

Available online at www.sciencedirect.com**ScienceDirect**

Cognitive Systems Research xxx (2016) xxx–xxx

**Cognitive Systems
RESEARCH**www.elsevier.com/locate/cogsys

Time-informed task planning in multi-agent collaboration

Michail Maniadakis*, Emmanouil Hourdakis, Panos Trahanias

Foundation for Research and Technology Hellas, FORTH, Crete, Greece

Received 14 April 2016; received in revised form 4 August 2016; accepted 20 September 2016

Abstract

Human-robot collaboration requires the two sides to coordinate their actions in order to better accomplish common goals. In such setups, the timing of actions may significantly affect collaborative performance. The present work proposes a new framework for planning multi-agent interaction that is based on the representation of tasks sharing a common starting and ending point, as petals in a composite daisy graph. Coordination is accomplished through temporal constraints linking the execution of tasks. The planner distributes tasks to the involved parties sequentially. In particular, by considering the properties of the available options at the given moment, the planner accomplishes locally optimal task assignments to agents. Optimality is supported by a fuzzy theoretic representation of time intervals which enables fusing temporal information with other quantitative HRI aspects, therefore accomplishing a ranking of the available options. The current work aims at a systematic experimental assessment of the proposed framework is pursued, verifying that it can successfully cope with a wide range of HRI scenarios.

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Keywords: Multi-criteria planning; Time-informed planning; Daisy planner; Multi-agent collaboration; Human-robot interaction

1. Introduction

Efficient and realistic human-robot collaboration encompasses crucial temporal and synchronization aspects which are typically overlooked in contemporary robot planning literature. In recent years though, there is a steadily increasing interest to explore the role of time in multi-agent collaboration setups (Chao, 2012; Effinger, Williams, Kelly, & Sheehy, 2009; Hoffman, 2013; Maniadakis & Trahanias, 2014). Multi-agent synchrony is typically achieved by introducing constraints that aim to maximize coincidence in the parallel activities of independent robots (Morris, 2014; Morris, Muscettola, & Vidal, 2001; Shah, Stedl, Williams, & Robertson, 2007; Smith, Gallagher, Zimmerman, Barbulescu, & Rubinstein, 2007).

Simple Temporal Networks (*STNs*) provide the basis to deal with temporal constraints in planning problems. To manage temporal constraints, *STNs* are typically mapped to the equivalent Distance Graphs (*DGs*) to check the existence of no negative cycles and thus prove the consistency and dispatchability of the plan (Dechter, Meiri, & Pearl, 1991). Along this line, recent works have considered back propagation rules to dynamically preserve dispatchability of plans (Morris, 2014; Shah et al., 2007), address temporal problems with choice (Shah & Williams, 2008), or reason between interacting agents (Boerkoel & Durfee, 2013).

Despite the effectiveness of relevant approaches, *STNs* exhibit an inherent limitation to deal with event sequences where start and end points coincide; such behaviors are termed “*daisy behaviors*” in the current work, as will be explained in Section 4 of the paper. The coincidence of start and end points creates *STN* loops which enable the

* Corresponding author.

E-mail address: mmaniada@ics.forth.gr (M. Maniadakis).

identification of negative circles in the equivalent *DG*, therefore suggesting the inconsistency of the relevant plans.

Disjunctive Temporal Constraint Networks (*DTCNs*) have been used as a basis for tackling such problems by considering the temporal properties of all possible assignments of tasks to agents, in order to select the best full plan (Shah & Williams, 2008). Besides the issue of computational complexity in case that many tasks have to be assigned to many agents, this approach is rather fragile, in the sense that unexpected events may destroy the execution of the plan and thus initiate re-planning. The same basic idea is followed in Effinger et al. (2009), by implementing an extensive AND/OR search tree over the possible plan executions.

A common limitation of all works outlined above, regards the treatment of time in isolation, without the ability to jointly consider other quantitative criteria that may affect collaboration. Along this line, the time-informed multi-criteria evaluation of plans is particularly new in the multi-agent interaction literature. The only work we are aware of is (Gombolay, Wilcox, & Shah, 2013), in which the planner aims at minimizing annoyance among agents and thus practically avoids team members collaboration.

The present work puts forward a new framework for studying multi-agent interaction, assuming the daisy-like representation of tasks and the use of fuzzy numbers to encode temporal information. The latter facilitates the direct use of time in mathematical calculations and thus the detailed analysis of graph properties, in order to take better-informed planning decisions.

In the literature, fuzzy times are used for many years in job scheduling problems (Deng, Chen, Zhang, & Mahadevan, 2012; Dubois, Fargier, & Prade, 1995). It is therefore surprising that, to the best of our knowledge, it is the first time they are employed in the context of dynamic multi-agent collaboration. Interestingly, fuzzy arithmetic facilitates mixing temporal criteria with any other numerically represented information regarding multi-agent interaction. The latter paves the way for pursuing immediate, locally optimal assignment of tasks to agents, in order to better direct human-robot collaboration towards the accomplishment of the common goal. Due to the simplicity of fuzzy number calculus, the current approach does not introduce any workload compared to contemporary approaches, therefore resulting to an easily implemented and particularly fast solution for multi-criteria, time-informed human-robot planning.

In contrast to previous works on scheduling multi-agent interaction that prepare full plans of agents' activities for all future moments (e.g. Gombolay et al., 2013; Shah & Williams, 2008), the proposed planner adopts an immediate, short-term planning approach, that enables taking locally optimal decisions, after considering the circumstances at the given moment. Accordingly, the planner operates as a light-weight process and at the same time

minimizes the chances for re-planning in the case of unexpected events.

To facilitate the systematic evaluation of the proposed framework, the planner is integrated into a simulated robot environment with two humanoids, one having the role of master (representing the human) and the other having the role of slave (representing the robotic partner). The proposed approach is assessed on “multi-agent collaboration for salad preparation”. The jobs assumed for the collaborating agents are mapped to a *daisy graph* with petals representing the tasks that can be undertaken by single agents. The proposed planner assigns tasks to the agents, considering (i) the time required for their execution and (ii) the quality of performance each agent may achieve. In that way, the planner takes short-term optimal decisions that, despite they cannot guarantee global optimality, result into very flexible and effective multi-agent synergies, as witnessed by the assumed results of the present study.

