Characterizing the Effects of Contextual Factors on the Complexity of Social Robot Navigation Scenarios

Andrew Stratton[1], Kris Hauser[2], Christoforos Mavrogiannis[1]

Abstract— **Social robot navigation algorithms are often demonstrated in overly simplified scenarios, prohibiting the extraction of practical insights about their relevance to real-world domains. Our key insight is that an understanding of the inherent complexity of a social robot navigation scenario could help characterize the limitations of existing navigation algorithms and provide actionable directions for improvement. We leverage an exploration of recent literature to identify a series of factors contributing to the complexity of a scenario, disambiguating between contextual and robot-related ones. We then conduct a simulation study investigating how manipulations of contextual factors impact the performance of a variety of navigation algorithms. We find that dense and narrow environments correlate most strongly with performance drops, while the heterogeneity of agent policies and directionality of interactions have a less pronounced effect. This motivates a shift towards developing and testing algorithms under higher-complexity settings.**

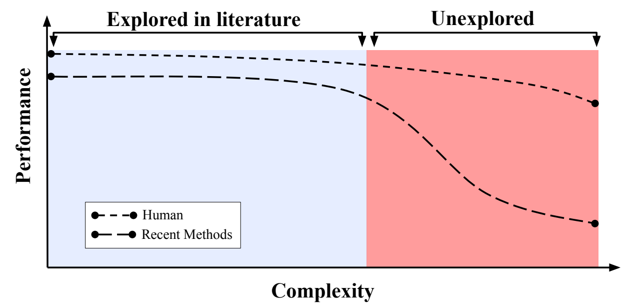


Fig. 1: We investigate how different parameters defining a social robot navigation scenario impact the performance of recent algorithms. We show that while many algorithms successfully handle a wide range of scenarios, their performance drops at a region of Complexity that is representative of real-world situations that humans effectively handle [12].

I. Introduction

Recent surveys on social robot navigation [8, 9, 22, 28] emphasize the need for a standardization of benchmarking practices. A major challenge in standardizing benchmarking in social navigation lies in the design of experiments, and specifically in balancing repeatability and natural interactions. While real-world experiments offer the most powerful insights, their high cost has motivated the experimentation within virtual environments [2]. Several works have developed photorealistic simulators [2, 14, 33], while others have focused on finding informative quantitative metrics [21, 27, 35]. However, as demonstrated by Fraichard and Levesy [7], simulation often introduces strong assumptions prohibiting the extraction of generalizable insights. Another challenge in designing benchmarks is the lack of understanding of what makes a social navigation scenario hard. In prior work, the crowd density [22, 30, 32] has often been used as a proxy for complexity. However, density on its own does not capture the complexity of the motion coupling between closely interacting agents. To capture that, metrics like Path Irregularity [11] and the Topological Complexity Index [6, 21] quantify aspects of geometric and topological richness of agents' trajectories.

In this paper we argue that to establish effective benchmarks for social robot navigation, the research community should better understand and control the *dimensions of problem complexity*. We enumerate a set of factors contributing to the complexity of a social robot navigation scenario, distinguishing contextual factors from robot-related ones. We then conduct an extensive simulated study to understand how robot-independent, contextual factors impact algorithm performance. Our findings underscore the need for algorithms that explicitly account for diverse, high Complexity scenarios, and benchmarks that explore scenarios with high Complexity across multiple factors.

[1]Department of Robotics, University of Michigan, Ann Arbor, USA. Email: {arstr, cmavro}@umich.edu

[2]Department of Computer Science, University of Illinois at Urbana-Champaign. Email: kkhauser@illinois.edu.

A. Stratton was partially funded by NSF Grant# NRI-2025782.

II. Factors of Complexity in the Literature

Based on recent literature [29], we identified the factors below as determinants of the complexity of a social robot navigation scenario (see Fig. 2). We distinguish the factors as *contextual* because they are agnostic to the robot platform used, as compared to *robot* factors such as maximum speed and size which are dependent on the choice of hardware.

Density. Density is often used as a proxy for the Complexity of a scenario [22, 30, 32]. We report it in the standard form of $agents/m^2$, although we note this does not account for the variable size of agents.

