Planning and Plan-execution for Human-Robot Cooperative Task achievement

Samir ALILI and **Matthieu WARNIER** and **Muhammad ALI,** and **Rachid ALAMI**
CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France.
Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France.
Email: firstname.lastname@laas.fr

Abstract

Human Robot cooperation brings several challenges to autonomous robotics such as adoption of a pro-active behavior, situation analysis and goal generation, intention explanation and adaptation and choice of a correct behavior toward the human. In this paper, we describe a decisional architecture for human robot interaction which addresses some of these challenges. The description will be centered on a planner called HATP (Human Aware Task Planner) and its capacity to synthesize plans for human robot teamwork that respect social conventions and that favour acceptable collaborative behaviors. We provide an overall description of HATP and its integration in a complete implemented architecture. We also illustrate the performance of our system on a daily life scenario achieved by a robot in interaction with a human partner in a realistic setup.

INTRODUCTION

The capacity to make decisions and to synthesize feasible plans that achieve the desired goals is necessary for the autonomy in robotics but is not sufficient to produce a "satisfactory" behavior. Human robot interaction requires more competence from robot like integration of social behaviors (Breazeal et al. 2005), exhibition of its intentions, intention reading (Croft 2003; Hofmann and Williams 2007), adoption of pertinent pro-active behaviors (Schmid, Weede, and Wörn 2007).Another relevant contribution is the planner developed by (Cirillo, Karlsson, and Saffiotti 2009) take into account explicitly the presence and the action of humans.

The paper consists in two parts. The first one deal with a planner called HATP (for Human Aware Task Planner) which has been specially designed for the synthesis of plans for a robot in presence or even in interaction with humans. It is based on a hierarchical task planning approach (Ghallab, Nau, and Traverso 2004) that integrates "social behavior rules" (Fong, Nourbakhsh, and Dautenhahn 2003), in order to favour the synthesis of socially acceptable plans. The second part explains the planner integration and utilization in the context of domestic activities where a robot is supposed to assist a human while respecting social rules. We describe the system architecture and give a short description of the components involved in the decisional layer. Then, we illustrate its use through several examples involving a mobile robot behaving proactively and acting in interaction with a person. Finally, we conclude and discuss future extensions.

Human Aware Task Planner - HATP

HATP is designed to synthesize plans for a robot that might have to achieve goals in a an environment where other "agents" are present (essentially humans). The robot will have to produce a plan taking into account its capacities and the current state of the environment. What makes HATP different is that it allows the robot to take into account also the state and the capacities of the other agents and even their preferences.

The result of a planning request is a plan with several synchronized streams, the stream of the robot as well as a set of streams that are devoted to anticipate the future potential actions of the other agents. Besides, we request that this plan exhibit some properties like the satisfaction of a set of so-called "social rules".

HATP data structure

HATP is based on a hierarchical task planning (Nau et al. 2003). We give here below a brief description of the representation of the world state and of the tasks.

World database In HATP the world is represented by a set of entities $Wb = \{En_1, En_2, En_3, ..., En_n\}$. Each entity is unique and is defined by a set of attributes that are similar to predicates or state variables in classical planning. Attributes can be static/dynamic and of type atom/vector. A static attribute represents an non-modifiable information about an entity whereas a dynamic one can be modified. An attribute of type atom can have only one value at a time whereas an attribute of type vector can store several values at the same moment. Figure 1 illustrates an example of an agent representation in HATP.

Tasks In HATP, similarly to HTNs, we can distinguish two types of tasks: Operators and Methods. An operator (also called action) is an atomic task. A method is a higher level task that must be decomposed to be realized.

