2022 IEEE International Conference on Robotics and Automation (ICRA)
May 23-27, 2022. Philadelphia, PA, USA

Joint Communication and Motion Planning for Cobots

Mehdi Dadvar[1], Keyvan Majd[1], Elena Oikonomou[1], Georgios Fainekos[1], and Siddharth Srivastava[1]

Abstract— **The increasing deployment of robots in co-working scenarios with humans has revealed complex safety and efficiency challenges in the computation of the robot behavior. Movement among humans is one of the most fundamental —and yet critical—problems in this frontier. While several approaches have addressed this problem from a purely navigational point of view, the absence of a unified paradigm for communicating with humans limits their ability to prevent deadlocks and compute feasible solutions. This paper presents a joint communication and motion planning framework that selects from an arbitrary input set of robot's communication signals while computing robot motion plans. It models a human co-worker's imperfect perception of these communications using a noisy sensor model and facilitates the specification of a variety of social/workplace compliance priorities with a flexible cost function. Theoretical results and simulator-based empirical evaluations show that our approach efficiently computes motion plans and communication strategies that reduce conflicts between agents and resolve potential deadlocks.**

I. Introduction

Technological breakthroughs of the past decade have led to increasingly common human-robot co-working environments [1]. Navigating among humans is an imperative task that most cobots, ranging from industrial to service robots, are expected to perform safely and efficiently. Although motion planning for autonomous robots has been studied from multiple perspectives [2]–[4], these approaches focus on movement actions and do not address the problem using communication to resolve situations that require extensive human-robot interaction. The objective of this paper is to develop a unified paradigm for computing movement and communication strategies that improve efficiency and reduce movement conflicts in co-working scenarios (see Fig. 1).

Although the problem of integrated task and motion planning has received significant research attention [5]–[10], the integration of these deliberative processes with communication has not been studied sufficiently. Prior work on this topic includes extensions to sampling based motion planning paradigms that model pedestrians as moving obstacles [11], [12]. While these extensions provide valuable enhancements of well-known and efficient algorithms, they view humans as impervious entities and have limited applicability in co-working scenarios where both the human and the robot need to adjust their behavior to allow feasible solutions. On the other hand, there are approaches that employ disjoint prediction models to establish simple interactions with humans to generate safer and more risk-aware motion plans [13]. Since these approaches neglect the effect of the robot's motion on the human's behavior, they suffer from *the robot freezing problem* where the robot cannot find any safe solution. To address this limitation, *socially compliant* methods consider potential human-robot cooperation via learning and planning techniques [14] to produce legible plans [15] or plans subject to stipulations on the information divulged during plan execution [16]. [17] employs inverse reinforcement learning (IRL) to learn interactive models of pedestrians in the environment for social compliant path planning. Further, [18] presents a social navigation framework that adapts the social force model (SFM) to generate human-friendly collision-free motion plans in unknown environments. Although these approaches model the effect of the robot's movement on the humans' behavior for legible motion planning, relying purely on motion actions, taxonomically known as implicit communication [19], could be misleading for the human [20] and may lead to deadlocks in confined environments.

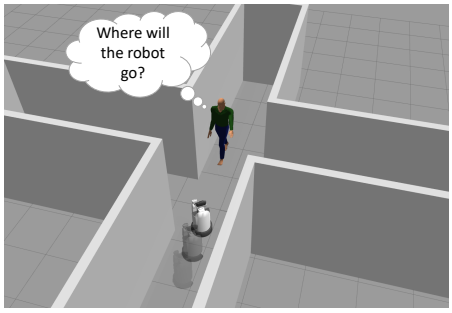

Fig. 1: An example of a social navigation scenario in a confined environment where the robot's movement can not reveal any information about its future intentions.

Employing explicit communication [21] coupled with the robot's movements would enrich the human-robot interaction. [22] uses IRL to model the effects of both explicit and implicit actions of the robot on the human's behavior. Further, a robot planner relies on this model to produce communicative actions to maximize the robot's clarity. Since this method assumes predefined behavior modes for the robot and human (robot priority and human priority), the solution always impels one agent to slow down, which degrades the planning effectiveness.

In contrast, we formalize a unified deliberative communication planning problem that addresses the joint problem of computing the robot's communication strategy and movements while taking into account the human's imper-

This work was supported in part by the NSF under grants IIP-1361926, IIS-1909370, IIS-1844325, OIA-1936997 and the NSF I/UCRC Center for Embedded Systems.

[1] The authors are with the School of Computing and Augmented Intelligence, Arizona State University, Tempe, AZ, USA. {mdadvar,majd,e.oikonomou,fainekos,siddharths}@asu.edu

2022 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9681-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICRA46639.2022.9812261

fect perception about the robot and its communications (Sec. III). We use a noisy communication model to estimate the results of robot's communications on the human's belief of the robot's possible locations. In contrast to the human prediction framework in [22], which requires the robot's future trajectories (the need for socially compliant planning illustrates the difficulty of obtaining such inputs), our approach supports arbitrary human movement prediction models that can predict human behaviors given a set of possible obstacles. Our solution paradigm derives estimates of the human's belief on the robot's positions to compute robot communication and movement plans (Sec. III-C)). This is done using a hierarchical search process in conjunction with a socially compliant motion planner Control Barrier Function enabled Time-Based RRT (CBF-TB-RRT) [23] (Sec. III-B). Theoretical results and extensive simulations on various environments show that this approach efficiently avoids deadlocks and computes mutually efficient solutions without requiring preset behavior modes.

