Article

The International Journal of Robotics Research
1–19

DOI: 10.1177/0278364918781016
journals.sagepub.com/home/ijr

Multi-agent path topology in support of socially competent navigation planning

Christoforos I Mavrogiannis[1] and Ross A Knepper[2]

Abstract

We present a navigation planning framework for dynamic, multi-agent environments, where no explicit communication takes place among agents. Inspired by the collaborative nature of human navigation, our approach encodes the concept of coordination into an agent's decision making through an inference mechanism about collaborative strategies of collision avoidance. Each such strategy represents a distinct avoidance protocol, prescribing a distinct class of navigation behaviors to agents. We model such classes as equivalence classes of multi-agent path topology, using the formalism of topological braids. This formalism may naturally encode any arbitrarily complex, spatiotemporal, multi-agent behavior, in any environment with any number of agents into a compact representation of dual algebraic and geometric nature. This enables us to construct a probabilistic inference mechanism that predicts the collective strategy of avoidance among multiple agents, based on observation of agents' past behaviors. We incorporate this mechanism into an online planner that enables an agent to understand a multi-agent scene and determine an action that not only contributes progress towards its destination, but also reduction of the uncertainty of other agents regarding the agent's role in the emerging strategy of avoidance. This is achieved by picking actions that compromise between energy efficiency and compliance with everyone's inferred avoidance intentions. We evaluate our approach by comparing against a greedy baseline that only maximizes individual efficiency. Simulation results of statistical significance demonstrate that our planner results in a faster uncertainty decrease that facilitates the decision-making process of co-present agents. The algorithm's performance highlights the importance of topological reasoning in decentralized, multi-agent planning and appears promising for real-world applications in crowded human environments.

Keywords

Motion planning, navigation, multi-agent systems, inference, topology, braids

1. Introduction

The traffic flow in human environments, such as crowded hallways, sidewalks, and rooms may often be characterized as unstructured and even unpredictable, as a result of the lack of formal rules to control traffic and the lack of explicit communication among agents. Nonetheless, humans are capable of traversing such environments with remarkable efficiency, without hindering one another's motion. Human navigation not only achieves collision avoidance; it does so while respecting several social considerations, such as the passing preference of others and their personal space (Hall, 1990). This behavior has largely been attributed to *trust*, with pedestrians trusting that others will behave *competently*, according to Wolfinger (1995). This form of trust enables humans to infer the intentions of others, under the assumption of rational action (Csibra and Gergely, 2007) but also effectively communicate their own intentions by leveraging various implicit communication channels, broadcasting and receiving information through their path shape, their body posture, their gaze, etc. This exchange of information through nonverbal, implicit communication, enables humans to negotiate and agree on a joint strategy of avoidance. Complying with this strategy yields a socially acceptable outcome for everyone in the scene.

Inspired by the outlined cooperative mechanisms that humans employ towards ensuring collision-free and socially compliant encounters, we design a navigation planning framework for artificial agents, navigating crowded environments. We explicitly incorporate the concept of

[1]Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, USA
[2]Department of Computer Science, Cornell University, Ithaca, NY, USA

Corresponding author:
Christoforos I Mavrogiannis, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Upson Hall, Room 568, 124 Hoy Road, Ithaca, NY 14853, USA.
Email: cm694@cornell.edu

cooperation into an agent's decision-making process by constructing an inference mechanism that reasons about joint strategies of avoidance. A joint strategy essentially corresponds to an avoidance protocol that all agents follow, while making progress towards their destinations (see Figure 1). Once a consensus over a joint strategy is established, the agents may be *trusted* to finesse the details of their path plans, in a compliant fashion. To accelerate the convergence to a state of consensus, the agents select behaviors that (1) communicate their own intentions but also (2) communicate acknowledgment of the intentions of others. This results in a faster decrease of uncertainty, which not only facilitates everyone's decision making but also often leads to overall efficiency improvement.

In this work, we represent joint strategies as topological patterns of agents' trajectories, which we model as topological *braids* (Birman, 1975) by employing the construction of braid groups from low-dimensional topology. Based on this representation, we design an analytical belief distribution over joint strategies that allows an agent to estimate what strategies are mutually acceptable by everyone and approximate the effect of its actions to the belief of others. We then employ an information-theoretic approach to manipulate this belief towards a state of low information entropy, which corresponds to higher confidence over the emerging evolution of the scene. We encode this specification into the agent's decision making policy, which determines an action by compromising between personal efficiency and contribution to consensus establishment.

In contrast to the majority of existing approaches that are either too myopic, only focusing on local collision avoidance resolution (e.g. Helbing and Molnár (1995), van den Berg et al. (2009), and related frameworks), or too specific, reproducing demonstrated behaviors in specific contexts (e.g. Chen et al., 2017; Kim and Pineau, 2016; Kretzschmar et al., 2016), our approach enables agents to execute local collision avoidance maneuvers with a global, multi-agent collision-avoidance horizon in mind. More specifically, we contribute: (1) a formal model that may represent and characterize topologically any, arbitrarily complex, multi-agent behavior in any environment with any number of agents; (2) a human-inspired inference mechanism for predicting collective, multi-agent navigation behaviors; (3) an online decision-making mechanism for decentralized, collaborative navigation planning that accelerates the rate of convergence to a mutually beneficial joint strategy; (4) simulation results of statistical significance that demonstrate the importance of incorporating a collective, global spatiotemporal understanding in the planning process. Our framework was designed according to the insights of sociology studies on pedestrian behavior (Wolfinger, 1995) and psychology studies on action interpretation (Csibra and Gergely, 2007), reflecting our scope of employing it on a mobile robot platform for navigation in crowded human environments. The topological structure that our model offers to the motion planning process is expected to reduce the emergence of undesired situations such as deadlocks and livelocks that are frequently observed in practice in human–robot pedestrian encounters. With this work, we aim at providing a thorough, proof-of-concept exploration of the aforementioned fundamental idea in a case study on a simplified, discretized board game setup that represents an abstracted version of the highly complex real-world problem.

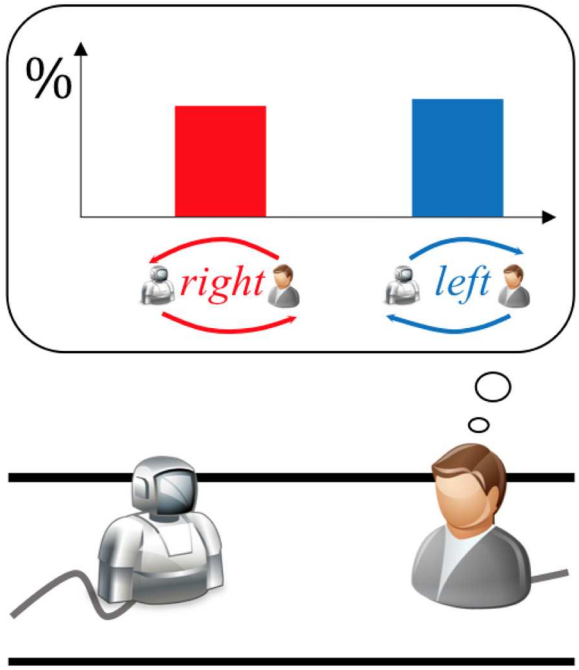


Fig. 1. A human and a robot are navigating towards opposing directions of a hallway. To avoid collision, they need to agree on an avoidance protocol (passing from the right- or left-hand side of each other). The jerky behavior of the robot so far and the smooth but non-committal, with respect to a passing side, path of the human agent yield a high-entropy belief distribution over an emerging avoidance protocol from the perspective of both agents. The goal of our planner is to generate a sequence of highly informative actions that will rapidly reduce the entropy and break a potential livelock or deadlock situation.

This paper extends our past work (Mavrogiannis and Knepper, 2016) with the incorporation of the following contributions: (1) a more thorough literature review; (2) a more rigorous presentation of the topological foundations of our approach; (3) advancements and improvements to our modeling representations; (4) an extended discussion on the inference mechanism and a more principled inference construction that incorporates new heuristics and handles uncertainty over agents' destinations; (5) a more detailed presentation of the algorithm design that includes a discussion of the algorithm's complexity; (6) an extensive simulated evaluation, demonstrating results with verified statistical significance.

2. Related Work

Our work focuses on the development of a navigation planning framework with the end goal of being employed by

robots operating in multi-agent human environments. As a result, it draws principles from several fields, such as action interpretation, human and robotic navigation, crowd simulation and tracking, and topology. In this section, we review relevant literature from these communities, discuss works that influenced our approach and highlight the features that make our framework unique.

2.1. Human navigation

Understanding and modeling human navigation accurately has been the focus of researchers from various fields for a long time. Karp et al. (1977), in their definition of the *minimax hypothesis of urban life*, specify that "urbanites seek to minimize involvement and to maximize social order." This idea is also present in Wolfinger's definition of the *pedestrian bargain* (Wolfinger, 1995), a concise, high-level protocol of foundational social rules that regulate pedestrian navigation: (1) "people must behave like competent pedestrians" and (2) "people must trust copresent others to behave like competent pedestrians." Trust to the rules of the bargain constitutes the basis of smooth co-navigation in human environments as it enables pedestrians to plan with the expectation that others will also behave competently and, thus, *cooperate* to resolve potential conflicts. This is enabled through a sophisticated mechanism of perception and action, enabled through information exchange mostly via path shape, body posture, and gaze (Goffman, 1966) that has been widely studied from a number of different fields. For example, Carton et al. (2016a) studied the trajectory planning horizon of humans in locomotion tasks towards informing the design of models for the prediction of human walking behaviors. In the field of psychology, Warren (2006) proposed a model that may describe organization in human behavior in a number of tasks by treating an agent and its environment as a pair of coupled interacting dynamical system. More broadly, Csibra and Gergely (2007) highlighted the teleological nature of human inference, suggesting that humans tend to attribute goals to observed actions.

Unlike human navigation which is heavily based on a multi-modal information exchange, our approach only considers the modality of path shape. Our planning framework aims at generating motion that maximizes social order through a local, collision-free action selection with a global lookahead. This is made possible with a principled design of a goal-driven inference that connects individual and collective behavior towards enabling artificial agents to understand the effects of their actions on the behaviors of others.

2.2. Multi-agent simulation

The problem of generating smooth, collision-free, multi-agent simulations has been central in a number of applications, ranging from city planning to the study of evacuation scenarios and even computer game design. To this end, a class of works models agents as interacting particles, attracted to their destinations and repulsed by others. Within this class, the *social force* model (Helbing and Molnár, 1995) has been one of the first and most influential approaches, whereas several works have employed similar models with additional considerations such as discomfort fields (Treuille et al., 2006), local predictive processes (Hoogendoorn and Bovy, 2003; Karamouzas et al., 2009), and cognitive heuristics (Moussaïd et al., 2011; Farina et al., 2017). Some works have employed data-driven techniques to learn the parameters of human navigation in different contexts from simulated (Henry et al., 2010) or real-world demonstrations (Karamouzas et al., 2014). Finally, van den Berg et al. (2009) and Knepper and Rus (2012) have proposed decentralized motion planners that explicitly leverage the expectation of cooperation.

Our planner is also cooperative by design. However, in contrast to most of the aforementioned approaches that either treat agents as moving obstacles or make purely local motion prediction, our inference mechanism enables agents to understand that they are part of a crowd of intelligent agents that cooperate to reach consensus over a collision avoidance protocol.

