Towards human-aware robot motions

Emrah Akin Sisbot

To cite this version:

Emrah Akin Sisbot. Towards human-aware robot motions. Automatic. Université Paul Sabatier - Toulouse III, 2008. English. NNT : . tel-00343633

HAL Id: tel-00343633

https://theses.hal.science/tel-00343633

Submitted on 2 Dec 2008

THESE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par *l'Université Toulouse III – Paul Sabatier*
Discipline ou spécialité : *Informatique*

Présentée et soutenue par *Emrah Akın ŞİŞBOT*
Le *17 Octobre 2008*

Towards Human-Aware Robot Motions

JURY

M. Jean-Pierre JESSEL	*Université Paul Sabatier, Toulouse*	*Président*
M. Michael BEETZ	*Technische Universität München*	*Rapporteur*
M. Christian LAUGIER	*INRIA, Grenoble*	*Rapporteur*
M. Thierry SIMÉON	*LAAS-CNRS, Toulouse*	*Examinateur*
M. Pierre-Yves OUDEYER	*INRIA, Bordeaux*	*Examinateur*
M. Rachid ALAMI	*LAAS-CNRS, Toulouse*	*Directeur de Thèse*

Ecole doctorale : *EDMITT*
Unité de recherche : *LAAS-CNRS*
Directeur(s) de Thèse : *Rachid Alami*

Abstract

The introduction of robots in our daily lives raises a key issue that is “added” to the “standard challenge” of autonomous robots: the presence of humans in the environment and the necessity to interact with them. In the factory, the robot is physically separated and a security distance is always kept from human workers. With this separation, the primary concern in such environments, the “safety”, is ensured. However this separation cannot be applied to future applications where the robot will be in a situation where it will have to assist humans. In a scenario where the robot has to move among people, the notion of safety becomes more important and should be studied in every detail.

Yet the biggest difference in these two environments does not come from the definition of their primary concern, the safety, but comes from a secondary concern. In factory, when the safety is ensured, the feasibility of the task gains in importance. The robot’s environment is perfectly structured and all the robots are perfectly coordinated in order to accomplish their tasks. On the contrary, the feasibility of the task leaves its place to the “comfort” for an interactive robot. For a robot that physically interacts with humans, accomplishing a task with the expense of human comfort is not acceptable even the robot does not harm any person.

The robot has to perform motion and manipulation actions and should be able to determine where a given task should be achieved, how to place itself relatively to a human, how to approach him/her, how to hand the object and how to move in a relatively constrained environment by taking into account the safety and the comfort of all the humans in the environment.

In this work, we propose a novel motion planning framework answering these questions along with its implementation into a navigation and a manipulation planner. We present the Human-Aware Navigation Planner that takes into account the safety, the fields of view, the preferences and the states of all the humans as well as the environment and generates paths that are not only collision free but also comfortable. We also present the Human-Aware Manipulation Planner that breaks the commonly used human-centric approaches and allows the robot to decide and take initiative about the way of an object transfer takes place. Human’s safety, field of view, state, preferences as well as its kinematic structure is taken into account to generate safe and most importantly comfortable and legible motions that make robot’s intention clear to its human partner.

Acknowledgements

First of all, I would like to express my gratitude to my supervisor, Rachid Alami, whose expertise, understanding, patience, and good humor added considerably to my graduate experience. Thank you Rachid, I would surely be lost in the vast sea of research without your guidance.

I would like to thank Michael Beetz and Christian Laugier for taking time out from their busy schedules to review my thesis. I would also like to thank all the members of the jury for accepting to take part in this once in a lifetime experience of mine.

I also would like to thank Raja Chatila, Director of LAAS-CNRS, for accepting me in his robotics group. And I am deeply grateful to Demir Kısakürek, Raset Soyer and SAJEV for the trust and financial support that they gave me in order to study in France.

I also would like to thank Ronald Arkin for the valuable discussions that we had during his stay in Toulouse.

This thesis could not have been accomplished without the assistance of many people whose contributions I gratefully acknowledge; Thierry Siméon and Juan Cortés for sharing their great knowledge in motion planning, Matthieu Herrb and Sara Fleury for providing their expertise to my “fragile” software, UH team for providing valuable inputs for this work, Cogniron@LAAS team for their tremendous efforts to make things work (and also for chocolatines), and Luis Marin for contributing this work from many “perspectives” and for tolerating my occasional bad temper.

I would like to express my warm thanks to Claudia, Nacho, Aurélie, Efrain, Gustavo, Thierry, Raquel, Teresa, Mathias, Vincent, Joan, Matthieu, Gaelle, Xavier, Mokhtar and many others for the friendly environment they create, making my Ph.D not an obstacle but a pleasure.

I also would like to thank Martine Labruyère, secretary of EDMITT, for lifting off the weight of bureaucracy from my shoulders.

I cannot go on without mentioning my friends in Istanbul. So I would like to thank my old and best friends, Ekin, Daghan, Ozan, Arzu and Ufuk for allowing me to empty my mind and enjoy my vacations in Istanbul. And also a big thanks to my brother Arda who seems to follow my steps, thus forcing me to be a good example.