Directionality. The directions in which humans encounter the robot contribute to the difficulty of the collision-avoidance task [36]. We identified four cases: Passing: Agents move parallel, paths do not intersect; Crossing: Agents move perpendicular, paths intersect; Random: Agent starts and goals are randomly sampled; Circle Crossing: Agent start and goals are sampled on opposite sides of the circumference of a circle.

Environment Geometry. We find that most often, evaluations take place in either hallways or medium to large office rooms, which place no constraints on the agents' movements.

Policy Mixture. The way humans navigate next to the robot is another complicating factor. Cooperative agents assume partial responsibility for collision avoidance, which simplifies the robot's task. In contrast, when rushing, being distracted, or changing intentions, humans may pose greater challenges to a robot [26]. We found that most real-world studies instruct participants to navigate cooperatively, and most simulation-based studies make use of cooperative crowd simulators like Social Force [13] and ORCA [34].

III. Investigating the Complexity of Social Robot Navigation Scenarios

We propose a definition of a *Social robot navigation scenario* based on parameters related to the factors of Complexity identified in Sec. II. We then describe an experiment design that investigates the performance of a variety of navigation algorithms under different scenarios.

A. Social Navigation Scenario

Consider a robot navigating next to $n \geq 1$ human agents in a workspace $\mathcal{W} \subseteq \mathbb{R}^2$ with a set of static obstacles $\mathcal{W}_{obs} \subseteq \mathcal{W}$. The robot starts from an initial configuration s_R and moves towards a goal g_R by following a policy π_R whereas humans are navigating from their initial configurations s_i towards their goals, g_i by following a policy π_i, $i \in \mathcal{N}$; agents' goals are unknown to one another. The robot occupies an area $A_R \in \mathcal{W}$, and each human occupies an area $A_i \in \mathcal{W}$. The objective of the robot is to reach its destination while avoiding collisions with static obstacles and abiding by social norms. We define a social navigation scenario as a tuple:

$$\mathcal{S} = (n, A_R, A_{i:n}, s_R, g_R, s_{i:n}, g_{i:n}, \pi_{i:n}, \mathcal{W}_{obs}). \quad (1)$$

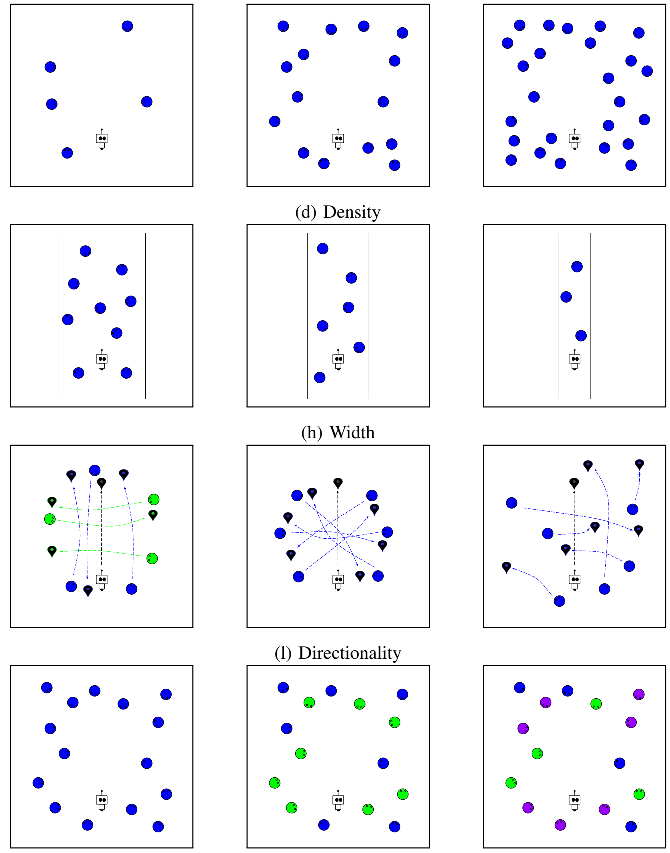
We denote by π_i the true policy for agent i, capturing the way they make decisions based on their objectives as well as behavioral and contextual aspects of their navigation profile.

B. Experiment Design

We design scenarios of varying complexity by manipulating each of the factors in Sec. II in isolation.

