Path Roughness, and Invasion of Personal Space) to better analyze the robot's trajectory and it's social norm compliance. Finally, we conducted exhaustive evaluations between Leader-Follower and a MPC based planner used our improved evaluation suite, and provided an analysis of the resulting behaviours, highlighting their strengths and weaknesses.

II. RELATED WORK

A. Crowd Simulation and Modeling

When testing robot crowd navigation algorithms in simulated environments, it is crucial that simulated crowds closely mirror real-world conditions. Social Force Models (SFMs) remain foundational for simulating pedestrian dynamics, treating pedestrian behavior as subject to attractive goal forces, repulsive obstacle forces, and inter-agent social forces [2]. These models continue as the standard for crowd simulation and have been thoroughly validated for their accuracy [3]. The fundamental principle is that pedestrians act as if subject to social forces including goal location attraction, position maintenance, desire to stay away from other pedestrians and static obstacles, and attractive forces that keep groups together or draw people to interesting objects.

B. Social Robot Navigation

Robot crowd navigation continues to be a significant barrier when integrating robots into social roles. [4] identifies two core challenges: decoupled versus coupled prediction and planning. In decoupled approaches, the system estimates how humans will move and plans motion to avoid intersecting expected paths. In coupled approaches, the system entangles human motion and robot motion by recognizing that robot motion affects human behavior. Expanding on coupled motion planning, [5] presents SICNav, which jointly solves robot motion and predicted crowd motion in closed loop using KKT reformation [6, 7].

A common problem is the robot freezing in dynamic environments. [8] formalizes the Freezing Robot Problem (FRP) and proposes Interacting Gaussian Processes (IGP) to model cooperative collision avoidance. [9] addresses this through CoMet, a group cohesion metric using weighted scores to identify and avoid freezing groups. [10] examines post-contact safety with reactive control methods, showing that RDS controllers achieve safer interaction distances with collision forces below safety thresholds.

Social conventions and human behavior play crucial roles in navigation. [11] incorporates personality trait theory, categorizing humans and adapting navigation accordingly, achieving 74.3% accuracy in low-density scenarios. [12] uses topological methods, decomposing obstacles into discrete homology classes and training an LSTM to predict human-preferred paths, outperforming handcrafted rules by 20%. [13] uses social conventions as mathematical cost functions, with user studies showing people prefer robots that follow social norms like respecting personal space and passing on the right. [14, 15] demonstrate that context-aware systems modeling crowd density, group formations, and predicting interaction outcomes enable smoother, safer navigation.

Learning-based approaches have shown significant promise. [16] proposes a crowd-aware DRL framework with attention mechanisms focusing on nearby humans. [17, 18] develop DRL policies that learn to behave as social actors, balancing collision avoidance, personal-space comfort, and efficiency. [19] uses Gaussian Mixture Models to learn motion prototypes, while [20] employs non-linear multi-objective optimization for social objectives. [21] presents a hybrid framework combining machine learning with Social Force Models, producing smoother paths than either method alone.

Non-verbal communication enhances navigation quality. [22] shows that gaze-control modules aligning robot head orientation with navigation cues significantly increase social presence and human comfort. [23] demonstrates that gaze and head orientation improve prediction of human intentions like turning or yielding, enabling robots to anticipate movement earlier.

C. Human-Robot Collaboration in Planning

[24] explores using humans as sensors in robotic search, concluding that humans should act as information fusion machines while robots handle planning and control. [25] provides a comprehensive survey demonstrating that incorporating social and psychological factors into motion planning helps robots navigate safely and comfortably. Key requirements include safety, legibility, predictability, and politeness, with open challenges in understanding human behavior, handling crowds, and evaluating social navigation.

D. Leader-Following Navigation

Leader-Follower navigation differs from traditional approaches by offloading difficult decision-making onto people in the crowd. Early work by [26] used GHMM predictors to identify goal points and select appropriate leaders, employing RiskRRT for path following with Social Force Model simulations. [27] developed a teach-and-repeat approach using CNN and LiDAR fusion to identify and track leaders, then smooth and repeat their paths. [1] demonstrates that following humans is an effective strategy in populated spaces.

[28] extends leader-following to any object type (robots, humans, objects), addressing failure modes when leaders move out of view. Using SAM2 for object identification and distance frame buffers for re-identification after occlusion, the

system implements five goal-aware adaptation mechanisms: Following, Chasing, Retreating, Planning, and Switching. [29] tackles view-loss through multimodal sensor fusion—RGB-D cameras with YOLO-v8, LIDAR, and custom angle-of-arrival sensors—achieving robust 50 Hz navigation using Riemannian Motion Policies with EKF tracking.

Specialized following behaviors include [30]'s "group surfing," where robots imitate optimal pedestrian groups for sidewalk navigation, and [31]'s frontal following using LSTM to predict human actions and A2C to learn optimal relative poses. [32] provides a categorical overview organizing person-following into Perception, Planning, Control, and Interaction, identifying operational challenges including sensor limitations in challenging environments like underwater settings.