Finally, I would like to thank my parents, Nüket and Sedat Şişbot, to whom I cannot overstate my gratitude and without whom none of this would have been even possible. To them I dedicate this thesis.

CONTENTS

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER ONE

Introduction

Robots, articulated machines, interacting with people have always been fascinating and drawn a lot of attention in every periods of history. Even though the term “robot” is first used in 1920, the examples of the idea of articulated machines, behaving, acting and moving like humans, date back to 1000BC. With the advances in science and technology, “automata” interacting with people had begun to appear in late 18th century. Although the early examples, like the chess playing automaton “the Turk”, were technologically far from the modern robot, the ideas that they were relied upon are still used in modern robotic (in case of “the Turk”, which was a chess playing humanoid with the help of a person hidden in its mechanical structure, the modern “Wizard of Oz” technique [Kelley 83] and the working of this “robot” show many conceptual similarities).

With the advances in technology in the last century, the early idea of robots “living” among people has begun to become a reality. The concept of “autonomy” has come to surface by allowing the robots to perceive, reason, act and “exist” by their own means. The notion of having autonomous robots in our daily lives raised also new questions and opened a whole new chapter in robotics research by giving birth to HRI, Human-Robot Interaction, research field.

HRI is a very fast growing field where the research goes in many directions towards a common goal: a robot that will perceive its environment, reason on the situation and act in a safe and comfortable way to facilitate people’s lives. The research in perception, in HRI point of view, is focused on creating and improving robot’s ability to “see” its environment, locate and identify objects and humans. The reasoning part of the field is mainly focused on developing robot abilities that will allow to decide “what to do” and “how to do” in a given situation. The acting part of the field, on the other hand, is towards a perfect execution of the results of robot reasoning and an ideal design of robot structure.

This thesis is placed in the reasoning part of the field by asking the question of “how the motion of the robot should be influenced by the fact that it acts in presence, in vicinity or even in close collaboration with humans?” and “how human and robot should share the space?” by proposing motions planning methods and algorithms that generate robot motions by not only reasoning on the environment but also reasoning explicitly on humans, thus allowing the robot to move in a safe and "comfortable" way.

1.1 Problem Statement

The introduction of robots in our daily lives raises a key issue that is "added" to the "standard challenge" of autonomous robots: the presence of humans in the environment and the necessity to interact with them. In the factory (figure 1.1.a), the robot is physically separated and a security distance is always kept from human workers. With this separation, the primary concern in such environments, the "safety", is ensured. However this separation cannot be applied to future applications[1] where the robot will be in a situation where it will have to assist humans. In a scenario where the robot has to move among people (figure 1.1.b), the notion of safety becomes more important and should be studied in every detail.

(a) Car assembly robots in Rover factory.

(b) A home environment.

Figure 1.1: *Two different usages of robots. The notion of safety gains more importance in a home environment, where the robot is in close proximity of humans, than a factory where the robots are completely isolated.*

Yet the biggest difference in these two environments does not come from the definition of their primary concern, the safety, but comes from a secondary concern. In factory, when the safety is ensured, the feasibility of the task becomes the most important concern (e.g. in figure 1.1.a's case, it is the assembly of a car). The robot's environment is perfectly structured and all the robots are perfectly coordinated in order to accomplish their tasks. On the contrary, the feasibility of the task leaves its place to the "comfort" for an interactive robot. For a robot that physically interacts with humans, accomplishing a task with the expense of human comfort (or "mental safety"[2]) is not acceptable even the robot does not hit any person. It is preferable that the robot gives a failure for a task even if it exists a way of doing it by causing fear/surprise/discomfort.

So, a robot that will serve as a helper among humans should not only be a machine but it should respect social rules and protocols [Chatila 02][Fong 03] ensuring a:

---

[1] In the future, even in the factory, robots and humans will have to work together shoulder to shoulder.

[2] In some literature, e.g. [Nonaka 04], comfort is also though as part of safety and categorized as mental safety.

- *Safe motion*, i.e., that does not harm the human,
- *Socially acceptable motion*, i.e., that takes into account the comfort of the human as well as his preferences and needs[3],
- *Reliable and effective motion*, i.e., that achieves the task adequately considering the motion capacities of the robot.

The robot has to perform motion and manipulation actions and should be able to determine where a given task should be achieved, how to place itself relatively to a human, how to approach him/her, how to hand the object and how to move in a relatively constrained environment in the presence of humans (an apartment for instance).

In this work, we propose solution methods to these questions with a motion planning framework with its implementation into a navigation and a manipulation planner:

- We present the Human-Aware Navigation Planner that takes into account the safety, the fields of view, the preferences and states of all the humans as well as the environment and generates paths that are not only collision free but also comfortable.
- We present the Human-Aware Manipulation Planner that breaks the commonly used human-centric approaches and allows the robot to decide and take initiative about the way of an object transfer takes place. Human's safety, field of view, state, preferences as well as its kinematic structure is taken into account to generate safe and most importantly comfortable and legible motions that make robot's intention clear to its human partner.

This works is also a part of a more general approach that has been conducted by our team in collaboration with other universities involving user studies with specialists, high level decision and low level perception systems.