We use the objective metrics proposed in Hoffman (2013) to assess the performance of the planner in four realistic human-robot interaction scenarios that simulate (i) human's leading role, (ii) self-motivated human actions, (iii) possible delays on task execution, and (iv) human preferences with respect to tasks. The obtained results show that the daisy planner is capable to reduce the idle time of agents and at the same time enforce their concurrent performance to improve collaboration.

The rest of the paper is structured as follows. The next section links the current work with the broader research in time perception and robotics. Then we discuss the representation of time intervals as fuzzy numbers, which enables making calculations with time. The presentation of the daisy planner comes in the following section, discussing in detail how the proposed architecture facilitates time-informed multi-agent coordination. Experimental results of the proposed planning framework in action are presented in Section 5, followed by discussion on the obtained results. The last section concludes the present work, highlighting also directions for extending the proposed framework.

2. Mind-time interactions

The sense of time is an essential capacity of humans, animals, birds, fishes, even plants, as described in Cashmore (2003). Time perception is among the first competencies evolved in biological systems, which means it has affected the subsequent evolution of nearly all cognitive modalities (Gerstner, 2012; Paranjpe & Sharma, 2005). Additionally, many time processing modalities mature very early in the human developmental procedure in order to provide a stable basis for other cognitive skills to develop (Droit-Volet, 2013). As a result, time is suggested to be the dimension that is dominantly used in the perception of complex stimuli (followed by space) (Navon, 1978). These remarks promote the notion of time as one of the most influential factors in the functionality of cognitive systems.

Understanding the time processing mechanisms in the brain of animals and humans is a timely and very challenging issue that has attracted rapidly increasing research interest in the neuroscience and cognitive science communities (Coull, Van-Wassenhove, & Coslett, 2013; Grondin, 2010). However, relevant research in artificial systems has not encountered particular progress (Maniadakis & Trahanias, 2014). Evidently, it is now high time to advance the temporal cognition of artificial agents as a means of accomplishing seamless and naturalistic human-robot interaction.

So far, cognitive systems research has mainly focused on time perception for very short time scales (in the order of milliseconds to few seconds). Further research is required to obtain insight into the mechanisms processing time in the order of minutes, and higher. In those scales, mind-time interactions have been mainly considered from a robotic, largely algorithmic point of view, as summarized in the previous section. Still, relevant procedures have rather limited potential to be used as an explanatory framework of temporal cognition in biological systems. The present work provides the means to combine time with other types of information and sensory input, therefore introducing a more general framework to explain time-related multi-modal cognition. Interestingly, the proposed interval theoretic framework can be universally applied to both short and long time scales, providing a new prism for studying well known properties of cognition in relation to time (e.g. the effect of emotion on time perception).

Without restricting the use of the proposed timing framework to a single cognitive task, the rest of the paper concentrates mainly on task planning. Core ideas on the representation and processing of time will be discussed from a planning point of view, being however applicable to other time-related cognitive procedures.

3. Fuzzy times

To facilitate time-informed planning, we use a fuzzy theoretic approach for the representation of time. The key idea regards representing time intervals as fuzzy numbers and exploit the power of fuzzy number calculus to compare alternative planning hypotheses. It is noted that the use of (non-fuzzy) interval arithmetic would theoretically give very similar results to the ones presented here. Still, in the present work we have adopted the fuzzified approach because it is well established, simple and computationally easy to implement. Moreover, it provides the potential to mix timing uncertainty with the uncertainty of other qualitatively defined measures in human-robot collaboration setups (e.g. fatigue), that is our ultimate, future goal.

A fuzzy number is a generalization of a regular, real number in the sense that it does not refer to one single value but rather to a connected set of possible values (Dubois & Prade, 1978). Fuzzy numbers allow us to treat a given planning problem as a fuzzy mathematical model.

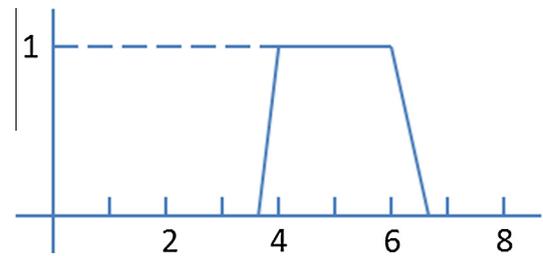


Fig. 1. Schematic representation of a fuzzy number representing the duration “approximately 4–6 min”.

In the present work we use trapezoidal¹ fuzzy numbers to represent the time boundaries an action may take to complete and we use fuzzy arithmetic to develop multi-criteria measures that enable comparing alternative planning scenarios.

To estimate the minimum a and maximum b time an action may take to implement, we assume an experimental procedure that involves a large number of repetitions as a means of estimating a and b values. Clearly the estimated a and b values are not absolute, but rather a rough approximation of the actual minimum and maximum durations for the given action (a larger number of repetitions could result into a slightly lower a or a slightly bigger b). To address this uncertainty, we assume that the actual minimum and maximum durations for the given action may possibly be 10% lower than a (i.e. $0.9 \times a$) and 10% higher than b (i.e. $1.1 \times b$). Following this assumption, it is possible to represent the duration of the action initially estimated as “approximately a to b ” with a fuzzy number. To this end, we use fuzzy numbers in trapezoidal form, represented by the quadruplet (p, m, n, q) . More specifically, m is assigned the estimated lower bound of the time interval, $m = a$, while n is assigned the estimated upper bound of the time interval, $n = b$. Subsequently, we use the assumed bounds to define $p = 0.9 \times a$ and $q = 1.1 \times b$. As an example, the fuzzy time “approximately 4 to 6 min” represented by the trapezoid $(3.6, 4, 6, 6.6)$ is shown in Fig. 1.

Following this formulation, a classic STN is transformed into its fuzzy form $fSTN$ by representing any edge labeled with $[a, b]$ in the original network, with a similar edge labeled with the fuzzy trapezoidal number $(0.9a, a, b, 1.1b)$. Interestingly, the fuzzified interpretation of temporal constraints had been theoretically investigated in the past (Vila & Godo, 1994), with rather rare practical applications.