Dynamic	Atom	Human.	posTopo	=	Position;
Dynamic	Atom	Human.	mood	=	numeric;
Dynamic	Atom	Human.	maxObjects	=	Numeric;
Static	Atom	Human.	type	=	String;
Dynamic	Atom	Human.	AgreecommunicationWithRobot	=	Boolean;
Dynamic	Atom	Human.	AgreePro-activity	=	Boolean;
Static	Atom	Human.	needGlasses	=	Boolean;
Dynamic	Atom	Human.	full	=	Boolean;
Dynamic	Atom	Human.	concerned	=	Boolean;
Dynamic	Vector	Human.	object	=	{list of hold object}

Figure 1: **HATP Human model:** In this example, the entity HUMAN is described by a set of attributes: for instance, his symbolic position, the degree of his desire to be involved in a task (mood), a boolean that indicates if the human allows the robot to be pro-active, the fact that he needs glasses or not, the list of objects that his is currently holding, etc.

Operators An operator is defined as a tuple $Op =< Name, Precd, E, C, Du >$ where *Name* is a name of the action, *Precd* is a precondition formula, E is the effect on the world after action execution, C is a positive cost function of action execution; it depends on the agent and on the context and Du is the duration of action execution. It also depends on the agent and on the context. The generic syntax is:

```
Action <action name>(<list of Entity>)
{
    preconditions {...};
    effects{...};
    cost{<cost_function_name>};
    duration{<duration_function_name>};
}
```

Methods are used by the planner to decompose a high-level task until it reaches atomic tasks (Operator). Methods are defined by pairs $M =< M\_goal, D >$ where M_goal represents a formula that specifies the goal. $D = \{D_1, \ldots, D_n\}$ is a set of alternative decompositions. Each decomposition D_i is a tuple $D_i =< Precd_i, TL_i, C_i >$ if the precondition $Precd_i$ is true the decomposition D_i is applicable and can be accomplished by performing the sub-tasks TL_i according to the constraints C_i which are generally precedence constraints.The generic syntax is:

```
method <method name>(<list of parameters>)
{
    M_goal{...};
    {
        preconditions {...};
        subtasks {...};
    }
    {
        preconditions {...};
        subtasks {...};
    }
    ...
}
```

C. Social rules

As mentioned above the consideration of abilities and preferences of agents is not sufficient to be sure that the planner will produce socially acceptable plans. We have defined a set of properties that a "good" plan should exhibit: "social rules" as a convention which associates penalties to a behaviour of an agent in a given context. In the current implementation of HATP, six types of rules have been identified:

Undesirable states correspond to world states which are dangerous, unpleasant (to humans) or inadequate.

Undesirable sequences correspond to some action combinations that can conduct to an uncomfortable feeling for robot partners.

Bad decompositions is a rule used to promote some choices at a high level of decomposition. The idea is some task decompositions could be preferred to others because they correspond, for instance, to more "legible" behaviour.

Effort balancing is used to establish a balance of effort among partners.

Timeouts is used to prevent long waiting periods of time between two actions done by a same agent.

Intricate synchronisation links between plan streams should be avoided; Indeed, such links make plans fragile and introduce a lot of dependency between agents.

D. HATP Algorithm

HATP planning process is composed of two threads. One thread is responsible for plan refinement (Montreuil, Clodic, and Alami 2007) and a second thread deals with plan evaluation. HATP plan refinement is inspired from SHOP2 procedure (Nau et al. 2003). The main differences between them are:

- HATP starts planning from a single task or from a tree of tasks where some choices have already been done
- HATP reduces time exploration by making the assumption that if two tasks are "parallel" (i.e. they do not have causal link between them), it will try to obtain a final plan where they are as independent as possible (for more details, refer to (Montreuil, Clodic, and Alami 2007)).
- HATP implements a multi-criterion plan evaluation .

Plan evaluation uses a metric based on the Decision by Objectives theory, more precisely The Analytic Hierarchy Process (AHP)(Forman and Selly 2001). It is a generic method (Saaty 2000; 1999), designed to facilitate decision making based on several heterogeneous criteria in complex environments. It decomposes the decision problem into a hierarchy of more easily comprehended sub-problems where each sub-problem can be independently analyzed.

Decisional architecture for Human Robot Interaction

We have devised a control architecture dedicated to robot decision and action in a human context (Figure 2). It has been developed as an instance of the generic the LAAS Architecture for Autonomous Systems (R.Alami et al. 1998).