II. Deliberative Communication Planning

We formulate the deliberative communication planning problem $\mathcal{P}_{\mathcal{DC}}$ as the problem of jointly computing communication signals with corresponding feasible motion plans for R in a social navigation scenario. As a starting point, we focus on settings with a single robot and a single human H. In such problems, R's actions $\mathcal{A}$ include communication as well as movement actions. In order to model realistic scenarios, we use potentially noisy models of H's movement (T_H) and of H's sensing (O) of R's communications. We use these models to evaluate possible courses of action while computing efficient, collision-free communication and movement plans for R.

Intuitively, T_H maps the current state of H and H's belief about the possible positions of R at the next planning cycle to possible motion plans for H. We model H's sensor model O as a variation of the standard noisy sensor paradigm used in planning under partial observability. O relates H's current state, R's communication action and R's intended next state to the observation signal that H receives. In this formulation, *H need not know R's current/intended states nor the exact communication that it executed – H* only receives an observation signal. Such sensor models are very general: they can capture a variety of scenarios ranging from perfect communication to imperfect communication settings where H may not have a perfect understanding or observation of R's communications and may conflate R's communication actions with each other.

Definition 1. A deliberative communication planning problem is a tuple $\mathcal{P}_{\mathcal{DC}} = \langle \mathcal{S}, s^0, \mathcal{A}, T, \mathcal{G}, O, J \rangle$, where:

- $\mathcal{S} = \mathcal{S}_R \times \mathcal{S}_H$ is the set of states consisting of R's and H's states, respectively.
- $s^0 = s_r^0 \times s_h^0$ are the initial states of R and H, respectively, where $s_r^0 \in \mathcal{S}_R$ and $s_h^0 \in \mathcal{S}_H$.
- $\mathcal{A}$ is the set of R's actions defined as $\mathcal{A} = \mathcal{A}_c \cup \mathcal{A}_m$, where $\mathcal{A}_c$ is a set of communication signals that includes the null communication, and $\mathcal{A}_m$ is the implicit uncountable set of R's feasible motion plans. Each feasible motion plan $\pi_R \in \mathcal{A}_m$ is a continuous function $\pi_R : [0,1] \to \mathcal{S}_R$ where $\pi_R(0) = s_r^0$ and $\pi_R(1) \in \mathcal{S}_R$.
- $\mathcal{G} = \langle \mathcal{G}_R, \mathcal{G}_H \rangle$ is the goal pair where $\mathcal{G}_R \subseteq \mathcal{S}_R$ is R's goal set and $\mathcal{G}_H \subseteq \mathcal{S}_H$ is H's goal set.
- $T = \langle T_R, T_H \rangle$ constitutes transition/movement models of both agents where T_R is R's transition function defined as $T_R : \mathcal{S}_R \times \mathcal{A}_m \to \mathcal{S}_R$, and $T_H : \mathcal{S}_H \times \mathcal{G}_H \times \mathcal{B}_H^{R'} \to 2^{\Pi_H}$ denotes H's movement model where $\mathcal{B}_H^{R'}$ is the set of possible beliefs over the state of R at the next planning cycle and Π_H is the set of feasible H movement plans within $\mathcal{S}_H$. T_H may be available as a simulator that yields a sample of the possible H plans.
- O is H's sensor model defined as $O : \mathcal{S}_H \times \mathcal{A}_c \times \mathcal{S}_R \to \Omega$, where Ω denotes H's observation. Situations where H cannot perfectly understand or observe R's communication can be modeled by mapping multiple tuples $\langle \mathbf{s}_h, a_c, \mathbf{s}_r \rangle$ to the same $\omega \in \Omega$, where $a_c \in \mathcal{A}_c$.
- $J : \mathcal{S}_H \times \mathcal{S}_R \times \mathcal{A} \to \mathbb{R}$ is a utility function denoting the value of a joint H-R state and a communication-motion action. In practice, we express J as a cost function.

A solution to $\mathcal{P}_{\mathcal{DC}}$ is a sequence of communication actions and motion plans that satisfy $\mathcal{G}_R$, and is defined as follow.

Definition 2. A solution to the deliberative communication planning problem $\mathcal{P}_{\mathcal{DC}} = \langle \mathcal{S}, s^0, \mathcal{A}, T, \mathcal{G}, O, J \rangle$ is a finite sequence of communication and movement actions: $\Psi = \langle (a_c^1, \pi_R^1), (a_c^2, \pi_R^2), \cdots, (a_c^q, \pi_R^q) \rangle$, where $a_c^i \in \mathcal{A}_c$, $\pi_R^i \in \mathcal{A}_m$, $\pi_R^1(0) = s_r^0$, $\pi_R^i(1) = \pi_R^{i+1}(0)$, and $\pi_R^q(1) \in \mathcal{G}_R$ for $i = 1, \cdots, q$.

III. Methodology

A. Overview

In the proposed paradigm of joint communication and motion planning, a motion planner (MP) returns a set of feasible and collision-free motion plans $\Pi_R \in \mathcal{A}_m$ considering the goal set $\mathcal{G}$. Accordingly, a communication planner (CP) uses a search tree to select a combination of a communication action and a motion plan at each planning cycle that minimizes J. Each node of this search tree is defined by $\langle \mathbf{s}, a_c, \pi_R \rangle$ where $\mathbf{s} \in \mathcal{S}$, $a_c \in \mathcal{A}_c$ and $\pi_R \in \Pi_R$. Here, a_c denotes the communication action being considered at this node while π_R denotes one of the plans returned by MP.