Defuzzification. There are many defuzzification procedures that map fuzzy numbers on ordinary crisp values. In the present work defuzzification is accomplished following the classic graded mean integration representation (Khadar, Madhusudhan, Ramanaiah, & Karthikeyan,

¹ The trapezoid representation of fuzzy numbers is not mandatory but simplifies calculations and therefore it is adopted in the present work.

2013), which assumes a fuzzy number $F = (p, m, n, q)$ to be represented by the crisp value:

$$\text{def}(F) = (p + 2m + 2n + q)/6. \quad (1)$$

Fuzzy time addition. The addition of two fuzzy numbers $F1 = (p1, m1, n1, q1)$ and $F2 = (p2, m2, n2, q2)$ is a new trapezoid fuzzy number of the form:

$$F1 + F2 = (p1 + p2, m1 + m2, n1 + n2, q1 + q2) \quad (2)$$

Fuzzy time subtraction. The difference between two trapezoid fuzzy numbers is again a fuzzy number defined as follows:

$$F1 - F2 = (p1 - q2, m1 - n2, n1 - m2, q1 - p2) \quad (3)$$

Mix fuzzy time and crisp values. Ordinary crisp numbers can be easily incorporated into fuzzy calculations. In particular the addition of a real r with a fuzzy number $F = (p, m, n, q)$ results into a new fuzzy number:

$$F + r = (p + r, m + r, n + r, q + r) \quad (4)$$

Similarly, multiplication by a crisp real number r is given by:

$$F \cdot r = (p \cdot r, m \cdot r, n \cdot r, q \cdot r) \quad (5)$$

Comparison of fuzzy numbers. In the present work we employ defuzzification as a means of comparing two fuzzy numbers $F1$ and $F2$. In particular, we assume that:

$$F1 > F2 \text{ iff } \text{def}(F1) > \text{def}(F2) \quad (6)$$

The latter provides a mechanism to compare and rank alternative task assignments and thus promotes local optimality in planning decisions.

4. Daisy behaviors

The current work considers the planning of complex behaviors separated into tasks that can be executed by different agents. For example, in the case of salad preparation, the collection of the ingredients assumes that one or more agents move to the cabinets, the fridge, the pantry, etc. to fetch the vegetables, the olive oil, the salt, etc. For an agent located in front of the countertop, the task “bring the salt” consists of a series of actions such as “move to the pantry”, “find the salt”, “grasp it” and “bring it back to the countertop”. Bringing the vegetables consists of another series of actions which will again start and end in front of the countertop. This type of daisy behaviors consisting of tasks sharing common start and end characteristics (similar to self-suspension in multiprocessor scheduling [Ridouard & Richard \(2006\)](#)) are in the main focus of the present work.

In order to graphically represent the composite behavior, each task is implemented as a sequence of actions that accomplish certain events. We assume that an “action” describes a motion sequence driven by a single basic goal. All steps of the motion sequence aim to accomplish the same goal, or the same change on the world state. A “task” is a higher level entity described by two or more actions,

which corresponds to accomplishing two or more successive changes on the world. For example, a complex task like “bring salad bowl” assumes that the robot will go to the cupboard, will grasp the salad bowl, will go back to the countertop, and will place the salad bowl on the countertop.

Following this typology, the graphical representation of ingredients collection results in a daisy-like scheme (see [Fig. 2](#)) where each task aiming at the collection of an ingredient is represented by one petal. Events are represented as vertices on petals linked by edges that represent actions. Such a daisy representation highlights the ability of parallel execution of tasks and it is proposed as a structured and scalable means to coordinate multi-agent setups.

Interestingly, task decomposition often includes pre- and post-actions that accompany the main-action. For example, to “add olive oil” in the salad, an agent should first grasp and move the olive oil bottle just above the salad bowl (pre-action). Then, the bottle must rotate to enable the flow of olive oil on the salad (main-action) and finally the agent must place the bottle back on the countertop (post-action). Even if task labeling concentrates on the main action, human common sense includes pre- and post-actions as secondary but still significant parts of the task that enable the same agent to undertake and implement a sequence of tasks. The three action types are provided as a useful conception which helps the experimenter in drawing the daisy plan that better summarizes the relationships between tasks. The planner treats pre- main- and post-actions without distinction.

Here we assume that the experimenter exploits background knowledge on the specific application domain, to devise a meaningful daisy plan that includes tasks separated into pre- main- and post-actions, which is provided to the planner as a basis for coordinating multi-agent activities. From this perspective, our approach shares common

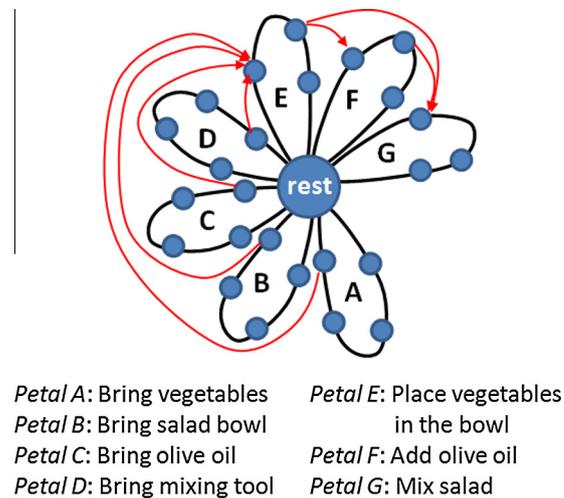


Fig. 2. A daisy graph representing the interaction of tasks that should be executed for the preparation of salad. Petals A, B, C and D represent tasks that can be implemented at any order. Petal E can be executed only after A, B, C, D, petal F can be executed only after A, B, C, D, E and petal G can be executed only after A, B, C, D, E, F. See text for details.

characteristics with the scheduling problems and indeed, we have been inspired by the relevant literature. Typically, in time-informed robot planning, task decomposition and the graph representing the scenario are developed offline, prior to the setup of the experiment (Abdeddam et al., 2007; Casanova, Lesire, & Pralet, 2015; Effinger et al., 2009; Gombolay et al., 2013). Likewise, we assume the decomposition of the behavior into tasks and actions to be developed offline prior to the setup of the experiment. The decomposition of behaviors into tasks and actions is an important research topic on its own right (Yan, Jouandeau, & Cherif, 2013), being however outside the scope of the present work.

