Social Momentum: A Framework for Legible Navigation in Dynamic Multi-Agent Environments

Christoforos I. Mavrogiannis
Sibley School of Mechanical &
Aerospace Engineering
Cornell University
Ithaca, NY
cm694@cornell.edu

Wil B. Thomason
Department of Computer Science,
Cornell University
Ithaca, NY
wbthomason@cs.cornell.edu

Ross A. Knepper
Department of Computer Science,
Cornell University
Ithaca, NY
rak@cs.cornell.edu

ABSTRACT

Intent-expressive robot motion has been shown to result in increased efficiency and reduced planning efforts for copresent humans. Existing frameworks for generating intent-expressive robot behaviors have typically focused on applications in static or structured environments. Under such settings, emphasis is placed towards communicating the robot's intended final configuration to other agents. However, in dynamic, unstructured and multi-agent domains, such as pedestrian environments, knowledge of the robot's final configuration is not sufficiently informative as it completely ignores the complex dynamics of interaction among agents. To address this problem, we design a planning framework that aims at generating motion that clearly communicates an agent's intended collision avoidance strategy rather than its destination. Our framework estimates the most likely intended avoidance protocols of others based on their past behaviors, superimposes them, and generates an expressive and socially compliant robot action that reinforces the expectations of others regarding these avoidance protocols. This action facilitates inference and decision making for everyone, as illustrated in the simplified topological pattern of agents' trajectories. Extensive simulations demonstrate that our framework consistently achieves significantly lower topological complexity, compared against common benchmark approaches in multi-agent collision avoidance. The significance of this result for real world applications is demonstrated by a user study that reveals statistical evidence suggesting that multi-agent trajectories of lower topological complexity tend to facilitate inference for observers.

KEYWORDS

Navigation; Expressive motion; Multi-Agent Systems; Topology.

1 INTRODUCTION

Over the past decades, a significant amount of research has focused on enabling robots to navigate human environments in a smooth and socially compliant fashion. The social navigation problem is inherently decentralized and precludes explicit communication. To address this complication, several works have proposed planners based on trajectory prediction [1, 2, 21, 34, 38], whereas others have contributed planners based on cognitive heuristics [9, 13, 20, 22, 25, 28, 29, 35], behavioral dynamics [4] and learning features of socially compliant navigation strategies from demonstrations [7, 18, 23]. Recent advances in the fields of crowd simulation and decentralized multi-agent planning [15, 17, 24, 36] have motivated inspiration and exchange of ideas and concepts between the fields.

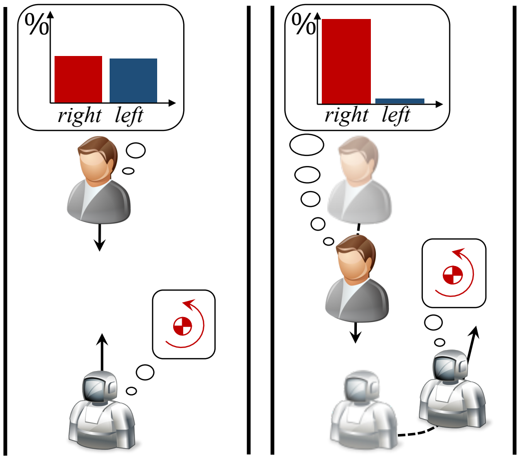


Figure 1: A human and a robot move towards opposing sides of a hallway. The initial configurations of the two agents make it hard for the human to predict the emerging avoidance strategy ("right" or "left"). The robot detects a slight advantage towards the "right" strategy and acts towards amplifying it and thus facilitating the inference of the human regarding the emerging (right) strategy of avoidance.

Despite the impressive performance benchmarks that these works have achieved, their deployment in human environments can yield undesired results. Humans tend to make inferences about the future behavior of others by attempting to attribute context-specific intentions to them [8] (see for example Fig. 1). Hence, robot motion that appears to lack a clear and consistent global intention complicates the inference for observers. Humans, driven by erroneous inferences, may then move in ways that are unexpected by the robot, which may in turn affect the subsequent inference and decision making of the robot, yielding an oscillatory pattern of joint behavior, commonly referred to as "reciprocal dance" [12].

The frequently observed emergence of this phenomenon highlights the need for equipping robots with models of human inference mechanisms. This could enable robots to anticipate the coupled dynamics of their interaction with humans and even act in a consistently readable fashion. This idea is not new; several collaborative robotics applications, including robot soccer [27], human-robot handovers [30], robot pointing [16], robot object picking [10], navigation [22] and language generation [31] have been centered around the generation of intent-expressive actions. These have inspired the design of further computational frameworks that generalize the generation of implicitly communicative actions to any collaborative task [19, 26]. Particular emphasis has been placed on the concept of *Legibility* [10] or *Readability* [6], a property of behavior that allows for fast and unambiguous inference of an actor's intent, which has been shown to lead to effective human-robot collaboration [10] and reduced planning effort for other agents [5, 6]. Dragan and Srinivasa [10] provided a general definition of *Legibility* as the property of motion that enables an *observer* to infer quickly and confidently the *goal* of an *actor*, given observation of the actor's past *action*.

Existing works on legible motion generation tend to associate the notion of a goal or intention with a point in a configuration space (e.g. [10, 22]). In a static and structured environment, where the dynamics of interaction among agents is predictable or known a priori, this is a well motivated modeling decision, as the observers' belief could be assumed to be an isolated relationship between an observed motion and a potential destination. However, in dynamic and unstructured environments, where the dynamics of interaction among multiple agents are complex, sole knowledge of an agent's destination may be insufficient to inform others of the agent's immediate behaviors. This highlights the need for a new consideration of legibility that captures the interactions with neighboring agents.

1.1 Contributions

In this paper, we present a planning framework for intent-expressive motion generation in dynamic multi-agent environments. Specifically, we contribute: (1) a novel topological consideration of legibility that captures the complexity of multi-agent dynamics; (2) a novel inference mechanism that reduces the trajectory prediction problem (continuous, infinite number of possible solutions) to the easier problem of topology prediction (discrete, bounded number of candidate solutions); (3) a cost-based planner, inspired by the cooperative nature of human navigation [37] and motivated by the goal-driven nature of human inference [8], that generates motion towards simplifying inference and planning for observers; (4) the introduction of a tool from low-dimensional topology for assessing the complexity of multi-agent trajectories and multi-agent motion planning; (5) extensive simulations demonstrating the topological and geometrical efficiency of our planner, compared against benchmarks in the area of multi-agent collision avoidance; (6) statistical evidence extracted from an online user study with human participants, demonstrating that executions of greater topological efficiency tend to be more *legible*, i.e., allow early and correct inference of agents' behaviors.

This paper constitutes a new step in our investigations of the use of topological methods and tools for modeling multi-agent interactions in navigation and planning socially competent robot behaviors in multi-agent domains. Our past works made explicit use of topological braids to plan socially competent robot behaviors [23, 24]. This work makes use of topological braids as an analytical tool for assessing the quality of multi-agent planning.

