DECAF: a Discrete-Event based Collaborative Human-Robot Framework for Furniture Assembly

Giacomuzzo, Giulio; Terreran, Matteo; Jain, Siddarth; Romeres, Diego

TR2024-138   October 04, 2024

Abstract

This paper proposes a task planning framework for collaborative Human-Robot scenarios, specifically focused on assembling complex systems such as furniture. The human is characterized as an uncontrollable agent, implying for example that the agent is not bound by a pre-established sequence of actions and instead acts according to its own preferences. Meanwhile, the task planner computes reactively the optimal actions for the collaborative robot to efficiently complete the entire assembly task in the least time possible. We formalize the problem as a Discrete Event Markov Decision Problem (DE-MDP), a comprehensive framework that incorporates a variety of asynchronous behaviors, human change of mind, and failure recovery as stochastic events. Although the problem could theoretically be addressed by constructing a graph of all possible actions, such an approach would be constrained by computational limitations. The proposed formulation offers an alternative solution utilizing Reinforcement Learning to derive an optimal policy for the robot. Experiments were conducted both in simulation and on a real system with human subjects assembling a chair in collaboration with a 7-DoF manipulator.

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2024

DECAF: a Discrete-Event based Collaborative Human-Robot Framework for Furniture Assembly

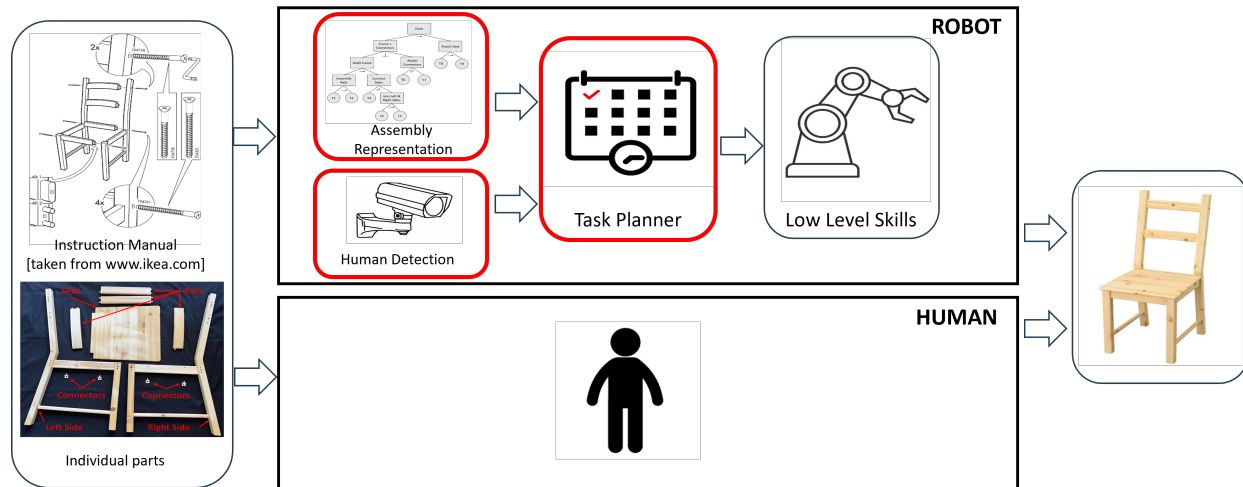Giulio Giacomuzzo[1], Matteo Terreran[1], Siddarth Jain[2], Diego Romeres[2]

Fig. 1: Schematic of a collaborative assembly workflow. The red highlighted components are the focus of this work.

Abstract— This paper proposes a task planning framework for collaborative Human-Robot scenarios, specifically focused on assembling complex systems such as furniture. The human is characterized as an uncontrollable agent, implying for example that the agent is not bound by a pre-established sequence of actions and instead acts according to its own preferences. Meanwhile, the task planner computes reactively the optimal actions for the collaborative robot to efficiently complete the entire assembly task in the least time possible.

We formalize the problem as a Discrete Event Markov Decision Problem (DE-MDP), a comprehensive framework that incorporates a variety of asynchronous behaviors, human change of mind, and failure recovery as stochastic events. Although the problem could theoretically be addressed by constructing a graph of all possible actions, such an approach would be constrained by computational limitations. The proposed formulation offers an alternative solution utilizing Reinforcement Learning to derive an optimal policy for the robot. Experiments were conducted both in simulation and on a real system with human subjects assembling a chair in collaboration with a 7-DoF manipulator.

I. INTRODUCTION

In recent years, the area of human-robot collaboration has gained increasing attention because of the advancement of robotics and artificial intelligence. The synergistic integration of human dexterity and robotic precision holds great potential, especially for small- and medium-sized enterprises with high-mix and low-volume productions. In this context, the assembly of furniture represents a challenging yet promising

test-bed application, since it involves long-term action sequences, dexterous manipulations, precise alignments, transportation of cumbersome objects, and three-hand operations. Reactive planners for robot actions play a crucial role in promoting smooth collaboration, yet robust frameworks for long-term tasks in unpredictable environments remain challenging and warrant further investigation. For effective collaboration, robots must adapt to human behavior while maintaining coordination, although explicit coordination may slow interaction. Additionally, unexpected events like task failures or sudden human change of mind can hinder team effectiveness and safety and thus need to be considered and handled at planning time.