4.1. Parallel task coordination

The proposed framework can easily address multi-agent coordination by introducing temporal constraints across petals. For example, salad preparation assumes placing vegetables in the bowl prior to adding olive oil. Such a dependency between the petal that implements “place vegetables in the bowl” and the petal that implements “add olive oil on top of vegetables” is represented by a temporal constraint linking the two events and thus informing the planner that a certain action should proceed another one.

Interestingly, the temporal constraints mentioned above typically describe relations between main-actions, therefore allowing pre- and post-actions from different petals to execute in parallel. Consider for example the petals E and F shown in Fig. 2 that are further analyzed in Fig. 3. The constraint (shown in red) indicates that F1 (Bottle at hand) is considered complete and the action towards F2 (Add olive oil on vegetables) can be initiated, only after E2 (Vegetables in the salad bowl) is complete. Temporal constraints are represented by an edge linking the relevant vertices, labeled with the minimum amount of time execution (usually one moment).

Clearly, the distinction between pre- main- and post-actions facilitates the parallel execution of non-conflicting activities. Assume for example two agents, each one working on a petal of the plan shown in Fig. 3. The first agent may grasp the olive oil bottle even before vegetables are

placed in the bowl. The addition of olive oil on vegetables can be executed only after the second agent has placed the vegetables in the salad bowl. After that, the two agents may execute the remaining actions in any order. While one agent throws away the vegetables bag, the other may place back the olive oil bottle. The agent that will finish first will be assigned the next task which regards salad mixing.

4.2. Multi-criteria assignment of tasks to agents

A particularly interesting aspect of daisy behavior planning regards the locally optimal assignment of tasks to agents, taking into account their specialized capabilities (speed of execution, quality of implementation, etc.). The fuzzy theoretic representation of time intervals introduced in Section 3 provides the substrate for mixing temporal constraints with other aspects of HRI, therefore enabling the multi-criteria optimized distribution of tasks to agents.

Taking advantage of fuzzy arithmetic it is possible to design composite time-informed criteria that enable ranking alternative assignments of tasks to agents. This is a newly introduced feature in multi-agent collaboration, which enables locally optimal planning decisions. Along this line, the present work considers the “time of execution” and the “robustness for task completion” as the main criteria for the attribution of tasks to agents. In particular, all tasks represented on the daisy graph are assigned for each agent (i) a triangular fuzzy number representing the time that may be spent by the agent for the implementation of the task and (ii) the corresponding level of robustness of the agent in the underlying task. We assume 5 levels of robustness (very low, low, medium, high, very high) represented by the numbers 1, 3, 5, 7, and 9. An illustrative example of fuzzy time and level of robustness assigned to the edges of petals is shown in Fig. 4. For the purposes of the present work we assume scalar robustness values which are empirically hand set, by the experimenter.

In order to assign a new task to an agent x the planner examines the daisy graph to identify the petals remaining for execution. It gives priority to the petals that are not constraint by the prior implementation of tasks, being directly and fully available for execution. The petals whose constraints are not fulfilled yet will be considered last by the planner.

For each fully available petal p it estimates the total time of execution $TT_{x,p}$ (following Eq. (2)) and the minimum

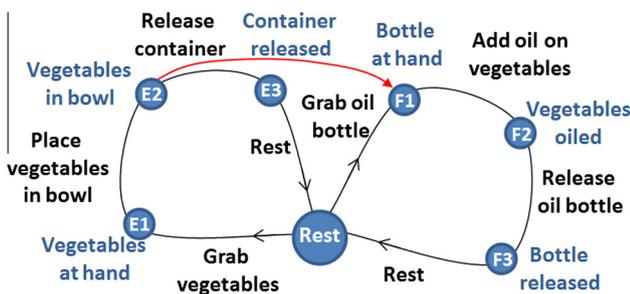


Fig. 3. Detailed description of the petals E and F shown in Fig. 2 that demonstrates the functionality of constraints among events, shown as red arrows. Action execution may depart from F1, only after E2 is completed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

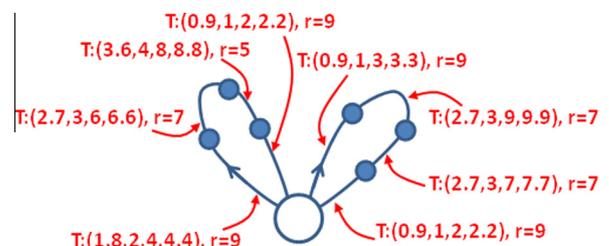


Fig. 4. A simple daisy graph illustrating the fuzzy times and robustness values assigned on the execution of each task in the petals.

level of robustness $MR_{x,p}$ of the agent x for all the actions involved in this petal. Following the example shown in Fig. 4, the total time of execution of the 1st petal is estimated to $TT_{x,1} = (1.8, 2, 4, 4.4) + (2.7, 3, 6, 6.6) + (3.6, 4, 8, 8.8) + (0.9, 1, 2, 2.2) = (9, 10, 20, 22)$ and the minimum level of robustness on this petal is estimated to $MR_{x,1} = \min\{9, 7, 5, 9\} = 5$. Similarly for the 2nd petal, the total time of execution is estimated to $TT_{x,2} = (0.9, 1, 3, 3.3) + (2.7, 3, 9, 9.9) + (2.7, 3, 7, 7.7) + (0.9, 1, 2, 2.2) = (7.2, 8, 21, 23.1)$ and its minimum robustness is $MR_{x,2} = \min\{9, 7, 7, 9\} = 7$.

The estimated cost for the assignment of agent x on petal p is estimated according to the formula:

$$C_{x,p} = TT_{x,p} / (1 + MR_{x,p}) \quad (7)$$

For the given example, the above equation results into $C_{x,1} = (9, 10, 20, 22) / (1 + 5) = (1.5, 1.66, 3.33, 3.66)$ and $C_{x,2} = (7.2, 8, 21, 23.1) / (1 + 7) = (0.9, 1, 2.6, 2.9)$. The two fuzzy measures can be compared as described in Eq. (6) which renders $C_{x,1} > C_{x,2}$. The latter suggests that agent x should be preferably assigned on the 2nd petal due to its lower estimated cost.