The decisional layer consists of two components:

- **HATP**: the task planner.

- **SHARY** which constitutes the decisional kernel. It is based on an incremental context-based task refinement in a human context. It uses two complementary subcomponents: the **Task Agenda**, a mechanism for robot higher level goal management based on **CRS**, a chronicle recognition system (Ghallab 1996; Dousson and Maigat 2007).

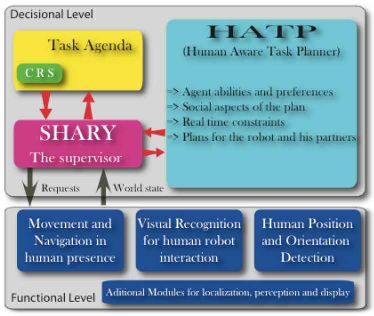


Figure 2: A control Architecture for a proactive robot assistant.

SHARY : The supervision and execution system

Shary (Clodic 2007; Clodic et al. 2008) originality, as a supervision system, lies in its ability to take into account not only the task achievement but also communication and monitoring needed to support interactive task achievement in a flexible way. SHARY allows to define a task or a hierarchy of tasks linked to more or less elaborated "communication policies" that enable to execute tasks given the possibility to deal with contingencies that could occur during its execution (or even to stop the task in case of unexpected events). This allows the robot to select a behaviour that is not only appropriate to achieve a task but also to monitor, for instance, human commitment to a task.

A communication scheme, for a given joint task, represents all possible turn taking steps and synchronization between the robot and its human partner (Clodic et al. 2007). Each time a state is visited the corresponding task recipe or atomic task is launched.

From a practical point of view, a communication scheme is a finite state automaton. Its states are communication acts expressed by the robot through dialog or by an expressive motion. Its transitions are communication acts directly expressed by the human or inferred from her/his behavior by monitoring tasks.

We have defined some generic communication schemes with a defined set of communication acts that are mandatory in the framework of task achievement (A.Clodic et al. 2007). This set takes inspiration from Joint Intention Theory (Cohen and Levesque 1991) that states that each partner should be informed of the beginning, realization and ending of a joint task.

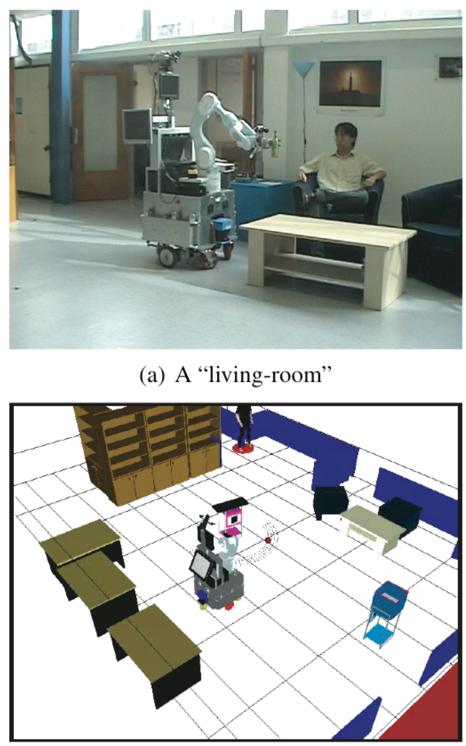
While executing a specific task this generic communication acts will be instantiated as an act_X_task with a recipe, an execution state, etc. For example, when the robot is proposing to give the human an object, it is realizing the act_X_task defined by the `Give Object` task and the `ASK-TASK` act.

Task Recipe: Task recipes are methods that compute the partially ordered list of subtasks of an act_X_task. This subtask tree contains both a set of tasks needed for the achievement of the act_X_task but also a list of tasks required for monitoring the execution. Recipes can be scripts, i.e. provided by the programmer, or can be synthesized by a planner such as HATP.

Toward Proactivity: Using the system proposed so far, it is possible to endow the robot with the capacity to achieve goals based on fetch-and-carry abilities in order to assist humans. Goals can be given to the robot by a human or instantiated proactively by the robot.