Scenario configurations. We first define a base scenario $\mathcal{S}_b$ defining with $n = 15$, $A_R, A_{i:n} = \pi(0.3)^2$, $s_R = (5, 1)$, $g_R = (5, 9)$, $\pi_{i:n} = ORCA, SFM, v_{pref} = 1.0m/s$, $\mathcal{W}_{obs} = \{[0, w] \times [0, l]\}^{\mathsf{c}} = \{[0, 10] \times [0, 10]\}^{\mathsf{c}}$; $s_{i:n}, g_{i:n}$ are sampled from passing and crossing. We then modify a single variable in each experiment, from low to high *intensity*:

- *Density*: {0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35}. We choose this range following prior work [22, 32].
- *Directionality*: {Passing only, crossing only, passing and crossing, circle crossing, random start/goal}. We found that prior evaluations frequently involve passing scenarios with one or two humans, although many papers set up simultaneous passing and crossing scenarios. We also find that while circle crossings are frequently used in simulation to force Complex interactions [3, 4, 22, 31], they do not appear as often in real robot evaluations.
- *Policy Mixture*: {SFM only, ORCA only, Mix 1, Mix 2, Mix 3}, where Mix 1 contains 8 ORCA and 7 SFM agents, Mix 2 contains 5 ORCA, 5 SFM, 2 CV, 3 Static agents, and Mix 3 contains 4 ORCA, 4 SFM, 4 CV, 3 Static agents, respectively. We add increasing inattentive agents at higher intensity to model Complex real-world scenarios in which the robot must navigate among cooperative and uncooperative agents simultaneously [20].
- *Environment Width*: {4.5, 4.0, 3.5, 3.0, 2.5, 2.0, 1.5}, which reflects the reported width of most hallways (e.g., [15, 24, 38]).



(d) Density

(h) Width

(l) Directionality

(p) Agent Policy Mixture

Fig. 2: Left to right: Scenarios of increasing complexity for each of the complexity factors considered (d-p). In the leftmost Directionality figure green agents are crossing and blue agents are passing, while different colors represent different policies in Policy Mixture.

Fig. 2 illustrates the configurations considered in our study.

Algorithms. We propose to evaluate the change in complexity of a given scenario using the performance of several navigation algorithms. Specifically, we employ:

- *Relational Graph Learning (RGL) [4]:* a reinforcement learning based approach that models pedestrian interactions using a graph neural network.
- *Social GAN Model Predictive Control (MPC-SGAN) [10]:* An approach integrating a recent crowd motion prediction model into an MPC framework, using the approach and implementation of Poddar et al. [26].
- *Constant Velocity Model Predictive Control (MPC-CV):* Identical to *MPC-SGAN*, but instead using CV motion predictions for each agent.
- *Social GAN Model Predictive Path Integral (MPPI-SGAN):* SGAN integrated into an MPPI controller.
- *Reactive Planner (RP)*: A myopic planner that attempts to avoid collisions while navigating towards the goal.
- *Optimal Reciprocal Collision Avoidance (ORCA) [34]*: A multiagent collision avoidance method that guarantees collision free movement among ORCA agents.
- *Social Forces Model (SFM) [13]*: A physics-inspired model of crowd motion.
- *Constant Velocity (CV)*: An unreactive agent that moves with constant velocity toward the goal.

Metrics. Based on literature [8, 9, 22], we use the following metrics to measure performance, collected only from

Successful trials:

- *Success:* The average number of trials in which the ego-agent reaches its goal without collision.
- *Time to goal:* The average time to goal across trials.
- *Distance to agent:* The Minimum Distance to the nearest other agent during a trial, averaged across trials.
- *Path Irregularity [11]:* The amount of unnecessary turning per unit path length, measured in $\frac{rad}{m}$, calculated as $\sum_{Path} \frac{Rotation - Min.\ rotation\ needed}{Path\ length}$.

Hypotheses. We expect that scenarios of higher Complexity will pose greater navigation challenges, and this will be reflected in significant performance drops across all algorithms. Additionally, as Density is often used as a proxy for Complexity as a whole, we anticipate that it will have the strongest correlation with performance drop. Based on these expectations, We formulate the following hypotheses:

- *H1*. Increasing the intensity of each of the four complexity factors (Density, Directionality, Environment Geometry, Policy Mixture) independently decreases performance with respect to collected metrics.
- *H2*. The negative correlation between Complexity and performance in *H1* will be strongest for Density.