2 MULTI-AGENT TRAJECTORY ANALYSIS

In a scene where multiple agents navigate towards their destinations while avoiding collisions with each other, their decision making over time may be represented as a geometrical pattern, formed by the spatiotemporal superposition of their trajectories. The topological properties of this pattern are indicative of the interactions that the agents have with each other throughout the course of the scene. In particular, the way agents avoid one another, by passing on the *left*, *right*, *front* or *back* of others results in an "entanglement" of their trajectories over time. Depending on the navigation strategies that agents follow, the complexity of this entanglement may range.

> *We hypothesize that the complexity of the entanglement of the trajectories of multiple agents that navigate simultaneously a workspace is related to the planning effort they spend.*

In particular, we aim to show that legible behaviors in multi-agent navigation result in trajectory entanglements of low complexity. This will allow us to employ as a measure of topological complexity as a proxy for assessing the legibility of multi-agent behaviors. Based on the work of Carton et al. [6], who showed that legible behaviors in navigation reduce the required planning effort of navigating agents, we aim to show that trajectories of low topological complexity require low planning effort for agents.

In the following subsections, we present a general model of multi-agent navigation, briefly recap our topological abstraction of collective multi-agent behavior [23, 24] and introduce a tool for evaluating the topological complexity of multi-agent trajectories.

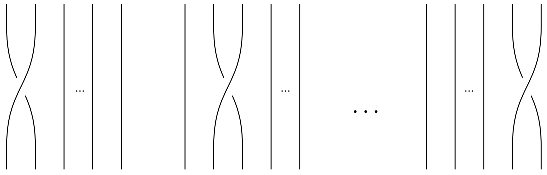
2.1 A Model of Multi-Agent Navigation

A set of agents $N = \{1, \dots, n\}$, with $n \geq 2$, navigate towards their intended destinations in a workspace $Q \subset \mathbb{R}^2$. Let us denote by $q_i \in Q$ the configuration of agent $i \in N$ and by v_i its velocity. Agent i starts from an initial configuration $q_i^s \in Q$ at time $t = 0$ and reaches a final configuration q_i^d at time $t = T_i$. We define the system state as a tuple $Q = (q_1, \dots, q_n) \in Q^n$. The system evolves from an initial configuration $Q^s = \left(q_1^s, \dots, q_n^s\right)$ to a goal configuration $Q^d = \left(q_1^d, \dots, q_n^d\right)$. These configurations define a *scenario* $\mathcal{S} = \left(Q^s, Q^d\right)$.

Executing a scenario requires that each agent $i \in N$ follows a collision-free trajectory $\xi_i : [0, T_i] \rightarrow Q$ that starts from $\xi_i(0) = q_i^s$ and reaches $\xi_i(T_i) = q_i^d$ at time $T_i > 0$. This results in a system trajectory $\Xi : [0, T] \rightarrow Q^n \backslash \Delta$, where $T = \max_{i \in N} T_i$ (it is assumed that agents remain stationary once they reach their destinations) and $\Delta = \{Q = (q_1, q_2, \dots, q_n) \in Q^n : q_i = q_j \text{ for some } i \neq j \in N\}$ is the set of states in collision. Δ partitions the space of system paths that execute the scenario $\mathcal{S}$, $\mathcal{Z}^{\mathcal{S}}$, into a set $\mathcal{T}^{\mathcal{S}}$ of classes of homotopically equivalent system paths (paths that can be continuously deformed to each other). Each one of these classes corresponds to a topologically distinct global strategy of collision avoidance that the system may follow to execute the *scenario* $\mathcal{S}$. Although the system of agents is not acting under a centralized decision making mechanism, the individual actions of agents form a collective behavior with the topological properties of a strategy $\tau \in \mathcal{T}^{\mathcal{S}}$. Following our past work [23, 24], we formally model the set of strategies $\mathcal{T}$ as the braid group [3]. The following subsections provide a brief introduction to braids and recap our topological abstraction of multi-agent navigation.

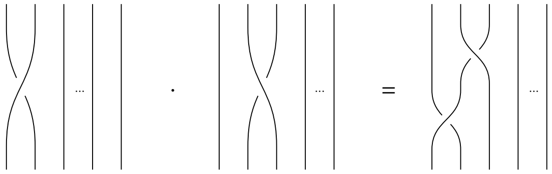
2.2 Background on Braids

Denote by x, y, z the Cartesian coordinates of a Euclidean space $\mathbb{R}^2 \times I$, where $I = [0, 1]$. A braid string is a curve $X(z) : I \rightarrow \mathbb{R}^2$ that increases monotonically in z, i.e., has exactly one point of



(a) σ_1 (b) σ_2 (c) σ_{n-1}

Figure 2: The generators of the Braid Group B_n.



σ_1 σ_2^{-1} $\sigma_1 \cdot \sigma_2^{-1}$

Figure 3: Composition operation $\sigma_1 \cdot \sigma_2^{-1}$ for $\sigma_1, \sigma_2^{-1} \in B_n$.

intersection $X(z) = (x, y)$ with each plane $z \in I$. A braid on n-strings or n-braid is a set of n strings $X_i(z)$, $i \in N = \{1, \ldots, n\}$ for which:

(1) $X_i(z) \neq X_j(z)$, for $i \neq j \; \forall z \in \mathbb{R}$
(2) $X(0) = (i, 0)$ and $X(1) = (p(i), 0)$,

where $p(i)$ is the image of an element $i \in N$, through a permutation $p : N \to N$ from the set of permutations of N, $Perm(N)$, defined as:

$$p = \begin{pmatrix} 1 & 2 & \ldots & n \\ p(1) & p(2) & \ldots & p(n) \end{pmatrix}. \tag{1}$$

The set of all braids on n strings, along with the composition operation, form a group B_n. The group may be generated from a set of $n-1$ elementary braids $\sigma_1, \sigma_2, \ldots, \sigma_{n-1}$ (see Fig. 2), called the generators of B_n, that satisfy the following *relations*:

$$\sigma_j \sigma_k = \sigma_k \sigma_j, \qquad |j - k| > 1, \tag{2}$$

$$\sigma_j \sigma_k \sigma_j = \sigma_k \sigma_j \sigma_k, \qquad |j - k| = 1. \tag{3}$$

A generator σ_i, $i \in \{1, 2, ..., n-1\}$ can be described as the crossing pattern that emerges upon exchanging the ith string (counted from left to right) with the $(i+1)$th string, such that the initially left string passes *over* the initially right one, whereas the inverse element, σ_i^{-1}, implements the same string exchange, with the difference that the left string passes over the right (see Fig. 3). An identity element, e, is a braid with no string exchanges. Two braids $b_1, b_2 \in B_n$ may be composed through the composition operation $(\cdot)$, which is algebraically denoted as a product $b_1 \cdot b_2$. Geometrically, this composition results in the pattern that emerges upon attaching the lower endpoints of b_2 to the upper endpoints of b_1 and shrinking each braid by a factor of 2, along the z axis (Fig. 3). Any braid can be written as a product of generators and generator inverses, a representation commonly referred to as a *braid word* (Fig. 3).