2.3. Multi-agent tracking

From a different perspective, the computer vision community has contributed a number of methods focused on tracking the local or global behavior of pedestrians in different environments. For example, Scovanner and Tappen (2009), Pellegrini et al. (2009), Alahi et al. (2016), and Ma et al. (2017) present data-driven models for local short-term trajectory predictions for interacting pedestrians, that make use of models of social interactions, whereas Zhou et al. (2012) predict large-scale, global pedestrian interactions. These contributions are of particular significance both for offline labeling of pedestrian datasets, but also for online tracking of multi-agent behaviors for robotics applications.

Similarly to some of the aforementioned approaches, our proposed inference mechanism enables agents to perceive the global nature of observed actions. Under the assumption of rationality (agents move efficiently towards intended destinations), our prediction model essentially scores the set of possible multi-agent collision avoidance strategies that the agents could follow to reach their destinations in a collision-free fashion. In a recent work, we showed how such a model can be learned from deep neural sequence-to-sequence architectures (Mavrogiannis et al., 2017).

2.4. Social robot navigation

Humans are intelligent agents, reacting to observed behaviors upon making inferences about their own future behaviors. Thus, enabling robots to navigate seamlessly alongside

humans in pedestrian environments requires an incorporation of our understanding of human navigation and decision making into the robot's motion planning processes.

To this end, a class of works have proposed optimization-based planners that consider different criteria related to social compliance. Sisbot et al. (2007) presented a cost-based planner that incorporates considerations of human comfort and context-specific social conventions, whereas Park et al. (2012) proposed an online model-predictive control framework that generates locally optimal, smooth, collision-free trajectories for autonomous robotic wheelchair navigation. Mead and Matarić (2017) considered the incorporation of psychophysical proxemics theory into a cost-based planner, aiming at driving the robot to configurations that facilitate speech and gesture-based social interactions between a human and a robot.

Another class of works has focused on reproducing observed human behaviors. Bennewitz et al. (2005), Sehestedt et al. (2010), and Ziebart et al. (2009) proposed planning frameworks that generate socially compliant robot motion through the use of human motion prediction models, learned from demonstrations of human trajectories in specific environments. Trautman et al. (2015) developed a planning framework that enables the generation of competent robot motion through dense human crowds via the consideration of a learned inference mechanism over cooperative collision-avoidance strategies. Kim and Pineau (2016) and Kretzschmar et al. (2016) proposed planning frameworks for generating humanlike robot motion, based on cost functions that model human preferences as observed in datasets collected from human demonstrations or from robots teleoperated in crowded environments. Chen et al. (2017) learned a model for generating motion that respects culture-specific social norms such as passing from the right-hand side. Finally, Bera et al. (2017) learned a probabilistic planning and prediction model that incorporates psychological and social constraints, based on identifying personality traits of pedestrians. For an extensive discussion of the state of the art in social robot navigation, the reader may refer to Thomaz et al. (2016).

The main contribution of this paper is an online, cost-based navigation planner for the generation of socially competent robot motion in multi-agent environments. Our framework aims at enabling a robot to interact (passively) with other agents, by *socially avoiding* them and does not consider the task of (active) social engagement. In contrast to some of the aforementioned data-driven approaches that imitate demonstrated human behaviors or social norms and are, thus, domain and context specific by design, our framework focuses on the underlying topological foundation of multi-agent collision avoidance, which represents an inherently generalizable basis across domains and contexts.

Our work leverages the foundational idea that, under the assumption of rationality, the collision avoidance constraints (with the environment and with others) introduce a topological structure on agents' global behaviors. To the best of the authors' knowledge, this work is the first to incorporate the existence of this structure into agents' inference mechanisms. This is done through the introduction of an inherently-generalizable representation of collective behavior, based on the braid group, a group of topological objects with a dual geometric and algebraic nature. This representation enables agents running our algorithm to reason about the topology of the spatiotemporal pattern of everyone's future trajectories instead of reasoning individually about the detailed trajectories of other agents.

Conceptually, the frameworks of Trautman et al. (2015) and Kretzschmar et al. (2016) are similar in that they also incorporate models of joint strategies into their planning processes. However, their models are not as expressive as our model of multi-agent collision-avoidance behaviors, based on topological *braids*. Braids allow our agents to understand the effect of their actions on the decision making of others and make decisions of global outlook that serve the social welfare.

2.5. Intention-aware and intent-expressive motion planning

Over the past few years, a significant amount of research has been devoted to the design of planning frameworks for explicit or implicit human–robot interaction. Ensuring safe, natural, seamless, and efficient interaction under this setting often necessitates the development of mechanisms for intention recognition and nonverbal communication. Towards this goal, roboticists have been inspired by the insights of studies on the principles underlying action interpretation by humans. Representative works include the studies of Csibra and Gergely (1998, 2007), Baker et al. (2009), and Wiese et al. (2012) who have highlighted the teleological nature of human inference, i.e. their tendency to interpret observed actions as approximately rational and hence to attribute context-specific intentions to them.

Based on these insights, Dragan and Srinivasa (2014) formalized the concepts of *legibility* and *predictability* as properties of motion that allow an observer to make a correct inference of another agent's *goal*, given observation of its *action* and of its future *action* given knowledge of its *goal*, respectively, under the principle of rational action. These formalisms may be used for encoding intentions into robot motion; intent-expressive robot motion has been shown to lead to increased efficiency in human–robot collaboration, but also to reduced planning effort for humans (Carton et al., 2016b). Generalizing these concepts to any human–robot collaboration task, involving any type of modality, Knepper et al. (2017) developed a formal framework for planning implicitly communicative actions.

Finally, the recent interest in autonomous driving has led to several works specializing on a street navigation context. For example, Bandyopadhyay et al. (2012) and Ferguson

et al. (2015) presented navigation frameworks for planning autonomous car navigation, based on models for predicting pedestrian intentions, whereas Sadigh et al. (2016) proposed a planning framework that reasons about human drivers' mental models to plan actions that influence their decision making towards desired outcomes.

Our planner aims at generating intent-expressive robot behaviors in dynamic multi-agent environments. In particular, it aims at manipulating an agent's path shape to convey its intentions to others. Our inference mechanism allows our agents to monitor the uncertainty over future multi-agent behaviors and take information-rich or *legible* actions that aim at reducing this uncertainty. In contrast to existing approaches that generate legible robot motion in static and structured environments (e.g. Dragan and Srinivasa, 2014), our framework is the first to address the problem of planning legible motion in a dynamic, multi-agent environment.

2.6. Braids in robotics

The foundation of this work is the topological construction of braids from the field of low-dimensional topology (Kassel and Turaev, 2008). The formalism of braids, first presented by Artin (1947a,b) and extensively studied by Birman (1975) has been an inspiration for applications in various disciplines. Our approach is specifically inspired by the use of braids as a model that captures the entanglement of particle trajectories in a fluid (Thiffeault, 2010); in a similar fashion, we employ braids to model the entanglement of the trajectories of navigating agents.

The idea of employing braids in planning problems in robotics is not new. It may be traced at least as far back to Ghrist (1999), who drew a parallel between braids and configuration spaces of robotic systems. Later, Diaz-Mercado and Egerstedt (2017) developed a framework for centralized multi-robot navigation, in which the agents are assigned trajectories that contribute to a topological pattern corresponding to a specified braid. Although we are also making use of braids to model multi-robot behaviors, the scope of our approach is inherently different, since our target application concerns navigation in dynamic environments where no explicit communication takes place among agents. In our case, the agents do not follow a pre-specified braid pattern, but rather employ a braid-based probabilistic reasoning to reach a topological consensus that best complies with everyone's intentions or preferences. For our purposes, braids provide an abstraction of the complex spatiotemporal multi-agent dynamics of interaction among agents. This abstraction enables artificial agents to reason about uncertainty in a principled fashion, as the dual geometric and algebraic representation of braids enables them to enumerate a set of diverse, topologically distinct scene evolutions. As a result, our algorithm generates *socially competent* behaviors, i.e. behaviors that explicitly take into consideration the *social welfare* of the system of all agents.

To showcase the benefits of braids for multi-agent planning problems in robotics, we consider a case study on an abstracted, discretized version of the real-world, multi-agent navigation problem. Simulation results demonstrate that our agents learn to coordinate faster than purely greedy agents. Recent work of ours (Mavrogiannis et al., 2017, 2018) has demonstrated the potential of the proposed planning architecture for real-world environments by considering continuous workspaces and comparing against other widely used baselines such as the social force model (Helbing and Molnár, 1995) and the optimal reciprocal collision avoidance (ORCA) framework (van den Berg et al., 2009). Ongoing work involves validation of our framework on an autonomous telepresence robot platform, designed to navigate around humans in crowded workspaces.

3. Foundations

Consider a set of $n \geq 2$ agents $N = \{1, \dots, n\}$ navigating a workspace $\mathcal{Q} \subset \mathbb{R}^2$. Denote by $q_i \in \mathcal{Q}$ the configuration of agent $i \in N$. Agent i starts from an initial configuration $q_i^s \in \mathcal{Q}$ at time $t = 0$ and reaches a final configuration q_i^d at time $t = T_i$. The final configuration q_i^d corresponds to a landmark d_i from a set of landmarks $D \subset Q$ (we assume that $d_i \neq d_j$ for any two agents $i, j \in N$). The path agent i follows to reach its destination is a function $\xi_i : [0, T_i] \rightarrow \mathcal{Q}$.

The agents do not explicitly exchange any kind of information with each other but are assumed to be acting rationally, which in our context means that (1) they always aim at making progress towards their destinations and (2) they have no motive for acting adversarially against other agents (e.g. blocking their paths or colliding with them). The notion of rationality is in line with the concept of *competence* as described by Wolfinger (1995) in his definition of the *pedestrian bargain* and also with the concept of *teleological reasoning* that appears to be foundational for human inference (Csibra and Gergely, 2007).

3.1. Game-theoretic setup

Inspired by Wolfinger's observations on the cooperative nature of human navigation, we approach the problem of robotic navigation in multi-agent, dynamic environments as a finitely repeated coordination game of imperfect information and perfect recall. The game is repeated a finite number of rounds K, which is unknown a priori and corresponds to the round at which the slowest agent reached its destination. At each round $k \in \{1, \dots K\}$, each agent i decides on an action a_i^k from a set of available actions (actions that could potentially lead to collisions and actions that violate the agent's dynamics are excluded) $\mathcal{A}_i^k$ by minimizing a cost function u_i. The agents are simultaneously selecting their actions and therefore they have no access to other agents' plans (imperfect information); we assume however that they maintain a history of all previous rounds (perfect recall). The result of all agents' decision making at round k is the

strategy profile $A^k = \{a_1^k, \ldots, a_n^k\}$. The sequence of strategy profiles of all rounds, from the beginning to the end of the game, $A_1 \ldots A_K$, forms a global *joint strategy* τ that the agents engaged in to avoid each other, while making progress towards their destinations.

Although the agents do not explicitly coordinate with each other to decide on a joint strategy, their strategy profiles at every round gradually reinforce and contribute to one. Imbuing artificial agents with an understanding of the collective dynamics of a multi-agent scene may allow them to make informed and *socially competent* decisions that contribute to the avoidance of undesired situations, such as hindering others' paths, deadlocks and livelocks. For this reason, in addition to personal efficiency, it is important that agents' cost functions incorporate a model of multi-agent efficiency, reflecting the *social welfare* of the system of agents.

In this paper, we develop a topological model of joint strategies, employing the formalism of braids (Birman, 1975), which we use to develop a human-inspired inference mechanism, supported by studies on human action interpretation. Our mechanism provides a principled prediction over the scene evolution that allows agents to take into consideration the effect of their decision making on any observers.