C. Implementation Details

We generate 500 scenarios for each condition within each experiment and report metrics as averages across all successful trials. We fix the random seed to ensure each method experiences the same scenarios. For the MPC methods, we used the formulation of Poddar et al. [26]. For MPC-SGAN and MPPI-SGAN, we use a checkpoint pretrained on the *Zara* portion of the UCY dataset [16].

D. Results

Fig. 3 summarizes our experimental results, organized by experiment types (rows) and metrics (columns). Based on this data, we investigate the validity of our hypotheses:

H1. We find a statistically significant correlation between Density and Success rate and Minimum Distance to agent ($\rho = -0.867, -0.735, p < 0.001$ using Spearman's r test), and a moderate correlation between Density and average Time and Path Irregularity ($\rho = 0.499, 0.463,\ p < 0.001$). We also find a correlation between *increasing* width and method Success ($\rho = 0.593,\ p < 0.001$), and a correlation between width and Minimum Distance ($\rho = 0.285,\ p < 0.05$). We do not find any statistically significant correlation with average Time or Path Irregularity. Regarding Policy Mixture and Directionality, we find strong and moderate correlations respectively between higher intensity and Success ($\rho = -0.715, -0.462,\ p < 0.002$), and a moderate correlation between Directionality and average Time ($\rho = -0.314,\ p < 0.05$). We see no statistically significant correlation for Minimum Distance, and Path Irregularity in either case. Thus, we find that while each factor affects at least one metric, Density and Environment Geometry appear to have the strongest correlation, giving partial support for *H1*.

H2. We observe that the correlations between Density and other collected metrics are all stronger than those of the other factors. Thus we find support for *H2*.

While Density correlates strongest with Complexity, we see that the Environment Geometry, Directionality, and agent policies all have at least some correlation with performance, and thus should be considered when assessing the Complexity of a scenario. While the support for **H2** vindicates prior use of Density as a proxy for Complexity, the partial support for **H1** suggests that a more accurate picture can be obtained by analyzing the rest of the contextual factors.

MPC-CV, MPC-SGAN, and MPPI-SGAN performance scales poorly with moderate to high crowd density. We do observe that MPPI-SGAN consistently has higher Success rates than MPC-SGAN, showing that a stronger controller implementation does indeed lead to better performance (albeit with slightly lower Minimum Distances). The steep decrease of all three methods, however, indicates that having an inaccurate prediction model becomes increasingly problematic as the number of agent trajectories predicted increases, even with a more robust controller. RGL, on the other hand, sees performance scale better with regard to density.

In nearly all experiments, RP and ORCA policies had low average Time, while still maintaining comparable or better Success rates to other methods. Even the CV agent was moderately Successful in lower Complexity scenarios, which matches prior experimental outcomes [22]. SFM's high Success rates and Minimum Distances indicate that with proper tuning it could be viable as a local controller for a socially navigating robot.

We do however see that the MPC methods maintain the highest Minimum Distance, while ORCA, RP, and CV have the lowest Minimum Distances. Additionally, the MPC methods, with the exception of the Environment Width experiments, maintain comparable Path Irregularity to ORCA, SFM, and RGL, showing they can maintain distance with comparably smooth paths. Thus we see that while the reactive and CV methods (with the exception of SFM) achieve efficient, collision-free navigation through other agents' cooperation, they are worse at respecting the personal space of others compared to predictive methods.

Qualitatively, we observed that passing scenarios are generally easier to navigate than crossing, and SFM agents are much more subservient than ORCA. An implicit hypothesis in our experimental ordering was that a mixture of two Directionalities or Policy Mixtures would be more Complex to navigate than either individually. We instead see improvement in the ORCA/SFM and Passing/Crossing scenarios compared to ORCA and Crossing only, which suggests the combinations are actually less Complex.