III. TECHNICAL APPROACH

A. Problem Definition and Method

We use the same problem formulation and method introduced by Liao et. al. for Leader Follower (LF) algorithm and provide a very brief overview here.

The state of the agent consists of its position and velocity $x = [p^\top, v^\top]^\top \in R^4$. H_i and R represent human and robot, and t timesteps. A unique ID H_i where $i \in \{1, \ldots, n\}$ is assigned to n humans, where n are total number of humans in our simulation. The robot's goal position is defined as $p_g \in R^2$, while $X_H^{0:t}$ represents the past trajectories of all humans. In a crowd navigation setting, the robot plan is generated by a planner $\mathcal{P}$, denoted as

$$x_R^{t+1} \leftarrow \mathcal{P}(x_R^t, x_H^t, \mathcal{O}, p_g). \tag{1}$$

They first identify a human leader H_L^t and subsequently define a subgoal $p_g^{t,s}$:

$$H_L^t = f_{\text{leader}}(x_R^t, X_H^{0:t}, \mathcal{O}, p_g), \tag{2}$$

$$p_g^{t,s} = f_{\text{subgoal}}(x_{H_L}^t, x_H^t). \tag{3}$$

Robot plan is then generated by base planner $\mathcal{P}$:

$$x_R^{t+1} \leftarrow \mathcal{P}(x_R^t, x_H^t, \mathcal{O}, p_g^{t,s}), \tag{4}$$

and the process repeats till the time robot reaches the goal.

LF consists of two major components: 1) Leader selection, which completes group identification and ensures reachability and 2) Subgoal Selection which defines a subgoal near the selected human leader and then uses a planner like Social Force Model (SFM) to reach there.

B. Pedestrian Environment Creation

In addition to the three environments proposed by Liao et. al, We develop three more environments (Convention, Museum, Black Friday) (see Fig. 1) to simulate more diverse and challenging social navigation scenarios to benchmark our LF and MPC based planner. We create these environments with the simulator introduced in the University of Michigan Computational HRI course as it matches with the state requirements for LF, and enables us to evaluate on parameters like Avg. Time to Goal (Tg), Avg. Min Human Distance, etc.

Firstly we introduce Convention with humans exhibiting clustering and de-clustering motion patterns in a circular area, and robot's goal position straight across this circular area. The idea behind this environment is to test if our algorithms can identify this pattern and plan an alternative path around the circumference of the circular area, avoiding the human crowd and getting stuck. We are particularly interested in the performance of the LF algorithm, as its human-centric planning means it may get stuck in these human crowds and facing trouble planning efficiently.

Secondly, we add a Museum scenario where the robot's goal is to navigate in an environment with many walls forcing a complex successful path with humans grouping and stopping in between their paths, akin to a very real museum.

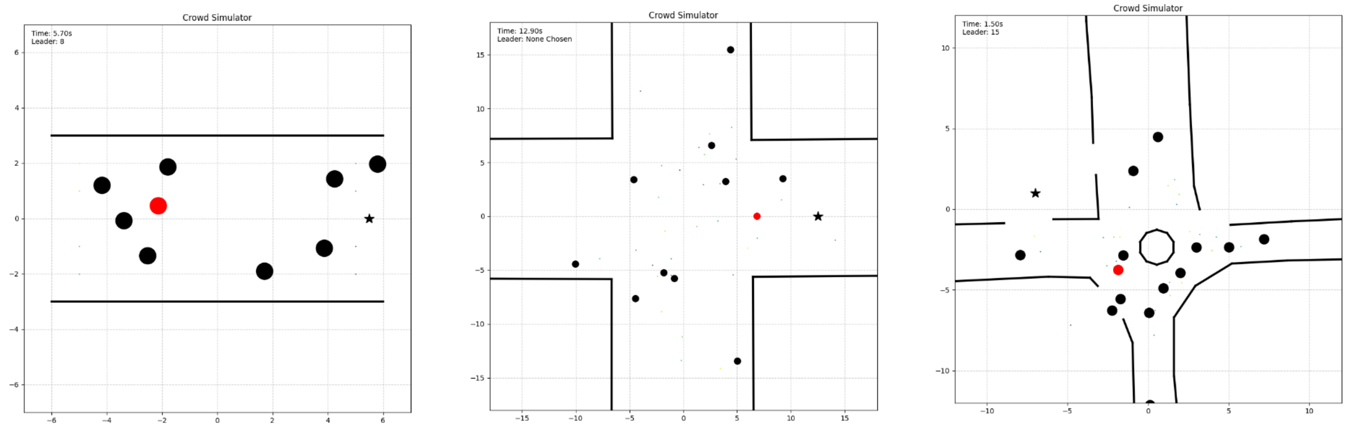
Finally, we develop a Black Friday scenario, with the intention of testing our algorithm's performance against a highly aggressive human environment. The scenario includes shelves as obstacles with humans moving in and out in an aggressive fashion, which we implemented by increasing their Goal seeking weight. This was specially interesting for LF algorithm as the authors mentioned its performance in presence of aggressive humans as one of its limitations.