Most of the existing literature either proposes to assign offline the tasks to the human and robot agents, which then work separately sharing only the workspace [1], or treat the collaboration as a multi-agent scheduling task [2], [3], [4]. Despite being effective, these approaches typically require the human to follow a prescribed plan. Single-agent frameworks have been also proposed, mainly relying on a leader-follower paradigm, where the robot actions are passively determined by the human leader through a dedicated interface [5], or learned from demonstrations [6]. These methods, generally, lack robustness against uncertainties like task failures and human changes of mind, common in practical applications. Additionally, they seldom account for agents collaborating on the same sub-task.

In this paper, we present a novel single-agent framework for planning long-term collaborative assembly tasks, such as furniture assembly, called Discrete-Event based Collaborative Human-Robot Assembly Framework for Furniture

[1]Department of Information Engineering, Università di Padova, Italy {giacomuzzo, matteo.terreran}@dei.unipd.it
[2]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA. {sjain ,romeres}@merl.com

(DECAF). In the proposed framework the robot agent observes the human and performs both joint and independent actions maximizing the task efficiency while reacting to unpredictable events. In detail, we rely on a Hierarchical Task Model (HTM) to encode the interdependence between the assembly sub-tasks, and on a Bayesian Inference module to observe the human agent behavior. The collaborative assembly is then modeled as a Discrete-Event Markov Decision Process (DE-MDP), which allows one to easily handle actions of different durations, as well as uncertain contingencies such as failures and changes of mind. To solve the DE-MDP and obtain a planning policy, we propose a Reinforcement Learning (RL) method.

Our contribution can be summarized as follows: i) We propose a novel DE-MDP formulation to model long-term collaborative assembly tasks, which accounts for asynchronous joint and independent actions with uncertain duration, and unpredictable events such as task failures and changes of mind. ii) We present DECAF, a complete planning framework that comprises the DE-MDP model, an HTM description of the assembly, and a Bayesian Inference module to discover human intention. iii) We assess our framework in both simulated and real-world environments. DECAF is validated through a real chair assembly task involving 10 human subjects. Results show decreased completion times compared to a human solving the task alone, along with reduced physical effort and an enhanced assembly experience for the human operator.

II. RELATED WORKS

There exists extensive literature on planning, scheduling, and task assignment for human-robot assembly tasks. Several approaches model the assembly task using AND/OR graphs and solve the planning as a search problem [7], [3], [4], [8]. Similarly, Behavior Trees have been also considered as task representation [9], [1], [10]. Despite they proved to be effective, all these works require the human to follow the specific plan prescribed by the scheduler. Moreover, all these methods are not robust to uncertain and unpredictable events.

Robustness to task failures is considered in [2], where the task is represented employing an HTM, and the cooperation is modeled as a multi-agent concurrent MDP. The MDP is solved by building a decision graph that also considers sub-task failure probability and the optimal path is selected defining a Mixed Integer Linear programming Optimization. However, this framework requires the human to adhere to pre-computed plans and does not consider joint actions. Additionally, integrating contingencies different from task failures proves challenging, as they should be manually added to the graph construction.

Differently, in [11], a robust planning algorithm based on a Partially Observable MDP is developed. The focus is on deriving a realistic human policy by considering the interplay between both agents' decision-making processes, which directly influences human intention. Their solution enables both agents to execute individual and joint actions, but overlooks actions of varying duration and uncertainty

robustness to uncertainty. In [12], an open-agent MDP is introduced to minimize the human's time in the task, but human has to follow an optimal policy. More similarly to our work, single-agent formulations have been explored in [13], [14] and [15]. In [13] a Robust Plan Recognition and Trajectory Prediction is proposed based on an HTM representation, where the human's intentions and plan are recognized using a perception module. The robot recognizes the human plan and performs its actions accordingly, but its role is limited to assistive operations, without the possibility to actively contribute to the task advancement. In [14] the same authors propose a collaborative framework where the HTM task scheduling is optimized to minimize the total task completion time and promote workspace separation. However, within this framework, the possibility to collaborate on the same joint task is not considered and the task is optimized only over a limited time horizon. Similarly in [15], the authors rely on a modified HTM task representation and their goal is to provide an adaptive framework where the overall assembly time is minimized. The robot observes the human's choice and selects its action to minimize the overall cost, averaged across all possible human actions. However, their graph-based approach is inadequate for handling failures and stochastic events.

III. PROBLEM FORMULATION

In this section, we formalize the long-term collaborative assembly planning problem we aim to solve, and we highlight the requirements and assumptions of our framework. Assembly tasks are defined as a hierarchical set of atomic tasks, hereafter denoted as actions, which must be executed complying with a set of ordering constraints. We assume that two agents, a human and a robot, cooperate to solve the task. Furthermore, we assume actions to be either individual or joint. Individual actions can be executed by a single agent, be it the human or the robot, while joint actions necessitate simultaneous execution by both agents. The framework is robust to actions of different durations, contingent upon the specific action and the agent undertaking it.