2.3 Braids of Entangled Agent Trajectories

Inspired by Thiffeault [32], who uses braids to analyze the trajectories of particles in fluids, we employ braids to analyze the trajectories that emerge in multi-agent scenarios.

Denote by $f_x : \mathcal{Q}^n \to Perm(N)$ a function that takes as input the system state $Q \in \mathcal{Q}^n$ and outputs a permutation $p \in Perm(N)$ corresponding to the arrangement of all agents in order of increasing x-coordinates. As the agents move towards their destinations, they employ navigation strategies – maneuvers to avoid collisions, forming collectively a system trajectory Ξ. By evaluating f_x on the waypoints of Ξ, we may obtain a corresponding path of permutations $\pi : [0, T] \to Perm(N)$. This path can be simplified into a sequence of permutations of minimal length, $\pi^* = (p_0, \ldots, p_K)$, i.e., consecutive waypoints $p_{j-1} \neq p_j$, $\forall j = \{1, \ldots, K\}$ are adjacent transpositions (i.e., they differ from each other by exactly one swap of adjacent elements). Due to continuity, a transition from the $(j-1)$th permutation, p_{j-1}, to the jth permutation, p_j, implies the occurrence of an *event* τ_j, which may be described as *the intersection of the x-projections of the paths of two agents that were adjacent in the permutation* p_{j-1}. The event τ_j may be represented as an elementary braid $\tau_j \in \sigma_i^{\pm 1}$, $i \in \{1, ..., n-1\}$, where i corresponds to the index of the leftmost swapping agent in permutation p_{j-1}.

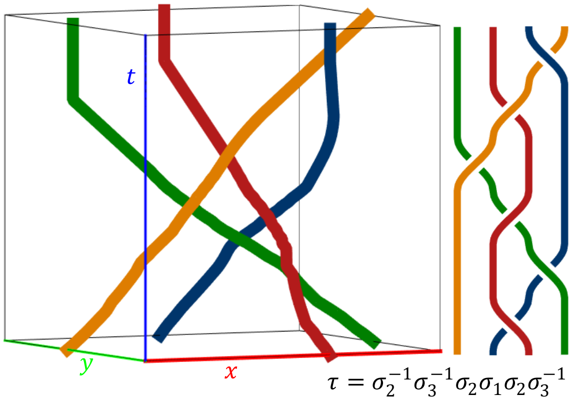

t
y
x
$\tau = \sigma_2^{-1}\sigma_3^{-1}\sigma_2\sigma_1\sigma_2\sigma_3^{-1}$

Figure 4: Space-time representation of a system path of 4 agents along with its corresponding braid diagram (right) and braid word (down), defined with respect to the path's x-projection. The visualization of the braid diagram and the extraction of the braid word were done using BraidLab [33].

Hence, the whole execution from $t = 0$ to $t = T$ may be abstracted into the braid that corresponds to the temporal sequence of events:

$$\tau = \tau_1 \tau_2 \ldots \tau_K \in B_n. \tag{4}$$

This braid word may serve as a compact abstraction of the system trajectory but also as a symbolic representation of the class of trajectories that are homotopy-equivalent with Ξ, i.e., the class of paths that execute the same scenario as Ξ and can be continuously deformed into each other. Fig. 4 illustrates our method for transitioning from a system trajectory into a braid word. The *braid word* representation retains the topological properties of the execution (relative locations of agents), while discarding any geometric properties (e.g. distances among agents). In the remainder of this paper, we will be referring to the sequence τ as the *topology* of the system trajectory. Essentially, we model the space of trajectory topologies $\mathcal{T}$ as the braid group, i.e., $\mathcal{T} := B_n$.

2.4 Complexity of Braids

We are interested in quantifying the topological complexity of a multi-agent trajectory Ξ. Assuming that such a trajectory may be abstracted into a braid word τ, with the method discussed in the previous subsection, an intuitive measure of complexity for Ξ would be the braid word *length* $l(\tau)$, defined as the number of generators that form it. In general, topologically complex braids correspond to braid words of larger *length* and equivalently, longer words indicate a higher topological complexity (see Fig. 4 for intuition on

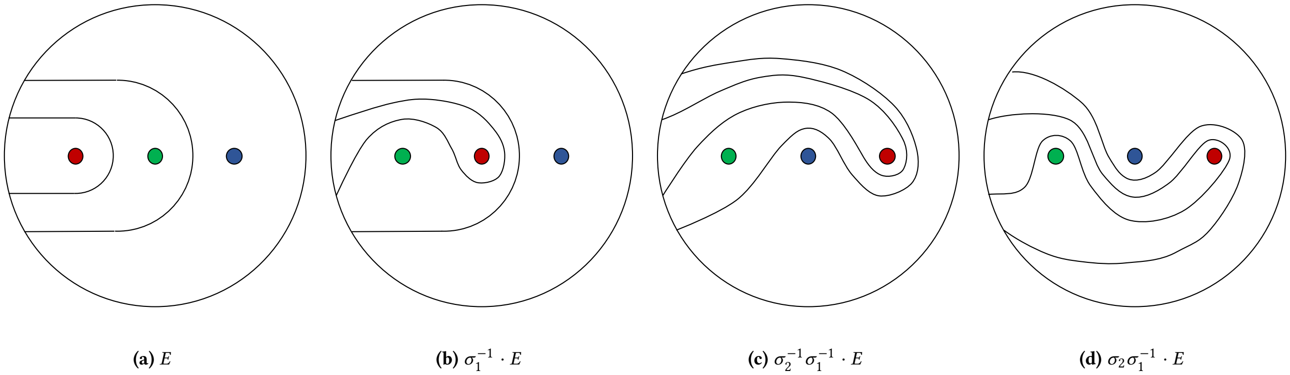

(a) E (b) $\sigma_1^{-1} \cdot E$ (c) $\sigma_2^{-1}\sigma_1^{-1} \cdot E$ (d) $\sigma_2\sigma_1^{-1} \cdot E$

Figure 5: Curve Diagrams of braids from B_3 on a disk with 3 punctures. From left to right, curve diagrams of: a trivial braid $\beta_a = e$ (Fig. 5a), $\beta_b = \sigma_1^{-1}$ (Fig. 5b), $\beta_c = \sigma_2^{-1}\sigma_1^{-1}$ (Fig. 5c) and $b_d = \sigma_2\sigma_1^{-1}$ (Fig. 5d), with complexities $c(\beta_a) = 0$, $c(\beta_b) = 1$, $c(\beta_c) = 1.5850$, $c(\beta_d) = 2$ respectively.