Fig. 2 illustrates the mechanism by which MP and CP interact. MP utilizes CBF-TB-RRT with H's movement model and R's dynamic model to produce a finite set of feasible motion plans $\Pi_R \subset \mathcal{A}_m$ (Sec. III-B). Starting with a node representing the current state, CP creates a successor node for each combination of a feasible plan in Π_R and a communication action from $\mathcal{A}_c$. For each such combination, it uses a belief update process to compute and store an estimate of H's next belief if R were to use the corresponding communication action. At each planning cycle, CP selects a node of tree that minimizes J (CP is described in Sec. III-C). An important property of this approach is that

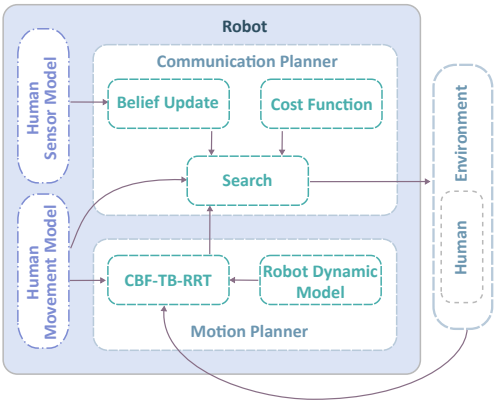

Fig. 2: An overview of our approach.

our solution algorithms are independent of the choice of R, the environment, and the H's movement and sensor models.

B. CBF-based TB-RRT as MP

We obtain a set Π_R of diverse plans by employing CBF-TB-RRT [23] as MP. This method provides a probabilistic safety guaranteed solution in real-time to the start-to-goal motion planning problem. At each time step, given a probabilistic trajectory of dynamic agents, this method extracts ellipsoidal reachable sets for the agents for a given time horizon with a bounded probability. This method extends time-based RRT (each node of TB-RRT denotes a specific state in a specific time), proposed in [24], with CBFs to generate path segments for R that avoid the agents' reachable sets while moving toward goal. If the probability distribution over the dynamic agents' future trajectory for a given finite time horizon is accurate, then the generated control by CBF-TB-RRT guarantees that the probability of collision at each time step is bounded.

In this paper, we modified the original CBF-TB-RRT method to better serve our hierarchical framework as follows. First, the set of possible future trajectories for H can either be given by a stochastic T_H or by a deterministic T_H with an ε tube around the predicted trajectory. We denote this region by $\mathcal{S}_R^{unsafe}$. MP maintains a continually updated estimate of R's safe states, $\mathcal{S}_R^{safe} = \mathcal{S}_R \backslash \mathcal{S}_R^{unsafe}$, where $\mathcal{S}_R^{safe}$ would be collision-free with respect to the predicted trajetories of H. Second, the original CBF-TB-RRT expands a tree for a finite time horizon and just apply the control for the first time-step at each planning cycle. In contrast, here, we let R to execute the full returned partial plan. Finally, instead of selecting one plan to execute, we select a set of p plans $\Pi_R \subseteq \mathcal{A}_m$ of $\bar{\pi}_{R,j} : [t_0, t_j] \rightarrow \mathcal{S}_R^{safe}$ for $j = \{1, 2, \cdots, p\}$. Here, each plan $\bar{\pi}_{R,j}$ represents a path segment from the initial vertex ν_0 at location $\bar{\pi}_R^{t_0}$ in time t_0 to another vertex ν_j at location $\bar{\pi}_R^{t_j}$ in time t_j. Assume c_j to be the cost of vertex ν_j in the set of all expanded RRT vertices $\mathcal{V}$. To select p diverse plans $\bar{\pi}_{R,j}$ with the minimum costs c_j for $j \in \{1, 2, \cdots, p\}$, we minimize $J_d = \sum_i \frac{w_c r_i c_i}{w_d \sum_{v, i \neq v} r_v d_{iv}}$ s.t. $\sum_i r_i = p$, and $r_i \in \{0, 1\}$ for $i, v = 0, \cdots, |\mathcal{V}|$. Here w_c and w_d are the numerator and denominator weights, respectively, d_{iv} is the Euclidean distance between vertices i and v, and $\mathbf{r}$ is a vector of binary values r_i for $i = 1, \cdots, |\mathcal{V}|$, that determines the selected plans (vertices).

Assumption 1. The future human motion remains within the unsafe region $\mathcal{S}_R^{unsafe}$ predicted by T_H.

Lemma 1. Following Assn. 1, all generated path segments $\bar{\pi}_{R,j}$ for $j = 1, \cdots, |\mathcal{V}|$ by CBF-TB-RRT are guaranteed to remain in $\mathcal{S}_R^{safe}$ if $\bar{\pi}_R^{t_0} \in \mathcal{S}_R^{safe}$.

Proof. This proof is immediate following [23, Prop. 1]. □

C. Communication Planner Module (CP)

As discussed in Sec. III-A, CP builds a search tree to select an optimal combination of communication action and motion plan. Recall that each node in the search tree consists of a state $\mathbf{s}$, a communication action a_c and a motion plan π_R. Here, π_R denotes the discretization of the continuous-time path segment $\bar{\pi}_R$ given by MP. We use a belief-space formulation to represent the set of locations where H might expect R to be at the next planning cycle $k+1$. Thus, the set of all possible beliefs of H, is the power set of $\mathcal{S}_R$. However, in practice H needs to keep track of only a subset of possible locations, in a small neighbourhood around H.