To recreate the scenarios by Liao et. al. and these three additional scenarios, we add additional features on top of the base simulator. One of these additions is the capability to add multiple agent goals with wait times for each of the goals, so our humans can go to multiple goals and wait for specified time before embarking for their next goal. This feature enables us to fairly mimic situations like Museum and Black Friday, where humans go to multiple destinations with different wait times.

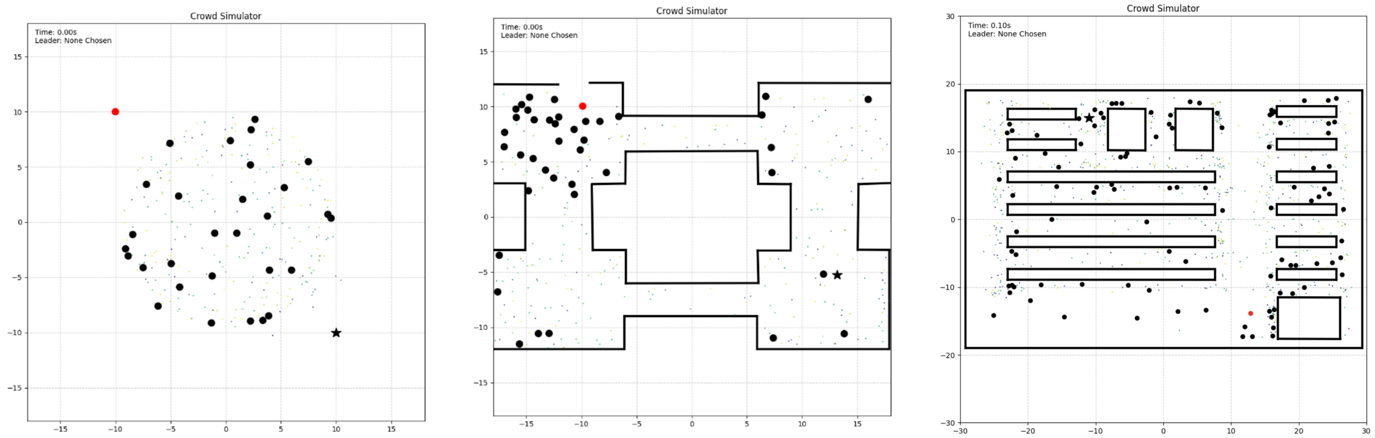
We use Blender [33] to create human trajectories and obstacles (see Fig. 2 for our simulated environments. The process includes drawing human trajectories as curve objects and environment obstacles as mesh objects. The human trajectories were divided by multiple sub-goals to mimic natural socially compliant behavior (like crossing a roundabout counterclockwise). A key point was to make sure to have enough subgoals to navigate around obstacles but few enough to still be able to use SFM to dynamically alter their paths if required. Finally, we convert these objects mesh information to euclidean coordinates compatible for our simulator. This entire workflow helps us build extremely complex social navigation scenarios like Black Friday with multiple obstacles and a hundred humans exhibiting social trajectories.

C. Baseline

We use a Model Predictive Control (MPC) based planner tuned to respect social navigation cues as our baseline. One of the major reason is because Liao et. al. also uses a sampling based MPC, Pred2Nav [34] as their baseline for comparison,



(a) Promenade (b) Crossing (c) Roundabout



(d) Convention (e) Museum (f) Black Friday

Fig. 1: Six benchmark environments for social navigation.The robot is in red and the robot goal is a black star

and we are interested for the reproduction of at least the same trend in the results for our scenarios.

Our MPC planner learned social compliance by respecting the following social navigation parameters:

- **Path horizon**: How much in the future the planner looks?
- **Goal weight**: How strongly the planner prioritizes reaching the goal?
- **Static obstacle weight**: How strictly static obstacles are avoided?
- **Dynamic obstacle weight**: How strictly dynamic obstacles (humans) are avoided?
- **Personal space weight**: How much the planner penalizes violating human personal space?