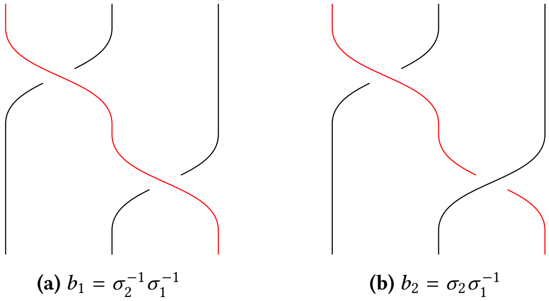

(a) $b_1 = \sigma_2^{-1}\sigma_1^{-1}$ (b) $b_2 = \sigma_2\sigma_1^{-1}$

Figure 6: Two braids of the same length, $b_1 = \sigma_2^{-1}\sigma_1$ and $b_2 = \sigma_2^{-1}\sigma_1^{-1} \in B_3$, with different entanglements. Qualitatively, it may be observed that b_2 is more complex than b_1. Fig. 5 formalizes the complexity measurement, using the Topological Complexity Index, defined on curve diagrams.

the relationship between braid length and topological complexity). However, the amount of *entanglement* induced by distinct braids of same length may vary. For example, consider the braids b_1 and b_2 shown in Fig. 6. Although $l(b_1) = l(b_2)$, it may be observed qualitatively that the entanglement of b_2 is more intense (or less trivial) than that of b_1, due to the action of its third strand. This observation is an intuitive indication that the braid word length cannot be used unambiguously as a universal complexity index.

To overcome this degeneracy, Dynnikov and Wiest [11] introduce the braid *Complexity Index*. This index quantifies the amount of entanglement induced on a canonical *curve diagram* (Fig. 5a) upon its application on it. An n-braid $b \in B_n$ may be *applied* on a canonical n-curve diagram by sequentially enforcing the braid's topological pattern on the diagram's fields (e.g. Fig. 5b, Fig. 5c, Fig. 5d). Intuitively, the canonical curve diagram can be thought of as a heterogeneous mixture, comprising n clearly separated substances. The application of a braid b can be thought of as the enforcement to the substances of a mixing pattern that follows the entanglement described by the braid b. In other words, the strings of the braid are matched to the substances of the curve diagram; as the braid progresses along the z-axis, the substances move with it, resulting in a new curve diagram. In the following subsections, we formally define curve diagrams and the braid *Complexity Index* [11].

2.4.1 *Curve Diagrams.*

Denote by D^2 the closed unit disk, centered at the origin of the complex plane $\mathbb{C}$ and let P_n be a set of n points, uniformly distributed along the intersection of the real axis with the disk, $\mathbb{R} \cap D^2$. The set of points in P_n are called *punctures*, whereas the set $D_n = D^2 \backslash P_n$, i.e., the region of the unit disk upon the removal of the punctures is called *punctured disk*. Finally, denote by E the union of $n - 1$ disjoint arcs on D_n, separating all the punctures, as shown in Fig. 5a. A *curve diagram* of a braid $\beta \in B_n$ is the image $D = \beta \cdot E$ of E under the homeomorphism corresponding to β. The image D is the union of arcs obtained from E through the action of β. This is only defined up to isotopies fixing ∂D^2 and P_n.

2.4.2 *The Complexity Index.*

The *norm* of a curve diagram D is defined as the number of intersections of D with the real axis and denoted as:

$$||D|| = \#(D \cap \mathbb{R}). \tag{5}$$

The *Complexity Index* of a braid $\beta \in B_n$ is then defined through the use of its corresponding curve diagram $D = \beta \cdot E$, as:

$$c(\beta) = \log_2(||\beta \cdot E||) - \log_2(||E||). \tag{6}$$

This expression is equal to the logarithm of the *gain* of intersections with the x-axis, upon the application of a braid. Looking at Fig. 5, it may be verified that the higher the number of intersections with the real axis, the higher the intensity the entanglement of the corresponding braid.

2.5 Complexity of Multi-Agent Planning

The *Complexity Index* quantifies the amount of entanglement that a braid induces to the canonical curve diagram. In this work, we employ braids as a representation of the topological pattern that a collection of trajectories forms. Thus, the *Complexity Index* may be used as a measure of the complexity of the braid corresponding to a specific execution of a multi-agent scenario. More importantly, it may serve as a characterization of the complexity of the solution to the motion planning problem of transitioning safely from Q^s to Q^d that the navigating agents converged to, without explicitly communicating with each other.

The setup of our problem is quite simple: agents move between two resting positions (from start to goal). However, the lack of explicit communication among them may result in complicated collision avoidance maneuvers, yielding undesired oscillatory behaviors but also potentially undesirably long and entangled paths for one or more agents. The former problem, commonly referred to as the “reciprocal dance” problem [12], has been widely addressed in the literature (see e.g. Trautman et al. [34]). The latter problem though, to the best of our knowledge, has not been modeled or approached appropriately. In this paper, we address it by (1) using the *Complexity Index* as a proxy to quantify the complexity of multi-agent planning, (2) proposing a planning framework that explicitly incorporates this understanding into the decision making stage, towards generating *legible* behaviors that reinforce plans of low trajectory entanglement and (3) investigating the effect of trajectory entanglement to human inference through a user study.

3 THE SOCIAL MOMENTUM PLANNING FRAMEWORK

In this section, we present a planner that enables agents to contribute to trajectory patterns of low topological entanglement, as they navigate towards their destination. Our planner is based on a cost function that detects the intentions of other agents over pairwise collision avoidance protocols (e.g. right or left) and favors actions that are in compliance with them. The cost function, named the *Social Momentum* cost, is defined as the weighted sum of the magnitudes of the pairwise angular momenta between the planning agent and all others. The optimization of the *Social Momentum* cost results in motion that tends to reinforce the current momenta by locally maximizing their magnitudes along their current directions, which corresponds to reinforcement of the currently established pairwise collision avoidance protocols between the planning agent and others. Our *Social Momentum* (SM) planning algorithm compromises between the *Social Momentum* cost and an *Efficiency* cost that drives the planning agent towards its destination. Throughout consecutive time steps, this policy results in a behavior that appears to be consistently compliant with the agent's past behavior and with the preferences of others over avoidance strategies. Effective communication of the agents' intended avoidance strategies results in behaviors that are easy to read and thus enable agents to implicitly cooperate efficiently to avoid each other, which leads to avoiding redundant trajectory entanglement.