1.2 Contributions

This work, in our knowledge, is the first work to tackle the problem of motion planning in presence of humans by proposing a general framework.

Although not being the main contribution, this work provides a very focused survey on motion generation methods in human presence and human/human - human/robot interaction user studies. These two fields, that are often dispersed, are merged in a single pot for the design of the Human-Aware motion planners. This survey, that shows the importance of robot motion in human-robot interaction, is a contribution towards both communities.

In this work, we showed that there are constraints in the path and in the final configuration of the robot that are far richer than simple obstacle avoidance.

Human-Aware Navigation Planner is one of the resulting implementation of this work. In the design of this planner, we studied in detail the meaning of safety and comfort in a human-robot physical interaction. With the help of user studies, these notions are interpreted and represented as cost functions that are used as metrics for such an interaction. The notion of visibility is one of the novelties along with the

---

[3] When the intention of the robot is clear (legible) then it adds also to safety.

inclusion of human states and preferences to the planning stage. With the help of the PerSpective Placement mechanism, robot navigation problem changes from finding a point-to-point motion to finding "a motion for a task".

In the navigation, taking into account not only the person with whom the robot interact but all of the people in the environment left the standard approaches of motion generation in human presence and provided a larger view to the problem.

The second main contributions of this thesis are the design of Human-Aware Manipulation framework and resulting planner that break the standard, human-centric way of interaction and give the robot the opportunity to take initiative. In the design of this planner, the notions of safety and comfort are extended. The 3-stage algorithm to solve the problem of "handing over an object" to human is a novel approach that enables the robot not only perform the reaching motion but also decide where the physical interaction will take place.

Our approach transformed the general problem of finding a path from a configuration to another, to finding a path to accomplish a task where final configurations are found automatically. This approach made a contribution in the effective motion planning use.

The usage of Generalized Inverse Kinematics at the last stage of the manipulation planner allows the robot not to be restrained by the arm motion and gives the possibility to include more coordinated motions between different parts of the robot. With this method, when handing over an object, the robot not only moves its arm but also moves its whole body, including its head, to better express its intention, thus fulfilling the legibility needs of the motion.

Using grids and cost functions as the basis of both planners gives sufficient expandability and enables to introduce further aspects of human-robot interaction.

Finally, both planners are integrated into two robotics platform interacting with multiple input-output modules.

1.3 Organization of the Thesis

This thesis is organized in following chapters:

Chapter 2 presents a short survey on motion planning in human presence. User studies on how a robot should move among humans are presented and interesting results are showed. Motion generation methods taking into account explicitly the existence of human are categorized by their contexts and their methods. These methods are then briefly presented in order to cover selected literature.

Chapter 3 presents the Human-Aware Navigation Planner, one of the main contributions of the thesis. After an introduction to the motion planning and challenges in the field, the methods and algorithms to model the human and to plan a path are illustrated. The paths generated by this planner in different scenarios and in different environments confirm the goals of such a planner.

The Human-Aware Manipulation Planner, being the other main contribution, is described in Chapter 4. After the problem statement, the 3-stage algorithm is explained in detail and is illustrated in examples where a robot hands over an object to a person. Resulting paths generated by this planner fulfill safety, comfort and legibility requirements of the interaction.

Chapter 5 presents the integration of both planners into two robotic platforms. The internal architectures of the planner modules as well as the general architectures of the robots are described. The robots are run in different scenarios and the resulting robot motions are illustrated.

Finally, Chapter 6 concludes this thesis and gives various perspectives for future research.

1.4 Publications

The following publications are associated with this thesis:

1. Emrah Akin Sisbot, Aurelie Clodic, Rachid Alami, Maxime Ransan, Supervision and Motion Planning for a Mobile Manipulator Interacting with Humans, *ACM/IEEE International Conference on Human-Robot Interaction (HRI), Amsterdam, Netherlands, 2008.*
2. Emrah Akin Sisbot, Luis Felipe Marin and Rachid Alami, Spatial Reasoning for Human Robot Interaction, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Diego, USA ,2007.*
3. Emrah Akin Sisbot, Luis F. Marin-Urias, Rachid Alami and Thierry Siméon, A Human Aware Mobile Robot Motion Planner, *IEEE Transactions on Robotics, Volume: 23, Issue: 5, pp: 874-883, 2007.*
4. K.L. Koay, E. A. Sisbot, D. A. Syrdal, M.L. Walters, K. Dautenhahn and R. Alami, Exploratory Study of a Robot Approaching a Person in the Context of Handling Over an Object, *AAAI Spring Symposia, Palo Alto, California, USA, 2007.*
5. Emrah Akin Sisbot, Luis F. Marin Urias, Rachid Alami and Thierry Siméon, A mobile robot that performs human acceptable motion, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 2006.*
6. Rachid Alami, Raja Chatila, Aurelie Clodic, Sara Fleury, Matthieu Herrb, Vincent Montreuil, Emrah Akin Sisbot, Towards Human-Aware Cognitive Robots, *The Fifth International Cognitive Robotics Workshop (The AAAI Workshop on Cognitive Robotics, COGROB),Boston, Massachusetts, USA, 2006.*
7. E. Akin Sisbot, Aurelie Clodic, Luis F. Marin Urias, Mathias Fontmarty, Ludovic Brèthes and Rachid Alami, Implementing a Human-Aware Robot System, *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Hatfield, UK, 2006.*
8. R. Alami, A. Clodic, V. Montreuil, E. A.Sisbot and R. Chatila, Toward Human-Aware Robot Task Planning, *AAAI Spring Symposia, California, USA, 2006.*
9. K. Dautenhahn, M. Walters, S. Woods, K. Lee Koay, E. A. Sisbot, R. Alami, T. Siméon, How may I serve you? A robot companion approaching a seated person in a helping context , *ACM/IEEE International Conference on Human-Robot Interaction (HRI), Salt Lake City, Utah, USA, 2006.*
10. R. Alami, A. Clodic, V. Montreuil, E. A.Sisbot and R. Chatila, Task planning for human-robot interaction, *Smart Objects and Ambient Intelligence (sOc-EUSAI), Grenoble, France, 2005.*
11. E. A. Sisbot, R. Alami and T. Siméon, K. Dautenhahn, M. Walters, S. Woods, K. L. Koay and C. Nehaniv, Navigation in the Presence of Humans, *IEEE-RAS International Conference on Humanoid Robots (Humanoids), Tsukuba, Japan, 2005.*

