# Interaction-Aware Autonomous Navigation among Pedestrians using Social Forces Response Dynamics

Luzia Knoedler, Nils Wilde, and Javier Alonso-Mora
Cognitive Robotics (CoR) department
Delft University of Technology
2628 CD Delft, The Netherlands
Email: {l.knoedler, n.wilde, j.alonsomora}@tudelft.nl

***Abstract*—Ignoring the interactions between agents during motion planning in multi-agent environments can result in overly conservative or opaque navigation behaviors, and in dense crowds, it may lead to the so-called Freezing robot problem. Although coupled planning can mitigate these issues, it typically incurs high computational costs, especially as the number of agents increases. To enhance interaction while limiting the computational complexity, we formulate the interactions as an underactuated system and propose to leverage the Social Forces Model (SFM) as the pedestrians' response dynamics. The SFM is a well-established model widely used for describing pedestrian behaviors. Unlike deep learning models that require extensive training data, SFM offers interpretability and adaptability across various environments. We use Model Predictive Path Integral Control to solve the optimization problem, demonstrating that by accounting for interactions, the robot can effectively leverage the behavior of other agents. Additionally, we show that when combined with a specific cost function, the robot is able to plan motions that decrease its impact on surrounding pedestrians.**

## I. INTRODUCTION AND RELATED WORK

The rise of autonomous mobile robots in our daily environments makes it essential to ensure their safe and efficient interaction with pedestrians. This necessitates that robots can accurately predict the behavior of the surrounding pedestrians. In recent years, prediction models based on various deep neural network architectures have demonstrated notable progress in both prediction accuracy and scalability [5, 8, 17]. Traditionally, these prediction models are used to forecast the future trajectories of surrounding pedestrians. Subsequently, a planning step follows, during which the robot plans its trajectory in response to these predictions. This invariably results in the pedestrians being solely treated as moving obstacles, fostering a one-way interaction where only the robot adjusts its behavior. This can give rise to overly conservative or opaque navigation behaviors, see the left column in Fig. 1, and in dense crowds, it may lead to the so-called Freezing Robot Problem (FRP) [14], even when perfect predictions are considered. Hence, performing coupled prediction and planning and, thus, joint collision avoidance is crucial for realizing decision-making that is more interactive and akin to human behavior.

An important body of work addresses multi-agent interactions from a game-theoretic perspective, specifically using general sum dynamic games [4, 7]. Since each agent's action depends on the decisions of the others, solving games presents

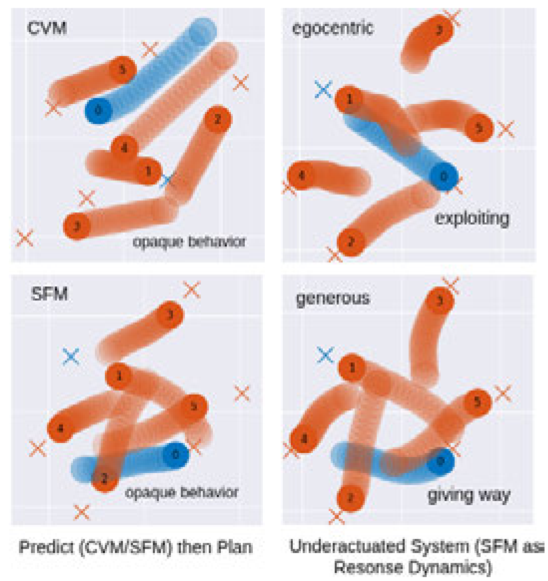


Fig. 1: Robot (blue) navigating among pedestrians (orange). Crosses represent the goals, respectively. Transparent circles indicate the future plan for the robot and the predicted behavior of the pedestrians.

a considerable computational challenge, particularly with an increasing number of agents. The computational complexity of these models imposes constraints on their applicability. The authors in [13] apply Model Predictive Path Integral Control (MPPI), a parallelizable sampling-based Model Predictive Control (MPC) algorithm, assuming knowledge of the other agents' objective functions and predicting their goals using the Constant Velocity Model (CVM). Nevertheless, the computational complexity scales linearly with the number of agents if a constant number of samples is assumed while the sample efficiency decreases.

A different approach is presented by [2], which bridges the gap between the use of prediction models discussed previously and coupled planning. They utilize predictions as an initial guess and incorporate an objective function to encourage proximity to these predictions. To reduce the computational complexity, [11] formulate the interaction between human-driven vehicles and autonomous vehicles as an underactuated dynamical system, meaning that the robot directly influences its own state and indirectly the state of the humans. The dynamics model thus includes the interaction dynamics between the agents. However, evaluating the dynamics model still requires solving for the optimal human response. Additionally, it requires the identification of the human objective function.