We perform a grid-based brute-force search through possible weights for each of these parameters. The final weights chosen represent the combination that maximized the reward function function J, defined as:

$$\begin{aligned} J = 60 &\left(\frac{\text{num of successful runs}}{\text{total num of runs}} \right) \\ &+ 20 \left(1 - \frac{\text{time to goal}}{\text{max time}} \right) \\ &+ 20 \cdot \min \left(\frac{\text{min human distance}}{\text{social force radius}}, 1 \right) \end{aligned} \tag{5}$$

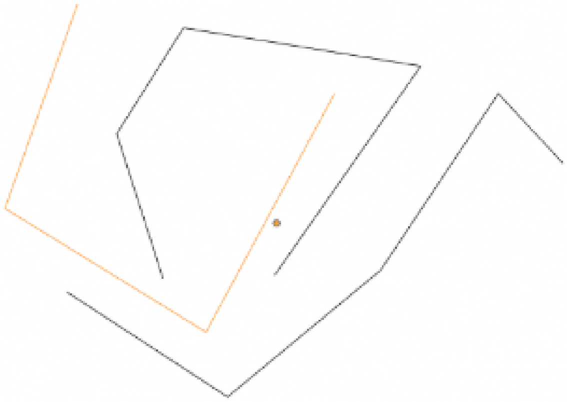
where social force radius was the radius of the circle under which human agents exhibited social behaviors.

The Museum environment is used for tuning, owing to its complexity for social navigation, with each combination of weight values evaluated with five runs of the simulation to obtain the performance metrics. We finally use the following MPC weights **(horizon range = 2, static weight=1, dynamic weight= 2, goal weight=41, personal radius weight=0)** for all MPC evaluations below.

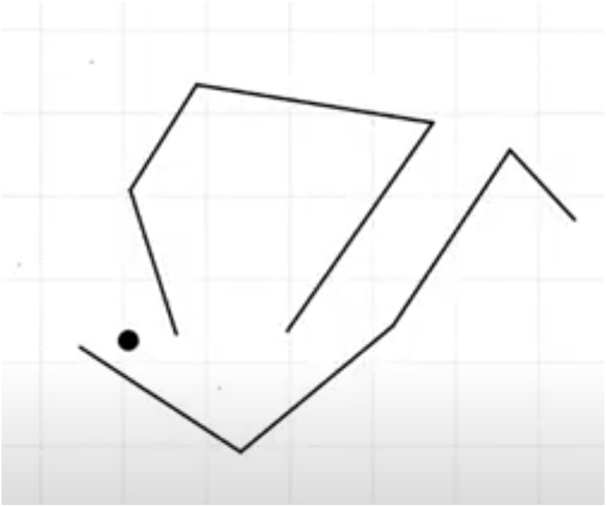
IV. EXPERIMENTS AND RESULTS

A. *Simulation Settings*

Scenarios: We performed simulation experiments on the original three social navigation scenarios from the Liao et.



(a) Hand-drawn obstacles and trajectories in Blender



(b) Rendered trajectories in our simulator

Fig. 2: Hand drawn obstacles (in black) and human trajectories (in orange) rendered from Blender to our simulator

al.'s work (Promenade, Crossing, and Roundabout) with our three additional pedestrian-centric scenarios (see Fig. 1).

1) Promenade has a single road, with the robot to moving along the road, 2) Crossing has a junction with people moving in all the four directions, and 3) Roundabout involves a street crossing on a circular roundabout. All of these scenarios were directly simulated from the Stanford Drone Dataset(SDD)[35].

Implementation Details: As mentioned earlier, we used the simulator introduced through the University of Michigan Computational HRI course for all of our experiments. We used a single robot with preferred velocity equal to 1.5 and used same radius circles to represent robot and humans. The humans had variable speeds depending on the simulations to give a more realistic benchmark. We used the LF parameters ”out of the box” from the Liao et. al. without any finetuning. Our baseline was the MPC based planner with tuned parameters as mentioned before. We ran and watched all of the simulations with these parameters three times each to make sure our results do not have discrepancies or ”hack” the rewards in any way. We then completed 100 simulations for each evaluation and took the average to report our quantitative results.

B. *Quantitative Measurement*

The four standard quantitative measurements for evaluating robot path performance are completion rate, time to goal, number of collisions, and minimum human distance.

The robot simulator finishes on either of two conditions: the robot successfully reaches the goal, or the maximum number of timesteps is reached. The maximum number of timesteps is determined for each simulation as at least 500 steps more than what would be required to optimally reach the goal location. This was to give the robot more than enough time to reach the goal while saving on unnecessary computation time. If the simulation ends early, that run is counted as a successful completion, if the simulation runs for the maximum number of timesteps, it is counted as a loss. The percentage of successful completions the robot does in 100 simulations determines the completion rate of the planner.

Of the scenarios where the robot successfully completes the simulation, the average time the simulation runs for is taken. This determines the time it took for the robot to reach the goal on an average successful run.

In each simulation, if the robot collides with a human, a flag is raised. The number of simulations that had a collision in them compared to the total number of simulations determines the collision rate.

After the simulation, the robot's position is compared to the positions of every human in the simulation at each timestep. The distance between the robot from each human is taken and the minimum distance across all timesteps determines the minimum human distance. The average of these minimums for each simulation is taken.

1) Path Roughness Measurement: The path roughness measurement operates as a replacement to a standard path efficiency measurement. In a path efficiency measurement, the distance the robot travels is compared to the optimal straight-line distance the robot could have traveled to reach the goal. Because many of our simulation environments contain obstacles that require the robot to deviate from straight line movement, this measurement would not be effective in analysis. While the distance the robot traveled (independent of the straight line distance) could be a useful measurement in measuring path efficiency, we chose to not use this measurement as to not penalize robots for taking longer routes, but potentially following social cues. This path distance would have encouraged beeline behavior.