3.2. A topological model of joint strategies

Let us collect the state of the system of all agents in a tuple $Q = (q_1, \ldots, q_n) \in \mathcal{Q}^n$. The system state evolves from a starting configuration $Q^s = (q_1^s, \ldots, q_n^s)$ to a final configuration $Q^d = (q_1^d, \ldots, q_n^d)$, by following a path $\Xi : [0, T] \rightarrow \mathcal{Q}^n$, from the space of system paths $\mathcal{Z}$, starting from Q^s and ending at Q^d. The system path is a function $\Xi : [0, T] \rightarrow \mathcal{Q}^n \setminus \Delta$, where $\Delta = \{Q = (q_1, q_2, \ldots, q_n) \in \mathcal{Q}^n : q_i = q_j \text{ for some } i \neq j \in N\}$ is the set of all system states with agents in collision and $T = \max_{i \in N} T_i$ (it is assumed that agents remain at their destinations until everyone reaches their own). Naturally Δ partitions the space of system paths $\mathcal{Z}$ into a set of classes of homotopically equivalent system paths. Each such class has distinct topological properties which indicate a distinct joint strategy that the agents followed to reach their destinations. To enumerate such classes of joint strategies but also to characterize topologically the collective behavior of a system of agents, we develop a model of joint strategies using the concept of braids (Birman, 1975). In the following paragraphs, we provide a primer on braids, establish a correspondence between braids and collective navigation behaviors and define a topological model of joint navigation strategies.

3.2.1. Background on braids. Braids are topological objects with algebraic and geometric presentations. We first introduce them as geometrical entities, following a presentation based on Artin (1947b) and continue with a discussion of their algebraic presentation and their group formation.

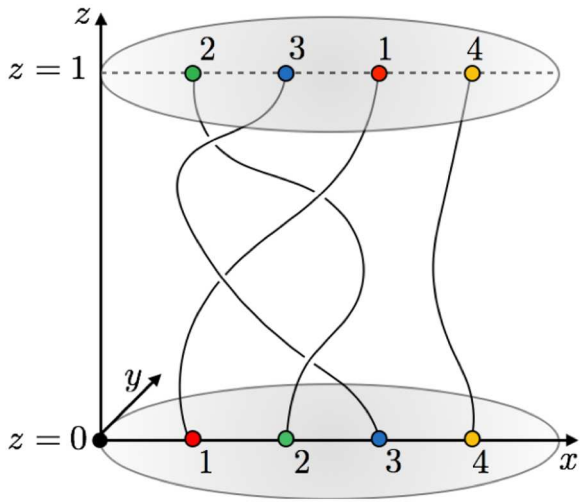


Fig. 2. A set of four particles are initially ($z = 0$) arranged along the x-axis, on the points 1, 2, 3, and 4. Through a sequence of rearrangements, the particles finally ($z = 1$) reach a final arrangement on the points 2, 3, 1, and 4, respectively. The pattern of their trajectories is a *geometric braid*.

Fig. 3. The generators of the braid group B_n: (a) σ_1; (b) σ_2; (c) σ_{n-1}.

Denote by x, y, z the Cartesian coordinates of a Euclidean space $\mathbb{R}^2 \times I$. 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 intersection $X(z) = (x, y)$ with each plane $z \in I$. A braid on n-strings or n-braid (see Figure 2) 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 $p : N \rightarrow N$ from the set of permutations of N, $Perm(N)$, defined as

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

This geometric representation of a braid is commonly referred to as a *geometric braid*. More formally, a geometric braid is often represented with a *braid diagram*, a projection of the braid onto the plane $\mathbb{R} \times 0 \times I$ (see, e.g., Figure 3).

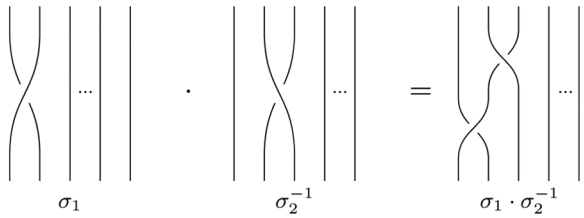


Fig. 4. Example of the composition operation $\sigma_1 \cdot \sigma_2^{-1}$ for $\sigma_1, \sigma_2^{-1} \in B_n$.

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 Figure 3), called the generators of B_n, that satisfy the following *relations*:

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

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

A generator σ_i, $i \in \{1, 2, \ldots, n-1\}$ 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 *under*[1] the right (see Figure 4). 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 two, along the z-axis (see Figures 3 and 4). Any braid can be written as a product of generators and generator inverses. This representation is commonly referred to as an *algebraic braid* or a *braid word* (Figure 4).

3.2.2. Abstracting joint strategies using braids. Denote by $f_x : \mathcal{Q}^n \rightarrow Perm(N)$ 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. These contribute to a system path Ξ that corresponds to a path of permutations $\pi : [0, T] \rightarrow Perm(N)$ that may be extracted by evaluating f_x throughout the whole path Ξ. This path can be represented by a sequence of permutations of minimal length $\pi^* = (p_0, \ldots, p_K)$, i.e. $p_{j-1} \neq p_j$, $\forall j = \{1, \ldots, K\}$ and consecutive waypoints are adjacent transpositions,[2] i.e. permutations that differ by exactly one swap of adjacent elements. Owing 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 $\tau_j \in \sigma_i^{\pm 1}, i \in \{1, \ldots, n-1\}$, where i corresponds to the index of the leftmost swapping agent in permutation p_{j-1}. Therefore, the whole execution from $t = 0$ to $t = T$ may be abstracted into the braid that corresponds to the temporal sequence of events:

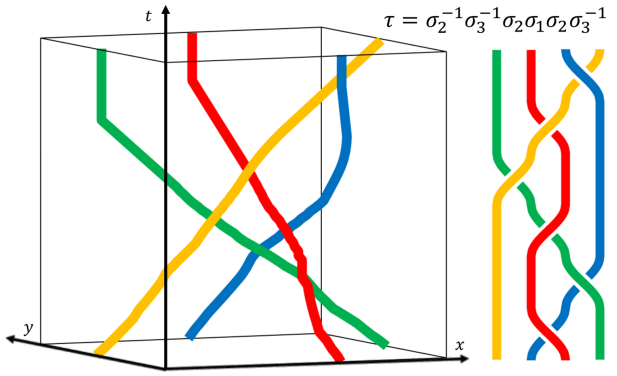


Fig. 5. A space–time representation of a system path in a workspace with four agents (left) along with its corresponding braid diagram (right) and braid word (top right), defined with respect to the path's x-projection. The visualization of the braid diagram and the extraction of the braid word was done using BraidLab (Thiffeault and Budišić, 2013–2017).

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

This braid word not only constitutes a topological characterization of the system path (see Figure 5 for an example of characterizing a system path as a braid) but it also represents a topological class of system paths that are homotopy-equivalent with the system path in consideration. In the remainder of this paper, we refer to the sequence τ as the *joint strategy* or the *entanglement* of the system path. Essentially, we model the space of joint strategies $\mathcal{T}$ as the braid group, i.e. $\mathcal{T} := B_n$.

Remark 1. *In our model, a braid constitutes a two-dimensional abstraction of a three-dimensional pattern of trajectories. Depending on the selection of the projection line, a different braid emerges. Although a change of projection line only changes the braid by conjugation (Thiffeault, 2010), in practice, this implies that a set of non-communicating agents might encode the same joint strategy with a different symbolic representations (braids). However, this does not affect the convergence to a consensus on a mutually acceptable joint strategy among agents; despite their different representations, they still take actions that contribute towards the same outcome, as will be shown in our simulation results. Therefore, the selection of the projection plane for a planning agent is not important, as long as its action selection process is consistent with it.*

4. Inference of collective behaviors

An individual agent has no sole control over a specific joint strategy. The joint strategy is an emergent behavior, resulting from the superposition of the individual strategies of all agents. In fact, since agents are not explicitly communicating or coordinating, they cannot have a priori knowledge or a precise sense of the actual joint strategy they are about to follow. However, understanding the dynamics of collective behavior may allow agents to adopt individual navigation strategies that allow others to infer their intentions more clearly, thus facilitating everyone's decision making by reducing uncertainty fast. Judging from our everyday life experience, we may argue that this is the case with humans as well. When humans encounter others in a hallway, they do not exactly know the specific joint strategy they will be following. However, they realize that their decisions are coupled with the decisions of others and are able to reach a consensus regarding an avoidance protocol that is comfortable for everyone.

In this section, we present a probabilistic intention inference mechanism that connects an observed system path with a future system path topology, designed according to the insights of psychology studies on human action interpretation. This design is motivated by our goal of employing our framework on an autonomous social robot that will be navigating in a safe and socially competent fashion around human pedestrians.

4.1. Teleological reasoning in multi-agent navigation

Csibra and Gergely (1998, 2007) argued that the mechanisms of human action interpretation are *teleological* in nature, i.e. humans tend to interpret observed actions as goal-directed in a given context. Following their insights, we design an inference mechanism of the form $P(\tau|\Xi_t, M_t)$, corresponding to a belief over an emerging joint strategy $\tau \in \mathcal{T}$ given a partial system path Ξ_t and the state of the context at time t. The joint strategy represents a collective *goal*, whereas the system path plays the role of the *action*. By context, we refer to publicly available information, such as a model of the static environment (e.g. a map, obstacles, points of interest, etc.) but also information extracted through processing, e.g. by employing secondary inference mechanisms regarding group formations, identification of reactive agents, etc.

4.2. Inferring joint strategies from context

From (4), the belief $P(\tau|\Xi_t, M_t)$ may be expanded as

$$P(\tau|\Xi_t, M_t) = P(\tau_1, \ldots, \tau_K|\Xi_t, M_t) \tag{5}$$

which, by applying the chain rule, may be factored as

$$P(\tau|\Xi_t, M_t) = \prod_{k=1}^{K} P(\tau_k|\bigcap_{j=1}^{k-1} \tau_j, \Xi_t, M_t) \tag{6}$$

This belief quantifies the likelihood of a sequence of events $\tau_1, \ldots, \tau_K$ given observation of agents' past behaviors and the context. Essentially, this corresponds to predicting the minimal sequence of permutations π^* but also the quality of the physical transitions between consecutive permutation waypoints (passing from the right-/left-hand side).

A joint strategy describes the avoidance protocol that the agents followed to avoid each other throughout the scene evolution, while navigating from Q^s to Q^d. These system path endpoints are not incorporated in the definition of the strategy as geometric entities but rather as the permutations $p^s = f_x(Q^s)$, $p^d = f_x(Q^d)$. This design decision reflects the observation that an agent navigating a multi-agent environment does not need to know the precise intended destinations of others to avoid collisions successfully; they just need to understand their passing preferences/intentions. However, the geometric arrangements of agents' intended final configurations greatly influence the convergence to a joint strategy τ. In particular, given the initial permutation p^s, only a subset $\mathcal{T} \in B_n$ may lead to p^d.

Given the importance of the final permutation in the prediction of a joint strategy, we may break the problem of predicting a joint strategy into two separate inference subproblems: (1) a prediction of the final permutation and (2) a prediction of a *compatible* system path *entanglement*, braid word, conditioned on the predicted final permutation. Following this reasoning, Equation (6) may be rewritten as the following product:

$$P(\tau|\Xi_t, M_t) = P(p^d, \tau^d|\Xi_t, M_t) \tag{7}$$

$$= P(\tau^d|p^d, \Xi_t, M_t)\, P(p^d|\Xi_t, M_t) \tag{8}$$

where $\tau^d \in \mathcal{T}$ represents a braid that is compatible with the prediction of a final permutation p^d, given the permutation $p_t = f_x(Q_t)$ corresponding to the current system state $Q_t = \Xi(t)$. Figure 6 depicts a graphic representation of the structure of our inference mechanism.