3.1 Angular Momentum for Collision Avoidance

Consider two agents *A* and *B* moving towards opposing sides of a hallway, as shown in Fig. 1. The geometry of the shared space renders agents' decisions coupled. In order to reach their destinations in a collision-free and socially acceptable fashion, they need to (1) agree on a passing side (right or left) and (2) respect the personal space [14] of each other by maintaining a comfortable minimum distance. To quantify how well the agents are doing with respect to both of these specifications, we construct an analogy with the physical quantity of Angular Momentum. Assuming unit masses for the two agents, the angular momentum of their system with respect to its center of mass *C*, may be defined as:

$$L^{AB} = r_A^C \times v_A + r_B^C \times v_B \tag{7}$$

where

$$r_A^C = q_A - r_C, \qquad r_B^C = q_B - r_C \tag{8}$$

are agents' positions, defined with respect to their center of mass

$$r_C = (q_A + q_B)/2. \tag{9}$$

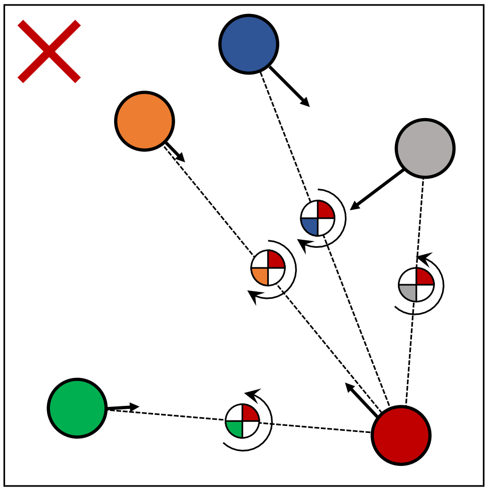


Figure 7: Social Momentum: The planning agent (red color) is moving towards the red target X, while complying with its pairwise momenta with all other agents.

For a system of agents navigating on the horizontal plane, the angular momentum is a vector perpendicular to the workspace, pointing along the positive direction of the *z*-axis for counterclockwise agent rotations and along the negative direction of the *z*-axis for clockwise rotations, thus encoding the right and left passings respectively. Its magnitude depends on the distance between the agents and also on the angle of their velocities, with larger distances and antiparallel velocities scoring higher. As a result, the angular momentum may be used (1) as a tool to monitor an emerging avoidance protocol (right/left passing) but also (2) as a tool to generate easily interpretable avoidance maneuvers in compliance with the preferences of the other agent and in consistency with previous behaviors of the agents.

3.2 Social Momentum for Legible Collision Avoidance

In a crowded multi-agent workspace, an agent interacts with multiple others at the same time, in the sense that every action taken broadcasts signals of intentions or preferences over avoidance strategies. Our framework enables an agent to read these preferences, associate them with its own, and act competently towards simplifying everyone's decision making. To this end, we introduce a novel index, comprising a weighted sum of the magnitudes of all pairwise momenta between the planning agent and the set of all other agents $N_i = N \backslash \{i\}$. Higher values indicate a higher certainty over the emerging pairwise avoidance protocols between the agent and all others. We call this cost the *Social Momentum* cost and formally define it for agent *i* as a real function $\mathcal{L} : \mathcal{A} \to \mathbb{R}$ over the agent's action space $\mathcal{A}$, as follows:

$$\mathcal{L}(a) = \begin{cases} \sum\limits_{j \in N_i} w_j ||\hat{L}^{ij}(a)||, & \text{if } sign\left((L^{ij})^T \hat{L}^{ij}(a)\right) \ \forall j \in N_i \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $\hat{L}^{ij}(a)$ denotes the expected pairwise momentum between agents *i* and *j*, upon agent *i* taking an action in consideration, $a \in \mathcal{A}$ and agent *j* moving with its current velocity, L^{ij} is their current momentum and $w_j \in \mathbb{R}$ is a weight, computed as the inverse of the distance between agents i and j. The quantity $sign\left((L^{ij})^T\hat{L}^{ij}(a)\right)$ indicates whether the expected evolution of the pairwise momentum between agents i and j is in compliance with their current momentum L^{ij}. A positive sign corresponds to an action that preserves the current momentum sign and thus the currently preferred pairwise avoidance protocol. A negative sign indicates inversion of the established pairwise avoidance protocol, which is undesired. For this reason, an action that results to inversion of a pairwise momentum is assigned a score of zero. Note that the only non-zero components of all pairwise momenta are their z-components, since we assume that the workspace is a horizontal plane.

Algorithm 1 SM($Q, \mathcal{A}, d, map, AtGoal, a$)

```
Input: $Q$ – current system state; $\mathcal{A}$ – action set; $d$ – agent's destination region; map; $AtGoal$ – boolean variable signifying arrival at agent's destination
Output: $a$ – action selected for execution
while ¬AtGoal do
    $\mathcal{A}_{cf} \leftarrow Collision\_Checking(\mathcal{R}, Q, \mathcal{A}, map)$
    $\mathcal{R} \leftarrow Get\_Reactive\_Agents(Q)$
    if $\mathcal{R} \neq \varnothing$ then
        $a \leftarrow Get\_Legible\_Action(Q, \mathcal{A}_{cf}, \lambda, d)$
    else
        $a \rightarrow Get\_Efficient\_Action(\mathcal{A}_{cf}, d)$
return $a$
```

3.3 Decision Making

In this section, we present the *Social Momentum* (SM) algorithm, a cost-based planning algorithm, built around the *Social Momentum* heuristic. The algorithm is based on frequent replanning; at every planning cycle, it picks an action that corresponds to the optimal compromise between progress to agent's destination and legible avoidance of others. We formalize this decision making strategy into the following optimization scheme:

$$a^* = \underset{a\in\mathcal{A}}{\operatorname{argmax}} \{\lambda\mathcal{E}(a) + (1-\lambda)\mathcal{L}(a)\}, \tag{11}$$

where $\lambda \in \mathbb{R}$ is a parameter accounting for proper scaling and weighting of the two quantities. We model the progress function $\mathcal{E} : \mathcal{A} \rightarrow \mathbb{R}$ to be the inverse of the length of the unobstructed line to destination. The action space $\mathcal{A}$ comprises a pre-sampled set of actions of finite duration that are executable by the agent.

Algorithm 1 describes the *SM* algorithm in pseudocode format. At each replanning cycle, the function `Collision_Checking` checks for collisions with other agents or bounds and returns a set $\mathcal{A}_{cf} \subseteq \mathcal{A}$ of collision-free actions. Then function `Get_Reactive` determines the subset of agents $\mathcal{R}$ to which the planning agent should be reacting: only agents that lie in front of the planning agent are considered (see Fig. 7). In case $\mathcal{R} \neq \varnothing$, the planning agent determines a legible action a by compromising between *Progress* to destination and *Social Momentum* (function `Get_Legible_Action`); otherwise, the algorithm switches to progress maximization mode (function `Get_Efficient_Action`). Termination occurs once the algorithm reaches a desired distance to destination.

4 EVALUATION

In this section, we evaluate our approach by investigating the following two hypotheses:

(1) "The Social Momentum Framework produces multi-agent trajectories of significantly simpler topological entanglement than existing approaches of multi-agent planning".
(2) "In multi-agent domains, multi-agent trajectories of simple topological entanglement are more legible".

Confirmation of these two hypotheses provides evidence that the Social Momentum framework results in legible navigation in multi-agent environments. To test hypothesis (1), we perform an extensive simulated evaluation. To test hypothesis (2), we conduct a study in which we ask users to predict the evolution of simulated multi-agent scenarios from partial observation.