IV. Discussion

Conducting high-Complexity evaluations. Our experiments demonstrate that most algorithms handle the most common evaluation scenarios in recent literature (low-medium density, passing only, medium-large rooms, cooperative agents [29]), but struggle in more Complex settings. This indicates the most frequently used scenarios are not the best for extracting useful insights; for example, had we not manipulated the Complexity factors towards the upper extremes, the severe performance drops at high density experienced by

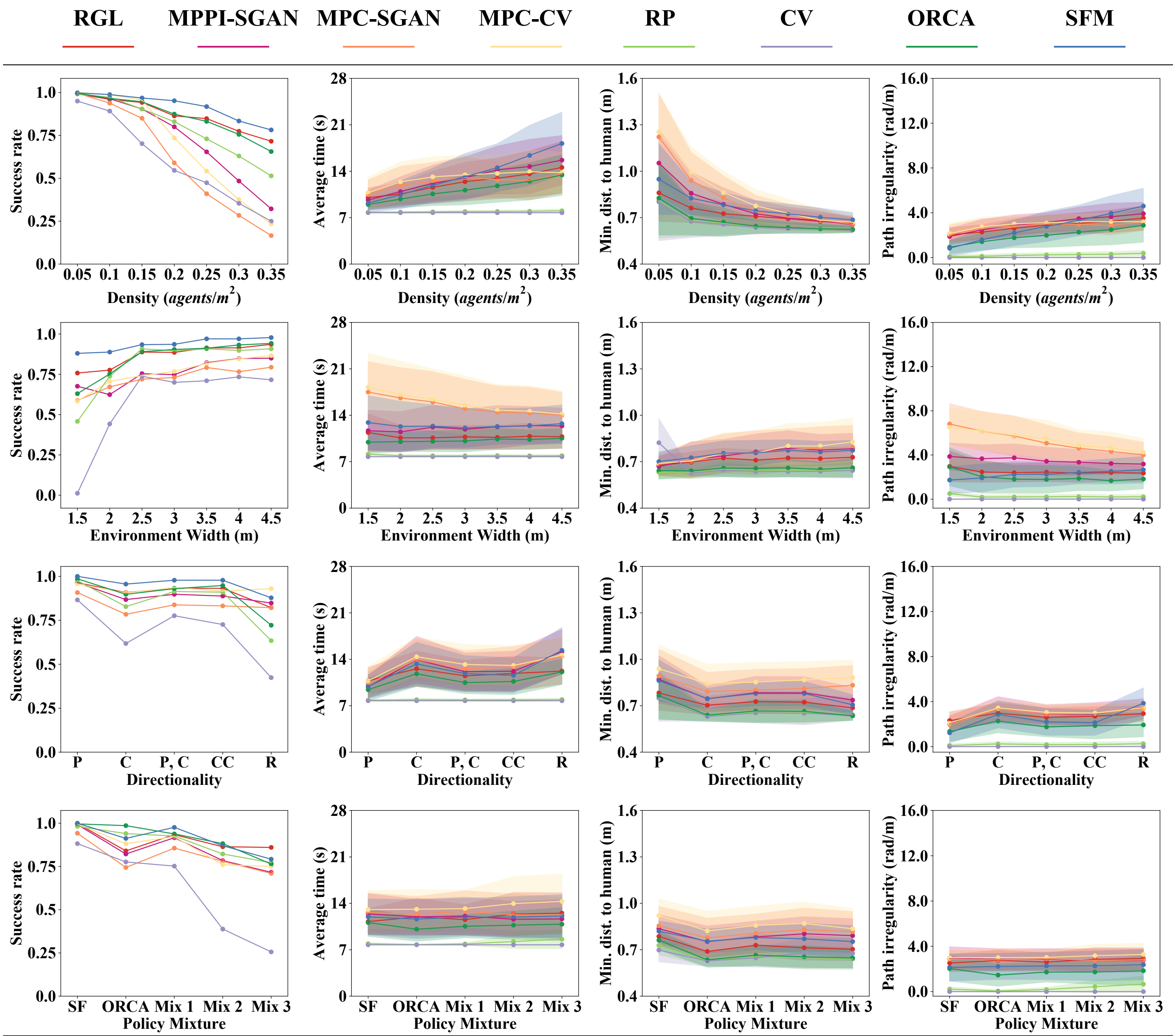


Fig. 3: Performance of methods across our experiments. Rows indicate experiments and columns correspond to different evaluation metrics. Each point represents the mean over 500 experiments; shaded regions indicate standard deviation. Mix 1 has 7 SFM and 8 ORCA agents. Mix 2 has 5 SFM, 5 ORCA, 2 CV, and 3 static agents. Mix 3 has 4 SFM, 4 ORCA, 4 CV, 3 static agents.