CHAPTER
TWO

State of the Art

This chapter aims to provide a literature survey on the methods and algorithms for robot motions in human presence. Section 2.1 opens this chapter with an introduction to the notion of safety in hardware and control levels as well as the methods of classification of motion generation methods. In Section 2.2, we discuss user studies in literature that provide interesting notions and metrics for a robot navigating and manipulating in a human populated environment. This section precedes Section 2.3 where different human models are presented and discussed. The following section (Section 2.4) presents methods and algorithms for safe (and comfortable) robot base motions by obeying a use case scenario categorization. In contrast to the previous section, Section 2.5 categorizes human friendly manipulation literature by their approaches to the problem. Finally Section 2.6 ends this chapter with a brief conclusion and perspectives for following chapters.

2.1 Introduction

Human-Robot Interaction (HRI) is a young research field compared to classical robotics. The increase of computation power as well as the advances in mechanical hardware and robotics allowed researchers to invest more resources to this new field. These intellectual investments (as well as financial ones) resulted and still resulting notable advances on methods and algorithms allowing the robots to approach more and more to a co-existence with humans. Yet these improvements bring new questions and concerns about safety of the humans sharing the same environment with the robot.

These concerns leads the notion of "safety" to be studied in detail (e.g. [Alami 06]) and to be evaluated through user studies (e.g. [Haddadin 08]). These studies converge towards a common classification of safety strategies as illustrated in Table 2.1, originally introduced by Ikuta et al. in [Ikuta 03]. In this classification, safety strategies are divided into two main categories: design strategies, where safety is ensured by the mechanical hardware design of the robot; and control strategies, where safety is ensured by controlling the motion of the robot.

We can study these strategies by their effects on a possible collision between human and robot. To avoid a collision, in hardware level, there is no solution except to

Table 2.1: *Classification of Safety Strategies.*

		control strategy	design strategy
before collision	avoid collision	distance	
	minimize impact force	speed moment of inertia	weight
after collision	attenuation diffusion	stiffness	cover joint compliance surface shape

isolate the robot. We rule out this solution because in an interaction context, isolating the robot would result in removing physical interaction from the scenario. Collision avoidance can be managed in control level, by controlling the distance between robot and humans. In order to minimize the impact force and to reduce the damage as much as possible in case of a collision, the weight of the robot plays an important role in design strategies in addition to speed and moment of inertia, which are the main actors in control level. In case of a collision, the robot can still reduce the damage given to human by controlling its stiffness [De Santis 08, Zollo 05]. From a design point of view, safer designs [Zinn 04, Bicchi 04] (cover, surface, shape and joints) combined with fault tolerant approaches [Lussier 05] may minimize the consequences of hurting a person in case of collision.

Although hardware design is a very important step towards safer and more comfortable physical human-robot interactions, it stays out of the scope of this thesis. With the following sections, this chapter focuses on the methods and algorithms for navigation and manipulation in human presence as well as interesting notions and concepts of human-robot interaction that can be found in selected literature[1].

In this thesis, we will not present nor detail the classic methods and algorithms of motion planning. Although we use these methods, they are now widely known and many high quality books (e.g. [Latombe 91, Laumond 97, Lavalle 06, Laugier 07]), surveying these methods, exist in literature.

2.2 Spatial Relationship Between Human and Robot

In order to equip the robot with methods and algorithms for a safe and comfortable motion, we must understand what kind of social behavior and what kind of motions, are accepted by people. In a non interactive context, safety is often reduced to a non-collision constraint where robot's priority is not to collide (in a non destructive context). For a robot that interacts closely with humans, safety gains a wider meaning still having the non-collision condition as essential.