Despite the aforementioned scenario includes two agents, we assume the human to be inherently uncontrollable. The robot observes the human actions and plan its actions based on the state of the environment which comprises the assembly state and behavior of the human partner. Consequently, the problem is formulated as a single-agent planning task.

The setting depicted above introduces interesting challenges, which we aim to address within our framework.
Synchronization: The human and the robot can work in parallel on different actions but have also to perform joint actions together. Since actions have different duration, agent synchronization is required to perform joint actions.
Human detection: As the human behaves uncontrolled and the robot has to adapt its actions to the human choices, human action detection is required. This inevitably introduces a delay into the robot action scheduling, which needs to be considered by the planner.
Human change of mind: The human could begin a task,

and then suddenly change its mind and perform a new task. In that case, the robot should reschedule its plan.

Failures: The actions performed by the agents could fail, which requires the introduction of recovery strategies.

To address the aforementioned challenges, we consider the following assumptions. i) Joint actions can be chosen and initiated only by the human operator. In case the human chooses a joint action, the robot joins them as soon as it finishes the ongoing action. ii) We introduce an additional action called *idle*, which can be performed by both agents. When *idle*, the agent is waiting, without contributing to the task advancement. We use the *idle* as a synchronization mechanism before a joint action: when the human chooses a joint action, they remain idle until the robot has completed its action. When the robot finishes, the two agents perform the joint action together. iii) If the human action is not detected, the robot must remain *idle*. v) Failures are detected at the end of an action, including those that fail before reaching the goal or due to incorrect execution. We do not consider failures that occur after an action is completed.

IV. DECAF FRAMEWORK

A. Hierarchical Task Model for Assembly task description

The DECAF framework relies on sequential/parallel Hierarchical Task Model (HTM) for task representation [13]. A HTM is a tree structure where the root node represents the entire assembly, while all the other nodes represent assembly subtasks. The leaves represent the actions to be executed. Each node can be categorized as parallel ($\parallel$), sequential ($\rightarrow$) or independent ($\perp$). Children of parallel nodes can be executed at the same time by the two agents, in any order. Children of sequential nodes must be individually executed in the specified order (from left to right). Finally, children of independent nodes must be individually executed but can be performed in any order. An example of HTM is in Fig. 3.

In our framework, the HTM is used to describe the sequential constraints on action executions, which are encoded in the DE-MDP described in Section IV-C.

B. Bayesian Inference for Human Intent Recognition

To recognize human actions, DECAF relies on a Bayesian Inference module. We assume each action to be related to a specific object in the assembly scene. Then, we use Bayesian Inference to detect the object of interest for the human, which in the following will be denoted as *goal*. The robot, during the assembly, task aims to infer the most likely goal g^* for the human from the set of possible goals g, given a set of observations. Following [16], [17], we formulate the intent inference problem as Bayesian filtering in a Markov model, which allows us to model the uncertainty over the candidate goals of the human agent as a probability distribution over the goals. We integrated human hand pose tracking through a deep learning model [18] finely-tuned within the experimental setup. We represent the goal g_t as the query variable and the observed features $\Theta_0,...,\Theta_t$ as the evidence variables, where Θ_t is a k-dimensional vector of k observations θ_t^i, $i = 1 : k$, at time t. The uncertainty over goals

is then represented as the probability of each goal hypothesis. In particular, we compute proximity and alignment to a goal as likelihood features, given the human hand pose and goal poses. We assume the conditional independence of observations, and thus, at time t, the belief $b_t(g)$ becomes,

$$b_t = P(g_t \mid \Theta_{0:t}) \propto \prod_{\theta_t \in \Theta_t} P(\theta_t \mid g_t) \sum_{g_{t-1} \in g} P(g_t \mid g_{t-1}) b_{t-1}(g_{t-1})$$

The posterior distribution at time t, denoted b_t, represents the belief after taking the observations into account. The set of prior probabilities $P(g_{t=0})$, $\forall g \in g$, initially represents the robot's belief over the goals. The beliefs then are continuously updated, by computing the posteriors, as more observations become available. Furthermore, $P(g_t \mid g_{t-1})$ is the conditional transition distribution of changing to goal g_t at time t given that the goal was g_{t-1} at time $t-1$. Finally, to predict the most likely goal $g_t^* \in g$, we select the goal class that is most probable according to the maximum *a posteriori* decision $g_t^* = \arg\max_{g_t \in g} P(g_t \mid \Theta_t)$.

C. Discrete-Event MDP

In this section, we describe the proposed method to model and solve the collaborative assembly scheduling. Our method models the system as a Discrete-Event Markov Decision Process (DE-MDP). A DE-MDP is an MDP in which the state transitions do not happen on a fixed time basis, but are determined by the occurrence of relevant events. In the case of collaborative assembly, the task advancement is related to the completion of actions that have different, possibly stochastic, durations. Moreover, the cooperation state is also modified by events such as human action recognition or human change of mind.

Consider an assembly task composed of N actions. The collaborative scheduling problem is modeled as a DE-MDP $(S, A, \mathcal{E}, \Gamma, \ell, T, R, \gamma)$, where S is the state space of the system; A is the action space; $\mathcal{E}$ represents the set of possible events; $\Gamma(s)$ is the set of feasible events at state $s \in S$; $\ell(s, a, e)$ is the event lifespan function, which gives the probability distribution over the time after which the event e is likely to occur at the current state $s \in S$ and robot action $a \in A$; $T(s, a, s', e)$ is the transition probability function which gives the probability of transitioning to state s' given the current state s, the current robot action a and the event e; $R(s, a, e)$ is the reward function and γ the discount factor.