4.2.1. Inferring the final permutation of the system. A planning agent knows with certainty its own destination but has no access to other agents' destinations. Although there is no need to make an inference regarding others' actual destinations, it is important to infer a final permutation p^d so as to make an informed inference regarding the emerging joint strategy, as discussed in the previous subsection. Under the assumption of rationality and given a model of the world, stored in the context M_t, the planning agent may infer the general directions of others through a belief:

$$P(p^d|\Xi_t, M_t) = P(p^d(1), \ldots, p^d(n)|\Xi_t, M_t) \tag{9}$$

where

$$p^d = \begin{pmatrix} 1 & \ldots & n \\ p^d(1) & \ldots & p^d(n) \end{pmatrix} \tag{10}$$

For simplicity, let us assume that the planning agent's ID is #1. Then, under the assumption that all agents are moving towards destinations from the known set D, the agent's

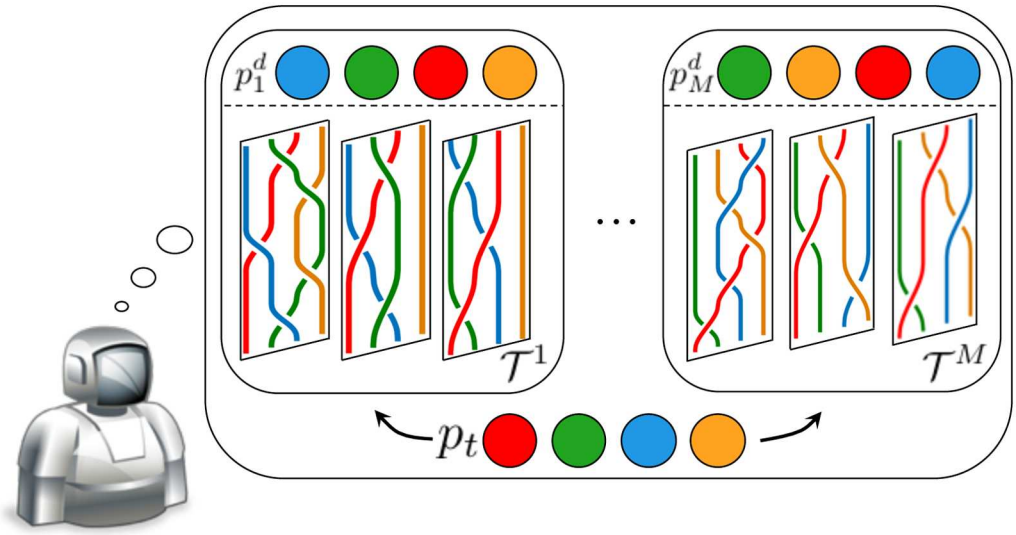


Fig. 6. Schematic representation of the inference mechanism from the perspective of a robot, navigating in a workspace with three other agents. From the perspective of the robot, the system state at time t corresponds to a permutation p_t (derived upon projecting on the x-axis of the robot's body frame), represented graphically with the color permutation at the bottom (the robot order in the permutation is denoted with red color). A set of M final permutations, taking the robot to its destination are considered and a set of three compatible joint strategies–braids are planned for each final permutation. The robot reasons over the set corresponding to the union of all sets of joint strategies $\mathcal{T} = \bigcup \mathcal{T}^m$, $m = \{1, \ldots, M\}$.

index under the permutation p^d, i.e. $p^d(1)$, is constrained. Leveraging this, the planning agent may conclude to a subset of feasible permutations $\mathcal{P} \subset Perm(N)$, by ruling out any incompatible permutations from $Perm(N)$ as unlikely. Under the assumption that the cardinality $|D| \geq |N|$, multiple assignments of destinations to agents may be possible for each permutation in $\mathcal{P}$.

More formally, for each compatible permutation $p_m^d \in Perm(N)$, $m \in \{1, \ldots M\}$, where $M = (n-1)!$, we may derive a set of possible destination assignments Δ_m that are compatible with (1) p_m^d and (2) the image of agent #1, through the permutation p_m^d, i.e. $p_m^d(1)$, corresponds to a final arrangement of all agents to destinations of D, with agent #1 at its destination d_1. Essentially, an assignment $\delta \in \Delta_m$ is an injective function $\delta : N \rightarrow D$ that maps all agents to a subset of landmarks from D. Upon marginalizing over all possible $\delta \in \Delta_m$, the probability that the final permutation $p^d = p_m^d$ may be derived as

$$P(p^d = p_m^d | \Xi_t, M_t) = \sum_{\delta \in \Delta_m} P(p^d = p_m^d | \Xi_t, M_t, \delta) P(\delta | \Xi_t, M_t) \tag{11}$$

By definition, $p^d = p_m^d$, if we know that agents are going to δ and therefore Equation (11) may be simplified as follows:

$$P(p^d = p_m^d | \Xi_t, M_t) = \sum_{\delta \in \Delta_m} P(\delta | \Xi_t, M_t) \tag{12}$$

The destination that an agent is aiming for is conditionally independent of the destinations of others, given Ξ_t and M_t, therefore we may express $P(\delta | \Xi_t, M_t)$ as

$$P(\delta | \Xi_t, M_t) = \prod_{j=2}^{n} P(q_j^d = \delta(j) | \Xi_t, M_t) \tag{13}$$

where we incorporated the fact that agent #1 is certain about its destination. Finally, under the assumption of rationality, we follow an approach similar to Dragan and Srinivasa (2014) to model $P(q_j^d = \delta(j) | \Xi_t, M_t)$ as

$$P(q_j^d = \delta(j) | \Xi_t, M_t) = \frac{1}{Z} \frac{\exp(-c(\xi_j) - c^*(q_j^s, \delta(j)))}{\exp(-c^*(q_j^s, \delta(j)))} \tag{14}$$

where ξ_j is the path agent j has followed so far, c measures the length of a path, c^* returns the shortest path between two points, and Z represents a normalizer across the set of landmarks D.

Algorithm 1 outlines the process of scoring all compatible final permutations from $Perm(N)$. Function `Get_Permutation` returns a permutation p_l of the set D, corresponding to the arrangement of landmarks in an order of increasing x-coordinates with respect to the agent's frame. Then, all permutations are accessed and checked for compatibility with the planning agent's destination d (function `Check_Perm`). In case a permutation is compatible, the set of possible assignments of agents to destinations that are in compliance with the permutation, is extracted with function `Get_Assignments` and then scored (function `Score_Assignments`). Otherwise, the corresponding permutation is assigned a zero score. The scores are

Algorithm 1. Score_Permutations($d, \Xi, D, perms$)

Input: d, agent's intended destination; Ξ, state history of all agents; D, list of landmark locations; *perms*, list of permutations.

Output: P, probability distribution over permutations.

```
p_l ← Get_Permutation(D)
for i = 1 : n! do
    compatible ← Check_Perm(perms[i], p_l, d)
    if compatible then
        Δ ← Get_Assignments(p_l, perms[i])
        S[i] ← Score_Assignments(Δ, Ξ)
    else
        S[i] ← 0
    end if
end for
P ← S/sum(S)
return P
```

finally normalized and returned in the form of a probability distribution P.

4.2.2. Inferring the path entanglement. The second distribution, over system path entanglements $P(\tau^d|p^d, \Xi_t, M_t)$ is a harder distribution to approximate. In particular, events that take place further than one event ahead may be impossible to be traced back to the decisions that agents have made until time t. For this reason, in this work, we approximate $P(\tau^d|p^d, \Xi_t, M_t)$ as

$$P(\tau^d|p^d, \Xi_t, M_t) = P(\tau_1^d, \ldots, \tau_K^d|p^d, \Xi_t, M_t) \quad (15)$$

$$\approx \frac{1}{\Gamma} P(\tau_1^d|p^d, \Xi_t, M_t) \quad (16)$$

that is, given the state of execution at time t, as expressed in Ξ_t and M_t, the probability of a braid word is set to be approximately proportional to the probability of the next generator, observed after time t, being equal to that prescribed by τ_1, where Γ is an appropriate normalizer. All entanglements that share the same first generator τ_1 are assigned the same probability. Finally, in case the current permutation of agents is equal to the predicted final permutation p^d, it is assumed that the only possible path entanglement is the trivial one, i.e. $\tau = e$, and the rest are assigned zero probabilities.

To model $P(\tau_1^d|p^d, \Xi_t, M_t)$, we first encode a generator $\tau_1^d \in \{\sigma_1, \sigma_1^{-1}, \ldots, \sigma_{n-1}, \sigma_{n-1}^{-1}\}$ into a tuple $g = (swap_g, sign_g)$, where $swap_g \in \{1, \ldots, n-1\}$ contains the generator's subscript (corresponding to the pair of agents that are exchanging sides) and $sign_g \in \{-1 + 1\}$ contains the generator's superscript (how they are exchanging sides). This model allows us to decompose the generator prediction problem into (1) a prediction of the immediately swapping agents and (2) a prediction of the type of their swap:

$$\begin{aligned} P(\tau_1^d = g|p^d, \Xi_t, M_t) =& P(swap_g, sign_g|p^d, \Xi_t, M_t) \\ =& P(sign_g|swap_g, p^d, \Xi_t, M_t) \\ & P(swap_g|p^d, \Xi_t, M_t) \end{aligned} \quad (17)$$

Regarding the prediction of the next swap, we employ the following model:

$$P(swap_g|p^d, \Xi_t, M_t) = \frac{1}{H} \prod_{i=1}^{n-1} R_{swap}(i) \quad (18)$$

where H is a normalizer across swaps and R_{swap} is defined as

$$R_{swap}(i) = \begin{cases} \frac{1}{1+\exp(\Delta x_i - \epsilon)}, & i = swap_g \\ \frac{1}{1+\exp(-(\Delta x_i - \epsilon))}, & i \neq swap_g \end{cases} \quad (19)$$

where $\Delta x_i = x_{i+1} - x_i$ represents the x-distance between the agents of the pair i (agents $p_t(i)$ and $p_t(i+1)$ in the current permutation p_t) and $\epsilon > 0$ is a constant. Smaller distances indicate swaps corresponding to generators that are exponentially more likely.