To address these issues, rather than estimating objectives explicitly and online, [3] build on top of the pedestrian prediction literature and learn an interactive multi-agent prediction policy. Using the policy they formulate the multi-agent motion planning problem as an optimization problem over only the ego agent's action sequence.

Instead of learning a multi-agent motion policy, we propose to use the Social Forces Model (SFM) [6]. The SFM is a widely used and well-established model for describing the motion of pedestrians, e.g. in simulations for benchmarking [15]. It offers several advantages: it provides a well-established and interpretable framework for modeling pedestrian interactions, it does not require training and therefore does not rely on available data of the considered context, and it can be easily adapted to different environments. We solve the multi-agent underactuated motion planning problem using MPPI.

## II. PROBLEM FORMULATION

### A. Preliminaries

*a) Social Forces Model (SFM):* The SFM [6] is a widely used and well-established model for describing the motion of pedestrians. It considers three main effects that determine the motion of a pedestrian $i$: the attraction towards their destination $\mathbf{f}^i_{\text{dest}}$, the repulsive effects of static obstacles $\mathbf{f}^i_{\text{static}}$, and the repulsive effects of other agents $\mathbf{f}^i_{\text{dyn}}$. At time $t$, the SFM describes the change in velocity through the composition of social forces resulting in

$$\frac{d\mathbf{v}^i}{dt} = \begin{bmatrix} a_x \\ a_y \end{bmatrix} = p_{\text{dest}}\mathbf{f}^i_{\text{dest}} + p_{\text{static}}\mathbf{f}^i_{\text{static}} + p_{\text{dyn}}\mathbf{f}^i_{\text{dyn}},$$

where $p_{\text{dest}}$, $p_{\text{static}}$, and $p_{\text{dynamic}}$ are weighting parameters implemented according to PedSim[1]. We make use of the SFM formulation presented in [9]. When a pedestrian encounters no disruptions, they move from their current position $\mathbf{p}$ towards their goal position $\mathbf{p}_g$ at a specific desired speed $v^{\text{des}}$. The destination force $\mathbf{f}^i_{\text{dest}}$ is determined according to

$$\mathbf{f}^i_{\text{dest}} = \frac{1}{\tau}(v^{\text{des}}\mathbf{e}^0 - \mathbf{v}),$$

where $\mathbf{e}^0 = (\mathbf{p}^i_g - \mathbf{p}^i)/||\mathbf{p}^i_g - \mathbf{p}^i||$ is the desired direction of motion, $\mathbf{v}$ is the current velocity, and $\tau$ is a relaxation time. The static obstacle repulsive force $\mathbf{f}^i_{\text{static}}$ is defined by:

$$\mathbf{f}^i_{\text{static}} = ae^{-\mathrm{d}^{\mathrm{i}}/\mathrm{b}},$$

[1] https://github.com/srl-freiburg/pedsim_ros/tree/master

where $d^i$ is the orthogonal distance of the $i$-th pedestrian to the obstacle. The pedestrian interaction force $\mathbf{f}^i_{\text{dyn}}$ is given by

$$\mathbf{f}^i_{\text{dyn}} = \sum_{j=0, j\neq i}^{N} \mathbf{f}^i_{\text{dyn,j}}, \quad (1)$$

$$\mathbf{f}^i_{\text{dyn,j}} = -Ae^{-d^{ij}/B}\left[e^{-(n'B\theta^{ij})^2}\mathbf{t}^{ij} + e^{-(nB\theta^{ij})^2}\mathbf{n}\right], \quad (2)$$

where $d^{ij}$ denotes the distance between two pedestrians $i$ and $j$, and $\theta^{ij}$ denotes the angle between the interaction direction $\mathbf{t}^{ij}$ and the vector pointing from pedestrian $i$ to pedestrian $j$. They are defined as follows:

$$d^{ij} = ||\mathbf{d}^{ij}|| = ||\mathbf{p}^j - \mathbf{p}^i||, \quad (3)$$

$$\mathbf{D}^{ij} = \lambda(\mathbf{v}^i - \mathbf{v}^j) + \mathbf{d}^{ij}/d^{ij}, \quad (4)$$

$$\mathbf{t}^{ij} = \mathbf{D}^{ij}/||\mathbf{D}^{ij}||, \quad (5)$$

$$B = \gamma||\mathbf{D}||, \quad (6)$$

where $A$, $\gamma$, $n$, $n'$, and  are model parameters.