Instead, path roughness penalizes robots for creating jittery paths through a moving window approach. At each timestep, the position of the robot is taken and the displacement vector between the robot's current position and its position ten steps in the future is calculated. In addition, the displacement vector between the robot's current position and it's position five steps in the future is also calculated. At each timestep, the angle between this long-term displacement vector and this short-term displacement vector is taken. This angle difference is aggregated for each timestep in the simulation, then averaged across all windows in the simulation. This causes robots that consistently wobble or make short turns to be penalized while robots with straight movements and large curves to be rewarded. The pseudocode for this computation is shown in Algorithm 1.

Because many of the scenarios require turns, a baseline path roughness is taken. We draw an optimal path for the robot and compute the path roughness of that path. We then subtract this baseline from the path roughness scores for each simulation to effectively normalize the results.

Algorithm 1 Path Roughness Computation

```
Total_Rotations ← 0
**for** each timestep $t$ **do**
    $\mathbf{d}^L \leftarrow \frac{\mathbf{x}_{t+10} - \mathbf{x}_t}{\|\mathbf{x}_{t+10} - \mathbf{x}_t\|}$
    $\mathbf{d}^S \leftarrow \frac{\mathbf{x}_{t+5} - \mathbf{x}_t}{\|\mathbf{x}_{t+5} - \mathbf{x}_t\|}$
    $\theta^L \leftarrow \arctan 2(d^L_y,\ d^L_x)$
    $\theta^S \leftarrow \arctan 2(d^S_y,\ d^S_x)$
    $R \leftarrow |\theta^L - \theta^S|$
    Total_Rotations ← Total_Rotations + $R$
**end for**
Path_Roughness ← $\frac{\text{Total\_Rotations}}{\text{num\_timesteps} - 10}$
```

2) Invasion of Personal Space Score: The Invasion of Personal Space Score (IPSS) seeks to provide more information on how the robot performs around humans. Ideally, the robot's presence does not interfere with the movement of the people around it. If a person must shift their movement to accommodate a robot's path, that path should be penalized.

At each timestep the IPSS compares the position of the robot to the position of every human in the simulation. For every human the robot is within the interaction range of (determined by the SFM model), the IPSS score for that model increases. This penalizes the robot for coming close enough to pedestrians such that it influences their SFM-based paths.

3) Comparison Table: The Leader-Follower (LF) and the Model Predictive Control (MPC) planners were run in each of the six scenarios for 100 simulations. The six quantitative measurements of each requirement are displayed in Table I and Table II.

V. DISCUSSION

We first evaluated on the original three scenarios from Liao et. al.'s work. For the Promenade scenario, the MPC dashed through the crossing consistently. On the other hand, the Leader Follower (LF) got distracted trying to find a leader, leading to a worse path roughness and slower time. For the crossing scenario, similar results were obtained. However, the MPC performed a straight-line movement, leading to perfect scores. On the other hand, the LF again became distracted by moving humans and chose a non-ideal trajectory. In the Roundabout, neither MPC nor the LF followed the social convention of going around the roundabout. Both beelined to the goal, which is especially a major failure of LF as it's goal of following pedestrians ideally leads to it matching social convention. LF again performed consistently worse than MPC, again due to its human-centric path planning, which made it highly distracted.

In the three custom designed scenarios, the comparison became more interesting. In the Convention environment, both MPC and LF entered the crowd, but LF was much more aggressive (in terms of collisions, minimum distance, and personal space). The LF became distracted by others and did not take an efficient route and therefore had a worse completion rate. In terms of the museum scenario, the MPC can be said to have "gotten lucky" in the videos we watched, as it clearly became entrapped by a moving group in the corridors which pushed it toward the goal. The LF was better at navigating obstacles by following leaders. In general, for the Black Friday scenario, the LF performed better than the MPC; it was significantly faster at reaching the goal, and scored a lower invasion of personal space score. The LF was less likely to come close to people, but when it did, it got much closer, as seen by its higher average minimum human distance and collision score. Both the LF and MPC got stuck behind obstacles often, but the LF was able to circumvent them by following humans.

These results clearly showed that the LF performed worse in simpler scenarios (Promenade, Crossing, Roundabout) and it struggled to follow social conventions that differed too far from a beeline route (as seen in Roundabout and Convention). However, it navigated better around obstacles in comparison to the MPC by following the peoples' path around it in our more complex scenarios like Museum where it reached it's goal in every simulation run. LF's core strength, leveraging the human's around it for navigation, started showing in complex scenarios, both in terms of obstacles and high crowd density as it performed much better than MPC in our most populated environment: Black Friday.