The notion of safety in an environment, where people and robots are not far from each other, must be studied and redefined in order to take into account not only the

---

[1] Of course, this chapter cannot and will not include all the work on control strategies for HRI. For a more general survey on HRI, we refer the reader to [Goodrich 07] and [Kemp 07].

collision freeness, but also the comfort and the naturalness of robot's movements. To find out what is a "safe" motion, user studies with (human-robot user studies) or without (human-human user studies) a robot are conducted.

2.2.1 How a mobile robot should place itself?

One of the major studies in human-human spatial placement behavior is conducted by Hall [Hall 66]. This study presented the "proxemics theory", where distances between people are categorized into four classes. These distances, named intimate, personal, social and public, provide spatial limits to different types of interactions. These distances are subject to change according to cultural differences. In Latin cultures, for instance, distances are smaller, and people tend to be more comfortable standing close to each other; in Nordic cultures, however, distances are greater. Proxemics also divides the space into three categories [Littlejohn 05, Low 03]: (1) Fixed-feature space which comprises things that are immobile, such as walls and territorial boundaries; (2) Semifixed-feature space which comprises movable objects, such as furniture; and (3) Informal space which comprises the personal space around the body, that travels around with a person as he/she moves, and that determines the personal distance among people.

The user study conducted by Yoda et al. [Yoda 95] made evidence that, in hallways, humans always move at constant velocity and their avoidance behavior can be approximated to a catenary. In [Pacchierotti 05, Pacchierotti 06a], Pacchierotti et al. described a user study where people in a hallway crossed a robot whose behavior is complying with the proxemics theory. The subjects in this user study evaluated the robot's behavior according to different robot speeds, "signaling distances" and "lateral distances". With the answers and reactions of four subjects, ideal values were found for these three parameters. Yet the validity of these values stays true only for those four subjects and cannot be easily generalized. Cultural differences as well as the context of interaction have an important effect on these distances and finding an optimal value valid for everybody is unlikely.

Another direction to understand how humans respond to robot motions is towards user studies on "robot approach". The goal of these studies is to understand people's expectations from the motions of an approaching robot and to determine common rules and protocols. Walters et al. [Walters 05a] studied relative human-robot distances. In this study, a number of children and adults are asked to approach the stationary PeopleBot$^{\text{TM}}$robot and to stop at a distance and at a point where they felt comfortable. This study showed that more than half of the children (53%) placed themselves in front of the robot at a distance varying from 1.5 m to 2 m (1.72 m in average) thus treating the robot as a social being. In contrast, 38% of adults approached the robot closer than 0.5 m treating the robot as a toy, an attraction. In the second part of this study the roles are inverted and the robot is allowed to approach freely to the people (adult subjects). Approximately 40% of subjects allowed the robot to approach right up until 0.5 m (limit set by the robot's safety system). An interesting result of this work is that people's spatial placement largely depends on how they see the robot (as a social being or as a toy), on how they evaluate robot's level of threat and on their personalities. This last item is studied in [Walters 05b] where a correlation between human-robot relative distances and human personalities is found. The results showed that the more creative and/or aggressive a subject rated him/herself, the closer he/she was likely to approach the robot. The authors also mention that the shape of the robot could have an important impact on approach distances because of the fact that, intuitively, a big manipulator arm is more threatening than an armless robot.

A follow-up study by Dautenhahn et al. [Dautenhahn 06] showed that people prefer the robot to approach from their left or right but not directly from their front (in the scenario sitting subjects evaluated robot's approach directions for a "bring object" task). In this study subjects evaluated the direct frontal approach as least comfortable by finding robot's motion threatening, aggressive or stopping too far away.

The spatial placement behavior given by people depends largely on the task and the context of the interaction. If the task requires a closer placement (e.g. handing an object) then people tend to tolerate the robot at closer distances unlike a task that has not such a requirement (e.g. talking). In a recent study conducted by Yamaoka et al. [Yamaoka 08], test subjects are asked to place themselves at a comfortable position to communicate with a robot whose duty is to show and present and object. The average human placement for this specific task was 1.19 m far from the robot. And since the task also focused on an object, people stayed at $\approx$1 m far from the object. The difference in spatial placement caused by the "context of interaction" (the scenario, the task and its related requirements) can be clearly seen if we compare the previous study to the study on robot approaching to a person in the context of handing an object by Koay et al. [Koay 07]. In this study, sitting test subjects were asked to decide a comfortable position for a robot whose task is to bring a can. The resulting value was an average of 0.67 m of distance from the robot. These two studies show a clear difference of robot's placement on human's comfort depending on the task.

2.2.2 Motion of a Mobile Manipulator

As commented previously, the robot's shape and the task to be performed are two important parameters that affect the placement of interaction. The shape of the robot is also linked to the task because of the fact that a task often requires a specific hardware component (e.g. a fetch-and-carry task requires a robotic arm, a navigation task, on the other hand, requires a mobile base). This link allows a person to predict the task and also to anticipate robot's abilities in order to evaluate the level of threat.