In this work, we make use of the SFM to represent the response dynamics of pedestrians. We choose the SFM parameters according to [9] which are summarized in Table I. We do not consider static obstacles; however, they can be easily incorporated if needed.

*b) Model Predictive Path Integral Control Algorithm:* MPPI can solve optimal control problems for discrete-time dynamical systems $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \tilde{\mathbf{u}}_t)$ with state $\mathbf{x}$, timmestep $t$, and noisy input $\tilde{\mathbf{u}}$ with variance $\mathbf{\Sigma}$ and mean $\mathbf{u}$. The mean input will be provided to the system. The algorithm generates $M$ input sequence samples $\tilde{U}_m$ with $m \in [1, M]$ from a distribution $\mathcal{N}(\mathbf{u}_t, \nu\mathbf{\Sigma})$ over a horizon $K$ with $\nu$ being a scaling factor. Using $\tilde{U}_m$ and the dynamics model, the state sequence is generated over a horizon $K$. For each sample, the cost consisting of a stage and a terminal cost is computed. The input sequence $U^*$, which approximates the optimal control input sequence, is computed using importance sampling. For more information, we refer to [13, 16].

### B. Problem Statement

We consider a scenario where a mobile robot, must navigate from an initial position $\mathbf{p}_0$ to a goal position $\mathbf{p}_g$ in the $\mathbb{R}^2$ plane populated by $N$ pedestrians, each also navigating towards its respective goal position. The robot's physical state at time step $t$ is denoted as $\mathbf{x}^0_t \in \mathbb{X}^R$, while each pedestrian's respective physical state is denoted by $\mathbf{x}^i_t \in \mathbb{X}^H$ for $i \in \{1, \ldots, N\}$.

**Notation:** We omit the indice when referring to the collection of variables over the indice. For example, $\mathbf{x}_t = (\mathbf{x}^0_t, \mathbf{x}^1_t, \ldots, \mathbf{x}^N_t)$ denotes the joint state of all agents at time step $t$ and $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K)$ the sequence of joint states over a time horizon which spans $K$ discrete steps. We use the superscript $\neg i$ to denote a collection of variables over all agents *except* agent $i$.

At every time step $t$, each agent influences the next joint state by applying the control input $\mathbf{u}^i_t$, respectively. We refer to the joint control input at time $t$ as $\mathbf{u}_t$. We assume that the joint state $\mathbf{x}_t$ is *markovian* and evolves dynamically according to the discrete-time dynamics

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t).$$

We seek to solve a motion planning problem for an under-actuated system:

$$\hat{\mathbf{u}}^0 = \arg\min_{\mathbf{u}^0} \sum_{k=0}^{K-1} c_k^0(\mathbf{x}_k, \mathbf{u}_k^0) + c_K^0(\mathbf{x}_K) \tag{8}$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k), && \text{(7a)} \\ & \mathbf{x}_0 = x_{\text{init}}, && \text{(7b)} \\ & \hat{\mathbf{u}}_k^i = \arg\min_{\mathbf{u}_k^i} c_k^i(\mathbf{x}_k, \mathbf{u}_k^i), && \text{(7c)} \\ & g^i(\mathbf{x}_k) \leq 0, && \text{(7d)} \\ & \forall i \in \{1, \ldots, N\}, \forall k \in \{0, \ldots, K-1\}, && \text{(7e)} \end{aligned}$$

where $\mathbf{x}_0$ denotes the joint state at the current time $t = 0$ and $\mathbf{u}^0 = (\mathbf{u}_0^0, \ldots, \mathbf{u}_{K-1}^0)$ is the sequence of robot control inputs. Collision avoidance constraints are imposed by (7d). With $\mathbf{u}^0$ the robot directly controls its state and indirectly influences $\mathbf{x}^{\neg 0}$ through (7c). The pedestrians' plans become a function of the robot's input. In contrast to formulating the interactions as a joint optimization, each pedestrian computes their best response to the other agents instead of trying to influence them. Note, that this formulation assumes that the pedestrians can estimate the robot's future states.

While the cost $c_k^0$ is a design parameter, the cost functions $c_k^i$ of the pedestrians are typically unknown. In this work, we propose to use the well-known SFM to provide analytical response dynamics instead of solving for the optimal pedestrian response $\tilde{\mathbf{u}}_k^i$. We consider a second-order point mass model for the dynamics of both the robot and the pedestrians.