Differing from the standard MDP framework where the state update occurs at every time step, the DE-MDP state transition happens only in the presence of an event. We use the index variable k to represent the time instant of the state evolution. Moreover, Δt^k denotes the time spent by the system to transition from state s^k to s^{k+1} and, for convenience, it is multiple of a discrete-time step $\bar{T}$. Therefore, the variable time step, Δt^k, at which the DE-MDP state updates depends on the event lifespan $\ell(s^k, a^k, e)$, with e being the event causing the system transition from s^k to s^{k+1}. In the following, for notation simplicity, we will explicit the dependence on the index k only when necessary.

In the following the elements of the DE-MDP are detailed.

a) Actions: the actions set A includes all the actions that can be executed by the robot. Let $\bar{A} = \{a_1, \ldots, a_N\}$ be the set of all possible actions to complete the task. Let us also introduce the set $A_r = \{a_{N+1}, \ldots, a_{2N}\}$ be the set of recovery actions. These actions are not necessary to complete the task but must be executed to recover from a failure. Finally, let us denote with a_{idle} the *idle* action. We model each action as a tuple $a_j = (o_j, \delta_j, p_j^f)$, where $o_j \in \{0, 1, 2, 3\}$ denotes if the action can be performed by the human (0), by the robot (1), by both the agents (2) or is a joint action (3); $\delta_j = [\delta_j^h, \delta_j^r]$ represents the action duration when performed by the human (δ_j^h) or by the robot (δ_j^r). Regarding the action duration modeling, we assume δ_j^h and δ_j^h to be normally distributed around a nominal action duration value; $p_j^f \in [0, 1]$ denotes the action failure probability. Note that in the case of joint actions, δ_j^h and δ_j^r are equal, while we assume the *idle* action to last until the next triggering event happens. With this formalism, we define the action space as $A = \{a_j \in \bar{A} | o_j \neq 0\} \cup \{a_j \in A_r | o_j \neq 0\} \cup \{a_{idle}\}$, which has cardinality N_r. For future convenience, let us define also the set of human actions $A_h = \{a_j \in \bar{A} | o_j \neq 1\} \cup \{a_j \in A_r | o_j \neq 1 \cup \{a_{idle}\}$, which has cardinality N_h.

Feasible Actions: A_f is the set of all feasible actions for the two agents at the current state. The nonfeasible actions are actions already completed, actions for which the requirements are still not satisfied, recovery actions for which the corresponding action is not failed, the current action being executed by the other agent, and the *idle* action for the robot if the human is already *idle*, to avoid infinite cycles.

b) States: the system state is $s = (s_a, a^h, t^h, t^r, d)$ where $s_a \in \{-1, 0, 1\}^N$ is the actions execution indicator, the i-th component representing the i-th action being completed (1), not attempted yet (0) or failed (-1); $a^h \in A_h \cup \{unknown\}$ is the current human action, which is part of the state because the agent cannot be controlled. Note that the human action is considered *unknown* only in case it has not been detected yet; $t^h \in \mathbb{N}$ is the time elapsed since the current human action started; $t^r \in \mathbb{N}$ is the time since the current robot action started; $d \in \{0, 1\}$ indicates if the current human action has been detected or not.

c) Events: the set $\mathscr{E} = \{H, R, D, C\}$ contains the events triggering the state transitions, where H represents the end of a human action, R represents the end of a robot action, D represents the human action detection and C represents a human change of mind. As a modeling choice, we assume the probability of an event happening $p(e|s, a)$ can be fully determined by the current state $s \in S$ and robot action $a \in A$. In particular, if $e = C$, we denote with $p_c \in [0, 1]$ the probability of a human change of mind to happen. Then, $p(C|s, a)$ represents the probability of having a change of mind before the next H or R event. The probability of having a change of mind before the next H event is p_c, as, if the change of mind happens, it will for sure happen before the end of the current human action. The probability of having a change of mind before the next R event, instead, depends on the time at which the next C and R will happen. Let Δ_C

and Δ_R be the time after which the next C and R events will happen, respectively. Note that Δ_C and Δ_R are random variables distributed as $\ell(s, a, C)$ and $\ell(s, a, R)$, respectively. Then, C will happen before R with probability $p_c p_{cr}$, with p_{cr} being the probability $p(\Delta_C < \Delta_R)$. See the paragraph regarding the lifespan function for the description of how the lifespan distributions are defined. Note that $p_{cr} = 1$ if $\Delta_R > \Delta_H$. We can in general conclude that $p(C|s, a) = p_c p_{cr}$. Then, if $e = H$, the probability $p(H|s, a)$ is 0 if the robot finishes its action before the human i.e., $\delta_a^r - t^r < \delta_{a^h}^h - t^h$, while it is $1 - p(C|s, a)$ if the human finishes before the robot. Analogous reasoning applies for the event $e = R$. Finally, if $e = D$, $p(D|s, a) = 1$ if $d = 0$, otherwise $p(D|s, a) = 0$.