EXPERIMENTS

The system has been implemented and integrated on a fully equipped mobile manipulator called Jido and tested in the experimental environment, shown in figure 3.



(a) A "living-room"

(b) 3D model used by Jido the robot

Figure 3: Robot working environment.

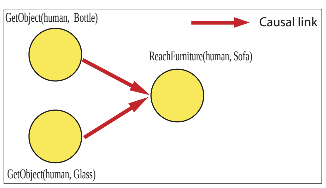
It simulates a living-room with different objects of everyday human life (tables, chairs,. . .). The robot goal is to assist human in his daily life, for this the robot able to perform a number of tasks Serving, Cleaning and maintaining an updated knowledge of the state of the world (detecting and tracking people in the room, detecting and recognizing new objects placed on tables by the persons. . . .).

Besides decisional capabilities that we have discussed, our robot has a complete set of functional capabilities such as navigation (Sisbot et al. 2007b), manipulation (Sisbot et al. 2007a), perspective placement (Marin, Sisbot, and Alami 2008) in presence of humans as well as perception of human activities (Burger, Ferrane, and Lerasle 2008) and face tracking(Germa et al. 2007). The challenge for the system is to perform these tasks showing proactive robot behavior and also interleaving it with social behavior when interacting with the human. We present, in the sequel, illustrative examples of Jido capabilities. A living room furnished environment (Coffee Table, Cupboard Table, chairs) and objects (Glasses for reading, Books, Bottles, Glasses todrink) is the setting for the examples.

Example 1: "Fetch-And-Carry" scenario

In this scenario we will see the capacity of HATP to synthesize plans for human robot teamwork while respecting social conventions and adopting acceptable collaborative behaviors. The planner takes into account the abilities and preferences of each agent (and more particularly the human) to produce pertinent plans.

Here, Jido goal is to satisfy the human desire to drink. The goal is described as a list of tasks (GetObject(human, Bottle), GetObject(human, Glass) and ReachFurniture(human, Sofa)) as shown in figure 4.



a.

```
node
{
  1:GetObject(human, Bottle);
  2:GetObject(human, Glass);
  3:ReachFurniture(human, Sofa) >1, >2;
}
```

b.

Figure 4: **HATP goal description.**: a.) graphical representation. b.) textual representation with > representing the precedence constraints

In this scenario the human in on the sofa, Jido is at its base, the glasses are on the table and the bottle is in the closed cupboard.

Figure 5.a illustrates the plan produced by HATP when social rules are inhibited. We can note that this plan is correct but it has a number of defects: the human opens the cupboard for the robot, the robot puts the bottle on the sofa and then the human picks it up. Now, if we introduce the following set of rules (avoid to have a bottle or glass on the sofa, avoid to leave the cupboard open, avoid awkward sequences such that robot puts down an object and human picks it up just after, avoid open/close sequences performed by different agents), we can see in Figure 5.b, that most defects have been corrected. However, this is not sufficient because we can observe that the human moves from the sofa to the table and comes back only for getting the glass, and there is a risk to have him wait for the robot to achieve its own part of the plan. If we introduce rules that try to avoid intricate synchronization between the robot and the human we observe (Figure 5.c) that all defects have been corrected.

Example 2: proactively ask human help to achieve a robot task

Through this scenario, we would like to show the ability of the system to take initiative and ask human help to remove an ambiguity or to escape from a blocked situation. The task consists in Jido cleaning the living-room table by picking up the bottle and throwing it into the trash.

Initially HATP produces a plan where Jido can do the task by itself. Shary executes the plan. It starts by execute the task "moveForManipulation" while ensuring that the movement is safe for the human (Sisbot et al. 2007a). The next task to perform is "pickup". Sometimes the object is unreachable, in this situation Shary sends a new request to HATP, which produces a new plan where human will reach the bottle and give it to Jido. Shary analyzes and detects that it is a double stream plan one for human and the other for robot. Jido asks human if the human agrees with this plan, and if human accepts, Shary starts the execution. Otherwise Shary informs human that there is no solution.