In the case that no petal can be identified that is directly and fully available for execution, e.g. because of constraints from tasks that are not completed yet (see for example the case of petals E and F in Fig. 2), the planner considers the properties of individual actions. In the current implementation the agent is assigned the task restricted by the least number of constraints, in the hope that these constraints will be fulfilled soon and the agent will proceed with the next action in the given petal. If the agent remains idle for long time that is beyond a threshold th_{st} , the currently assigned action is released asking the re-assignment of a new action by the planner.

4.3. Immediate optimal planning

Multi-agent collaboration in unstructured environments may be often interrupted by unexpected events that disturb the execution of actions. In such cases it is rather impractical to devote resources into the estimation of global plans that describe a long sequence of actions, similar to Shah and Williams (2008) and Effinger et al. (2009). It seems more effective to adopt a progressive approach that develops optimal short-term plans based on the here and now of the world and the current state of the collaborating team. We term such an approach “Immediate Optimal Planning” (IOP). In essence, IOP aims to keep all available agents busy by better exploiting their own skills for the benefit of the team. To this end, it develops short-term optimal matches between the capacities of the agents not charged with a job and the tasks to be computed at the given moment.

The proposed approach fits perfectly to multi-agent scenarios where collaboration evolves according to the master-slave teamwork mode. This is typically the case in

human-robot interaction. The human partner plays the role of the master and the robotic partner plays the role of the slave. IOP can easily address such situations by giving priority to the human, properly directing robot actions to fit the human needs and choices.

Through appropriate parameterization, the planner can be informed that the human partner dislikes a certain task. Due to the multi-criteria ranking of agent-task pairs, the planner will avoid assigning the given task to the human, but enforce its assignment to the robotic partner.

The planner suggests to the human a task (optimally selected for her/him according to the planner’s criteria) and the human is free to adopt or not this suggestion. Following to the actual human choice, the planner shifts attention to the robot considering the available tasks. It optimally selects the new task for the robot maximizing its fit to the human decisions and activities. Moreover, in case that the human undertakes the execution of a task, but for some unexpected reason the task is not completed after a specified time period, the task is withdrawn from the human and may be re-assigned to the robot to be implemented in the immediate future.

5. Results

The proposed framework introduces new features in human-robot interaction, which enable time-informed, multi-criteria, locally optimal planning. We have considered experiments involving up to four agents, which is an adequate upper limit for human-robot collaborative task execution.

To explore the performance of the planner in a broad range of scenarios, a simulation environment was used. More specifically, the studied simulation assumes collaboration of two Aldebaran NAO’s in a master-slave mode, with the master agent representing the human partner that is free to adopt or not planner’s suggestions, and the slave agent representing the robot that adapts its performance to match human decisions.

Two-agent interaction is studied in a naturalistic scenario that assumes the collaborative preparation of a salad. This is a very common real-world task for humans which involves a relatively small number of objects therefore enabling a realistic and thorough transfer of the overall setup into the simulated world. Salad preparation has been used throughout the text as the main motivating example and the key aspects regarding the daisy representation of the scenario and the IOP procedures have been already discussed in detail. The collaborative behavior is split into seven well defined tasks as shown in Fig. 5, which are further separated into actions as illustrated in Table 1. For each action, the minimum and maximum time of execution for the two agents and its graded robustness in the action are shown in columns 3–6 of the same table. According to the assumed role of Robot1 as the human partner, it is assigned high robustness values on all actions.