Given a prediction of a $swap_g$, the next step is to determine the type of swapping, i.e. $sign_g$. To model $P(sign_g|swap_g, p^d, \Xi_t, M_t)$ we employ a heuristic based on the angular momentum of the system of the agents in consideration, which, assuming unit masses, may be defined as

$$\vec{L}(swap_g) = \vec{r}_i^c \times \vec{v}_i + \vec{r}_{i+1}^c \times \vec{v}_{i+1} \quad (20)$$

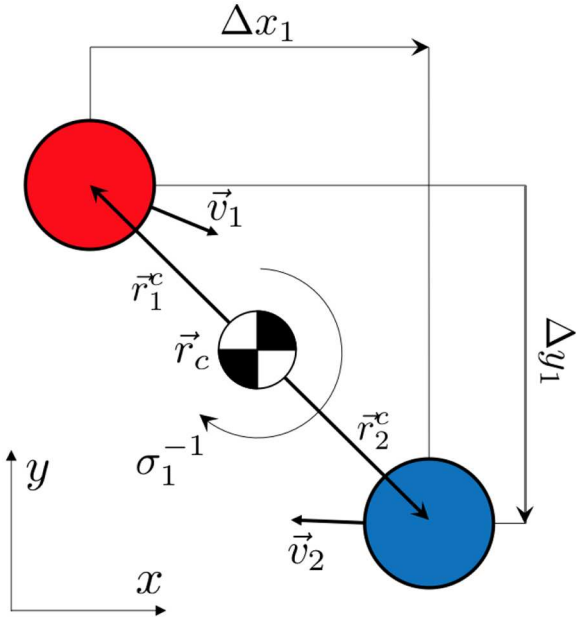


Fig. 7. Demonstration of the momentum heuristic for predicting a generator superscript: the x-projections of the agents' paths are about to cross, forming a σ_1^{-1} generator. The z component of their angular momentum is negative, indicating a tendency for a counterclockwise rotation, which indicates a negative braid exponent; likewise, in case the z component of the angular momentum were positive, the emerging exponent would be positive.

where $\vec{r}_i^c$ and $\vec{r}_{i+1}^c$ are the positions of the currently right and left agents, respectively, defined with respect to their current center of mass $\vec{r}_c = (\vec{r}_i^c + \vec{r}_{i+1}^c)/2$, whereas $\vec{v}_i$ and $\vec{v}_{i+1}$ are their respective velocities. For two masses moving on the same plane, the angular momentum is a vector, normal to the plane, along the direction of rotation. If the masses are about to rotate counterclockwise, with respect to an axis of reference, the z-component of the momentum, L_z, is positive, and negative in case the masses are about to rotate clockwise. Based on this observation, we model $P(sign_g|swap_g, p^d, \Xi_t, M_t)$ as

$$P(sign_g|swap_g, p^d, \Xi_t, M_t) = \begin{cases} \frac{1}{\Theta}\frac{1}{1+\exp(-sign_g L_z(swap_g))}, & \text{if } |\vec{v}_i| + |\vec{v}_{i+1}| > 0 \\ \frac{1}{\Theta}\frac{1}{1+\exp(-sign_g \Delta y_i)}, & \text{otherwise} \end{cases} \quad (21)$$

where Θ is an appropriate normalizer. The more positive L_z is, $P(sign_g = +1|swap_g, p^d, \Xi_t, M_t)$ gets exponentially closer to 1, whereas in the opposite case, $P(sign_g = -1|swap_g, p^d, \Xi_t, M_t)$ gets closer to 1. In case the velocities of both agents are currently zero, the corresponding scores only depend on their distance along the y-axis, $\Delta y_i = y_{i+1} - y_i$. Figure 7 demonstrates schematically the concept of momentum (for non-zero velocities) and how it is used to predict $sign_g$.

Remark 2. *It should be noted that the model of inference presented in this section constitutes is an extended version of the one presented by Mavrogiannis and Knepper (2016), as it may handle (a) uncertainty over destinations, (b) redundancy of destinations (case with no unique mapping from a permutation to the set of destinations), and (c) incorporates a novel heuristic for predicting the exponent of braid generators. However, it should also be noted that this distribution is a simplified approximation that cannot guarantee robust performance and generalization. We are using it to provide a proof of our concept. In recent work of ours (Mavrogiannis et al., 2017) we presented a data-driven framework for directly learning to predict future trajectory topologies from simulated demonstrations of challenging multi-agent scenarios.*

5. Decision making

In multi-agent environments, where no explicit communication takes place among agents, uncertainty regarding everyone's actions is typically high, which complicates decision making. Humans usually overcome such a complication by communicating implicitly, mostly through motion. Doing so is made possible through inference mechanisms that allow them to read the intentions of others and select socially compliant actions that reduce uncertainty. This enables them to reach a consensus over an avoidance protocol that serves everyone and ensures comfort, while making progress towards their destinations. The superposition of these considerations represents what, to our interpretation of the pedestrian bargain (Wolfinger, 1995), constitutes socially *competent* behavior in a pedestrian context.

To generate socially competent behaviors, we design a cost-based policy. Our cost function enables an artificial agent to take actions that not only contribute progress towards its destination but also towards a consensus over a joint strategy that appears to be mutually beneficial for everyone in the scene. Regarding the first specification, a distance-based efficiency cost is employed, whereas for the second specification, the entropy of the distribution over joint strategies $P(\tau|\Xi, M)$ is used. The distribution allows the planning agent to estimate the long-term effects of an action in consideration and how it might influence the decision making of others. The reduction of the entropy of the distribution may allow the agent to select an action that reduces the uncertainty for everyone.

In the following subsections, we describe our decision-making framework in detail.

5.1. Modeling agents' cost functions

We model the interests of an agent i with a cost function $u_i : \mathcal{A}_i \to \mathbb{R}$ that maps an action $a_i \in \mathcal{A}_i$ to a real number. We design this cost to comprise two terms: (1) E_i, which represents the agent's personal *efficiency* and (2) C_i, which represents the state of *consensus* over a joint strategy among agents, from the perspective of agent i, upon taking an action $a_i \in \mathcal{A}_i$:

$$u_i(a_i) = \lambda E_i(a_i) + (1-\lambda)\, C_i(a_i) \quad (22)$$

We define the personal efficiency term E_i, to be the length of the shortest path to the agent's destination, whereas C_i is modeled as the information entropy of the belief distribution over joint strategies $P(\tau|\Xi, M)$, from the perspective of agent i, i.e.

$$C_i(a_i) = -\sum_{\tau \in \mathcal{T}} P(\tau|\Xi^+, M) \log_2 P(\tau|\Xi^+, M) \quad (23)$$

where Ξ^+ denotes the system path so far, Ξ, augmented with the action in consideration a_i. Finally, λ is a weighting factor, expressing the compromise between efficiency and consensus. Formally, the decision-making policy may be described as a minimization of (22):

$$a_i^* = \arg\min_{a_i \in \mathcal{A}_i} u_i(a_i) \quad (24)$$

Note that the cost function u_i plays the role of a utility function, with the difference that lower values are better.

Overall, this policy enables an agent to make decisions that not only contribute progress towards its destination but also towards a mutually beneficial consensus over a scene outcome. The faster such a consensus is established, the lower the uncertainty will be for all agents throughout the remainder of the execution. The efficiency term represents

agents' intention of reaching their destinations by spending low energy and is in line with the principle of rational action as highlighted in the definitions of the pedestrian bargain (Wolfinger, 1995) and the teleological reasoning (Csibra and Gergely, 1998). The consensus term scores the current state of the global consensus among agents regarding the joint strategy to be followed and, therefore, it directly incorporates a form of social understanding into the agent's decision-making policy. The lower the entropy, the lower the uncertainty regarding the emerging joint strategy. Thus, by consistently picking actions that contribute to entropy reduction, an agent communicates its intention of complying with a subset of scene outcomes that appear to be preferable by everyone according to the model $P(\tau|\Xi, M)$. As a result, the agents are expected to reach a consensus over τ easier and faster, avoiding ambiguous situations such as livelocks or deadlocks and reach their destinations with lower planning effort.

5.2. Planning joint strategies

In practice, making use of the distribution $P(\tau|\Xi, M)$ requires determining a set of candidate joint strategies $\mathcal{T}$. The braid group B_n is countably infinite; in practice, however, only a subset of joint strategies (braids) are meaningful under the context of a scene M_t and given observations of agents' past behaviors Ξ_t. In particular, as discussed in Section 4, given any compatible final permutation $p^d_m \in \mathcal{P}$, only a subset $\mathcal{T}^m \subset B_n$ may be achievable from the current permutation $p_t = f_x(Q_t)$. Consequently, there arises the problem of planning a set of joint strategies $\mathcal{T} = \bigcup \mathcal{T}^m$, $m = \{1, \ldots, M\}$, compatible with the set of different final permutations in consideration from $\mathcal{P}$.

Planning a joint strategy that transitions the system from a permutation corresponding to the current state of the system, p_t, to a permutation corresponding to the final state of the system, p^d_m, may be decomposed into the following subproblems: (1) planning a path of permutations, $\pi^m = (p_t, \ldots, p^d_m)$ connecting p_t with p^d_m through a sequence of adjacent transpositions; and (2) assigning compatible elementary braids to the transitions between consecutive permutations. Each transition may be implemented in two different ways, i.e. by a compatible generator or its inverse. For example, a path π of length l_π (the number of transitions required to reach p^d_i from p^t, through π) may be implemented by 2^{l_π} different braids.

Assuming that we have concluded to a set $\mathcal{P} \in Perm(N)$ of potential final permutations, for each one of them $p^d_m \in \mathcal{P}$, we plan a set of paths of adjacent transpositions, $\mathcal{P}^m$, based on which we generate a batch of joint strategies $\mathcal{T}^m$. These are used to form the final set of joint strategies $\mathcal{T} = \bigcup \mathcal{T}^m$, $m = \{1, \ldots, M\}$.

In this paper, we convert the problem of planning a topological joint strategy into a search in a graph of permutations. In the following sections, we describe the construction of the graph and the planning procedure.

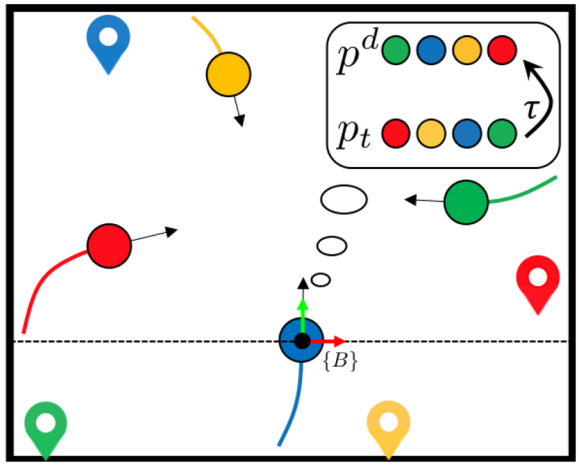


Fig. 8. A multi-agent scene from the perspective of the planning agent (blue). At time t, the agent arranges all agents in the scene in an order of increasing x-coordinates with respect to the x-axis of its body frame $\{B\}$ and derives a corresponding permutation p_t. Based on observation of all agents' past trajectories (solid lines) and given knowledge of existing landmarks in the scene, the blue agent makes a prediction of everyone's destination (colored pointers) and derives a corresponding final permutation p^d. Transitioning from p_t to p^d may be implemented with a joint strategy $\tau \in \mathcal{T}$.

5.3. Permutation graph search

The set of all permutations on N, $Perm(N)$, along with the composition operation, form the symmetric group S_n. Here S_n is a group of order $n!$, that can be generated by the set of adjacent transpositions $\beta_j = \begin{bmatrix} j & j+1 \end{bmatrix}$, for $1 \leq j < n-1$.

We make use of the symmetric group to construct a graph $G = (V, E)$, where $V = Perm(N)$ and any pair of nodes $v_a, v_b \in V$ is only connected iff $\exists\ \beta_{ab} \in S_n$ that permutes v_a into v_b. The graph G may be represented as a $(n-1)$-dimensional polytope, embedded in a n-dimensional space, which is commonly referred to as a *permutohedron* (Ziegler, 1995). Figure 9 depicts a permutohedron of order four, along with example paths and indications of braid transitions.

Planning a path from a permutation p^a to a permutation p^b, corresponding to the vertices $v_a, v_b \in V$, respectively, is equivalent to finding a path of vertices–permutations that connect them. Figure 8 illustrates the concept of planning a joint strategy. At planning time t, the agents have already followed trajectories Ξ_t. The planning agent (blue color) has predicted that they are aiming at reaching the destinations denoted by pointers of corresponding colors. Transitioning from the current system configuration to the predicted final system configuration corresponds to transitioning from the current permutation p_t to a permutation p^d, both defined with respect to the dashed line, parallel to the x-axis of the agent's body frame $\{B\}$.