Figure 6 illustrates the different steps.

CONCLUSION AND FUTURE WORK

We have presented a robot task planner called HATP that has been designed for interactive and collaborative robotic applications. We have illustrated its ability to produce so-called "socially acceptable plans" for several agents while taking into account the agents abilities, preferences and desires and respecting social aspects related to the interaction of a robot with humans. We have seen its integration in a complete robot control architecture for a proactive robot assistant.

We have show through implemented examples various aspects of the system scenario recognition, planning based initiative taking and execution through HRI dedicated robot supervisor.

Deciding on whether to take initiative and whether to ask permission or inform about initiative taking is not easy. These issues were simplified here. The robot always took initiative if there are some plans involving robot and always asked for permission before execution.

Concerning future work, several aspects will be investigated. The first one is the development of heuristics in the refinement process in order to explore the most promising parts of the solution space and to allow the use of HATP in more complex situations with a more elaborate model of its activities.

In the current model all agents are assumed to have the same knowledge of world state. This is clearly a simplification that does not correspond to the reality. It would be interesting to distinguish the agents knowledge and to act, when necessary to inform or to resolve ambiguous situations. Another aspect which is complementary to the preceding is to establish a principled link between symbolic task planning and geometric planning. This would allow to

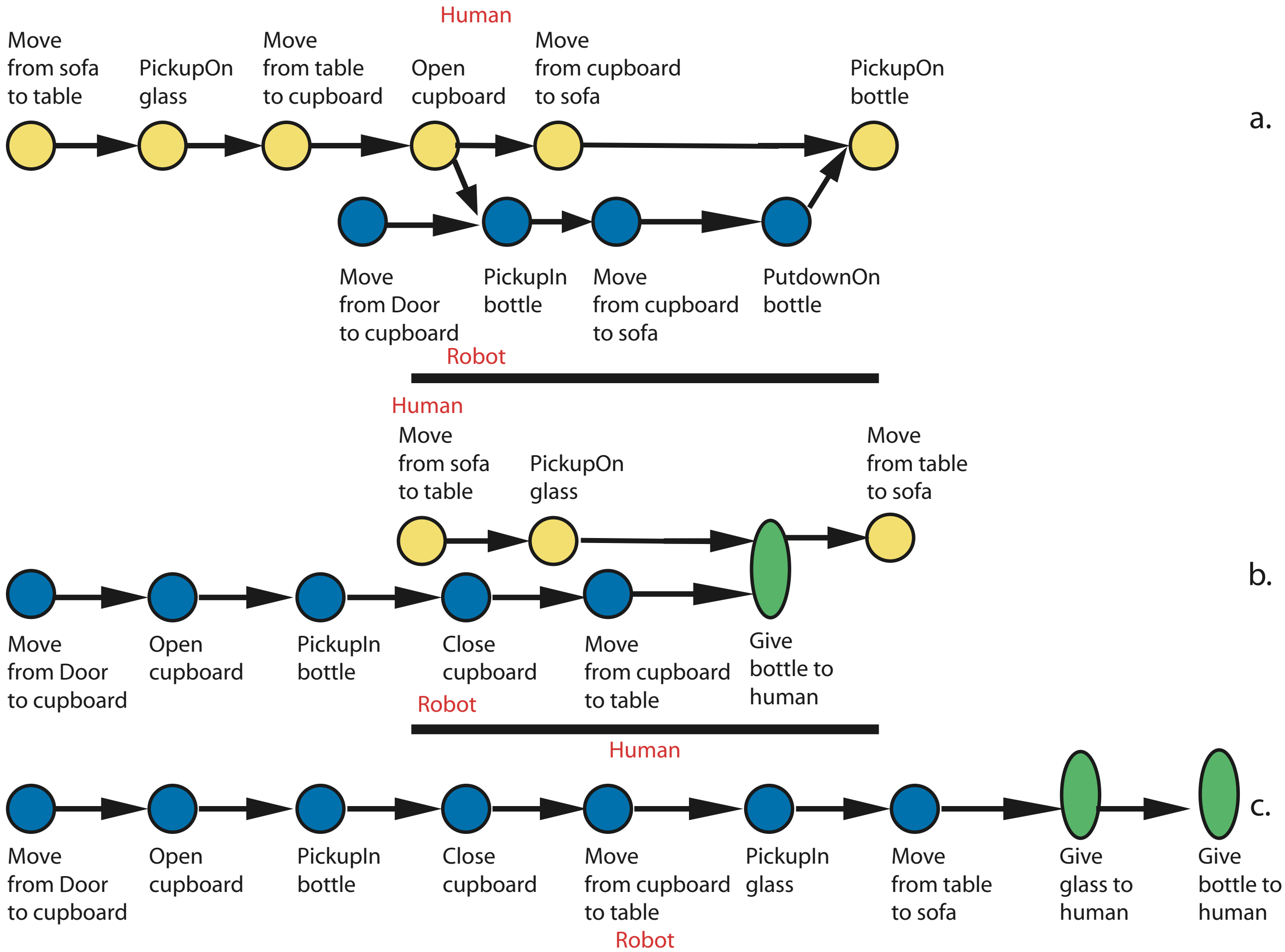