4.1 Simulations

We evaluate our planner in simulation by comparing against common benchmarks in multi-agent planning. Specifically, we consider the *Social Force* (SF) model [15] and the *Optimal Reciprocal Collision Avoidance* (ORCA) framework [36]. It should be noted that these frameworks were designed to produce fast and realistic simulations of multi-agent navigation scenarios and not to produce legible behaviors. However, they still constitute relevant baselines due to (a) their wide dissemination and existence of ready implementations, (b) their proven performance in various types of scenarios, (c) the fact that they constitute common benchmarks and thus common works of reference, (d) their ability to handle any number of agents.

4.1.1 Experimental Setup. We consider multi-agent scenarios involving sets of homogeneous agents navigating a circular workspace. Each scenario is generated as follows: (1) the workspace circumference is partitioned to n arcs of equal length, (2) each arc is assigned to an agent, (3) the agent is placed at a random, collision-free position on the arc assigned to it and (4) each agent is assigned a destination that is antipodal to its starting location and lies on the workspace circumference. The workspace considered is a circle with a diameter of $5m$, whereas the agents were discs of diameter $0.6m$. This setup was specifically selected as it reinforces the emergence of challenging agent encounters. The occurrence of such scenarios highlights the value of intent-expressiveness as a feature that reduces uncertainty by reinforcing implicit coordination. We consider 4 different classes of scenarios, each corresponding to a different number of agents, ranging from 3 to 6. For each class, we generate 200 intense multi-agent scenarios at random and execute them with each of the planners considered. Note that each of the planners considered can be tuned to yield qualitatively different behaviors. In order to ensure a fair comparison, we assumed similar tunings with respect to sensitivity to obstacles and kept them constant across all trials.

4.1.2 Quality Measures. To evaluate the quality of an execution, we consider two different criteria: (1) the braid *Complexity Index*, which serves as a measure of the topological efficiency of the execution and (2) the *Path Irregularity* index [13], defined as the total amount of unnecessary rotation (divergence of agent's heading from its direction to its destination) per unit path length, averaged per agent, which serves as a measure of the geometrical efficiency of the execution. *Complexity Index* computations were implemented

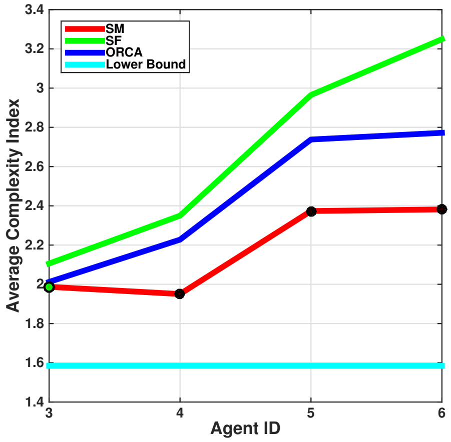


Figure 8: Average *Complexity Index* of trajectories generated by executing 200 scenarios with 3, 4, 5 and 6 agents with the *Social Momentum* (SM), *Social Force* (SF) and *Optimal Reciprocal Collision Avoidance* (ORCA) models. A theoretical lower bound baseline is also included for reference. Datapoints marked black correspond to significantly lower average *Complexity* of SM than both SF and ORCA, whereas the datapoint marked green indicates significantly lower average *Complexity* of SM than SF, according to paired Student's T-test. Test statistics can be found at table 1.

using the BraidLab package [33], assuming a projection onto the global x-t coordinate plane.

In the evaluation of the topological entanglement, we also include a theoretical *Lower Bound* baseline, which returns a topological path of minimal topological entanglement that executes the scenario in consideration. This baseline may be described as follows: (1) find the minimal path of transpositions, π^* that connects the initial permutation of the system, $p^s = f_x(Q^s)$, with the final permutation of the system, $p^d = f_x(Q^d)$ and (2) derive a sequence of generator transitions for all consecutive waypoints in π^*, i.e., a braid $\beta^* \in B_n$ that yields the lowest *Complexity Index* $c(\beta^*)$ for the scenario. This baseline can be thought of as an ideal case of perfect communication and compliance (or centralized planning).

4.1.3 Performance Comparison. Fig. 8 depicts the average *Complexity Index* for each planner and class of scenarios considered. The *Complexity Index* of SF and ORCA appears to be consistently rising with the number of agents. In contrast, SM exhibits a slower rise; the transitions between 3 and 4 agents and between 5 and 6 agents are done with almost constant complexity, with the only rise taking place in the transition between 3 and 4 agents. Overall, SM achieves consistently lower topological entanglement with statistical significance, except from the case of 3 agents, where the scenarios are not geometrically challenging to yield significantly diverse behaviors. Detailed statistics of paired t-tests conducted for the SM-SF and SM-ORCA pairs are reported in table 1. Despite this result, the theoretical Lower Bound consistently outperforms all planners, providing an illustrative demonstration of their suboptimality in terms of *topological efficiency* which reflects the price of no explicit communication in multi-agent planning. Note that the constant *Complexity Index* value of 1.5850 that the Lower Bound achieves is an artifact of the symmetry of the considered scenarios (agents traveling to antipodal points in the workspace).

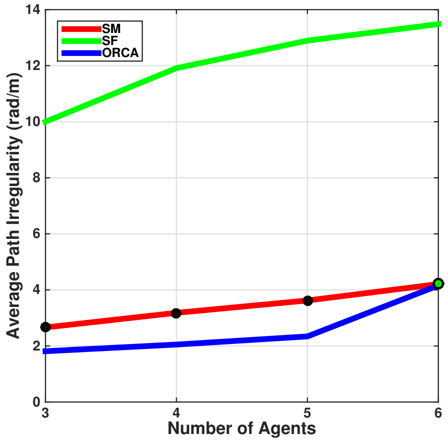


Figure 9: Average *Path Irregularity* per agent extracted by executing 200 scenarios with 3, 4, 5 and 6 agents. Datapoints marked black denote different irregularity of SM than both SF and ORCA, whereas the datapoint marked green indicates lower irregularity of SM than SF, according to paired Student's T-tests. Test statistics can be found at table 1.

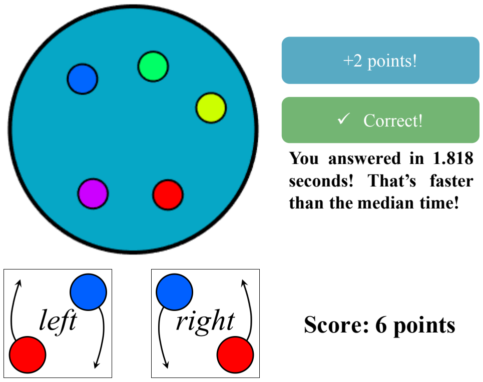
t-Tests		*Complexity Index*		*Path Irregularity*	
Agents	*Pair*	t-value	p-value	t-value	p-value
3	SM-SF	-2.497	0.013	-26.397	<0.001
	SM-ORCA	-0.593	0.553	9.197	<0.001
4	SM-SF	-7.963	<0.001	-34.514	<0.001
	SM-ORCA	-5.740	<0.001	17.336	<0.001
5	SM-SF	-9.424	<0.001	-41.400	<0.001
	SM-ORCA	-5.395	<0.001	7.934	<0.001
6	SM-SF	-11.561	<0.001	-51.430	<0.001
	SM-ORCA	-5.250	<0.001	0.152	0.879

Table 1: Statistics of paired t-tests between SM and SF, ORCA for different agent numbers. We considered $N - 1$ degrees of freedom, where $N = 200$ is the number of scenarios per class.