Definition 3. A ***δ-local neighborhood*** of H is a subset $\mathcal{L} \subseteq \mathcal{S}_R$ s.t. the Euclidean distance from S_H $d(\mathbf{s}_{xyz}, \mathcal{S}_H)$ of R's base coordinates $\mathbf{s}_{xyz}$ in state $\mathbf{s}$ is less than δ $\forall S \in \mathcal{L}$.

We maintain a bounded, discretized set of regions to approximate H's belief about R's presence in their δ-local neighborhood. Let $\mathcal{L}_H$ be the set of these discretized zones $\{l_1, \ldots, l_\ell\}$. Collectively these regions can represent neighborhoods in domain-specific configurations (e.g., an H-centered forward-biased cone or a rectangular region around H with discretized cells). Given a state $(\mathbf{s}_R, \mathbf{s}_H) \in \mathcal{S}$ we use $\mathbf{s}_R \in l_i(\mathbf{s}_H)$ to express that when R's state is $\mathbf{s}_R$ and H's state is $\mathbf{s}_H$, R will be in the region l_i in H's local neighborhood. In this notation, H's belief is a Boolean vector of dimension $|l_H|$, so that $b_i = 1$ in a belief $\mathbf{b}$ represents a belief that $\mathbf{s}_R \in l_i(\mathbf{s}_H)$ is possible at the next time step.

Given a starting belief $\mathbf{b}_k$ and an observation symbol ω_k, we can invert the sensor model and the transition function to derive a logical filtering based belief update equation for computing $\mathbf{b}_{k+1}$ as follows. Let $\varphi_1(\mathbf{s}_R^{k+1}, i)$ state that R at $\mathbf{s}_R^{k+1}$ would be in H's i^{th} neighborhood zone, i.e., $\mathbf{s}_R^{k+1} \in l_i(\mathbf{s}_H^k)$; $\varphi_2(\mathbf{s}_R^k, j)$ state that b_j^k was 1 with R at $\mathbf{s}_R$, i.e., $b_j^k = 1 \wedge \mathbf{s}_R^k \in l_j(\mathbf{s}_H^{k-1})$; $\varphi_3(\mathbf{s}_R^k, \mathbf{s}_R^{k+1})$ state that R can move from $\mathbf{s}_R^k$ to $\mathbf{s}_R^{k+1}$, i.e., $\exists a_m \in \mathcal{A}_m, T_R(\mathbf{s}_R^k, a_m) = \mathbf{s}_R^{k+1}$; and $\varphi_4(\mathbf{s}_H^k, \omega, \mathbf{s}_R^{k+1})$ state that R may have executed a communication action a_c that resulted in observation ω, i.e., $\exists a_c \in \mathcal{A}_c, o(\mathbf{s}_H^k, a_c, \mathbf{s}_R^{k+1}) = \omega$. Inverting the sensor model and the transition function gives us $b_i^{k+1} = 1 \textit{iff} \exists \mathbf{s}_R^k, \mathbf{s}_R^{k+1} \in \mathcal{S}_R; j \in [1, \ell] : \varphi_1(\mathbf{s}_R^{k+1}, i) \wedge \varphi_2(\mathbf{s}_R^k, j) \wedge \varphi_3(\mathbf{s}_R^k, a_m, \mathbf{s}_R^{k+1}) \wedge \varphi_4(\mathbf{s}_H^k, \omega, \mathbf{s}_R^{k+1})$. CP uses this expression to compute R's estimate of H's belief $\mathbf{b}^{k+1}$ given a belief $\mathbf{b}^k$ at the parent node and the observation ω that H would receive as a result

of the communication action being considered at that node. We use $\mathbf{b}(n)$ to denote this belief for node n.

CP uses a cost function J to evaluate a node $n = \langle \mathbf{s}, a_c, \pi_R \rangle$ in the search tree. Intuitively, J needs to consider H and R's future paths Γ_H and Γ_R, respectively. $\tilde{\Gamma}_R(n)$ is an estimate for Γ_R based on π_R. However, we do not have an accurate future path for H and we use $\mathbf{b}(n)$ and H's movement model T_H to obtain an estimate $\tilde{\Gamma}_H(n)$. We omit the node argument unless required for clarity.

For computational efficiency, we discretize Γ_R and Γ_H as sequences of waypoints: $\Gamma_R = \{\gamma_R^i\}_{i=1}^{i_{max}}$ and $\Gamma_H = \{\gamma_H^i\}_{i=1}^{i_{max}}$. W.l.o.g., both sequences have the same length as the agent with the shorter path can be assumed to stay at their final location for remainder of the other agent's path execution. Let $c(\tilde{\Gamma})$ be the sum of pairwise distances between successive waypoints in a path $\tilde{\Gamma}$ and let $\delta(\tilde{\Gamma}_1, \tilde{\Gamma}_2)$ be $\delta(\tilde{\Gamma}_1, \tilde{\Gamma}_2) = max(d_{min}(\tilde{\Gamma}_1, \tilde{\Gamma}_2) - \sigma^{safe}, 0)$, where σ^{safe} denotes the safety threshold and $d_{min}(\tilde{\Gamma}_1, \tilde{\Gamma}_2)$ is the minimum Euclidean distance between $\tilde{\Gamma}_1$ and $\tilde{\Gamma}_2$: $min_{i=1,\ldots,i_{max}}\{d(\gamma_1^i, \gamma_2^i)\}$. Besides, let $c_C(a_c)$ be the cost of executing the communication action a_c, and η_R, η_H, η_P, and η_C be the weights of the cost function. Using this notation, we define $J(n)$ as follows:

$$J(n) = \eta_R c(\tilde{\Gamma}_R(n)) + \eta_H c(\tilde{\Gamma}_H(n)) + \eta_p 1/\delta(\tilde{\Gamma}_R(n), \tilde{\Gamma}_H(n)) + \eta_C c(a_c) \quad (1)$$

In Alg. 1, at each planing iteration (lines 3-20), CP gets a library of motion plans Π_R from MP. In lines 7-11, a branch of the tree is created for each a_c and π_R. As explained in (1), the path-to-goal of H and R are required to compute a cost value for each branch. Γ_H is thoroughly given by T_H, as mentioned in line 9. On the other hand, since a π_R is likely a partial path, T_R is utilized in line 10 to compute a completed path-to-goal for R given π_R (see an illustrative example in the extended version [25]).