## III. METHOD

In this section, we introduce the **So**cial-**F**orces-**I**nformed **I**nteraction-**A**ware **M**odel **P**redictive **C**ontrol (SoFIIA-MPC) framework. Instead of solving for the optimal pedestrian response at each iteration like [11], we make use of a response policy that implicitly encodes the pedestrians' cost function. Contrary to [3], we do not learn a policy of the other agents but make use of the well-established SFM.

### A. Social-Forces-Informed Interaction-Aware Model Predictive Control

We assume that a parameterized approximation $\hat{\pi}_\theta^i(\mathbf{x}_t) = \arg\min_{\mathbf{u}_k^i} c_k^i(\mathbf{x}_k, \mathbf{u}_k^i)$ of the pedestrians' response dynamics exists. This reduces the general multi-agent interaction problem to a single-agent optimization problem:

$$\hat{\mathbf{u}}^0 = \arg\min_{\mathbf{u}^0} \sum_{k=0}^{K-1} c_k^0(\mathbf{x}_k, \mathbf{u}_k^0) + c_k^0(\mathbf{x}_K) \tag{8}$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k), && \text{(8a)} \\ & \mathbf{x}_0 = x_{\text{init}}, && \text{(8b)} \\ & \tilde{\mathbf{u}}_k^i = \hat{\pi}_{\theta^i}(\mathbf{x}_k), && \text{(8c)} \\ & g_k^i(\mathbf{x}_k) \leq 0, && \text{(8d)} \\ & \forall i \in \{1, \ldots, N\}, \forall k \in \{0, \ldots, K-1\}. && \text{(8e)} \end{aligned}$$

We approximate $\pi_\theta^i(\mathbf{x}_t)$ using the SFM resulting in

$$\hat{\pi}_\theta^i(\mathbf{x}_t) = p_{\text{dest}} \mathbf{f}_{\text{dest}}^i(\mathbf{x}_t) + p_{\text{static}} \mathbf{f}_{\text{static}}^i(\mathbf{x}_t) + p_{\text{dyn}} \mathbf{f}_{\text{dyn}}^i(\mathbf{x}_t).$$

### B. Cost Function

We consider two cost function designs: one part addresses the costs related to the ego agent $c_{\text{ego}}$, and the other part aims to influence the behavior of other agents $c_{\text{affect}}$. We design $c_{\text{ego}}$ to encourage the robot to reach its goal, to avoid collisions, and to maintain a velocity limit. The cost terms are defined as follows:

$$c_{\text{ego}} = c_{\text{goal},0} + c_{\text{vel-limit}} + c_{\text{collision}}, \tag{9}$$

$$\begin{aligned} c_{\text{affect}} = {} & (w_{ego} c_{\text{goal},0} + w_{others} c_{\text{goal}\neg\text{i}})/N \\ & + c_{\text{vel-limit}} + c_{\text{collision}}. \end{aligned} \tag{10}$$

Collisions between the robot and a pedestrian and velocities higher than a maximum velocity are penalized with a constant cost. The goal cost of agent $i$ is defined as

$$c_{\text{goal,i}} = ||\mathbf{p}_k^i - \mathbf{p}_g^i|| / ||\mathbf{p}_0^i - \mathbf{p}_g^i||, \tag{11}$$

and the goal cost of the other agents $\neg i$ is defined as

$$c_{\text{goal}\neg\text{i}} = \sum_{j \neq i}^{N} c_{\text{goal,j}}. \tag{12}$$

## IV. RESULTS

We consider two versions of our planner: SoFIIA-MPC using $c_{ego}$ and SoFIIA-MPC-affect using $c_{affect}$. Specifically, we consider a case with $w_{ego} = 0.8$ and $w_{others} = 1$. In this section, we compare our planners with the following baselines: 1) **MPC-CVM**: Predict-then-Plan approach. Assumes that the pedestrians continue moving with their current velocities and uses $c_{ego}$, 2) **MPC-SFM**: Predict-then-Plan approach. The behavior of the pedestrians is predicted using the SFM assuming that the robot also follows the SFM, uses $c_{ego}$.

We evaluate the different planners in simulation, with the pedestrian behavior modeled using the SFM. While the current evaluations do not provide insights into how well SoFIIA performs in environments with real pedestrians, these experiments offer valuable insights into the benefits of accounting for interactions in planning. MPC-SFM incorporates the true model for the simulated pedestrians. However, an assumption about the robot's future behavior has to be made to predict the pedestrians' future behaviors. To generate the SFM predictions, we assume that the robot behaves as an SFM agent.