Fig. 9 depicts the average Path Irregularity per agent, for each planner and class of scenarios considered. Although for all planners the path irregularity rises with the number of agents, each planner performs differently. The different performance of each planner is indicative of the distinct philosophy with which they have been designed. SF, lacking predictive mechanisms yields significantly more irregular paths than SM and ORCA. ORCA achieves consistently the lowest path irregularity, as a result of its geometrically optimal behavior, which in practice results in minimal divergence from the unobstructed line connecting an agent with its destination at any time. SM performs slightly worse ORCA, as a result of its different consideration of collision avoidance as a rotation; SM agents diverge from their shortest paths more often to convey intent. For the case of 6 agents the geometric complexity of the scenarios is too intense even for ORCA which performs almost equally to SM.

4.2 User Study

We conducted a user study, in which we asked users to watch a series of videos (shown from a top view) of simulated executions of scenarios involving 5 agents navigating a circular workspace. For



You answered in 1.818 seconds! That's faster than the median time!

Score: 6 points

Figure 10: Study interface: A video of a scenario execution is shown and users predict how the red agent is going to avoid the blue agent by pressing the corresponding button at the bottom. The display of user's score and performance statistics aim to to incentivize fast and accurate responses.

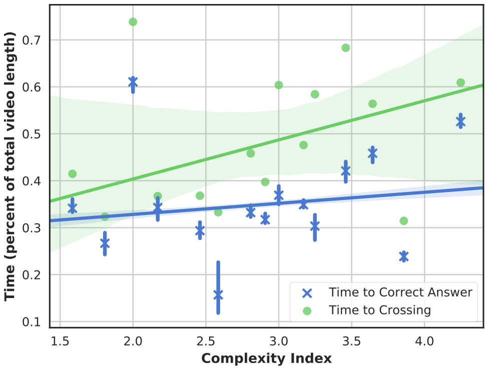


Figure 11: Relation between the *Complexity Index* and (a) time until two specific agents pass each other (green points/line) and (b) median time until users give a correct prediction of the passing (blue crosses/line). Times are normalized to the total length of the relevant video.

each video, users were asked to predict the way two agents were going to avoid each other (right or left side). Speed and correctness (the basis of the legibility definition) were incentivized through a scoring system that awarded points for quick and accurate answers and deducted points for wrong or slow responses (Fig. 10 depicts the study interface). The study used a total of 15 videos, with duration ranging from 6.3 to 15.7 seconds, corresponding to scenarios of *Complexity Index* uniformly ranging from 1.585 to 4.250. More than 180 users, recruited from the social media platforms of Reddit and Facebook, contributed a total of 2704 video views and clicks. An analysis of the collected dataset is presented in Fig. 11.

The blue trend shows the relation between the *Complexity Index* and the median time of correctly predicting the side on which one agent will pass another. We fit a linear model to the data using iteratively reweighted least squares, shown in Fig. 11 as a blue line with a 95% confidence interval. The effect of the *Complexity Index* on click time is positive, with a slope of 0.0236, and significant (Student's t-test, $t = 5.60$, $p < 0.001$). In other words, as the topological entanglement intensifies, users take more time to accurately predict the side of passing, i.e., more complex scenarios are less legible. We verified that the rate of incorrect answers for a video is not correlated with the *Complexity Index* of that video via computation of a Pearson correlation coefficient ($r^2 = 0.1017$, $p = 0.7185$).

The green trend shows the relation between the *Complexity Index* and the time of passing between the two agents. We fit a linear model to the data, shown as the green line with a 95% confidence interval. The trend is positive (slope of 0.0833) and nearly significant ($t = 1.93$, $p = 0.0538$). Increased *Complexity Index* correlates positively with increased time of passing, and thus with longer, less efficient interactions. For each video, most users were able to correctly predict the passing side before the passing occurred. We see a trend towards predicting the passing with greater lead times for greater *Complexity Index*. However, we argue that the measure of legibility that matters is not lead time but lag time after navigation begins before the user is able to predict the correct passing.

4.3 Discussion

Our simulated evaluation confirmed our first hypothesis by revealing statistical evidence that SM achieves consistently lower trajectory entanglement than other representative multi-agent collision avoidance approaches. Furthermore, our user study confirmed our second hypothesis by demonstrating a positive correlation between trajectory entanglement and the time taken for a correct prediction of a passing between two agents. In other words, *executions of lower trajectory entanglement are more legible.* From these, we may assert that the Social Momentum framework appears to produce legible behaviors in multi-agent environments. This feature is of particular importance for robots navigating crowded human environments, where no explicit communication takes place among agents and no formal rules are guiding traffic, such as pedestrian environments.

5 CONCLUSION

We presented a planning framework for legible motion generation in multi-agent environments. We approached legibility from a topological perspective and introduced the concept of *legible avoidance* as a maneuver that clearly indicates the way an agent plans to avoid another (e.g. *right* or *left* side). Based on this idea, we designed the *Social Momentum* planning framework that enables agents to generate intent-expressive and socially compliant behaviors in multi-agent environments. Statistical evidence, extracted from extensive simulations and from a user study with human participants demonstrated the ability of our framework to produce legible behaviors in multi-agent environments. This result is particularly important for operation in human environments with no explicit communication among agents, such as pedestrian environments. Ongoing work involves validating our approach by conducting experiments on an autonomous social robot, navigating academic hallways.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant IIS - 1526035. Wil Thomason was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. We are grateful for this support.

REFERENCES

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. 2016. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '16).*

[2] Maren Bennewitz, Wolfram Burgard, Grzegorz Cielniak, and Sebastian Thrun. 2005. Learning motion patterns of people for compliant robot motion. *International Journal of Robotics Research* 24 (2005), 31–48.

[3] Joan S. Birman. 1974. *Braids Links And Mapping Class Groups.* Annals of Mathematics Studies 82, Princeton University Press.

[4] Stephane Bonneaud and William H. Warren. 2014. An Empirically-Grounded Emergent Approach to Modeling Pedestrian Behavior. In *Pedestrian and Evacuation Dynamics 2012*. Springer International Publishing, 625–638.

[5] Baptiste Busch, Jonathan Grizou, Manuel Lopes, and Freek Stulp. 2017. Learning Legible Motion from Human–Robot Interactions. *International Journal of Social Robotics* (Mar 2017).

[6] Daniel Carton, Wiktor Olszowy, and Dirk Wollherr. 2016. Measuring the Effectiveness of Readability for Mobile Robot Locomotion. *International Journal of Social Robotics* 8, 5 (2016), 721–741.