Figure 5: **Example 1 (Effect of social rules on the plans)**: the yellow circles represent human actions, the blue circles represent robot actions and the green ellipses represent joint actions.

build far more pertinent robot behaviours and less vulnerable symbolic plans.

Acknowledgments: The work described here has been partially conducted within the EU Project CHRIS (Cooperative Human Robot Interaction Systems) funded by the E.C. Division `FP7-IST` under Contract `215805`. We also acknowledge support of Agence Nationale de la Recherche ANR (AMORCES/PSIROB project).

References

A.Clodic; R.Alami; V.Montreuil; and et al. 2007. A study of interaction between dialog and decision for human-robot collaborative task achievement. In *16th IEEE RO-MAN*.

Breazeal, C.; Buchsbaum, D.; Gray, J.; Gatenby, D.; and Blumberg, B. 2005. Learning from and about others: Towards using imitation to bootstrap the social understanding of others by robots. *Artif. Life* 11(1-2):31–62.

Burger, B.; Ferrane, I.; and Lerasle, F. 2008. Multimodal interaction abilities for a robot companion. In *ICVS*.

Cirillo, M.; Karlsson, L.; and Saffiotti, A. 2009. Human-aware task planning for mobile robots. *in the 14th International Conference on Advanced Robotics ICAR09*.

Clodic, A.; Ransan, M.; Alami, R.; and Montreuil, V. 2007. A management of mutual belief for human-robot interaction. In *IEEE SMC*.

Clodic, A.; Cao, H.; Alili, S.; Montreuil, V.; Alami, R.; and Chatila, R. 2008. Shary: a supervision system adapted to human-robot interaction. *11th International Symposium on Experimental Robotics 2008, ISER 2008*.

Clodic, A. 2007. *Supervision pour un robot interactif : Action et Interaction pour un robot autonome en environnement humain*. Ph.D. Dissertation, University of Toulouse.

Cohen, P. R., and Levesque, H. J. 1991. Teamwork. *Nous* 25(4):487–512.

Croft, D. K. E. A. 2003. Estimating intent for human-robot interaction. *In IEEE International Conference on Advanced Robotics* 810–815.