VI. LIMITATIONS AND LESSONS LEARNED

While our experiments do properly compare the Leader-Follower algorithm to the MPC planner, there are some limitations to the approach conducted. First, though Path Roughness can work as a metric for identifying the efficiency of curved paths, it is a noisy metric due to natural variations in the robotic movement. While this noise was largely removed by subtracting a baseline simulation value from each run, the randomness of these deviations makes baseline subtraction an imperfect solution. In addition, the window sizes of 10 for long term movement and 5 for short term movement were

	Promenade		Crossing		crossing	
Metric	**LF**	**MPC**	**LF**	**MPC**	**LF**	**MPC**
Completion Rate	100	100	100	100	100	98
Avg. Time to Goal	213.35	71	399.97	164	359	134
Avg. Min Human Distance	0.72	0.97	0.94	2.87	0.78	2.03
Percentage of Collisions	1	0	0	0	0	0
Avg. Path Roughness	0.68	0.07	0.54	0	1.58	0.59
Invasion of Personal Space Score	688.39	68.78	202.91	5.53	310.77	187.36

TABLE I: Leader-Follower vs MPC Results in Promenade, Crossing, and RoundAbout across 100 simulations

	Convention		Museum		Black Friday	
Metric	**LF**	**MPC**	**LF**	**MPC**	**LF**	**MPC**
Completion Rate	96	100	100	95	6	7
Avg. Time to Goal	543.92	168.67	573.98	300.31	186.83	946.85
Avg. Min Human Distance	0.684	0.922	0.67	0.95	0.69	1.047
Percentage of Collisions	2	0	3	0	6	0
Avg. Path Roughness	0.72	0.19	0.79	0.52	2.41	1.45
Invasion of Personal Space Score	1765.14	433.48	2577.61	856.51	1327.33	1996.77

TABLE II: Leader-Follower vs MPC results in Convention, Museum, and Black Friday across 100 simulations

arbitrarily chosen based on the sizes of the necessary curves in the simulations. These values should be evaluated and refined in future experiments.

Second, the brute force approach to determining the proper MPC parameters required us to limit the number of potential parameter values that were tested. This led to us discretizing these values and choosing relatively small windows for potential values. In the future, a gradient-based optimization approach should be used to better tune the parameters.

Similarly, the Leader-Follower parameters were left with the original values from Liao et. al. when they should have also been tuned. In further testing, the Leader-Follower's internal SFM model should be tuned as well as the parameters in the Leader-Follower algorithm itself. This could lead to the Leader-Follower algorithm performing much better in our tests.

Finally, there were some issues with the created simulations. Both Museum and Black Friday featured long stretches of corridors with people moving through them. Unfortunately, both simulations had instances of groups of people all moving together through a corridor. This large group was unpassable by MPC, leading to it getting pushed through the corridor. In doing so, the MPC planner often succeeded in complex environments by being pushed to the goal. In further testing, the size of corridors should be increased and large waves of pedestrians all moving in one direction should be limited.

Outside of these lessons that directly relate to the experimentation, there were additional things we learned in completing this project. First, our work in both running many simulations for data collection and also in brute-force evaluating parameters for MPC taught us importance of parallelizing code and minimizing memory leaks in long-running programs. This lesson led to programs that originally needed to run for over 40 hours being able to finish overnight. Second, we learned the importance of version control when simultaneously developing software. Because much of our work was done in tandem, many versions of key files were often in development at the same time. This led to merging issues that were exacerbated by VSCode's and Git's struggle to interact with Python notebooks. We learned to separate our notebooks into many files for different simulations to allow for better version control. Finally, despite the late nights this project led to, we learned that social navigation is a very exciting topic that encourages hard work. We were asked to continuously solve problems and develop new techniques. This was a very interesting and rewarding project to work on.

VII. CONCLUSION

This report presents an evaluation of the Leader-Follower (LF) algorithm introduced by Liao et al, benchmarked using a variety of diverse social navigation scenarios introduced by us. We learned from our experiments that LF, surprisingly, is worse against MPC based planner on simple navigation scenarios, contrary to original paper author's results. However, LF closes the gap, and performs better than MPC, when deployed in more socially complex scenarios with high number of humans, possibly owing to its people powered planning architecture.

ACKNOWLEDGMENTS

A special thanks to Professor Christoforos Mavrogiannis, Andrew Stratton, and the free coffee in the FRB kitchnette for helping us complete this project.

REFERENCES

[1] Yuwen Liao, Xinhang Xu, Ruofei Bai, Yizhuo Yang, Muqing Cao, Shenghai Yuan, and Lihua Xie. Following is all you need: Robot crowd navigation using people as planners. *arXiv preprint arXiv:2504.10828*, 2025. URL https://arxiv.org/pdf/2504.10828. Preprint.

[2] Dirk Helbing and Péter Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995. doi: 10.1103/PhysRevE.51.4282. URL https://arxiv.org/pdf/cond-mat/9805244.