Assumption 2. The predicted trajectories Γ_H given by T_H in the discretized domain is an over-approximation of the predicted trajectories by T_H in the continuous domain.

Assumption 3. The discretized projection of $\bar{\pi}_R$ on Γ_R (π_R in discretized domain) is an over-approximation of $\bar{\pi}_R$ in the continuous domain.

Theorem 1. Let $P_{DC} = \langle \mathcal{S}, s^0, \mathcal{A}, T, \mathcal{G}, O, J \rangle$ be a deliberative communication problem and let $\Psi^* = \langle (a_c^i, \pi_R^i) \rangle_{i=1}^q$ be its solution computed by Alg. 1 using the cost function J in (1). Let Γ_R be the discretized waypoints of R in Ψ^* defined as $\Gamma_R = \langle \pi_R^i \rangle_i$, and Γ_H be a corresponding discretized waypoint sequence of a trajectory for H predicted by T_H and starting at s^0 with the goal G_H. If Assn. 1-3 hold, Γ_R will either lie within $\bar{\mathcal{S}}_R^{safe}$ or it will satisfy $d_{min}(\Gamma_R, \Gamma_H) > \sigma^{safe}$.

Proof. Readers are referred to the extended version of this paper [25] for the proof. □

IV. Empirical Evaluation

We conducted extensive experiments in various simulation environments to evaluate the proposed method. These experiments 1) draw a comparison between the proposed method and the baseline method CBF-TB-RRT, and 2) illustrate the performance of the proposed method in deadlock situations.

Algorithm 1: Communication Planner

Input: $\mathcal{P}_{\mathcal{DC}}$
Output: Ψ

```
1  initialize: b_0 and S^0
2  while GOAL_TEST(G_R, S_R) == FALSE do
3      Π_R ← get the plans from the MP
4      MIN_COST ← ∞
5      for π_R ∈ Π_R do
6          for a_c ∈ A_c do
7              ω_{k+1} ← O(a_c, S^k)
8              b_{k+1} ← UPDATE(b_k, ω_{k+1})
9              Γ̃_H ← T_H(S_H^k, G_H, b_{k+1})
10             Γ̃_R ← T_R(S_R, π_R)
11             c_branch ← J(Γ̃_R, Γ̃_H, a_c)
12             if c_branch < MIN_COST then
13                 MIN_COST ← c_branch
14                 BEST_ACTION ← ⟨π_R, a_c⟩
15             end
16         end
17     end
18     EXECUTE(BEST_ACTION)
19     S^k ← S^{k+1}
20     Ψ.APPEND(BEST_ACTION)
21 end
```

A. Implementation

1) CBF-TB-RRT Design: In our implementation, we consider the nonholonomic unicycle model [23] for R dynamics where states are $\mathbf{s}_r = [x_r, y_r, \theta_r]^T \in \mathcal{S}_R \subseteq \mathbb{R}^2 \times [-\pi, \pi)$ and control inputs are $\mathbf{a}_r = [v_r, \omega_r]^T \in \mathcal{A}_R \subseteq \mathbb{R}^2$. The parameters x_r, y_r, θ_r denote the longitudinal and lateral positions of R and heading angle, respectively. The controls v_r and ω_r also represent the linear and angular velocities of R, respectively. While expanding the RRT tree, the cost $c_i = w_d^G c_d^G + w_d^H c_d^H + w_g c_g + w_t c_t$ is assigned to each vertex $\nu_i \in \mathcal{V}$ for $i = 0, 1, \cdots, |\mathcal{V}|$. Here, c_d^G is the Euclidean distance between vertex i and the goal point, c_d^H is the Euclidean distance between vertex i and H, c_g is the difference between the heading of sampled vertex and the heading toward goal, and c_t is the number of waypoints that lie within the occupied regions in a straight line that connects the sampled vertex to the goal point. w_d^G, w_d^H, w_g, and w_t are weight terms. Readers are referred to [23] for further details on CBF-TB-RRT tree expansion.