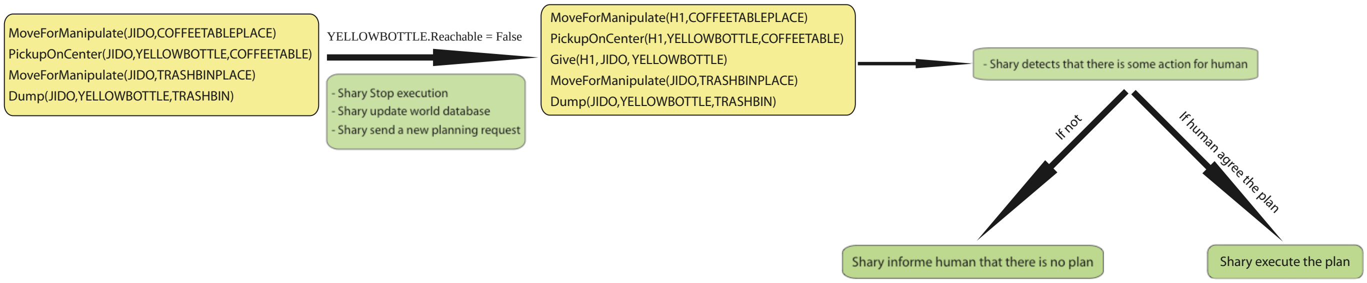


Figure 6: **Example 2 (In difficulty ask Help, if possible):** `clean-up` task execution: The yellow square represents HATP plan's, the green square represent decision and action. The first plan failed due to robot inability to take the bottle (even when it has perceived it), then Shary asks a new feasible plan. HATP finds a plan with a higher cost and two streams and where the person is requested to participate by giving the bottle to Jido. Shary asks human if he agrees with the now plan. if the answer is positive, Shary starts execution of plan else it informs the human that there is no plan.

Dousson, C., and Maigat, P. L. 2007. Alarm driven supervision for telecommunication network : Ii- on-line chronicle recognition. In *In proc. of the 20th International Conference on Artificial Intelligence (IJCAI-2007).*, 324–329.

Fong, T. W.; Nourbakhsh, I.; and Dautenhahn, K. 2003. A survey of socially interactive robots. *Robotics and Autonomous Systems*.

Forman, E., and Selly, M. A. 2001. *Decision By Objectives*. World Scientific.

Germa, T.; Brèthes, L.; Lerasle, F.; and Simon, T. 2007. Data fusion and eigenface based tracking dedicated to a tour-guide robot. *ICVS*.

Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated Planning - theory and practice*. Morgan Kaufmann Publishers.

Ghallab, M. 1996. On chronicles: Representation, online recognition and learning. In *In Luigia Carlucci Aiello, Jon Doyle, and Stuart Shapiro, editors, Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning*, 597–607.

Hofmann, A. G., and Williams, B. C. 2007. Intent recognition for human-robot interaction. *Interaction Challenges for Intelligent Assistants AAAI 2007 Spring Symposium* 60–61.

Marin, L.; Sisbot, E. A.; and Alami, R. 2008. Geometric tools for perspective taking for human-robot interaction. In *Mexican International Conference on Artificial Intelligence (MICAI 2008)*.

Montreuil, V.; Clodic, A.; and Alami, R. 2007. Planning human centered robot activities. In *IEEE Int. SMC*.

Nau, D.; Au, T. C.; Ilghami, O.; Kuter, U.; and et al. 2003. Shop2: An htn planning system. *Journal of Artificial Intelligence Research*.

R.Alami; R.Chatila; S.Fleury; M.Ghallab; and F.Ingrand. 1998. An architecture for autonomy. *Internatonal Journal of robotics Research, Special Issue on Integrated Architectures for Robot Control and Programming* 17(4).

Saaty, T. L. 1999. *The Analytic Hierarchy Process for Decisions in a Complex World*. RWS Publications.

Saaty, T. L. 2000. *Fundamentals of the analytic hierarchy process*. RWS Publications.

Schmid, A. J.; Weede, O.; and Wörn, H. 2007. Proactive robot task selection given a human intention estimate. *Robot and Human interactive Communication, RO-MAN 2007. The 16th IEEE International Symposium* 726–731.

Sisbot, E. A.; Urias, L. F. M.; Alami, R.; and Siméon, T. 2007a. Spatial reasoning for human-robot interaction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*.

Sisbot, E.; Marin-Urias, L.; Alami, R.; and Simeon, T. 2007b. A human aware mobile robot motion planner. *IEEE Transactions on Robotics* 23(5):874–883.