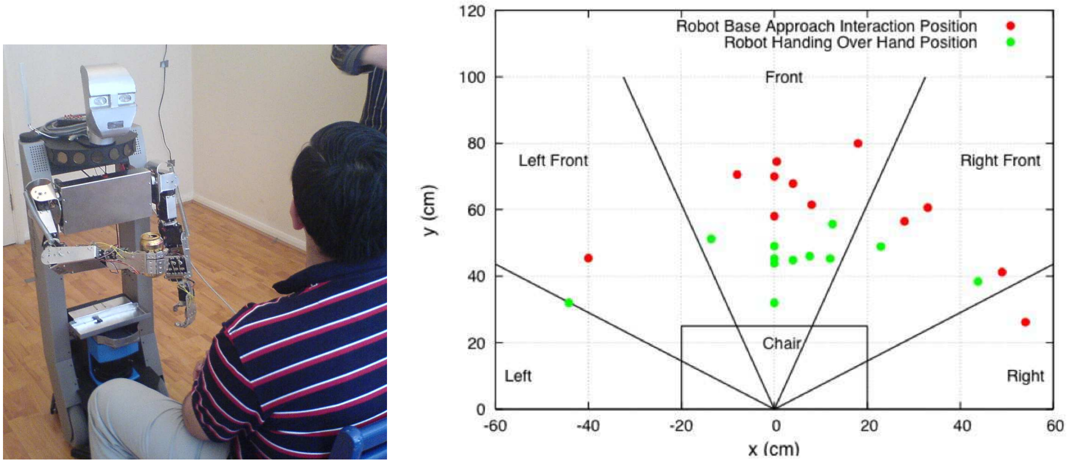
Manipulation tasks, like fetch-and-carry tasks, involve close interaction between robot and human. Distance boundaries for these types of tasks are shorter than simple navigation/communication tasks. In order to understand the spatial requirements for a manipulation scenario, we can no longer simplify the robot's placement to a 2D point (circular in case of a cylindric shaped robot. More generally, it is a 2D projection of robot's shape to the ground.), but we should take into account its whole structure.

Let us consider a robot handing over a bottle to a person. User studies are necessary to find answers to the following questions:

- How far the robot's base should be placed from the human?
- How far the robot's hand (gripper) should be placed from the human (or from which part of the body)?
- What are the gestures and protocols to hand over an object?
- How to synchronize the robot's base and the arm motion?
- What kind of human factors effect the robot motion?

One of the major problems of user studies for robot manipulation is the concern on the security of the test subjects. In these types of scenarios, where the robot and people are in a close interaction, any kind of unexpected robot motion can harm, cause serious injuries or at least can frighten subjects around the robot. Due to this issue, user studies in this field are often based on robot observation from a distance or at some cases based on Virtual Reality (VR).

In [Sakata 04], Sakata et al. evaluated the motions of a humanoid robot (HRP-2 in this case) for a simple pick and place task. The robot's motions are categorized into three classes of patterns: arm, head and upper body motions. The pick and place task is executed either as a sequence, or as a combination of these motion patterns with various speeds. The results of this study indicated that moving the arm and the upper body as well as moving the head at the same time for a pick and place task is more comfortable than a sequence of these actions. Similar results are obtained also by Boekhorst et al. with a smaller robot by observing the behavior of a large group of children in front of a robot moving its head and its arm [Boekhorst 05].



(a) Experiment Setup.

(b) Robot base and Robot hand placements registered during the user trial.

Figure 2.1: *Robot "handing over an object" User Trials*

In [Koay 07], Koay et al. presented a detailed study[2] on how a robot should hand an object to a person. The experiment was a Wizard of Oz[3] setup including a one armed can holding PeopleBot and a sitting person without any obstacles around (Figure 2.1.a). The robot's task was to approach the person and give him/her the can. The subject was asked to evaluate the movements of the robot and also to indicate a more comfortable for performing the task way if possible. 58% of test subjects preferred a frontal approach of the robot and 75% preferred that the can handed directly in front. The most comfortable position for handing the object was at ≈0.5 m far from the chest of the person and at ≈0.8 m height (Figure 2.1.b). And interesting result indicated no correlation between person's height and height of object's handing position. This

---

[2] Details of this study can be found in annex A.

[3] In the field of human-robot/computer interaction, a Wizard of Oz experiment is a research experiment in which subjects interact with a computer system which is believed to be autonomous, but which is actually being operated or partially operated by an unseen human being.

study also showed that 58% of participants evaluated to feel more comfortable when the robot's arm begins to move during the base motion at the last 1 m of its whole trajectory.

A recent study conducted by Haddadin et al. [Haddadin 08, Haddadin 07] pointed out the importance of robot speed and mass in human-robot close interaction scenarios. In this work, crash test scenarios are held between different weighted robots with different speed and a crash test dummy. The results showed that the role of mass (and the load) of the robot becomes serious only if a human body part clamps or the speed of the robot exceeds 2 m/s.

The importance of dynamics is also pointed out by Huber et al. in [Huber 08], where two different robot arm speed profiles are compared in an object handing over scenario. Test subjects found that the motion with limited jerk speed profile more natural and comfortable compared to the trapezoidal speed profile. Compared to human-human handing over trials, the motion with limited jerk caused the same reactions, thus showing that the robot's movement was as predictable as that of a human.