2) Human Movement Model: We assumed H's movement is described by a deterministic kinematic motion transition function (T_H) and we used the Dynamic Window Approach (DWA) in MP, proposed in [26], to predict H's shortest trajectory to the goal for a finite time horizon. Since DWA is a deterministic prediction method, we assumed an ε bound around H's predicted trajectory following Assn. 1

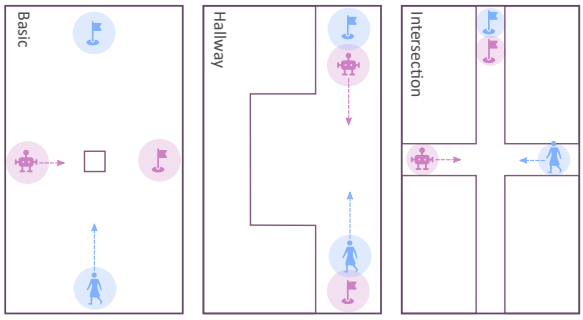

Fig. 3: Schematic illustration of diversified test environments that capture various conflicting situation.

to derive the CBF safety constraints. Given H's predicted trajectory $\mathbf{s}_h$, we define the safe set $\mathcal{S}_R^{safe} \subseteq \mathcal{S}_R$ as $\mathcal{S}_R^{safe} = \{\mathbf{s}_r \in \mathcal{S}_R, \mathbf{s}_h \in \mathcal{S}_H \mid B(\mathbf{s}_r, \mathbf{s}_h) \geq 0\}$, where $B(\mathbf{s}_r)$ is a continuously differentiable safety measure defined as

$$B(\mathbf{s}_r, \mathbf{s}_h) = \|[x_r, y_r]^T - \mathbf{s}_h\|_2^2 - (\varepsilon + r_h + r_r)^2, \quad (2)$$

r_h, and r_r are the radii of H and R, respectively. The safety measure $B(\mathbf{s}_r)$ is employed as a CBF to impose the safety constraints on the control input $\mathbf{a}_r$ in a Quadratic Program (QP) to generate safe plans π_R [23].

As illustrated in Sec. III-A, CP also utilizes T_H to predict a trajectory-to-goal for H for each branch of the search tree. Besides, in contrast to the requirements of the motion planning module, H movement prediction must be provided for the whole horizon in CP. Therefore, for the sake of computational efficiency, CP utilizes another H movement model rather than DWA. CP considers a grid-based abstraction of the environment and utilizes A* search algorithm to predict a path-to-goal for H.

Assumption 4. Predictions drawn from A* and DWA approaches complied with the Assn. 2 in all our experiments.

3) Human Motion Execution Model: We utilized the Social Forces model [27] to simulate H's movement, as it is very fast, scalable, and yet describes observed pedestrian behaviors realistically. We modeled H and R both as pedestrians. To mimic H's reactivity to R's communication action a_c, the model creates multiple virtual agents moving from the R's current position to all x-y projections of discretized zones $l_i \in l_H$ in the CP's belief model for which $b_i = 1$. If $\mathbf{b}_k = \varnothing$, R's goal is computed as a linear projection from its current position based on its current velocity, i.e. H makes no assumptions over R's future trajectory. Thus, in our experiments, the models used by H are different from the model H used by R, which is likely in real-world setting.

B. Experimental Setup

Test environments: Fig. 3 the environments used in our experiments. The basic floor map exemplifies spacious environments, while the hallway and intersection floor maps model more restricted and confined environments.

Measurements: Aside from cost-to-goal of R and H, there are four more quantitative measures to evaluate the performance and effectiveness of the proposed method. 1) R's normalized speed (RNS): $RNS = {}^{c_R^*}/_{time_R^{actual}}$ measures R's normalized mean speed from $\mathbf{s}_r^0$ to $\mathcal{G}_R$, where c_R^* and $time_R^{actual}$ denote the optimal cost-to-goal of R and the R's actual travel time respectively. 2) H's normalized speed (HNS): $HNS = {}^{c_H^*}/_{time_H^{actual}}$ measures H's normalized average speed from $\mathbf{s}_h^0$ to $\mathcal{G}_H$, where c_H^* and $time_H^{actual}$ denote the optimal cost-to-goal of H and H's actual travel time respectively. 3) Planning iterations (PI): PI denotes the number of iterations of lines 2 to 17 in Alg. 1. And 4) Proximity cost (PC): PC measures the closeness of R and H during an experiments. Let $\Gamma_R = \{\gamma_R^i\}_{i=1}^{i_{max}}$ be R's discretized trajectories given by a solution Ψ and $\Gamma_H = \{\gamma_H^i\}_{i=1}^{i_{max}}$ be the corresponding discretized waypoint sequence of an actual trajectory for H. We defined PC using (2) as follows.

$$Z = \{\zeta_i \mid \zeta_i = B(\gamma_R^i, \gamma_H^i) < thresh\}_{i=1}^{i_{max}} \quad (3)$$

$$PC = \begin{cases} \infty & \text{if } \exists \zeta_i \in Z,\ \zeta_i < 0 \\ {}^1/_{\sum_{i=1}^{i_{max}} \zeta_i} & \text{otherwise} \end{cases}, \quad (4)$$

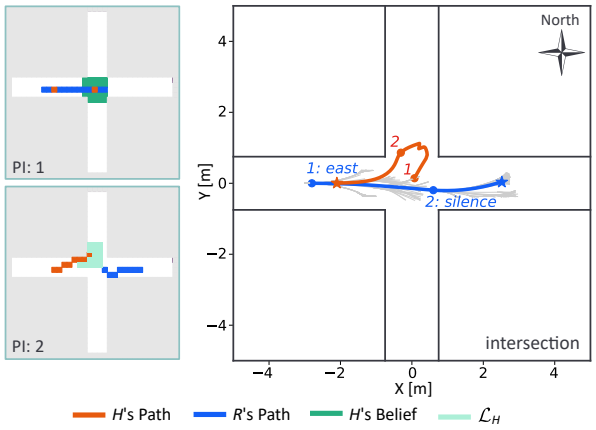

Fig. 4: An example of a potential deadlock.