[3] Stefan Seer, Christian Rudloff, Tomislav Matyus, and Norbert Brändle. Validation of crowd models including social groups. *Transportation Research Procedia*, 2: 724–732, 2014. doi: 10.1016/j.trpro.2014.09.080. URL https://www.researchgate.net/publication/267748709_Validation_of_Crowd_Models_Including_Social_Groups.

[4] Christoforos Mavrogiannis, Francesca Baldini, Allan Wang, Dapeng Zhao, Pete Trautman, Aaron Steinfeld, and Jean Oh. Core challenges of social robot navigation: A survey. *ACM Transactions on Human-Robot Interaction*, 12(3):1–39, September 2023. doi: 10.1145/3583741. URL https://doi.org/10.1145/3583741.

[5] Sepehr Samavi, James R. Han, Florian Shkurti, and Angela P. Schoellig. Sicnav: Safe and interactive crowd navigation using model predictive control and bilevel optimization. *IEEE Transactions on Robotics*, 2024. doi: 10.1109/TRO.2024.3360229. Early Access.

[6] Peter Trautman, Jeremy Ma, Richard M. Murray, and Andreas Krause. Robot navigation in dense human crowds: The case for cooperation. In *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2153–2160, Karlsruhe, Germany, May 2013. IEEE. doi: 10.1109/ICRA.2013.6630866.

[7] Anirudh Vemula, Katharina Muelling, and Jean Oh. Modeling cooperative navigation in dense human crowds. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1685–1692, Singapore, May 2017. IEEE. doi: 10.1109/ICRA.2017.7989199.

[8] Peter Trautman and Andreas Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 797–803, Taipei, Taiwan, October 2010. doi: 10.1109/IROS.2010.5654369. URL https://ieeexplore.ieee.org/document/5654369.

[9] Adarsh Jagan Sathyamoorthy, Utsav Patel, Moumita Paul, Nithish K. Sanjeev Kumar, Yash Savle, and Dinesh Manocha. Comet: Modeling group cohesion for socially compliant robot navigation in crowded scenes. *arXiv preprint arXiv:2108.09848*, 2021. URL https://arxiv.org/abs/2108.09848. Preprint.

[10] Diego Paez-Granados, Yujie He, David Gonon, Dan Jia, Bastian Leibe, Kenji Suzuki, and Aude Billard. Pedestrian-robot interactions on autonomous crowd navigation: Reactive control methods and evaluation metrics. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, October 2022. Accepted Version.

[11] Anika Bera, Tanmay Randhavane, Rohan Prinja, and Dinesh Manocha. Sociosense: Robot navigation amongst pedestrians with social and psychological constraints. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7018–7025, Vancouver, BC, Canada, September 2017. doi: 10.1109/IROS.2017.8206628. URL https://ieeexplore.ieee.org/document/8206628.

[12] Diego Martinez-Baselga, Oscar de Groot, Luzia Knoedler, Luis Riazuelo, Javier Alonso-Mora, and Luis Montano. Shine: Social homology identification for navigation in crowded environments. *The International Journal of Robotics Research*, 2025. doi: 10.1177/02783649251344639. URL https://journals.sagepub.com/doi/10.1177/02783649251344639. OnlineFirst.

[13] Rachel Kirby. *Social Robot Navigation*. Ph.d. thesis, Carnegie Mellon University, Robotics Institute, Pittsburgh, PA, USA, May 2010. URL https://www.ri.cmu.edu/pub_files/2010/5/rk_thesis.pdf.

[14] Frederik Haarslev, William Kristian Juel, Avgi Kollakidou, Norbert Krüger, and Leon Bodenhagen. Context-aware social robot navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12461–12467, London, UK, May 2023. IEEE. doi: 10.1109/ICRA48891.2023.10160502.

[15] Yuki Kobayashi, Takuma Sugimoto, Keisuke Tanaka, et al. Robot navigation based on predicting of human interaction and its reproducible evaluation in a densely crowded environment. *International Journal of Social Robotics*, 14:373–387, 2022. doi: 10.1007/s12369-021-00791-9. URL https://doi.org/10.1007/s12369-021-00791-9.

[16] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6015–6022, Montreal, QC, Canada, May 2019. IEEE. doi: 10.1109/ICRA.2019.8794134.

[17] Daniel Flögel, Lars Fischer, Thomas Rudolf, Tobias Schürmann, and Sören Hohmann. Socially integrated navigation: A social acting robot with deep reinforcement learning. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7801–7808, Prague, Czech Republic, September 2021. IEEE. doi: 10.1109/IROS51168.2021.9636455.

[18] Bozhi Xue, Ming Gao, Chao Wang, et al. Crowd-aware socially compliant robot navigation via deep reinforcement learning. *International Journal of Social Robotics*, 16:197–209, 2024. doi: 10.1007/s12369-023-01071-4. URL https://doi.org/10.1007/s12369-023-01071-4.