d) Feasible event set: $\Gamma(s) : S \rightarrow E \subseteq \mathscr{E}$ provides the set of feasible events given the current state $s \in S$. In particular, we have: $\Gamma(s|d = 0) = \{D\}$, namely if the human action has not been detected, the only feasible next event is the detection, while if the human action has been detected all the other events are feasible, namely $\Gamma(s|d = 1) = \{H, R, C\}$.

e) Event lifespan: the event lifespan function $\ell(s, a, e)$ provides a probability distribution over the time spent by the system in the current state $s \in S$ with the current robot action $a \in A$, before transitioning to the next state due to the event $e \in \mathscr{E}$, namely $\ell(s, a, e) = p(\Delta_e | s, a)$, with Δ_e being the time after which the event e will occur given s and a. In the assembly scenario we are considering, if $e = D$ then $\Delta_D = \bar{\Delta}_D$ with probability 1, where $\bar{\Delta}_D$ represents the human action detection time, which we assume to be deterministic and known. If $e = H$, $\Delta_H = \delta_{a^h}^h - t^h$. If $e = R$, $\Delta_R = \delta_{a^r}^r - t^r$, where δ_{a^h} and δ_{a^r} represent the human's and robot's actions duration, while t^h and t^r represent the time elapsed from the human and the robot started their current action. Note that, as δ_{a_h} and δ_{a_r} are random variables, also Δ_H and Δ_R are random variables with the same distribution as δ_{a_h} and δ_{a_r}, with mean shifted by the elapsed times t^h and t^r, respectively. Finally, we model the lifespan Δ_C of the event $e = C$ as a discretized truncated exponential distribution with support spanning the time interval from detection to end of the human action.

f) Transition: The transition function $T(s, a, s', e)$ provides the probability of transitioning to the next state $s' \in S$ given the current state s and the current robot action a due to the occurrence of the event e. If $e = D$, namely the event is a human action detection, the effect on the state is the deterministic transition of d from 0 to 1 and the transition of a^h to the *unknown* state. If $e = H$, s_a will transition to $s_a' = s_a \pm 1_{a^h}$, with $+$ if the action ends successfully and $-$ if it fails, where $1_{a_j} \in \{0, 1\}^{N+1}$ is the vector containing all zeros except for the component corresponding to a_j. a^h will transit to the *unknown* state, while $d' = 0$ and $t^{h'} = 0$. If $e = R$, the transition will affect s_a and t^r, in particular $s_a' = s_a \pm 1_a$ and $t^{r'} = 0$. Finally, a change of mind $e = C$ will cause the human and consequently the robot to change action, namely $a^{h'}$ will transit to the *unknown* state, d will transit to 0, while $t^{h'} = 0$ and $t^{r'} = 0$.

g) Reward: as we aim to minimize the execution time, we model the reward as the negative transition time, namely $R(s^k, a^k, e) = -\Delta_t^k$. Different rewards could be possible,

which for example consider human ergonomics or previous action correlation.

D. DE-MDP Solution

The DE-MDP can be solved by RL[1] that optimizes a robot policy $\pi_\theta(\cdot)$ mapping the current state s to the next robot action a. In particular, in the presence of predictable stochastic events, RL can learn the stochastic dynamics and take actions that consider also the possible future evolution. Moreover, parametric function approximators can be exploited in the presence of high dimensional state-action spaces to reduce the computational and storage burden.

V. SIMULATED EXPERIMENTS

In this section, we assess the advantages of DECAF in simulated collaborative assembly scenarios. Both toy examples and a real chair assembly are considered.

In all the following experiments, the RL policy is trained with the Proximal Policy Optimization (PPO) algorithm [19]. To this aim, we implemented the DE-MDP described in Section IV-C as a custom Gym environment [20] and used the Stable Baselines library [21] for policy training. The training process is sped up using action masking to limit the action exploration only to A_f, as explained in [22].

Tasks: We designed 4 toy assembly examples with number of actions varying from $n = 8$, to $n = 32$. For each experiment, we generated a random *HTM*, composed of $\frac{n}{4}$ joint actions and $\frac{n}{2}$ only robot actions. The remaining actions can be performed by either the human and the robot. The nominal action duration is randomly generated between $4\bar{T}$ and $16\bar{T}$. Additionally, we considered the simplified but realistic example of the IKEA Ivar chair assembly, whose HTM is reported in Fig 3.

Baselines: DECAF's performance is benchmarked against that of two baselines policies: the first is a greedy policy inspired to [1], which selects, among the feasible actions, the one with the lowest nominal duration. The second, instead, is a random policy which randomly selects the next robot action from the feasible actions set with uniform probability.

A. Stochastic setting

In the stochastic setting, we introduce human change of mind and action failures.

Tab. I shows the execution time at the increase of the Human Change of Mind probability p_c, with p_c ranging from 0.1 to 0.4. For each value of p_c, we trained a PPO policy and tested it with a Monte Carlo experiment composed of 1000 simulations. The RL policy demonstrates its effectiveness in addressing changes of mind by computing viable solutions in all cases. As expected, at the increase of the change of mind probability, the average completion time increases.