Hypotheses: We evaluate the following hypotheses 1) In confined environments, the chances of a deadlock are higher without communication. Therefore, the effect of communication to avoid such deadlocks is more effective. 2) The proposed approach not only results in less conflicting social navigation, but also prevents deadlock situations where non-communicative approaches fail to find a solution. 3) By adjusting the weight vector of the cost function J, H or R can be prioritized. Accordingly, the non-prioritized agent is expected to have a decreased normalized average speed due to an increased cost-to-goal.

C. Results

1) Comparison with CBF-TB-RRT: This section aims to demonstrate that the proposed method performs as optimally as CBF-TB-RRT, in terms of the traveled distances, while it reduces the conflict between H and R. In Table I, the results are presented as the range of 10 experiments the experiments for each test environments of Fig. 3, where $\eta_R = 1.5, \eta_H = 0.25, \eta_P = 3$, $\eta_C = 1$, and $A_c = \{north, south, east, west\}$.

TABLE I: Comparison with CBF-TB-RRT.

Measures / Maps	Our approach				CBF-TB-RRT			
	R cost-to-goal	H cost-to-goal	PI	PC	R cost-to-goal	H cost-to-goal	PI	PC
Basic	5.65–5.68	7.33–7.52	2–2	0.50–0.53	5.51–6.27	6.90–6.99	46–113	0.21–0.57
Intersection	3.63–3.88	6.10–6.29	2–2	0.22–0.24	4.20–4.24	5.76–5.90	51–98	0.34–∞
Hallway	10.12–10.58	6.85–7.39	4–4	0.72–0.89	10.27–10.30	6.65–6.77	121–123	∞ –∞

The results show the range of the measurements in 10 trials per map; PI: planning iterations; PC: proximity cost.

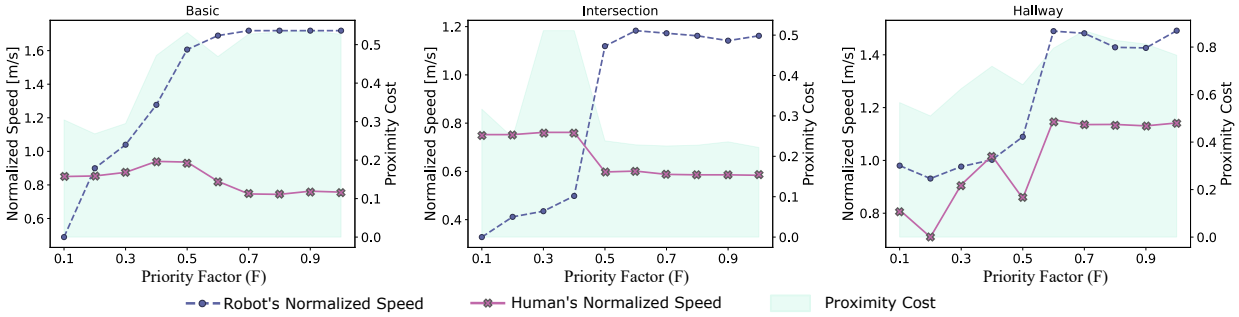

Fig. 5: Flexible prioritization of the H and the R in different test environments, where $F = 1$ prioritizes R.

Our results show that PC of the baseline drastically increases in more confined environments. E.g., PC has a finite range in the basic environment since the room is spacious, while the PC range is infinity in the intersection environment where the floor map is confined and only one agent can pass through a corridor at a time. The situation is even more severe in the hallway environment in which the baseline method results in an infinite PC for all 10 experiments. These observations validate Hypothesis 1. In contrast, the proposed method handles conflicting situations of the intersection and hallway environments effectively. The PC values of our method in all environments are dramatically lower compared to the baseline method, while cost-to-goal of R and H do not increase noticeably. Moreover, employing the proposed method eliminates the necessity for frequent re-planning as PI drops significantly compared to the experiments with the baseline method.

2) Handling potential deadlocks: According to IV-C.1, the proposed method is significantly more effective in reducing PC in confined environments while maintaining the efficiency. This property is particularly imperative in preventing potential deadlocks in narrow passages, where a lower PC implies less conflicting path for H and R. Fig. 4 demonstrates a pervasive case where lack of communication leads to a freezing situation. In this example, at the first planning iteration, the R transmits an "east" signal, automatically selected by CP, to H by which H is informed about R's plan before she enters the narrow corridor. As shown in Fig. 4 (top left), this communication signal updates H's belief about R's next location adequately and impels H to clear the passage. At the second planning iteration, R has already passed the intersection, so it remains silent and H's belief indicates no collisions, as depicted in Fig. 4 (bottom left).

In the same scenario, the baseline method performs ineffectively since H enters the left corridor before R departs it. When H gets closer to R, there won't be enough room for the RRT to be expanded and a deadlock happens since the passage will be blocked for R permanently. This analysis supports Hypothesis 2 regarding the capability of the proposed method to handle potential deadlocks.

3) Flexible prioritization: H or R can be prioritized flexibly by adjusting the weights of J. A parameter study on η_R and η_H reveals the way that each agent is favored in different social navigation scenarios, as shown in Fig. 5. In these experiments, the weights are adjusted as $\eta_R = F\eta_{const}$, and $\eta_H = (1 - F)\eta_{const}$, where $F \in [0, 1]$ denotes the priority factor (R is fully prioritized for $F = 1$), and $\eta_{const} = 1.5$. In Basic and Intersection environments, prioritizing an agent, increases the agent's normalized speed significantly. However, in the Hallway environment, the whole H-R interaction is relatively smoother and less conflicting when R has a higher priority leading to a higher normalized speed for H as well. Together, the present findings confirm Hypothesis 3. Furthermore, the results support the fact that the proposed method maintains a reasonably low PC in all test environments no matter which agent is prioritized. In other words, the proposed method can be used to identify appropriate priorities for smooth social navigation.