[19] Matthias Luber, Luciano Spinello, Jens Silva, and Kai O. Arras. Socially-aware robot navigation: A learning approach. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 902–907, Vilamoura-Algarve, Portugal, October 2012. IEEE. doi: 10.1109/IROS.2012.6385716.

[20] Santosh Balajee Banisetty, Scott Forer, Logan Yliniemi, Monica Nicolescu, and David Feil-Seifer. Socially-aware navigation: A non-linear multi-objective optimization approach. *IEEE Robotics and Automation Letters*, 6 (2):4127–4134, April 2021. doi: 10.1109/LRA.2021.3068880.

[21] Óscar Gil, Anais Garrell, and Alberto Sanfeliu. Social robot navigation tasks: Combining machine learning techniques and social force model. *Sensors*, 21(21):7087, 2021. doi: 10.3390/s21217087. URL https://doi.org/10.3390/s21217087.

[22] Kerry He, Wesley P. Chan, Akansel Cosgun, Albin Joy, and Elizabeth A. Croft. Robot gaze during autonomous navigation and its effect on social presence. In *2023 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 246–254, Stockholm, Sweden, March 2023. ACM/IEEE. doi: 10.1145/3568162.3576991.

[23] Benjamin Holman, Akash Anwar, Avyay Singh, Mariela Tec, Justin Hart, and Peter Stone. Watch where you're going! gaze and head orientation as predictors for social robot navigation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3553–3559, Xi'an, China, May 2021. IEEE. doi: 10.1109/ICRA48506.2021.9561286.

[24] Michael Lewis, Katia Sycara, and Paul Scerri. Using humans as sensors in robotic search. In *Proceedings of the 12th International Conference on Information Fusion (FUSION)*, pages 1249–1256, Seattle, WA, USA, July 2009. URL https://www.cs.cmu.edu/~pscerri/papers/Fusion09Lewis.pdf.

[25] Thibault Kruse, Amit Kumar Pandey, Rachid Alami, and Alexandra Kirsch. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12):1726–1743, 2013. doi: 10.1016/j.robot.2013.05.007.

[26] Procopio Silveira Stein, Vitor Santos, Anne Spalanzani, and Christian Laugier. Navigating in populated environments by following a leader. In *2013 IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, Gyeongju, South Korea, August 2013. URL https://inria.hal.science/hal-00917074/PDF/2013_stein_Soumission_roman.pdf.

[27] Alessandro Antonucci, Paolo Bevilacqua, Stefano Leonardi, Luigi Paolopoli, and Daniele Fontanelli. Humans as path-finders for mobile robots using teach-by-showing navigation. *Autonomous Robots*, 47(8):1255–1273, 2023. doi: 10.1007/s10514-023-10134-4. URL https://arxiv.org/pdf/2107.03079.

[28] Qianyi Zhang, Shijian Ma, Boyi Liu, Jingtai Liu, Jianhao Jiao, and Dimitrios Kanoulas. Follow everything: A leader-following and obstacle avoidance framework with goal-aware adaptation. *arXiv preprint arXiv:2504.19399*, 2025. URL https://arxiv.org/abs/2504.19399. Preprint.

[29] Carmen Scheidemann, Lennart Werner, Victor Reijgwart, Andrei Cramariuc, Joris Chomarat, Jia-Ruei Chiu, Roland Siegwart, and Marco Hutter. Obstacle-avoidant leader following with a quadruped robot. *arXiv preprint arXiv:2410.00572*, 2024. URL https://arxiv.org/pdf/2410.00572. Preprint.

[30] Yuqing Du, Nicholas J. Hetherington, Chu Lip Oon, Wesley P. Chan, Camilo Perez Quintero, Elizabeth Croft, and H. F. Machiel Van der Loos. Group surfing: A pedestrian-based approach to sidewalk robot navigation. *arXiv preprint arXiv:2104.05933*, 2021. URL https://arxiv.org/pdf/2104.05933. Preprint.

[31] S. Leisiazar, S. R. R. Rohani, E. J. Park, A. Lim, and M. Chen. Adapting to frequent human direction changes in autonomous frontal following robots. *IEEE Robotics and Automation Letters*, 10(3):2934–2941, March 2025. doi: 10.1109/LRA.2025.3537860.

[32] Md Jahidul Islam, Junaed Hong, and Junaed Sattar. Person-following by autonomous robots: A categorical overview. *The International Journal of Robotics Research*, 38(14):1581–1618, 2019. doi: 10.1177/0278364919881683. URL https://arxiv.org/abs/1803.08202.

[33] Blender Foundation. Blender. https://www.blender.org, 2024. Version 4.0.

[34] Sriyash Poddar, Christoforos Mavrogiannis, and Siddhartha S. Srinivasa. From crowd motion prediction to robot navigation in crowds, 2023. URL https://arxiv.org/abs/2303.01424.

[35] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *European conference on computer vision*, pages 549–565. Springer, 2016.