Similarly, Tab. II shows the execution time at the increase of the action failure probability. For simplicity, we considered each action to have the same probability of failing, $p_f \in [0.1, 0.4]$. The occurrence of a failure introduces

---

In the extended version of the paper we consider also a solution based on a decision graph as a baseline TODO:Arxivlink

p_c	8 Actions	16 Actions	24 Actions	32 Actions	Chair
0.1	100.2 [3.2]	177.7 [4.4]	277.9 [5.8]	344.8 [6.2]	135.0 [6.0]
0.2	101.5 [4.8]	180.2 [6.0]	279.7 [7.5]	363.1 [6.2]	139.9 [9.5]
0.3	102.7 [6.0]	183.2 [7.6]	281.5 [11.1]	359.5 [15.2]	144.1 [12.9]
0.4	103.9 [6.8]	187.8 [10]	286.5 [15.1]	380.6 [17.5]	151.0 [16.8]

TABLE I: Completion time at the increase of change of mind probability. The table reports mean and standard deviation (between square brackets) in terms of number of time steps $\bar{T}$, obtained after 1000 trials with the DECAF RL policy.

in the feasible action set a recovery action, which needs to be performed to complete the previously failed action. The DECAF framework with RL policy leads to a feasible assembly plan also with failures and recovery actions. As expected, also in this case the overall average assembly time is increased at the increase of the failure probability.

Failure p	8 Actions	16 Actions	24 Actions	32 Actions	Chair
0.1	111.3 [15.0]	196.8 [17.7]	327.6 [22.6]	511.2 [26.6]	149.4 [13.6]
0.2	125.4 [19.1]	222.7 [24.9]	400.5 [30.1]	556.7 [34.2]	174.0 [17.9]
0.3	137.6 [21.6]	295.7 [28.5]	453.1 [33.6]	602.9 [37.9]	184.2 [21.4]
0.4	153.0 [23.2]	400.5 [30.1]	485.5 [36.6]	649.7 [41.7]	209.6 [21.1]

TABLE II: Completion time at the increase of action failure probability. The table reports mean and standard deviation (between square brackets) in terms of the number of time steps $\bar{T}$, obtained with 1000 trials and the DECAF RL policy.

VI. REAL USER STUDY

DECAF efficacy is validated in a real-world experiment involving human subjects. The experiment consisted of the human participants assembling an IKEA Ivar chair together with a 7 Degrees of freedom Franka Emika Panda robotic arm. As a baseline, participants undertook the assembly task individually to gauge the enhancements in both performance and work experience resulting from the introduction of the robot equipped with the DECAF planner.
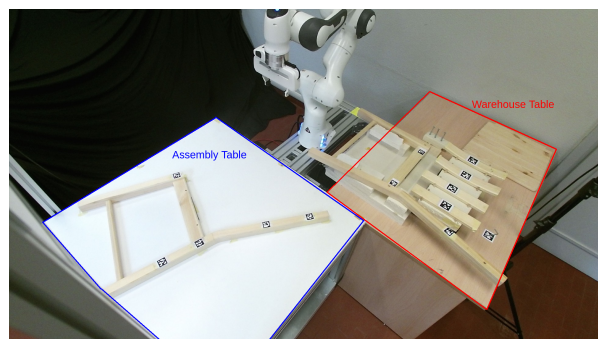

Fig. 2: Experimental setup for the Ivar chair assembly task.

A. Experimental Setup and Task Description

The experimental setup consists of the robotic work-cell shown in Figure 2, including a Franka Emika Panda manipulator and two tables: a *warehouse table* where the components to be assembled are stored, and an *assembly table* where the collaborative assembly process takes place. The area of each table is monitored by an RGB-D camera

(i.e., Intel Realsense LiDAR Camera L515), using April-Tag [23] fiducial markers to detect and localize all the chair components. The chair is composed of several individual parts: left and right sides, a wooden panel for the seat, and rails connecting the two sides through dowel pins and screws. To simplify rails insertion robot operations, the dowel pins were replaced with neodymium magnets. The HTM representing the chair assembly task is shown in Fig. 3.
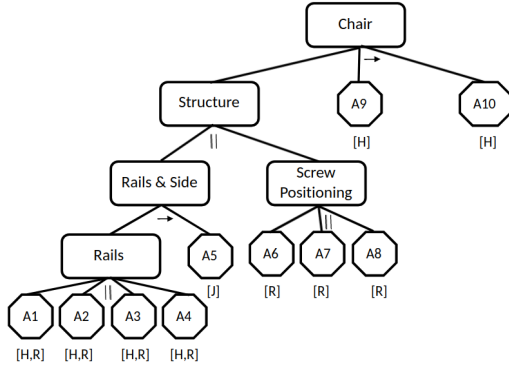

Fig. 3: The HTM of the Ivar chair assembly task.

The task is composed of the following actions: (i) pick and place of the rails, which can be assigned to either the human or the robot (A1-A4); (ii) collaborative transportation of the left side of the chair, where the robot helps the human carrying the weight and assists in the precise positioning of the side (A5); (iii) pick and place of the screws, a robot-only operation due to the significant distance between the human operator and the screws on the warehouse table (i.e., to promote a better ergonomics). The robot places the screws based on the information provided by AprilTag markers. (A6-A8); (iv) screwing, a human-only operation where the human operator tightens the screws placed by the robots (A9); (v) placement of the seat, a human-only action (A10).