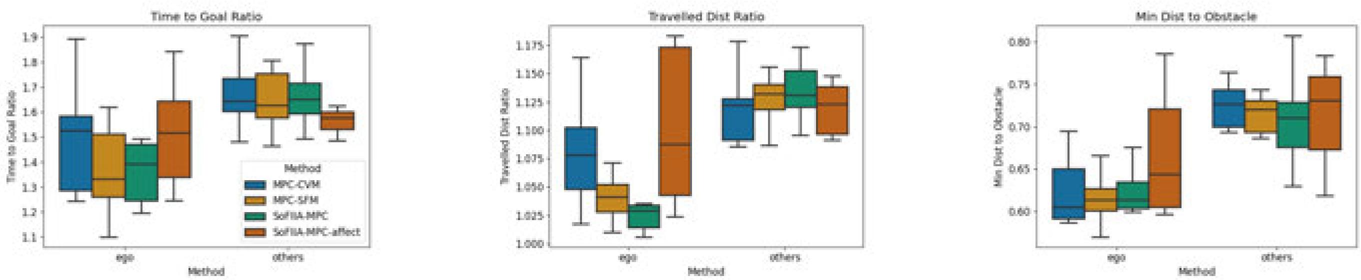


Fig. 2: Metrics over 10 random scenarios. We show each metric for the different planners separated by agent type, i.e., robot/ego-agent and other agents. For the other agents we consider the mean value.

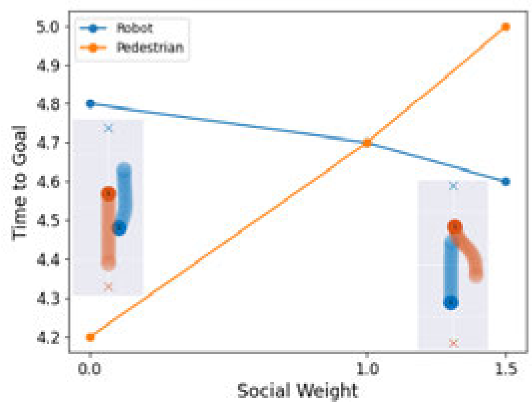


Fig. 3: SoFIIA-MPC: Time to goal over different social weights $p_{dyn}$ for head-on scenario with two agents.

Since we consider the robot as an SFM agent, the SFM predictions inherently account for interactions between the robot and the pedestrians. The CVM was shown to outperform even state-of-the-art learning-based prediction models [12] and was applied in many state-of-the-art motion planners [1]. We solve the MPC problem using the MPPI-torch implementation[2] [10] with a horizon $K$ of 20 time-steps, and a step-size $\Delta t = 0.1s$. Furthermore, we evaluate how the cost function can be used to adapt the behavior.

### *A. Comparative Analysis of Motion Planning: Non-Reactive vs. Reactive Agent Models*

We first show that by considering the interactions the robot applying SoFIIA-MPC can exploit the other agents, see Fig. 3. The achieved time to goal for the robot decreases with increasing socialness of the other agent.
Furthermore, we compare the navigation metrics over 10 random scenarios. We consider the following metrics:

- Travelled Distance Ratio: Distance to the goal divided by the straight line distance to the goal,
- Time to goal ratio: Time to goal divided by the time required to reach the goal in a straight line with maximum speed,
- Minimum distance: The minimum distance between agents.

The results, presented in Fig. 2, show that the Travelled Distance Ratio of the robot decreases as the planner accounts more for interactions.

### *B. Exploiting Interactions for Desired Agent Behaviors*

By explicitly considering the interactions, it is possible to influence the other agents to certain behaviors. While previous works in the autonomous driving field [11] demonstrate that cars can be slowed down or influenced to merge into another lane, these strategies are not directly applicable in the social navigation context. This is because the considered driving scenarios are more structured, e.g., by considering lanes. Thus, we set the robot's objective to navigate while trying to reduce its influence on the other agents. In Fig. 2 it can be seen that we were able to decrease the Time to Goal ratio for the other agents. However, these are preliminary results, which have to be further evaluated for a higher number of scenarios.