MPC-CV, MPC-SGAN, and MPPI-SGAN would not have been identified. These observations suggest that the social navigation community should shift to studying denser, more geometrically-constrained, highly-mixed and high-traffic environments, similar to many context-rich public domains.

Handling test-time distribution. The high Success rates of CV and RP suggest that in lower-Complexity regimes, it might be sufficient to use non-reactive classical planning techniques. As the Complexity increases, all algorithms experience a steep decline; the magnitude of the decline is related to how far out of distribution an algorithm operates. Our evaluation (Fig. 3) captures the sensitivity of data-driven approaches to their training distribution. For instance, we see that RGL, and MPC/MPPI-SGAN experience a substantial performance decline as the test-time Density moves away from the training/tuning Density $(0.05, 0.10\ agents/m^2)$. In practice, data-driven approaches will often face distribution shift at deployment, which motivates the development of algorithms for overcoming distribution shifts to be an important direction for future research [18, 39]. Alternative approaches include techniques for proactive, expressive motion generation [1, 21, 35] or techniques that embrace graceful touch [25], similar to how human crowds resolve close interactions in train stations or airports [12].

Simulating complex settings. While realistically simulating pedestrian-robot interactions is challenging [7, 22], we view it as essential for testing high-Complexity settings that are difficult to safely replicate in the real world. This requires revisiting conventional assumptions, such as that humans are non-reactive to the robot [4, 5, 17, 19, 23, 37], which is unrealistic since most robots are large enough to be observed [29]. Thus, simulation should focus on *visible* robot settings, leveraging metrics and considerations of users' perceptions [21, 27, 35] to compare algorithms' social performance.

References

[1] J.-L. Bastarache, C. Nielsen, and S. L. Smith. On legible and predictable robot navigation in multi-agent environments. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5508–5514, 2023.

[2] A. Biswas, A. Wang, G. Silvera, A. Steinfeld, and H. Admoni. SocNavBench: A grounded simulation testing framework for evaluating social navigation. *Transactions on Human-Robot Interaction*, 11(3), 2022.

[3] C. Chen, Y. Liu, S. Kreiss, and A. Alahi. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6015–6022, 2019.

[4] C. Chen, S. Hu, P. Nikdel, G. Mori, and M. Savva. Relational graph learning for crowd navigation. In *Proceesings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10007–10013, 2020.

[5] Y. F. Chen, M. Liu, M. Everett, and J. P. How. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning. In *Procedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 285–292, 2017.

[6] I. Dynnikov and B. Wiest. On the complexity of braids. *Journal of the European Mathematical Society*, 9(4):801–840, 2007.

[7] T. Fraichard and V. Levesy. From crowd simulation to robot navigation in crowds. *IEEE Robotics and Automation Letters*, 5(2):729–735, 2020.

[8] A. Francis et al. Principles and guidelines for evaluating social robot navigation algorithms. *arXiv:2306.16740* [cs.RO], 2023.

[9] Y. Gao and C.-M. Huang. Evaluation of socially-aware robot navigation. *Frontiers in Robotics and AI*, 8, 01 2022.

[10] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2255–2264, 2018.

[11] J. Guzzi, A. Giusti, L. M. Gambardella, and G. Di Caro. Human-friendly robot navigation in dynamic environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 423–430, 05 2013.

[12] D. Helbing and A. Johansson. Pedestrian, crowd and evacuation dynamics. In *Encyclopedia of Complexity and Systems Science*, pages 6476–6495. Springer New York, 2009.

[13] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.

[14] J. Holtz and J. Biswas. Socialgym: A framework for benchmarking social robot navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11246–11252, 2022.

[15] R. Kirby. *Social Robot Navigation*. PhD thesis, Carnegie Mellon University, 2010.

[16] A. Lerner, Y. Chrysanthou, and D. Lischinski. Crowds by Example. *Computer Graphics Forum*, 2007.

[17] S. Liu, P. Chang, Z. Huang, N. Chakraborty, K. Hong, W. Liang, D. L. McPherson, J. Geng, and K. Driggs-Campbell. Intention aware robot crowd navigation with attention-based interaction graph. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 12015–12021, 2023.