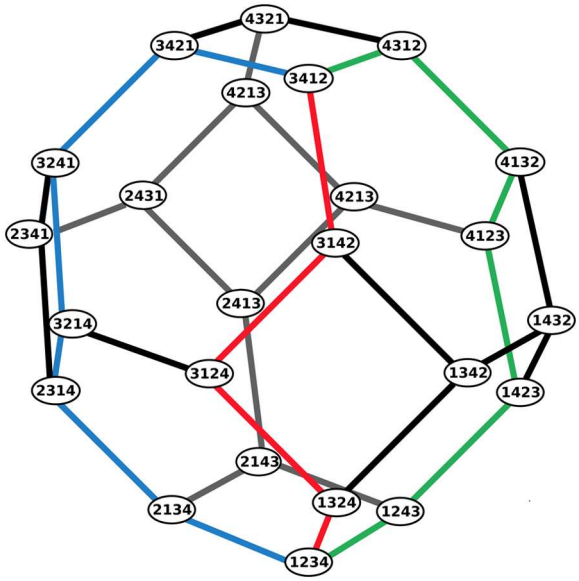


Fig. 9. A permutatohedron of order four for a scene with four agents. Three alternative paths implementing the transition from the permutation 1234 to the permutation 3412 are depicted in different colors. Each path consists of a sequence of transitions, each of which can be implemented topologically with a braid generator or its inverse.

Algorithm 2. SCN(D, Q, d, Ξ, At_Goal, M)

Input: D, list of landmarks; Q, system state; d, planning agent's intended destination; Ξ, state history of all agents; At_Goal, Boolean variable signifying arrival at agent's destination; M, context.
Output: a, action selected for execution

while ¬AtGoal **do**
  $R \leftarrow Get\_Reactive\_Agents(\Xi)$
  $\mathcal{A} \leftarrow Collision\_Check(\Xi, M, R)$
  if $R \neq \varnothing$ **then**
    $\mathcal{T} \leftarrow Get\_Strategies(d, Q, D, R)$
    $a \leftarrow Cost\_Optim(\mathcal{A}, \mathcal{T}, d, \Xi, M)$
  else
    $a \leftarrow Efficiency\_Optim(\mathcal{A}, d, M)$
  end if
end while
return a

5.4. Online algorithm

In this section, we describe our algorithm design for online navigation planning that makes use of the components detailed in the previous sections. The algorithm compromises between making progress towards the agent's destination and being respectful of everyone's intentions, as inferred by their past behaviors. In case the planning agent does not observe other agents on its way to its destination, it switches to efficiency optimization.

The SCN algorithm (Algorithm 2) is our online algorithm for *socially competent navigation*. At every replanning cycle, the function `Get_Reactive_Agents` first returns a list of agents, R, which the planning agent should be avoiding collisions with. Then, function `Collision_Check` generates a set of implementable, collision-free actions $\mathcal{A}$. If $R \neq \varnothing$, the algorithm proceeds by planning a set of joint strategies $\mathcal{T}$ (function `Get_Strategies`) and then picking the minimizer of (22) (function `Cost_Optim`). In case $R = \varnothing$, the function `Efficiency_Optim` computes the most efficient action. The algorithm terminates when the boolean *AtGoal* becomes true, indicating that the planning agent has reached its destination.

5.4.1. A discussion of complexity. The most computationally expensive part of the proposed algorithm is the computation of the set of joint strategies $\mathcal{T}$. Obtaining $\mathcal{T}$ involves determining the set of all compatible final *permutations* $\mathcal{P} = \{p_1^d, \ldots, p_M^d\}$ that correspond to final system states with the planning agent at its destination. This set has cardinality $|\mathcal{P}| = M = (n-1)!$ and may be computed in time $O(n!)$. For each permutation p_m^d, we are computing the following.

1. A set of $\mathcal{K}$ *permutation paths* Π^m that connect the current system permutation p_t with $p_m^d \in \mathcal{P}$. For computing this path set, we employ an algorithm for finding $\mathcal{K}$-shortest paths. Such algorithms typically make use of a shortest path algorithm, e.g. Dijkstra's (Dijkstra, 1959); therefore, their complexity depends on the number of calls to the shortest path algorithm. Katoh's algorithm (Katoh et al., 1982) appears to be the most efficient among them, with a runtime complexity of $O(\mathcal{K}(|E| + |V|\log|V|))$, where $|V|$ and $|E|$ represent the number of nodes and edges in the graph, respectively, which for a permutohedron of order n are equal to $|V| = n!$ and $|E| = n!(n-1)$. For a constant $\mathcal{K}$, this computation runs in time $O(n!n\log n)$.
2. A set of *braids* $\mathcal{T}^m$, consistent with each permutation path $\pi^m \in \Pi^m$. For each permutation path in Π^m, we derive 2^l different braids (where l is the number of edges in the path), by taking all possible permutations of consistent generator assignments on the permutation path edges. For a constant $\mathcal{K}$ and considering a maximum path length $\frac{n(n-1)}{2}$, this computation runs in time $O(2^{\frac{n(n-1)}{2}})$ (worst-case complexity).

The dominant term in the expression of the overall worst-case complexity for computing the set of possible braids $\mathcal{T}^m$ for one permutation p_m^d is the exponential term $O(n!)$. This implies that the complexity of the present algorithm does not scale well with large numbers of agents. Furthermore, considering that the aforementioned computation needs to run for all possible permutations in $\mathcal{P}$, the complexity becomes $O((n!)^2)$. However, we argue that for our purposes, i.e. eventual deployment on a social robot navigating in real-world human environments, the following

considerations may enable us to restrict ourselves to low n and $|\mathcal{P}|$ and $\mathcal{K}$.

(a) A real robot has limited sensing capabilities, usually corresponding to a local radius of a few meters. Therefore, even in crowded environments, the surrounding agents will be considered as they enter the sensing radius and not universally.
(b) The rationality assumption for agents, which is supported by studies on human inference (Csibra and Gergely, 2007) and human navigation (Wolfinger, 1995). Rational agents aim at avoiding undesired divergences from the direction pointing towards their destinations (Moussaïd et al., 2011). This implies that (i) humans end up following short permutation paths and (ii) a good final permutation prediction can be done by simply projecting forward agents' current velocity on the boundary of the robot's sensing radius (the robot does not need to know exactly where others are going). These observations motivate low $\mathcal{K}$ and low $|\mathcal{P}|$, respectively.
(c) The rationality assumption also allows us to assume that once two agents pass each other, they stop reacting. For this reason, at replanning time, we restrict ourselves to considering only the number of agents that are ahead and are assumed to be observing the planning agent. Therefore, as the execution progresses and the robot approaches its destination, the number of reactive agents is expected to drop, allowing the robot to switch to efficient, and less computationally intense, execution.
(d) Our algorithm is based on frequent replanning. Plans have a short horizon but are made with a global reasoning (over joint strategies). The short planning horizon has been shown to be in compliance with human locomotion according to Carton et al. (2016a), who presented evidence that humans employ a shorter planning horizon as they navigate complex environments, to avoid collisions that could emerge from unexpected disturbances. The global planning horizon ensures that the motion of the robot will be consistent throughout the whole sequence of consecutive planning cycles.

For reference, in a game with five agents, a replanning cycle of SCN that generates three permutation paths per permutation runs at an average of $\sim$ 185 ms, with the worst case being $\sim$ 402 ms in a non-optimized MatLab implementation on a MacBook Pro of 2015 with an Intel Core i7 processor of 2.5 GHz, running macOS Sierra. These times appear to be encouraging for real-time execution on a mobile robot platform, upon the transfer to a faster language and the appropriate code optimizations.

6. Evaluation

In this section, we present simulation results, demonstrating the benefits of our approach. Section 6.1 describes the experimental setup and provides implementation details, whereas Section 6.2 presents results extracted by testing our algorithm under different settings.

Algorithm 3. GREEDY($Q, d, \Xi, AtGoal, M$)

Input: Q, system state; d, agent's destination; Ξ, state history of all agents; *At_Goal*, Boolean variable signifying arrival at agent's destination; M, context.
Output: a, action selected for execution

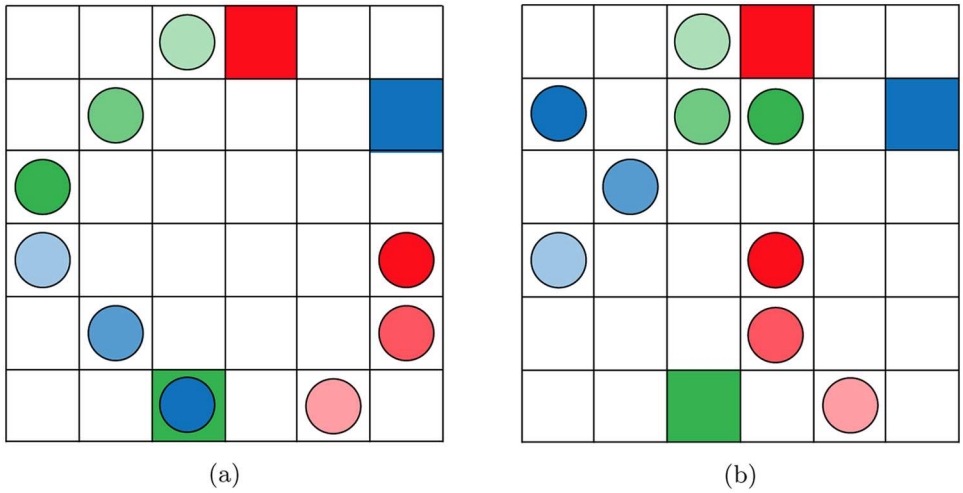
```
while ¬AtGoal do
    A ← Collision_Check(Q, M)
    a ← Efficiency_Optim(A, Q, d, M)
end while
return a
```

6.1. Setup

We consider a setup where a set of n agents navigate a discretized square workspace, partitioned into a set of N_t^2 tiles, where N_t is the number of tiles per side (Figure 10). Each agent $i \in \{1, \ldots, n\}$ starts from an initial tile q_i and moves towards a final tile d_i. The game is played in rounds until all agents reach their destinations.

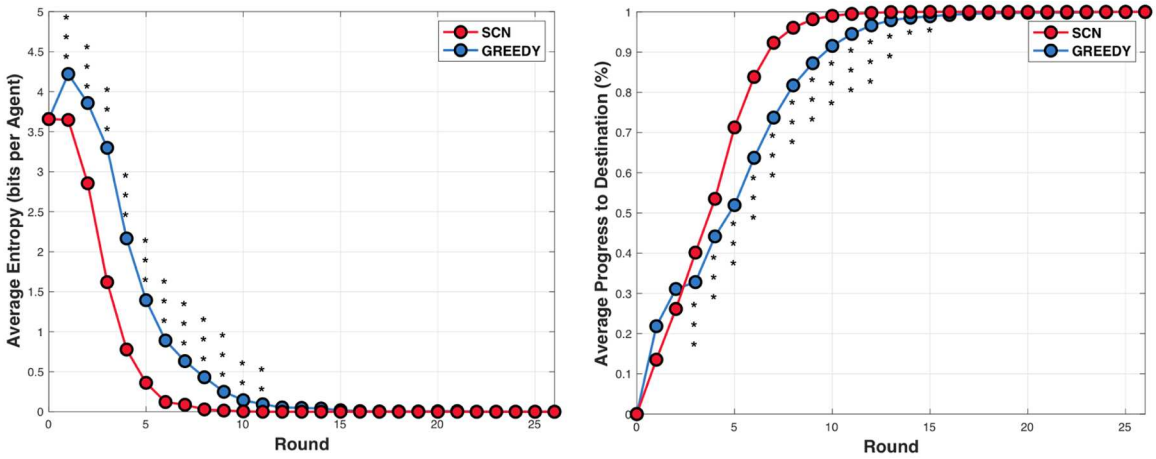
To assess our approach and demonstrate its benefits, we consider challenging game scenarios that reinforce intense encounters among agents. We do so by positioning agents on the sides of the workspace and having them navigate towards opposing sides. Each game scenario is sampled at random from the space of scenarios of size $N_t^n \times N_t^n$, corresponding to the number of distinct assignments of agents to initial and final configurations. At every round, the players simultaneously pick an action, which corresponds to a neighboring, unoccupied square. Forward, backward, left, right, and diagonal, collision-free transitions are allowed. Since at planning time each agent has no access to the plans of others, in order to ensure collision avoidance, transitioning to a square that is adjacent to a square currently occupied by another agent is not allowed. Depending on the number of agents and the size of the workspace, this setup might result in deadlocks. In our evaluation, executions that result in deadlocks are discarded. However, note that the purpose of our evaluation is to study how agents behave when they have **multiple** *actions* available. We examine how the adoption of different strategies in action selection may affect the evolution of the game qualitatively and quantitatively.