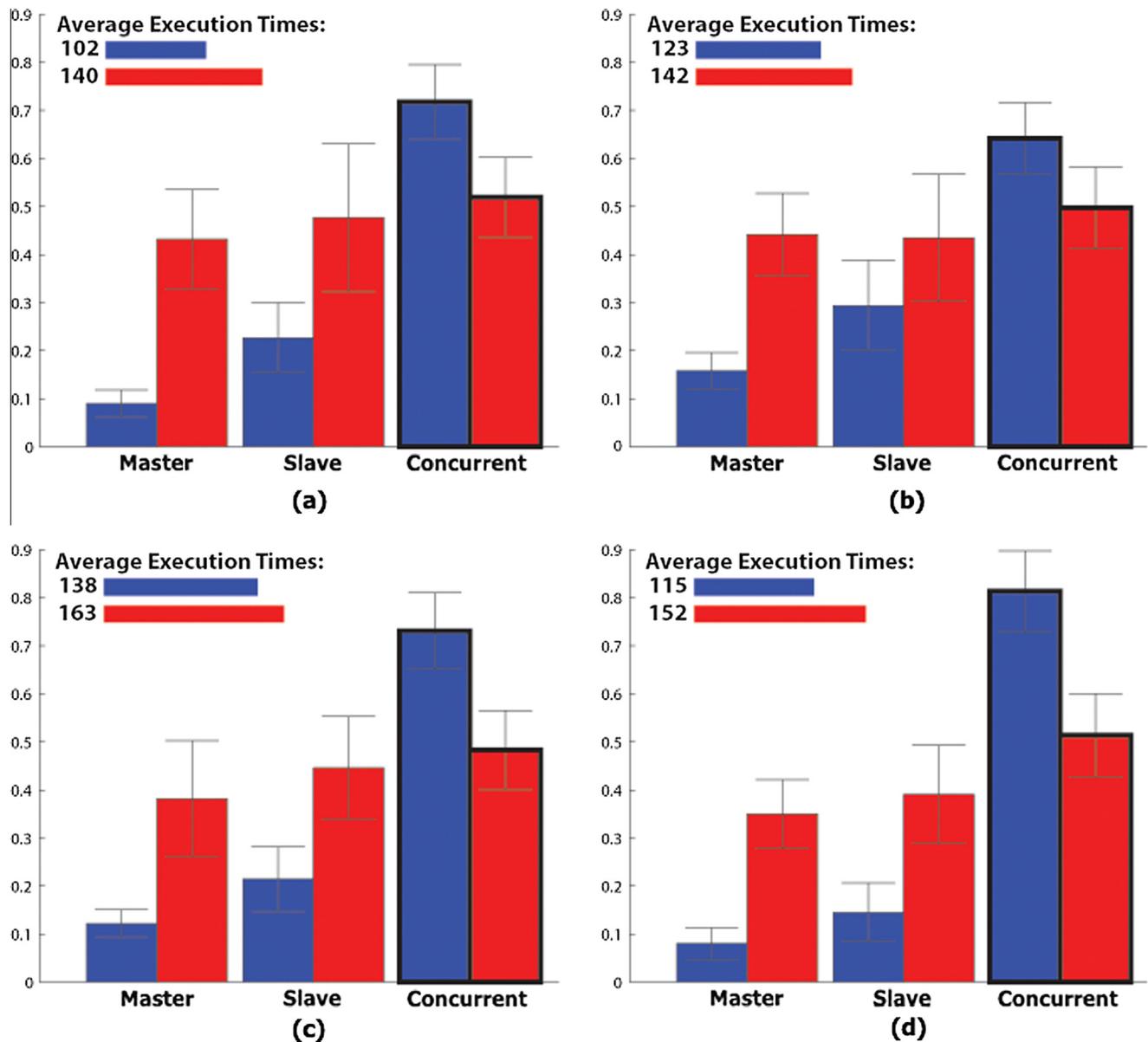


Fig. 5. The mean and standard deviation of (a) case c1 - normal evolution, (b) case c2 - task assignment noise, (c) case c3 - execution delay and (d) case c4 - varying robustness. In all plots, the first pair of bars shows master's idle time, the second slave's idle time and the third the time of concurrent activity for the two agent. Each plot summarizes the results of 100 randomly generated runs for the proposed planner shown in blue and for a random planner shown in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The assessment of the daisy planner assumes investigating its performance in multiple and different lines of scenario evolution. The present study considers four such cases:

- **c1 - normal evolution:** the planner directs actions of both agents,
- **c2 - task assignment noise:** the master agent may arbitrarily ignore planner suggestions,
- **c3 - execution delay:** the implementation of the master's actions is delayed beyond the known upper limits,
- **c4 - varying robustness:** both agents exhibit limited robustness in the execution of actions.

The simulated environment enables automatically testing the planner on a large number of randomized instances. In the current study we consider 100 different instances for each one of the four cases mentioned above.

The quantitative assessment of the planner is based on the objective metrics introduced in Hoffman (2013), evaluating the fluency of human-robot collaboration. In particular, the statistical information extracted from each run regards the master's idle time, the slave's idle time, and the time of concurrent activity. Due to the formulation of the daisy plan, at any moment of the experiment at least one of the agents remains active, therefore, the fourth measure proposed in Hoffman (2013), namely functional delay,

Table 1
Agents' time and robustness level for each action.

	ACTION	ROBOT 1		ROBOT 2	
		Time	Robustness	Time	Robustness
A1	Go to cupboard	[21, 32]	9	[22, 38]	9
A2	Get salad	[4, 9]	9	[6, 10]	7
A3	Go to countertop	[21, 32]	9	[22, 38]	9
A4	Place salad on countertop	[3, 7]	9	[4, 8]	7
B1	Go to cupboard	[21, 32]	9	[22, 38]	9
B2	Get salad bowl	[6, 9]	9	[7, 10]	5
B3	Go to countertop	[21, 32]	9	[22, 38]	9
B4	Place salad bowl on countertop	[4, 7]	9	[5, 7]	7
C1	Go to cupboard	[21, 32]	9	[22, 38]	9
C2	Get olive oil bottle	[4, 8]	9	[7, 10]	7
C3	Go to countertop	[21, 32]	9	[22, 38]	9
C4	Place olive oil bottle on countertop	[3, 5]	9	[3, 6]	7
D1	Go to cupboard	[21, 32]	9	[22, 38]	9
D2	Get mixing tool	[3, 8]	9	[5, 9]	7
D3	Go to countertop	[21, 32]	9	[22, 38]	9
D4	Place mixing tool on countertop	[4, 6]	9	[3, 8]	5
E1	Get salad	[4, 9]	9	[6, 10]	7
E2	Put salad in the bowl	[7, 10]	9	[7, 12]	7
E3	Release salad container	[2, 4]	9	[3, 6]	9
F1	Get olive oil	[4, 8]	9	[7, 10]	7
F2	Put olive oil in salad	[5, 8]	9	[5, 7]	9
F3	Release olive oil bottle	[3, 5]	9	[3, 6]	9
G1	Get mixing tool	[3, 8]	9	[5, 9]	7
G2	Mix salad	[5, 11]	9	[6, 14]	7
G3	Release mixing tool	[3, 5]	9	[3, 8]	9

that is based on the time that both agents remain idle is not applicable for the current experiments. Our results consider a fourth task-specific measure that is related to the average total time spent for the preparation of the salad.

The proposed approach is contrasted against a basic planner that assumes the random assignment of petals to agents. Apart from that, all other parameters (stuck time, petal reset threshold, etc.) remain the same for the two planners.

The results obtained for the two planners are summarized in Table 2. Additionally, Fig. 5 provides a graphical illustration of the same results. Plot (a) corresponds to the normal master slave collaborative planning. The daisy planner (shown in blue) results into lower idle times for the master agent and adequately low idle times for the slave agent. The master leads the plan letting the slave agent to

adjust on it. This is the reason why the variance of slave's idle time is relatively higher. The daisy planner facilitates concurrent activity of the two agents as witnessed by the sufficiently high third blue bar. As it is shown in the graph, planning agents' activities with a random petal assignment impedes collaboration among agents resulting in a longer implementation of the composite behavior. This is because randomness practically abolishes leadership of the master agent and ignores ordering constraints among tasks.

Plot (b) corresponds to the case that the master agent ignores petal assignments with a probability 25% (this might be the case for a human member of the team). In this case the total time of execution slightly increases because the effect of local optimality in the plan is partially eliminated (i.e. the master may choose tasks that should be better assigned to the slave). The idle time of both agents

Table 2
Performance results for the Daisy Planner (DP) and the Random Planner (RP).