## V. CONCLUSION

In this work, we addressed the challenge of enhancing interaction in multi-agent motion planning while maintaining computational efficiency. Specifically, we formulate the interactions as an underactuated system and leverage the Social Forces Model (SFM) to represent pedestrians' response dynamics. Since our aim was to evaluate the effect of accounting for interactions, we assumed the parameters of the SFM as well as the pedestrians' goals to be known. How the parameters can be estimated remains to be explored. Future work will focus on further validating our approach in more diverse scenarios including static obstacles and in real-world scenarios.

TABLE I: Parameters

| | | | |
|---|---|---|---|
| SFM $A$ | 4.5 | SFM $\tau$ | 0.54 |
| SFM $\gamma$ | 0.35 | Horizon $K$ | 20 |
| SFM $n$ | 2.0 | Radius agents | 0.3 m |
| SFM $n'$ | 3.0 | Time step $\Delta t$ | 0.1 s |
| SFM $\lambda$ | 2 | Horizon $K$ | 20 |

[2]https://github.com/tud-airlab/mppi_torch/tree/main

## VI. Acknowledgements

This paper has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101017008. All content represents the opinion of the author(s), which is not necessarily shared or endorsed by their respective employers and/or sponsors.


## References


[1] Bruno Brito, Michael Everett, Jonathan P How, and Javier Alonso-Mora. Where to go next: Learning a subgoal recommendation policy for navigation in dynamic environments. *IEEE Robotics and Automation Letters*, 6 (3):4616–4623, 2021.

[2] Yuxiao Chen, Sushant Veer, Peter Karkus, and Marco Pavone. Interactive joint planning for autonomous vehicles. *IEEE Robotics and Automation Letters*, 2023.

[3] Jose Luis Vazquez Espinoza, Alexander Liniger, Wilko Schwarting, Daniela Rus, and Luc Van Gool. Deep interactive motion prediction and planning: Playing games with motion prediction models. In *Learning for Dynamics and Control Conference*, pages 1006–1019. PMLR, 2022.

[4] David Fridovich-Keil, Ellis Ratner, Lasse Peters, Anca D Dragan, and Claire J Tomlin. Efficient iterative linear-quadratic approximations for nonlinear multi-player general-sum differential games. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 1475–1481. IEEE, 2020.

[5] Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15303–15312, 2021.

[6] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[7] Simon Le Cleac'h, Mac Schwager, and Zachary Manchester. Lucidgames: Online unscented inverse dynamic games for adaptive trajectory prediction and planning. *IEEE Robotics and Automation Letters*, 6(3):5485–5492, 2021.

[8] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From goals, waypoints & paths to long term human trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15233–15242, 2021.

[9] Mehdi Moussaïd, Dirk Helbing, Simon Garnier, Anders Johansson, Maud Combe, and Guy Theraulaz. Experimental study of the behavioural mechanisms underlying self-organization in human crowds. *Proceedings of the Royal Society B: Biological Sciences*, 276(1668):2755–2762, 2009.

[10] Corrado Pezzato, Chadi Salmi, Max Spahn, Elia Trevisan, Javier Alonso-Mora, and Carlos Hernández Corbato. Sampling-based model predictive control leveraging parallelizable physics simulations. *arXiv preprint arXiv:2307.09105*, 2023.

[11] Dorsa Sadigh, Nick Landolfi, Shankar S Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for cars that coordinate with people: leveraging effects on human actions for planning and active information gathering over human internal state. *Autonomous Robots*, 42:1405–1426, 2018.

[12] Christoph Schöller, Vincent Aravantinos, Florian Lay, and Alois Knoll. What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robotics and Automation Letters*, 5(2):1696–1703, 2020.

[13] Lucas Streichenberg, Elia Trevisan, Jen Jen Chung, Roland Siegwart, and Javier Alonso-Mora. Multi-agent path integral control for interaction-aware motion planning in urban canals. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1379–1385. IEEE, 2023.

[14] Peter Trautman and Andreas Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 797–803. IEEE, 2010.

[15] Nathan Tsoi, Alec Xiang, Peter Yu, Samuel S Sohn, Greg Schwartz, Subashri Ramesh, Mohamed Hussein, Anjali W Gupta, Mubbasir Kapadia, and Marynel Vázquez. Sean 2.0: Formalizing and generating social situations for robot navigation. *IEEE Robotics and Automation Letters*, 7(4):11047–11054, 2022.

[16] Grady Williams, Paul Drews, Brian Goldfain, James M Rehg, and Evangelos A Theodorou. Information-theoretic model predictive control: Theory and applications to autonomous driving. *IEEE Transactions on Robotics*, 34(6):1603–1622, 2018.

[17] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021.