Another tool for conducting manipulation user studies is to use Virtual Reality (VR) systems with immersive 3D environments. The great advantage of such systems is the absence of a real danger and thus the possibility of studying dangerous cases where using a real robot would be too risky. Inoue et al. has showed in [Inoue 05] that there is not a significant difference of responses given by participants to a VR robot than to a real robot and thus the results obtained from VR user studies can be generalized to real robots. This last result makes the utilization of VR interesting and more practical in some cases.

In the VR user study conducted by Nonaka et al. [Nonaka 04], a virtual robot is placed in front of a sitting person. Participants evaluated a coordinated motion of body, arm and head as more comfortable compared to a single motion of robot arm. This study showed clearly the importance of head, arm and body coordination during a manipulation task (precisely a handing object task) not only for being natural, but also for clearly showing robot's intentions.

Research towards understanding spatial relations between humans and robots is a very fast growing field. Basing the strong foundations of human social behaviors and human psychology, challenges of this type of user studies [Walters 05c] are being overcome and this field discovers more and more distinct human behaviors towards robots which can later be used in motion generation methods and algorithms to synthesize safer and more comfortable robot motions.

2.3 Human Models for a Friendly Motion

As it results from user studies, a robot moving in an environment where humans are present, has to take them into account explicitly by obeying social distances and protocols. This fact implies that humans cannot be considered just as objects and cannot be simplified to moving obstacles. In order to generate safe and comfortable motions, humans must be taken into account as another type of entity having following, not only but most important, characteristics:

Position and Orientation: Each human within the robot's environment has a position (x, y) that can be considered as the projection of the chest's 3D coordinates on a plane. Legs, chest and head are extremities that can be used to obtain the orientation.

Hands and manipulation capability: Human's hands interact with the robot and therefore they provide positions and orientations. Right and left-handedness provides an information on which hand the human prefers.

Motion: When a human is changing its position, he/she is moving. His/her motion has a direction, a speed and a goal.

Attention: Generally, when awake, humans have a point of focus where they give their attention.

Field of view: The field of view determines the zone that the human can perceive and can give his/her attention to. It can also indicate the human's zone of awareness within the environment.

Activity: A human performs many activities during the day. Some of these are less important than others, but they all affect the human's behavior.

Plan and a Goal: Activities have goals to satisfy. To achieve this goal, a plan must be followed.

Human modeling is a very important step for creating human aware motions. This model provides the human-sided inputs to the system and plays a major role in the robot behavior design (i.e. determines the robot movements in a motion planning context. But in a larger schema, where not only robot motions but also its whole behavior is human friendly and interactive, human model is still a very important element and should be richer with the inclusion of not only a geometric representation but also a symbolic one). Figure 2.2 illustrates a scenario where various important characteristics of a human can be seen.

The following section will briefly describe different models used in selected literature[4] .

2.3.1 Position and Orientation

The most common human representation is a point and an orientation (x, y, θ). This model allows the robot to localize persons based on their 2D positions and orientations [Nakauchi 02, Takemura 07, Yoshimi 06] and constitutes the basis of almost all the 2D human models for navigation.

The position of a person is obtained by the projection of the position of his/her center of mass to a plane parallel to the ground. The orientation, on the other hand, can be interpreted in different ways. The most common orientation choice is to take the chest's direction as the direction of the person [Yoda 97]. If the person is moving then a common choice is to consider the direction of the movement as the direction of the person [Hoeller 07]. In practical applications, this choice largely depends on the sensor capabilities of the robot. For instance a robot equipped with only a laser scanner, may only calculate human's orientation using the positions of his/her individual legs. Yet, as with two legs we obtain two orientations facing at opposite directions, it is impossible to guarantee the correct human orientation. In contrast, the orientation of a moving person usually corresponds to the direction of his/her motion. Still many of the biggest challenges lie on perception capabilities which still constitute a big limitation nowadays.

---

[4] This section will only review human models of the selected literature listed in Navigation (§ 2.4) and Manipulation (§ 2.5) sections of this chapter.