	CASE-1		CASE-2		CASE-3		CASE-4	
	DP	RP	DP	RP	DP	RP	DP	RP
Robot	Mean (STD)	Mean (STD)	Mean (STD)	Mean (STD)	Mean (STD)	Mean (STD)	Mean (STD)	Mean (STD)
Master	0.09 (0.03)	0.43 (0.10)	0.15 (0.04)	0.44 (0.08)	0.12 (0.03)	0.382 (0.12)	0.08 (0.03)	0.35 (0.07)
Slave	0.22 (0.07)	0.47 (0.15)	0.29 (0.09)	0.43 (0.13)	0.214 (0.07)	0.44 (0.11)	0.14 (0.06)	0.39 (0.10)
Concurrent	0.71 (0.08)	0.52 (0.08)	0.64 (0.07)	0.49 (0.08)	0.73 (0.08)	0.48 (0.08)	0.81 (0.08)	0.51 (0.09)

slightly increases because the master may be occasionally stuck on unsuccessfully chosen, constrained petals. This is also the main reason why the amount of concurrent agents' activity gets worst. The use of the randomized planner for this case provides actually the same results as in the previous case. The random selection of tasks by the master has no direct effect over the already randomized assignment of tasks by the planner.

The plot (c) in Fig. 5, corresponds to the case of delayed execution of tasks by the master agent. In particular, the agent executes actions on average 1.5 times longer than normal, which is often more than the known maximum time of actions. This makes the master agent look relatively busy, despite the increase in the total time of implementing the collaborative behavior. However, this is not the case for the slave agent. Even if the slave now undertakes a larger part of the composite behavior, the introduced delays by the master that leads the plan, makes the idle time of the slave agent occasionally exceed the specified threshold of being idle th_{st} , causing a task assignment reset. The employment of the planner that operates with random petal assignments significantly delays the total time of implementation. This is because execution delays in combination with random assignments significantly increase the probability that an agent gets stuck.

The last plot in Fig. 5, examines the case of using varied robustness values for the master agent (i.e. the third column of Table 1 is randomly initialized with values 5, 7, or 9). In terms of action execution, lower robustness corresponds to adding gradually more noise on the actuators of the simulated agent. Intuitively, the current setup is similar to the third case considering action delays, but now the planner gets informed of the expected delayed implementation of tasks. The latter highlights the beneficial effect of multi-criteria optimality in the planner which assigns tasks to the agents after taking into account the possibility of delays during action execution. Therefore, besides the slight increment in the total time of composite behavior implementation, both agents remain idle for relatively short times, keeping high the amount of time they are concurrently active. The overall picture of the randomized planner is a bit better than in case (c) because the delays on action execution are now less, reducing the probability that any of the agents gets stuck.

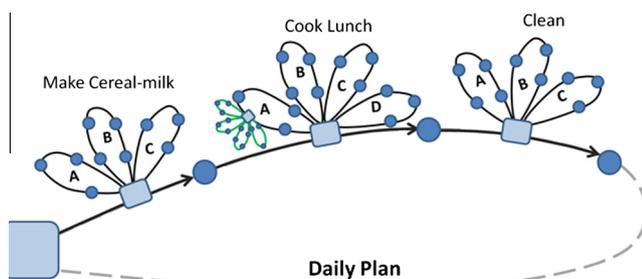


Fig. 6. The multi-level extension of the daisy architecture which will enable planning at multiple temporal scales.

6. Discussion

The current work introduced a new, time-informed, multi-criteria planner that is able to flexibly coordinate multi-agent interaction in highly dynamic environments. The obtained results demonstrate that the proposed planner can adequately cope with scenarios exhibiting varying characteristics, inspired by realistic human-robot interaction setups. More specifically, studied scenarios involve (i) leading role of human - case c1, (ii) self motivated human action - case c2, (iii) task execution delays - case c3, and (iv) human preferences with respect to tasks - case c4.

Currently, the daisy plans are broken down into well defined petals describing circular tasks that start and end at the same state. The latter may seem restrictive for the proposed approach, rendering the daisy representation applicable only to a bounded set of behaviors. To address this issue, on-going work explores the properties of multi-level daisy graphs which assume actions being further analyzed as daisy behaviors at a more fine level (see Fig. 6). Multi-level architectures significantly increase the taxonomy of problems that can be effectively addressed by the daisy planner. Interestingly, such a representation enables encoding and planning agent actions at multiple time scales which is another very interesting feature for human-robot interaction setups. For example, daily jobs may consist of cereal-milk preparation, cooking and cleaning, which are further analyzed into daisy plans as discussed throughout the paper.

The daisy planner is perfectly coupled with an analytical method that accomplishes the locally optimal selection of tasks for each agent. The fuzzified representation of time intervals and the use of the relevant arithmetic, enable considering time as an ordinary metric that can be used in equations and optimality criteria. It is noted that the adoption of an alternative, probabilistic representation of time, would require a much more complex mathematical formulation for implementing basic arithmetic operations - for example, differentiate according to the (in)dependence of the random variables representing duration. The present work adopts the fuzzified representation mainly because of its simplicity in making arithmetic that combines time with other numerical features. This is a particularly unique property of the proposed approach which enables mixing time with criteria such as the priority of tasks, the effectiveness of agents on tasks, even fatigue or task-dislike in the case of human partners. Such a capacity is particularly useful in symbiotic human-robot interaction.

The present study did not focus particularly on the “uncertainty” encoded in the fuzzified representation of time. As it has been already mentioned in Section 2, the use of any other interval calculus for the addition, subtraction and comparison of time intervals would also allow the daisy planner to accomplish very similar results in terms of multi-agent collaboration. Moreover, the issue of the number of petal-constraints has not been extensively considered

in this work, since the current implementation has been mainly designed on the basis of the examined collaborative behavior. The more thorough study of the uncertainty encoded in fuzzy representations and the systematic consideration of petal-constraints number is within our future research priorities for the daisy planner.