To demonstrate the importance of incorporating a topological understanding into agents' decision-making process, we compare the performance of our algorithm (SCN) against a *greedy* baseline (see Algorithm 3) that plans actions, greedily seeking to maximize its efficiency (the progress to the agent's destination) at every round. The GREEDY algorithm makes use of the same collision checking function as SCN. Their main difference lies in how they select an action when multiple of them are available.



(a) (b)

Fig. 10. A game with three agents in a 6×6 workspace. (a) and (b) depict partial executions of the same scenario (same start and end positions for all agents) with SCN and GREEDY respectively. The current system state is denoted with non-transparent system circles, whereas faded configurations correspond to configurations of past time steps. (a) SCN: By the end of round 2, the agents have reached a system configuration corresponding to a clear consensus over a joint strategy. (b) GREEDY: By the end of round 2, the agents are in a system configuration that is about to lead to conflicting encounters.



(a) (b)

Fig. 11. Comparative diagrams, generated upon running 200 experiments with three agents in a workspace of size 6×6. (a) Average entropy profile per agent per experiment over 200 scenarios involving three agents. On average, SCN agents reached a consensus over a joint strategy faster. (b) Average efficiency profile per agent per experiment over 200 scenarios involving three agents. On average, SCN agents made greater progress towards their destinations per round. The red curves correspond to agents running SCN and the blue curves to agents running GREEDY. The compromise between *efficiency* and *consensus* was set to $\lambda = 0.2$ and the number of paths per permutation to three. Student's t-tests were performed on all rounds to determine the statistical significance of the profiles difference. The symbols $*$, $**$, and $***$ denote rounds on which the difference in the performance between SCN and GREEDY was found to be significant to a degree described by p-values < 0.05, 0.01, and 0.001 respectively, according to a paired Student's t-test. Owing to space constraints, the significance symbols were placed vertically.

Considering a homogeneous setup (n agents running SCN versus n agents running GREEDY), we show that explicitly reasoning about the emerging joint strategy at planning time, benefits everyone in the scene, as it leads to a faster uncertainty decrease that simplifies everyone's decision making. Note that despite the different braid convention that each agent is making, they still manage to converge to a mutually beneficial consensus on a joint strategy of avoidance much faster than GREEDY agents. Qualitatively, our algorithm leads to less ambiguous system configurations, which result in higher average progress per round and lower average time to destination.

6.2. Simulation results

In this section, we present the behavior that our algorithm generates and demonstrate its benefits by comparing its performance with the performance of the GREEDY baseline.

6.2.1. Qualitative behavior. Figure 10 depicts partial executions, after two rounds, of a scenario from a game involving three agents. Figure 10a depicts the play of agents running SCN, whereas Figure 10b shows the corresponding play of the agents running GREEDY. It can be observed that the agents running our algorithm have led the game to a configuration that is more beneficial for everyone, as all of their encounters are essentially resolved by the end of the second round. This was achieved by planning informative actions that rapidly led to a significant entropy decrease and accelerated convergence to a consensus over a joint strategy. In contrast, the agents running the baseline, having initialized their game by focusing on efficiency, are now reacting suboptimally to their constrained action spaces. A video presentation, demonstrating the concept of our approach as well as our system in action, is provided as Extension 1.

6.2.2. Performance. Figure 11 depicts comparative performance diagrams, derived upon running 200 randomly sampled game scenarios involving three agents navigating a small workspace of size 6×6. The quality of agents' decision making is illustrated in the profiles of average entropy and average progress to destination, depicted in Figure 11a and (b), respectively. It can be observed that the systems of agents running SCN achieve faster entropy reduction and higher average progress towards destinations, compared to the systems running the baseline, with statistical significance noted in the diagrams. Figure 13a depicts comparative plots of average time to destination (left) and average time to "get free," which corresponds to the time an agent first has full control over the scene evolution, i.e. the first time when no other agents are ahead.

Similar comments can be made for the case of four agents, navigating a workspace of the same size 6×6. Figure 12 presents comparative performance diagrams for entropy and progress to destination, whereas Figure 13b depicts a comparison of average time to destination and average time to get free.

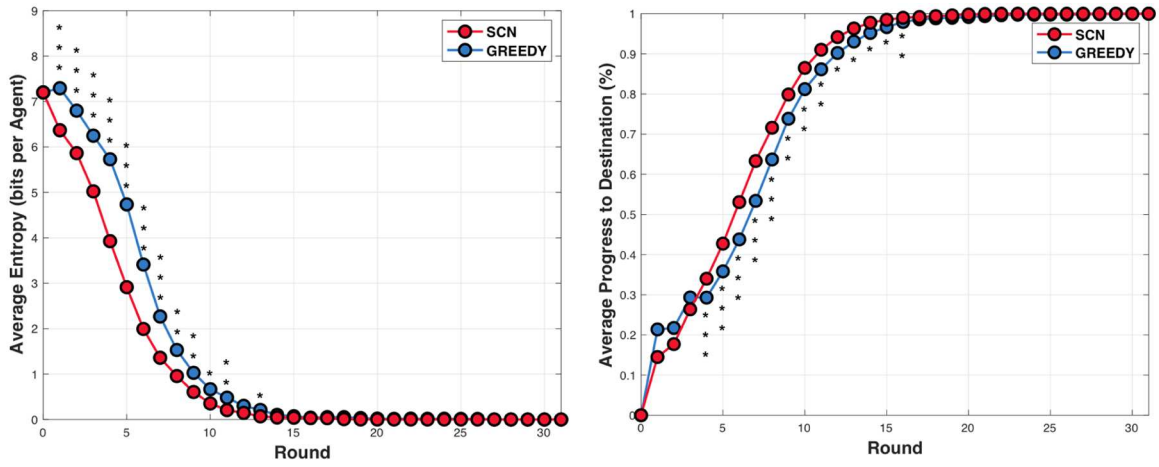
For the simulated examples presented, each agent models joint strategies as braids, defined with respect to a projection line that is parallel to its starting side. For the case of the GREEDY agents, entropy was evaluated by employing the inference mechanism of the agents running SCN, i.e. we computed what their belief would be if they had access to the inference mechanism of SCN agents. To ensure proper comparison, the same set of braids was considered at each time step for the same agent in both setups. The weighting factor λ was set to 0.2, as it was found experimentally to lead to a desired compromise between efficiency and consensus. For deriving multiple candidate paths in the permutation graph, we use the algorithm of Yen (1971) for finding *K shortest paths*.

7. Discussion and future work

We have considered the problem of decision making in a navigation scenario involving multiple rational and cooperative agents that do not explicitly communicate with each other. In such a scenario, the uncertainty over the exact strategies of other agents make it hard for an agent to predict their behaviors and, thus, to make safe and socially compliant decisions over its own actions. These settings may be found in a variety of real-world application, such as robotic navigation in crowded human environments.

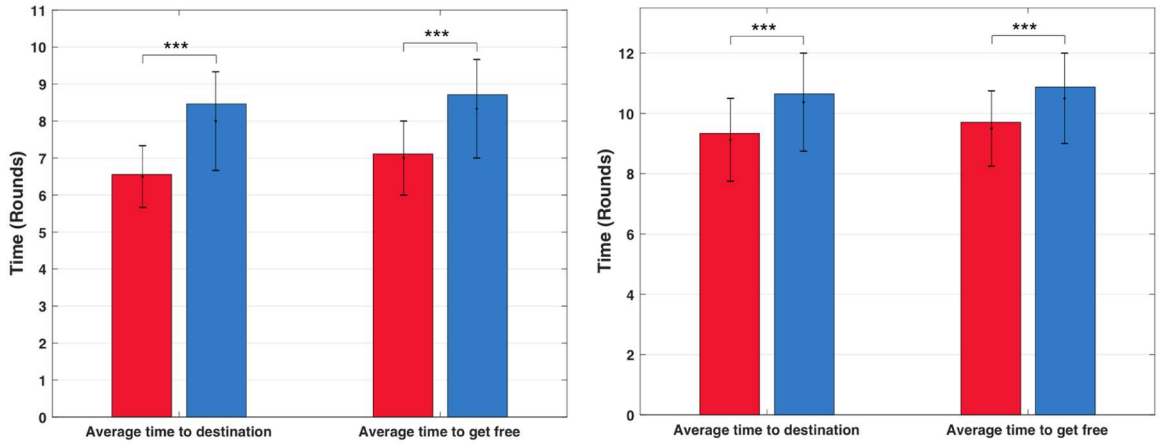
To address this problem, we presented an online planning framework, inspired by the insights of recent studies on the cooperative nature of pedestrian behavior (Wolfinger, 1995) and the goal-directed inference of humans (Csibra and Gergely, 2007). Our framework explicitly incorporates the concept of cooperation by modeling multi-agent collective behaviors as topological global joint strategies, using the formalism of braids (Birman, 1975). Our topological model forms the basis of an inference mechanism that associates observed behaviors with future collective topologies. In the decision-making stage, each agent decides on an action that corresponds to a compromise between its personal efficiency (progress towards destination) and a form of joint efficiency (the status of a consensus on a joint strategy of avoidance). To clearly showcase the benefits of our decision-making concept, we deliberately studied a simplified version of the real-world problem by considering an abstract, discrete setup, involving artificial agents playing a cooperative game. Extensive trials over randomly generated, challenging scenarios demonstrated the benefits of reasoning about joint strategies over a baseline that greedily prescribed actions of high efficiency. Our algorithm was shown to lead to a faster decrease of uncertainty regarding the scene evolution, which resulted in efficiency increase and lower execution times with high statistical significance.

Our purpose in this paper was to demonstrate the benefits of a topologically based decision-making mechanism for dynamic multi-agent environments. Future work involves modifications of the proposed approach towards enabling online, sensor-based, real-world operation in unstructured environments. We contributed a first step towards that direction by developing a data-driven approximation of our inference mechanism with a deep neural model, learned from demonstrations of simulated, challenging, multi-agent scenarios in continuous settings (Mavrogiannis et al., 2017); a comparative evaluation against the social force model (Helbing and Molnár, 1995) demonstrated promising results in simulation. Another step towards approaching more realistic execution settings was done with our social momentum motion planner (Mavrogiannis et al., 2018), which also



(a) (b)

Fig. 12. Comparative diagrams, generated upon running 200 experiments with four agents in a workspace of size 6×6. (a) Average Entropy profile per agent per experiment over 200 scenarios involving four agents. On average, SCN agents reached a consensus over a joint strategy faster. (b) Average Efficiency profile per agent per experiment over 200 scenarios involving four agents. On average, SCN agents made greater progress towards their destinations per round. The red curves correspond to agents running SCN and the blue curves to agents running GREEDY. The compromise between *efficiency* and *consensus* was set to $\lambda = 0.2$ and the number of paths per permutation to three. Student's t-tests were performed on all rounds to determine the statistical significance of the profiles difference. The symbols $*$, $**$, and $***$ denote rounds on which the difference in the performance between SCN and GREEDY was found to be significant to a degree described by p-values < 0.05, 0.01, and 0.001, respectively, according to a paired Student's t-test. Owing to space constraints, the significance symbols were place vertically.