The robot actions are implemented as pick and place motion primitives using ROS and MoveIt. The cameras are extrinsically calibrated wrt the robot base through hand-eye calibration [24], allowing to express all the AprilTag poses detected by the cameras in the robot reference frame.

B. Participants and Experiment Design

The experiments involved 10 volunteers (3 women and 7 men) aged between 24 and 46. The group consists of 3 non-robotic experts and 7 experts in the general field of robotics. Each participant performed 3 trials on the system: one trial assembling the chair alone without any collaboration with the robot, and two trials in collaboration with the robot arm. Upon executing all trials, users completed two short surveys: the NASA-TLX [25] to assess the workload and a custom 5-point Likert scale questionnaire to evaluate user satisfaction about the collaborative assembly with robot intervention.

C. Results

We assess collaboration by examining assembly completion time and the outcomes of the aforementioned question-naires. Our objective is to determine whether introducing the

robot equipped with the DECAF planning framework not only enhances execution time but also improves the quality of the assembly experience from the user's perspective.
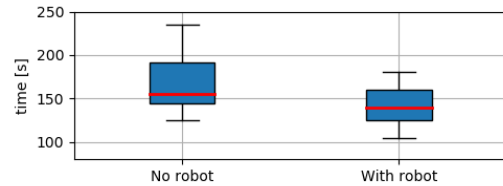

Fig. 4: Completion time distribution obtained with the real user experiments described in section VI.

Fig. 4 compares the completion time distribution obtain by the 10 subjects when assembly the chair alone or with the robot. It is important to note that integrating a robotic collaborator into the assembly process has introduced certain delays. Understanding the robot's intentions typically requires the user to pause and observe it, while sharing the workspace with a moving robot necessitates slower human movements. Furthermore, since our primary focus is not on implementing optimal robot movements, the robot did not operate at maximum speed during testing. As a result, joint actions like frame transportation were slower with the human-robot team compared to a solo human. However, the overall average assembly time was still lower with the robot, highlighting its efficiency benefits. Despite minor drawbacks in human-robot collaboration, the reduced assembly times indicate a positive impact on productivity.
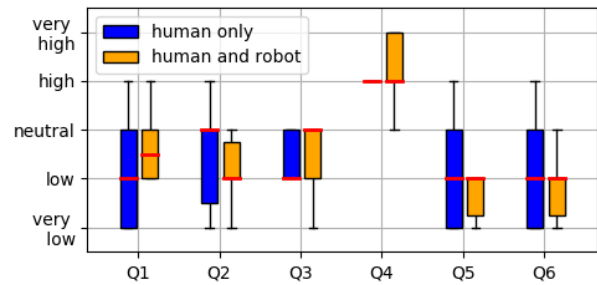

Fig. 5: Nasa TLX questionnaire. **Q1**: How mentally demanding was the task? **Q2**: How physically demanding was the task? **Q3**: How hurried or rush was the pace of the task? **Q4**: How successfully were you in accomplishing the task? **Q5**: How hard did you have to work to accomplish the task? **Q6**: How insecure, discouraged, irritated, stressed, or annoyed were you?

Fig. 5 shows the distribution of responses to the NASA-TLX questionnaire. Overall, participants found working with the robot less physically demanding and easier. However, some participants reported that working with the robot was more mentally demanding than working alone (see Q1 and Q4). As expected, we attribute this to the challenges of sharing the workspace with a moving robot and understanding the robot's intention, which could be counter-intuitive.
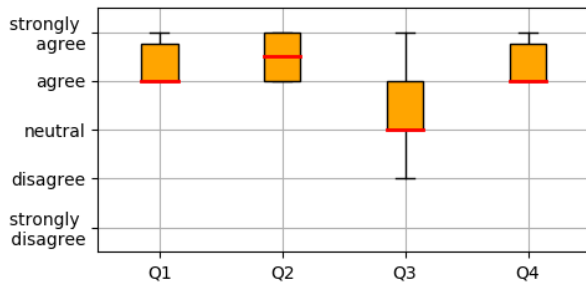
Fig. 6: Robot experience questionnaire. **Q1**: The robot and I collaborated fluently together to accomplish the task. **Q2**: I feel the robot had a good understanding of the task. **Q3**: I was never surprised by the robot actions. **Q4**: I feel satisfied with the performance of the system.

Finally, the results of the custom questionnaire reported in Fig 6 show that the collaboration with the robot has been appreciated by all the participants. Some of them felt surprised by the robot's action choice. We attribute this to the fact that the robot plan is optimized only considering the nominal action duration. This result in a process of action choice that is very different from the decision process of the human, which in turn expects the robot to reason in a human-like fashion. Some users, for example, expected the robot to perform all similar actions consecutively (e.g. position all the screws, then position all the rails) and also would prefer the robot to perform actions uncomfortable for them (e.g. involving objects difficult to reach). A possible way to mitigate this discordance is to add metrics measuring ergonomics or action correlation into the reward function.