Throughout the paper, a rough estimate of the agents' robustness on the available action set represented by preset scalar values, is employed. Devising a function that measures agents' robustness would require a large number of robot data to be obtained and statistically analyzed, without any impact though on the actual study of this work, i.e. the expressive power of the daisy planner. The manually-set robustness values used herewith provide a stable proof of concept for the ability of the daisy planner to flexibly coordinate multi-agent setups. In the future we also plan to develop a fuzzy robustness measure (e.g. using the inverse of the mean error rate) that will be combined with the fuzzy representation of time to provide a better informed tool for ranking alternative task assignments in multi-agent collaboration problems.

7. Conclusions

The broader vision of our research aims at time-aware social robots that successfully interact with humans for the collaborative accomplishment of mid-term goals. Besides the fundamental role of time in cognition, the perception and processing of time in association with the information obtained by other sensory modalities remains rather poorly investigated. The present work provides a holistic and comprehensive framework for studying mind-time interactions. The fuzzy theoretic consideration of time enables the fusion of temporal information with other quantitative criteria, therefore resulting in a particularly effective mechanism for studying temporal cognition in both artificial and biological systems.

Acknowledgment

The current work was partially supported by the EU FET Proactive grant (GA: 641100) TIMESTORM – Mind and Time: Investigation of the Temporal Traits of Human–Machine Convergence.

References

- Abdeddam, Y., Asarin, E., Gallien, M., Ingrand, F., Lesire, C., & Sighireanu, M. (2007). Planning robust temporal plans: A comparison between CBTP and TGA approaches. In *International conference on automated planning and scheduling, ICAPS*. .
- Boerkoel, J., & Durfee, E. (2013). Distributed reasoning for multiagent simple temporal problems. *Journal of Artificial Intelligence Research*, 47, 95–156.
- Casanova, G., Lesire, C., & Pralet, C. (2015). Managing dynamic multi-agent simple temporal network. In *Proc. 15th international conference on autonomous agents and multi-agent systems, AAMAS*. .
- Cashmore, A. (2003). Cryptochromes: Enabling plants and animals to determine circadian time. *Cell*, 114, 537–543.
- Chao, C. (2012). Timing multimodal turn-taking for human-robot cooperation. In *14th ACM international conference on multimodal interaction (ICMI 12)* (pp. 309–312).
- Coull, J., Van-Wassenhove, V., & Coslett, H. (2013). Special issue: How does the brain process time? *Neuropsychologia*, 51, 187–384.
- Dechter, R., Meiri, I., & Pearl, J. (1991). Temporal constraint networks. *Artificial Intelligence*, 61–95.
- Deng, Y., Chen, Y., Zhang, Y., & Mahadevan, S. (2012). Fuzzy dijkstra algorithm for shortest path problem under uncertain environment. *Applied Soft Computing*, 12, 1231–1237.
- Droit-Volet, S. (2013). Time perception in children: A neurodevelopmental approach. *Neuropsychologia*, 51, 220–234.
- Dubois, D., Fargier, H., & Prade, H. (1995). Fuzzy constraints in job-shop scheduling. *Journal of Intelligent Manufacturing*, 6, 215–234.
- Dubois, D., & Prade, H. (1978). Operations on fuzzy numbers. *International Journal of Systems Science*, 9, 613–626.
- Effinger, R., Williams, B., Kelly, G., & Sheehy, M. (2009). Dynamic controllability of temporally-flexible reactive programs. In *19th international conference on automated planning and scheduling (ICAPS 09)*. .
- Gerstner, J. (2012). On the evolution of memory: A time for clocks. *Frontiers in Molecular Neuroscience*.
- Gombolay, M., Wilcox, R., & Shah, J. (2013). Fast scheduling of multi-robot teams with temporospatial constraints. In *Proceedings of the robots: Science and systems (RSS)*. .
- Grondin, S. (2010). Timing and time perception: A review of recent behavioral and neuroscience findings and theoretical directions. *Attention, Perception, & Psychophysics*, 72, 561–582.
- Hoffman, G. (2013). Evaluating fluency in human-robot collaboration. In *International conference on human-robot interaction (HRI), workshop on human robot collaboration*. .
- Khadar, B. A. R., Madhusudhan, R., Ramanaiah, M., & Karthikeyan, K. (2013). Statistical optimization for generalised fuzzy number. *International Journal of Modern Engineering Research*, 3, 647–651.
- Maniadakis, M., & Trahanias, P. (2014). Time in consciousness, memory and human-robot interaction. In *From animals to animats 13 (SAB 13)* (pp. 11–20).
- Morris, P. (2014). Dynamic controllability and dispatchability relationships. In *Integration of AI and OR techniques in constraint programming. Lecture notes in computer science* (Vol. 8451, pp. 464–479). Springer International Publishing.
- Morris, P., Muscettola, N., & Vidal, T. (2001). Dynamic control of plans with temporal uncertainty. In *Int joint conf on AI (IJCAI)*. .
- Navon, D. (1978). On a conceptual hierarchy of time, space and other dimensions. *Cognition*, 6, 223–228.
- Paranjpe, D., & Sharma, V. (2005). Evolution of temporal order in living organisms. *Journal of Circadian Rhythms*, 3.
- Ridouard, F., & Richard, P. (2006). Worst-case analysis of feasibility tests for self-suspending tasks. In *Proc. of the 14th real-time and network systems* (pp. 15–24).
- Shah, J., Stedl, J., Williams, B., & Robertson, P. (2007). A fast incremental algorithm for maintaining dispatchability of partially controllable plans. In *Proc. 18th int conf on automated planning and scheduling*. .
- Shah, J., & Williams, B. (2008). Fast dynamic scheduling of disjunctive temporal constraint networks through incremental compilation. In *Proc. 19th int. conf. on automated planning and scheduling*. .
- Smith, S., Gallagher, A., Zimmerman, T., Barbulescu, L., & Rubinstein, Z. (2007). Distributed management of flexible times schedules. In *Int conf on autonomous agents and multiagent systems (AAMAS)*. .
- Vila, L., & Godo, L. (1994). On fuzzy temporal constraint networks. *Mathware and Soft Computing*, 3, 315–334.
- Yan, Z., Jouandeau, N., & Cherif, A. A. (2013). A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10.