[18] Y. Liu, R. Cadei, J. Schweizer, S. Bahmani, and A. Alahi. Towards robust and adaptive motion forecasting: A causal representation perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17081–17092, June 2022.

[19] S. Matsuzaki and Y. Hasegawa. Learning crowd-aware robot navigation from challenging environments via distributed deep reinforcement learning. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 4730–4736, 2022.

[20] S. Matsuzaki, S. Aonuma, and Y. Hasegawa. Dynamic window approach with human imitating collision avoidance. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 8180–8186, 2021.

[21] C. Mavrogiannis, P. Alves-Oliveira, W. Thomason, and R. A. Knepper. Social momentum: Design and evaluation of a framework for socially competent robot navigation. *Transactions on Human-Robot Interaction*, 11(2), 2022.

[22] C. Mavrogiannis, F. Baldini, A. Wang, D. Zhao, P. Trautman, A. Steinfeld, and J. Oh. Core Challenges of Social Robot Navigation: A Survey. *Transactions on Human-Robot Interaction*, 12(3), 2023.

[23] Y.-J. Mun, M. Itkina, S. Liu, and K. Driggs-Campbell. Occlusion-aware crowd navigation using people as sensors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12031–12037, 2023.

[24] E. Pacchierotti, H. I. Christensen, and P. Jensfelt. Evaluation of passing distance for social robots. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 315–320, 2006.

[25] D. Paez-Granados, V. Gupta, and A. Billard. Unfreezing social navigation: Dynamical systems based compliance for contact control in robot navigation. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 8368–8374, 2022.

[26] S. Poddar, C. Mavrogiannis, and S. S. Srinivasa. From crowd motion prediction to robot navigation in crowds. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6765–6772, 2023.

[27] P. T. Singamaneni, A. Favier, and R. Alami. Towards benchmarking human-aware social robot navigation: A new perspective and metrics. In *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 914–921, 2023.

[28] P. T. Singamaneni, P. Bachiller-Burgos, L. J. Manso, A. Garrell, A. Sanfeliu, A. Spalanzani, and R. Alami. A survey on socially aware robot navigation: Taxonomy and future challenges. *The International Journal of Robotics Research*, 0(0):02783649241230562, 2024.

[29] A. Stratton, K. Hauser, and C. Mavrogiannis. Characterizing the complexity of social robot navigation scenarios, 2024.

[30] M. Sun, F. Baldini, P. Trautman, and T. Murphey. Move beyond trajectories: Distribution space coupling for crowd navigation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2021.

[31] M. Sun, F. Baldini, P. Trautman, and T. Murphey. Mixed-strategy nash equilibrium for crowd navigation. *arXiv:2403.01537* [cs.RO], 2024.

[32] P. Trautman, J. Ma, R. M. Murray, and A. Krause. Robot navigation in dense human crowds: Statistical models and experimental studies of human-robot cooperation. *International Journal of Robotics Research*, 34(3):335–356, 2015.

[33] N. Tsoi, A. Xiang, P. Yu, S. S. Sohn, G. Schwartz, S. Ramesh, M. Hussein, A. W. Gupta, M. Kapadia, and M. Vázquez. Sean 2.0: Formalizing and generating social situations for robot navigation. *IEEE Robotics and Automation Letters*, 7(4):11047–11054, 2022.

[34] J. van den Berg, S. J. Guy, M. C. Lin, and D. Manocha. Reciprocal n-body collision avoidance. In *Proceedings of the International Symposium on Robotics Research (ISRR)*, pages 3–19, 2009.

[35] N. Walker, C. Mavrogiannis, S. S. Srinivasa, and M. Cakmak. Influencing behavioral attributions to robot motion during task execution. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2021.

[36] A. Wang, C. Mavrogiannis, and A. Steinfeld. Group-based motion prediction for navigation in crowded environments. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2021.

[37] R. Wang, W. Wang, and B.-C. Min. Feedback-efficient active preference learning for socially aware robot navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11336–11343, 2022.

[38] B. Yang, J. Jiao, L. Wang, and M. Liu. An online interactive approach for crowd navigation of quadrupedal robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 13556–13562, 2022.

[39] H. Yu, C. Hirayama, C. Yu, S. Herbert, and S. Gao. Sequential neural barriers for scalable dynamic obstacle avoidance. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11241–11248, 2023.