V. Conclusion

This paper proposes a joint communication and motion planning framework that selects from an arbitrary input set of communication signals while computing the robot motion plans. The simulation results demonstrated that the presented framework avoids potential deadlocks in confined environments by leveraging explicit communications coupled with robot motion plans. We found that producing less conflicting trajectories for the robot in confined environments, which led to drastically lower proximity costs, indicates lower chances of a deadlock. We also observed that the proposed method does not degrade the robot's efficiency (in terms of traveled distances) compared to CBF-TB-RRT. In contrast, the non-communicative baseline method resulted in high proximity cost overall, which shows its incapability of generating viable solutions when extensive human-robot interaction is required.

REFERENCES

[1] J. Cheng, H. Cheng, M. Q.-H. Meng, and H. Zhang, "Autonomous navigation by mobile robots in human environments: A survey," in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1981–1986, IEEE, 2018.

[2] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1343–1350, IEEE, 2017.

[3] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation.," in *Robotics: science and systems*, 2012.

[4] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: the case for cooperation," in *2013 IEEE international conference on robotics and automation*, pp. 2153–2160, IEEE, 2013.

[5] L. P. Kaelbling and T. Lozano-Pérez, "Hierarchical planning in the now," in *Workshops at the Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[6] C. R. Garrett, T. Lozano-Pérez, and L. P. Kaelbling, "Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning," in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 30, pp. 440–448, 2020.

[7] S. Srivastava, E. Fang, L. Riano, R. Chitnis, S. Russell, and P. Abbeel, "Combined task and motion planning through an extensible planner-independent interface layer," in *2014 IEEE international conference on robotics and automation (ICRA)*, pp. 639–646, IEEE, 2014.

[8] N. Shah, D. K. Vasudevan, K. Kumar, P. Kamojjhala, and S. Srivastava, "Anytime integrated task and motion policies for stochastic environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9285–9291, IEEE, 2020.

[9] N. T. Dantam, Z. K. Kingston, S. Chaudhuri, and L. E. Kavraki, "An incremental constraint-based framework for task and motion planning," *The International Journal of Robotics Research*, vol. 37, no. 10, pp. 1134–1151, 2018.

[10] N. T. Dantam, S. Chaudhuri, and L. E. Kavraki, "The task-motion kit: An open source, general-purpose task and motion-planning framework," *IEEE Robotics & Automation Magazine*, vol. 25, no. 3, pp. 61–70, 2018.

[11] I. A. Sucan, M. Moll, and L. E. Kavraki, "The open motion planning library," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, 2012.

[12] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 2, pp. 995–1001, IEEE, 2000.

[13] H. Nishimura, B. Ivanovic, A. Gaidon, M. Pavone, and M. Schwager, "Risk-sensitive sequential action control with multi-modal human trajectory forecasting for safe crowd-robot interaction," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11205–11212, IEEE, 2020.

[14] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 335–356, 2015.

[15] A. Kulkarni, S. Srivastava, and S. Kambhampati, "A unified framework for planning in adversarial and cooperative environments," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 2479–2487, 2019.

[16] Y. Zhang, D. A. Shell, and J. M. O'Kane, "Finding plans subject to stipulations on what information they divulge," in *International Workshop on the Algorithmic Foundations of Robotics*, pp. 106–124, Springer, 2018.

[17] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *The International Journal of Robotics Research*, vol. 35, no. 11, pp. 1289–1307, 2016.

[18] H. Kivrak, F. Cakmak, H. Kose, and S. Yavuz, "Social navigation framework for assistive robots in human inhabited unknown environments," *Engineering Science and Technology, an International Journal*, vol. 24, no. 2, pp. 284–298, 2021.

[19] R. A. Knepper, C. I. Mavrogiannis, J. Proft, and C. Liang, "Implicit communication in a joint action," in *Proceedings of the 2017 acm/ieee international conference on human-robot interaction*, pp. 283–292, 2017.

[20] S. Habibian, A. Jonnavittula, and D. P. Losey, "Here's what i've learned: Asking questions that reveal reward learning," *arXiv preprint arXiv:2107.01995*, 2021.

[21] K. Baraka and M. M. Veloso, "Mobile service robot state revealing through expressive lights: formalism, design, and evaluation," *International Journal of Social Robotics*, vol. 10, no. 1, pp. 65–92, 2018.

[22] Y. Che, A. M. Okamura, and D. Sadigh, "Efficient and trustworthy social navigation via explicit and implicit robot–human communication," *IEEE Transactions on Robotics*, vol. 36, no. 3, pp. 692–707, 2020.

[23] K. Majd, S. Yaghoubi, T. Yamaguchi, B. Hoxha, D. Prokhorov, and G. Fainekos, "Safe navigation in human occupied environments using sampling and control barrier functions," *arXiv preprint arXiv:2105.01204*, 2021.

[24] A. Sintov and A. Shapiro, "Time-based rrt algorithm for rendezvous planning of two dynamic systems," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6745–6750, IEEE, 2014.

[25] M. Dadvar, K. Majd, E. Oikonomou, G. Fainekos, and S. Srivastava, "Joint communication and motion planning for cobots," *arXiv preprint arXiv:2109.14004*, 2021.

[26] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[27] D. Helbing and P. Molnar, "Social Force Model for Pedestrian Dynamics," *Physical Review E*, vol. 51, pp. 4282–4286, May 1995.