[7] Yu Fan Chen, Michael Everett, Miao Liu, and Jonathan P. How. 2017. Socially Aware Motion Planning with Deep Reinforcement Learning. *CoRR* abs/1703.08862 (2017). arXiv:1703.08862 http://arxiv.org/abs/1703.08862

[8] G. Csibra and G. Gergely. 2007. 'Obsessed with goals': Functions and mechanisms of teleological interpretation of actions in humans. *Acta Psychologica* 124, 1 (2007), 60–78.

[9] José Grimaldo Da Silva Filho and Thierry Fraichard. 2017. Human Robot Motion: A Shared Effort Approach. In *European Conference on Mobile Robotics*. Paris, France. https://hal.inria.fr/hal-01565873

[10] Anca D. Dragan and Siddhartha Srinivasa. 2014. Integrating human observer inferences into robot motion planning. *Autonomous Robots* 37, 4 (2014), 351–368.

[11] Ivan Dynnikov and Bert Wiest. 2007. On the complexity of braids. *Journal of the European Mathematical Society* 009, 4 (2007), 801–840.

[12] Franck Feurtey. 2000. *Simulating the collision avoidance behavior of pedestrians.* The University of Tokyo, School of Engineering. Department of Electronic Engineering.

[13] J. Guzzi, A. Giusti, L. M. Gambardella, G. Theraulaz, and G. A. Di Caro. 2013. Human-friendly robot navigation in dynamic environments. In *2013 IEEE International Conference on Robotics and Automation (ICRA '13)*. 423–430.

[14] E.T. Hall. 1990. *The Hidden Dimension.* Anchor Books.

[15] Dirk Helbing and Péter Molnár. 1995. Social force model for pedestrian dynamics. *Physical Review E* 51 (1995), 4282–4286. Issue 5.

[16] Rachel Holladay, Anca Dragan, and Siddhartha Srinivasa. 2014. Legible Robot Pointing. In *Proceedings of the International Symposium on Robot and Human Interactive Communication (Ro-Man '14)*.

[17] Ioannis Karamouzas, Brian Skinner, and Stephen J. Guy. 2014. Universal Power Law Governing Pedestrian Interactions. *Physical Review Letters* 113 (Dec 2014), 238701. Issue 23.

[18] Beomjoon Kim and Joelle Pineau. 2016. Socially Adaptive Path Planning in Human Environments Using Inverse Reinforcement Learning. *International Journal of Social Robotics* 8, 1 (01 Jan 2016), 51–66.

[19] Ross A. Knepper, Christoforos I. Mavrogiannis, Julia Proft, and Claire Liang. 2017. Implicit Communication in a Joint Action. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (HRI '17)*. 283–292.

[20] Ross A. Knepper and Daniela Rus. 2012. Pedestrian-inspired sampling-based multi-robot collision avoidance.. In *Proceedings of the International Symposium on Robot and Human Interactive Communication (RO-MAN '12)*. 94–100.

[21] Henrik Kretzschmar, Markus Spies, Christoph Sprunk, and Wolfram Burgard. 2016. Socially compliant mobile robot navigation via inverse reinforcement learning. *The International Journal of Robotics Research* 35, 11 (2016), 1289–1307.

[22] Thibault Kruse, Patrizia Basili, Stefan Glasauer, and Alexandra Kirsch. 2012. Legible robot navigation in the proximity of moving humans. In *Proceedings of the 2012 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO '12)*. 83–88.

[23] Christoforos I. Mavrogiannis, Valts Blukis, and Ross A. Knepper. 2017. Socially Competent Navigation Planning by Deep Learning of Multi-Agent Path Topologies. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '17)*.

[24] Christoforos I. Mavrogiannis and Ross A. Knepper. 2016. Decentralized Multi-Agent Navigation Planning with Braids. In *Proceedings of the Workshop on the Algorithmic Foundations of Robotics (WAFR '16)*.

[25] Mehdi Moussaïd, Dirk Helbing, and Guy Theraulaz. 2011. How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences* 108, 17 (2011), 6884–6888.

[26] Stefanos Nikolaidis, David Hsu, and Siddhartha Srinivasa. 2017. Human-robot mutual adaptation in collaborative tasks: Models and experiments. *The International Journal of Robotics Research* 36, 5-7 (2017), 618–634.

[27] Enrico Pagello, Antonio D'Angelo, Federico Montesello, Francesco Garelli, and Carlo Ferrari. 1999. Cooperative behaviors in multi-robot systems through implicit communication. *Robotics and Autonomous Systems* 29, 1 (1999), 65 – 77.

[28] Masahiro Shiomi, Francesco Zanlungo, Kotaro Hayashi, and Takayuki Kanda. 2014. Towards a Socially Acceptable Collision Avoidance for a Mobile Robot Navigating Among Pedestrians Using a Pedestrian Model. *International Journal of Social Robotics* 6, 3 (2014), 443–455.

[29] Emrah Akin Sisbot, Luis Felipe Marin-Urias, Rachid Alami, and Thierry Siméon. 2007. A Human Aware Mobile Robot Motion Planner. *IEEE Transactions on Robotics* 23, 5 (2007), 874–883.

[30] Kyle Strabala, Min Kyung Lee, Anca Dragan, Jodi Forlizzi, Siddhartha S. Srinivasa, Maya Cakmak, and Vincenzo Micelli. 2013. Toward Seamless Human-robot Handovers. *Journal of Human-Robot Interaction* 2, 1 (2013), 112–132.

[31] Stefanie Tellex, Ross Knepper, Adrian Li, Daniela Rus, and Nicholas Roy. 2014. Asking for Help Using Inverse Semantics. In *Proceedings of the Robotics: Science and Systems (RSS '14)*.

[32] Jean-Luc Thiffeault. 2010. Braids of entangled particle trajectories. *Chaos* 20, 1 (2010). arXiv:0906.3647

[33] Jean-Luc Thiffeault and Marko Budišić. 2013–2017. Braidlab: A Software Package for Braids and Loops. (2013–2017). http://arXiv.org/abs/1410.0849 Version 3.2.1.

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

[35] X. T. Truong and T. D. Ngo. 2017. Toward Socially Aware Robot Navigation in Dynamic and Crowded Environments: A Proactive Social Motion Model. *IEEE Transactions on Automation Science and Engineering* 14, 4 (2017), 1743–1760.

[36] Jur van den Berg, Stephen J. Guy, Ming C. Lin, and Dinesh Manocha. 2009. Reciprocal *n*-Body Collision Avoidance. In *Proceedings of the International Symposium on Robotics Research (ISRR '09)*. 3–19.

[37] Nicholas H. Wolfinger. 1995. Passing Moments: Some Social Dynamics of Pedestrian Interaction. *Journal of Contemporary Ethnography* 24, 3 (1995), 323–340.

[38] Brian D. Ziebart, Nathan Ratliff, Garratt Gallagher, Christoph Mertz, Kevin Peterson, J. Andrew Bagnell, Martial Hebert, Anind K. Dey, and Siddhartha Srinivasa. 2009. Planning-based Prediction for Pedestrians. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*.