(a) (b)

Fig. 13. Average time to destination and average time to get free, i.e. reach a configuration at which no agents are ahead, generated after running 200 experiments involving (a) three and (b) four agents, in a workspace of size 6×6. Red bars correspond to agents running SCN and blue bars to agents running GREEDY. On average, the SCN agents (red color bars) reached their destination faster and managed to "get free" faster than the GREEDY agents (blue color bars). The error bars indicate 25–75 percentiles. For these experiments, the compromise between efficiency and consensus was set to $\lambda = 0.2$ and the number of paths per permutation to 3. The $***$ symbol denotes a highly significant timing difference, according to Student's t-test (p-value < 0.001).

follows the principles of topological braid theory to generate legible behaviors in multi-agent environments; this approach was also shown to outperform social force and ORCA (van den Berg et al., 2009) in terms of intent-expressiveness, social compliance, and topological complexity of executions.

Ongoing work involves learning a data-driven model of our inference mechanism from *human* trajectory datasets (e.g. Brščić et al., 2013) that would enable agents to form a humanlike belief over the future behavior of systems of multiple agents. We are also in the process of planning an extensive user study that will measure the effects of the

behaviors generated by our planning architecture on human subjects in a controlled lab experiment.

Funding

This material is based upon work supported by the National Science Foundation (grant numbers IIS-1526035 and IIS-1563705). We are grateful for this support.

ORCID iD

Christoforos I Mavrogiannis, https://orcid.org/0000-0003-4476-1920

Notes

1. This specific convention is not particularly important, as long as one is consistent. There are works that use the inverse convention when defining the positive and negative generator exponents. Our selection facilitates the exposition of further concepts in the remainder of the paper.
2. A transposition can be described as a permutation involving exactly one swap of a pair of elements. An adjacent transposition is a transposition involving an exchange of two adjacent elements. An adjacent transposition implementing an exchange of the elements with order j and $j+1$, with $1 \leq j < n-1$ in a list of n elements, is commonly denoted as $\beta_j = \begin{bmatrix} j & j+1 \end{bmatrix}$.

References

Alahi A, Goel K, Ramanathan V, Robicquet A, Fei-Fei L and Savarese S (2016) Social LSTM: Human trajectory prediction in crowded spaces. In: *Proceedings of the 2016 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 961–971.

Artin E (1947a) Braids and permutations. *Annals of Mathematics Second Series* 48: 643–649.

Artin E (1947b) Theory of braids. *Annals of Mathematics* 48(1): 101–126.

Baker CL, Tenenbaum J and Saxe R (2009) Action understanding as inverse planning. *Cognition* 113(3): 329–349.

Bandyopadhyay T, Won KS, Frazzoli E, Hsu D, Lee WS and Rus D (2012) Intention-aware motion planning. In: *Proceedings of the 2012 International Workshop on the Algorithmic Foundations of Robotics (WAFR '12)*.

Bennewitz M, Burgard W, Cielniak G and Thrun S (2005) Learning motion patterns of people for compliant robot motion. *The International Journal of Robotics Research* 24: 31–48.

Bera A, Randhavane T, Prinja R and Manocha D (2017) Sociosense: Robot navigation amongst pedestrians with social and psychological constraints. In: *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '17)*, pp. 7018–7025.

Birman JS (1975) *Braids Links And Mapping Class Groups*. Princeton, NJ: Princeton University Press.

Brščić D, Kanda T, Ikeda T and Miyashita T (2013) Person tracking in large public spaces using 3-d range sensors. *IEEE Transactions on Human–Machine Systems* 43(6): 522–534.

Carton D, Nitsch V, Meinzer D and Wollherr D (2016a) Towards assessing the human trajectory planning horizon. *PLOS ONE* 11(12): 1–39.

Carton D, Olszowy W and Wollherr D (2016b) Measuring the effectiveness of readability for mobile robot locomotion. *International Journal of Social Robotics* 8(5): 721–741.

Chen YF, Everett M, Liu M and How JP (2017) Socially aware motion planning with deep reinforcement learning. In: *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '17)*, pp. 1343–1350.

Csibra G and Gergely G (1998) The teleological origins of mentalistic action explanations: A developmental hypothesis. *Developmental Science* 1(2): 255–259.

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

Diaz-Mercado Y and Egerstedt M (2017) Multirobot mixing via braid groups. *IEEE Transactions on Robotics* 33(6): 1375–1385.

Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numerische Mathematik* 1(1): 269–271.

Dragan AD and Srinivasa S (2014) Integrating human observer inferences into robot motion planning. *Autonomous Robots* 37(4): 351–368.

Farina F, Fontanelli D, Garulli A, Giannitrapani A and Prattichizzo D (2017) Walking ahead: The headed social force model. *PLOS ONE* 12(1): 1–23.

Ferguson S, Luders B, Grande RC and How JP (2015) Real-time predictive modeling and robust avoidance of pedestrians with uncertain, changing intentions. In: *Algorithmic Foundations of Robotics XI: Selected Contributions of the Eleventh International Workshop on the Algorithmic Foundations of Robotics*. Cham: Springer International Publishing, pp. 161–177.

Ghrist RR (1999) Configuration spaces and braid groups on graphs in robotics. Preprint https://arxiv.org/abs/math/9905023.

Goffman E (1966) *Behavior in Public Places: Notes on the Social Organization of Gatherings*. Free Press.

Hall E (1990) *The Hidden Dimension*. Anchor Books.

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

Henry P, Vollmer C, Ferris B and Fox D (2010) Learning to navigate through crowded environments. In: *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 981–986.

Hoogendoorn S and Bovy PH (2003) Simulation of pedestrian flows by optimal control and differential games. *Optimal Control Applications and Methods* 24(3): 153–172.

Karamouzas I, Heil P, van Beek P and Overmars MH (2009) A predictive collision avoidance model for pedestrian simulation. In: *Motion in Games*. Berlin: Springer, pp. 41–52.

Karamouzas I, Skinner B and Guy SJ (2014) Universal power law governing pedestrian interactions. *Physical Review Letters* 113: 238701.

Karp D, Stone G and Yoels W (1977) *Being urban: a social psychological view of city life*. Heath.

Kassel C and Turaev V (2008) *Braid Groups* (*Graduate Texts in Mathematics*, vol. 247). New York: Springer.

Katoh N, Ibaraki T and Mine H (1982) An efficient algorithm for k shortest simple paths. *Networks* 12(4): 411–427.

Kim B and Pineau J (2016) Socially adaptive path planning in human environments using inverse reinforcement learning. *International Journal of Social Robotics* 8(1): 51–66.

Knepper RA, Mavrogiannis CI, Proft J and Liang C (2017) Implicit communication in a joint action. In: *Proceedings of the 2017 ACM/IEEE International Conference on Human–Robot Interaction (HRI '17)*, pp. 283–292.

Knepper RA and Rus D (2012) Pedestrian-inspired sampling-based multi-robot collision avoidance. In: *Proceedings of the 2012 IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN '12)*, pp. 94–100.

Kretzschmar H, Spies M, Sprunk C and Burgard W (2016) Socially compliant mobile robot navigation via inverse reinforcement learning. *The International Journal of Robotics Research* 35(11): 1289–1307.

Ma WC, Huang DA, Lee N and Kitani KM (2017) Forecasting interactive dynamics of pedestrians with fictitious play. In: *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '17)*, pp. 4636–4644.

Mavrogiannis CI, Blukis V and Knepper RA (2017) Socially competent navigation planning by deep learning of multi-agent path topologies. In: *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '17)*, pp. 6817–6824.

Mavrogiannis CI and Knepper RA (2016) Decentralized multi-agent navigation planning with braids. In: *Proceedings of the 2016 International Workshop on the Algorithmic Foundations of Robotics (WAFR '16)*.

Mavrogiannis CI, Thomason WB and Knepper RA (2018) Social momentum: A framework for legible navigation in dynamic multi-agent environments. In: *Proceedings of the 2018 ACM/IEEE International Conference on Human–Robot Interaction (HRI '18)*. New York: ACM Press, pp. 361–369.

Mead R and Matarić MJ (2017) Autonomous human–robot proxemics: socially aware navigation based on interaction potential. *Autonomous Robots* 41(5): 1189–1201.

Moussaïd M, Helbing D and Theraulaz G (2011) How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences of the USA* 108(17): 6884–6888.

Park JJ, Johnson C and Kuipers B (2012) Robot navigation with model predictive equilibrium point control. In: *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '12)*, pp. 4945–4952.

Pellegrini S, Ess A, Schindler K and Gool LJV (2009) You'll never walk alone: Modeling social behavior for multi-target tracking. In: *Proceedings of the International Conference on Computer Vision (ICCV '09)*, pp. 261–268.

Sadigh D, Sastry S, Seshia SA and Dragan AD (2016) Planning for autonomous cars that leverages effects on human actions. In: *Proceedings of the 2016 Conference on Robotics: Science and Systems*, pp. 239–248.

Scovanner P and Tappen MF (2009) Learning pedestrian dynamics from the real world. In: *2009 IEEE International Conference on Computer Vision (CVPR '09)*, pp. 381–388.

Sehestedt S, Kodagoda S and Dissanayake G (2010) Robot path planning in a social context. In: *Proceedings of the 2010 IEEE Conference on Robotics, Automation and Mechatronics*, pp. 206–211.

Sisbot EA, Marin-Urias LF, Alami R and Siméon T (2007) A human aware mobile robot motion planner. *IEEE Transactions on Robotics* 23(5): 874–883.

Thiffeault JL (2010) Braids of entangled particle trajectories. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 20(1): 017516.

Thiffeault JL and Budišić M (2013–2017) Braidlab: A software package for braids and loops. Version 3.2.3. Preprint http://arXiv.org/abs/1410.0849.

Thomaz A, Hoffman G and Cakmak M (2016) Computational human–robot interaction. *Foundations and Trends in Robotics* 4(2–3): 105–223.

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

Treuille A, Cooper S and Popović Z (2006) Continuum crowds. *ACM Transactions on Graphics* 25(3): 1160–1168.

van den Berg J, Guy SJ, Lin MC and Manocha D (2009) Reciprocal *n*-body collision avoidance. In: *Proceedings of the International Symposium on Robotics Research (ISRR '09)*, pp. 3–19.

Warren WH (2006) The dynamics of perception and action. *Psychological Review* 113(2): 358–389.

Wiese E, Wykowska A, Zwickel J and Müller HJ (2012) I see what you mean: How attentional selection is shaped by ascribing intentions to others. *PLOS ONE* 7(9): 1–7.

Wolfinger NH (1995) Passing moments: Some social dynamics of pedestrian interaction. *Journal of Contemporary Ethnography* 24(3): 323–340.

Yen J (1971) Finding the *k* shortest loopless paths in a network. *Management Science* 17(11): 712–716.

Zhou B, Wang X and Tang X (2012) Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents. In: *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2871–2878.

Ziebart BD, Ratliff N, Gallagher G, et al. (2009) Planning-based prediction for pedestrians. In: *Proceedings of the 2009 International Conference on Intelligent Robots and Systems (IROS '09)*, p. 3931–3936.

Ziegler GM (1995) *Lectures on polytopes*. New York: Springer-Verlag.

Appendix. Index to multimedia extensions

Archives of IJRR multimedia extensions published prior to 2014 can be found at http://www.ijrr.org, after 2014 all videos are available on the IJRR YouTube channel at http://www.youtube.com/user/ijrrmultimedia

Table of Multimedia Extension

Extension	Media type	Description
1	Video	A simulated demonstration of our system, illustrating the main concepts underlying our approach.