VII. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed the DECAF framework for task planning in human-robot collaborative assembly. The framework relies on an HTM description of the task alongside a Bayesian inference module for human intent recognition. We have formalized the collaborative assembly between a Human and a Robot as a Discrete Event Markov Decision Problem, which models at the same time all the properties of Parallel-, Sequential-, Joint- Independent-, Asynchronous- and with Variable Duration- actions, and includes Human intent detection change of mind and handling of failures. We validated the framework through simulations on both toy problems and a real IKEA Ivar Chair. Additionally, an extensive evaluation with real human users confirmed DECAF's benefits in both collaboration efficiency and quality.

REFERENCES

[1] F. Fusaro, E. Lamon, E. De Momi, and A. Ajoudani, "A human-aware method to plan complex cooperative and autonomous tasks using behavior trees," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2021, pp. 522–529.

[2] N. Dhanaraj, S. V. Narayan, S. Nikolaidis, and S. K. Gupta, "Contingency-aware task assignment and scheduling for human-robot teams," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5765–5771.

[3] K. Darvish, E. Simetti, F. Mastrogiovanni, and G. Casalino, "A hierarchical architecture for human–robot cooperation processes," *IEEE Transactions on Robotics*, vol. PP, pp. 1–20, 11 2020.

[4] L. Johannsmeier and S. Haddadin, "A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 41–48, 2016.

[5] H. Fukui, S. Yonejima, M. Yamano, M. Dohi, M. Yamada, and T. Nishiki, "Development of teaching pendant optimized for robot application," in *2009 IEEE Workshop on Advanced Robotics and its Social Impacts*. IEEE, 2009, pp. 72–77.

[6] Z. Zhu and H. Hu, "Robot learning from demonstration in robotic assembly: A survey," *Robotics*, vol. 7, no. 2, p. 17, 2018.

[7] L. H. De Mello and A. C. Sanderson, "And/or graph representation of assembly plans," *IEEE Transactions on robotics and automation*, vol. 6, no. 2, pp. 188–199, 1990.

[8] R. A. Knepper, D. Ahuja, G. Lalonde, and D. Rus, "Distributed assembly with and/or graphs," in *Workshop on AI Robotics at the Int. Conf. on Intelligent Robots and Systems (IROS)*, 2014.

[9] F. Fusaro, E. Lamon, E. De Momi, and A. Ajoudani, "An integrated dynamic method for allocating roles and planning tasks for mixed human-robot teams," in *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*. IEEE, 2021.

[10] M. Iovino, E. Scukins, J. Styrud, P. Ögren, and C. Smith, "A survey of behavior trees in robotics and ai," *Robotics and Autonomous Systems*, vol. 154, p. 104096, 2022.

[11] Y. You, V. Thomas, F. Colas, R. Alami, and O. Buffet, "Robust robot planning for human-robot collaboration," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023.

[12] P. S. Suresh, S. Jain, P. Doshi, and D. Romeres, "Open human-robot collaboration using decentralized inverse reinforcement learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024.

[13] Y. Cheng, L. Sun, C. Liu, and M. Tomizuka, "Towards efficient human-robot collaboration with robust plan recognition and trajectory prediction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2602–2609, 2020.

[14] Y. Cheng, L. Sun, and M. Tomizuka, "Human-aware robot task planning based on a hierarchical task model," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1136–1143, 2021.

[15] K. Ramachandruni, C. Kent, and S. Chernova, "Uhtp: A user-aware hierarchical task planning framework for communication-free, mutually-adaptive human-robot collaboration," *ACM Transactions on Human-Robot Interaction*, 2023.

[16] S. Jain and B. Argall, "Probabilistic human intent recognition for shared autonomy in assistive robotics," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 9, no. 1, pp. 1–23, 2019.

[17] S. Jain and B. Argall, "Recursive bayesian human intent recognition in shared-control robotics," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 3905–3912.

[18] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "Mediapipe hands: On-device real-time hand tracking," *arXiv preprint arXiv:2006.10214*, 2020.

[19] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.

[20] M. Towers, J. K. Terry, A. Kwiatkowski, J. U. Balis, G. d. Cola, T. Deleu, M. Goulão, A. Kallinteris, A. KG, M. Krimmel, R. Perez-Vicente, A. Pierré, S. Schulhoff, J. J. Tai, A. T. J. Shen, and O. G. Younis, "Gymnasium," Mar. 2023.

[21] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, 2021.

[22] S. Huang and S. Ontañ ón, "A closer look at invalid action masking in policy gradient algorithms," *The International FLAIRS Conference Proceedings*, vol. 35, may 2022.

[23] D. Malyuta, C. Brommer, D. Hentzen, T. Stastny, R. Siegwart, and R. Brockers, "Long-duration fully autonomous operation of rotorcraft unmanned aerial systems for remote-sensing data acquisition," *Journal of Field Robotics*, p. arXiv:1908.06381, Aug. 2019.

[24] D. Evangelista, D. Allegro, M. Terreran, A. Pretto, and S. Ghidoni, "An unified iterative hand-eye calibration method for eye-on-base and eye-in-hand setups," in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2022.

[25] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," *Human mental workload*, vol. 1, no. 3, pp. 139–183, 1988.