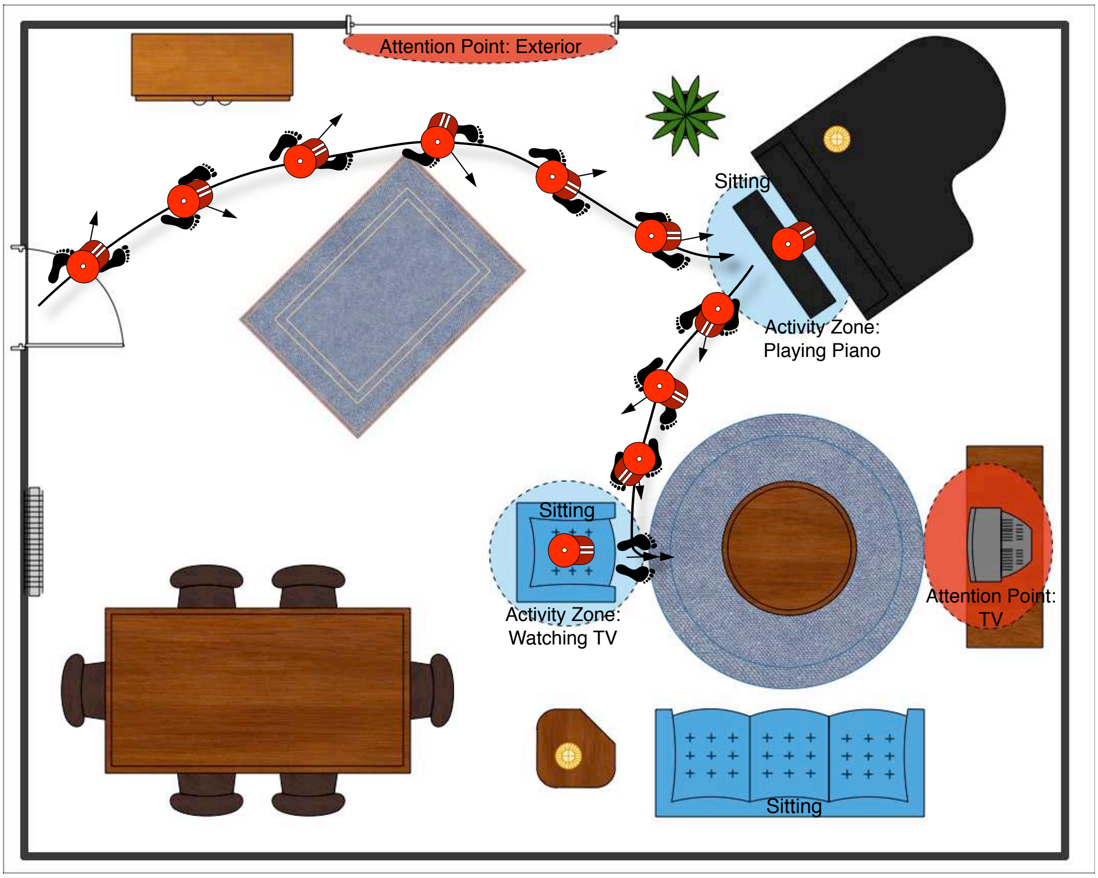


Figure 2.2: *Various aspects of a human model. In this scenario a person enters a room to play the piano. Even though his/her chest follows his/her path, the orientations obtained from his/her paces (presented by the arrows between his/her feet) differ largely. The following question rises "Which one to consider as the human orientation?". A third orientation, that does not follow neither of these two, is the looking direction (presented with the orientation of his/her hat). During his/her motion, the person looks to the window changing his/her attention towards the event occurring outside. Once arrived to the piano, he/she proceeds to start the activity of playing the piano which is done in a sitting position. The activity of the person not only describes what the human is doing but also it defines geometric constraints about human (in this case, sitting). The same notions are followed when the person leaves the piano and walks to sit on the chair to watch TV. This scenario shows different possible components of a human model; a path, an orientation, a looking direction, an activity and geometric constraints linked to the activity (or not linked in case of watching TV and sitting, which are both independently feasible).*

2.3.2 Whole Body Representation

Unlike the scenarios where the robot only navigates in an environment where humans are present, in manipulation scenarios, where robot and human are near to each other, modeling the human with only his/her position is not enough. The whole body of the person must be taken into account as well.

A common model for manipulation scenarios is to consider the important parts of human body. 3D positions of the head, the chest and the hands of the person are commonly used to build a model in order to have “global” presentation of human’s extremities [Kulić 05]. This model is also enriched by considering these parts as ellipsoids thus having a proxemics representation.

2.3.3 Motion

Since humans are generally moving in the environment a more accurate representation would be to include their speed (v, w) to their position component [Pacchierotti 06b, Althaus 04, Martinez-Garcia 05, Hoeller 07, Zender 07]. In robot navigation, only the global displacement of the human is taken into account (motion of body parts are ignored). The motion of the human is represented in a 2D plane by linear and angular velocity component (v, w) and they are added to the position information to form (x, y, θ, v, w). Using this representation, it is possible to obtain not only the position but also the speed and heading direction of a person. When the person is moving, his/her orientation θ is considered the same as the direction of his/her movement.

2.3.4 Path and Destination

The human’s complete position information (x, y, θ, v, w) is also used to learn his/her motion patterns [Gockley 07]. This model allows us to anticipate the human path [Hoeller 07, Sasaki 06] and the human’s destination [Bennewitz 05, Panangadan 04]. Since the robot is able to predict where the human will be at a given time in the future, it will have a higher probability to generate paths that do not interfere with humans’ paths. This model can also be extended to a large number of moving entities not only limited by moving humans but covering also dynamic objects in the environment ([Laugier 08]). Furthermore, knowing the possible human destinations and activities could also enable the robot to predict human activity and goal and therefore let the robot to plan its further activities in harmony with humans’.

2.3.5 Personal Spaces

Personal spaces are modeled as distance thresholds [Hall 66, Pacchierotti 05] where (1) “intimate distance” ranges up to 0.45 m from the body and interaction within this space might include physical contact; (2) “personal distance” ranging from 0.45 m to 1.2 m used for interacting with family members and close friends; (3) “social distance” ranging from 1.2 m to 3.5 m used for formal interaction; and finally (4) “public distance” extending beyond 3.5 m where no interaction occurs.

These distances, coming from the proxemics theory, give valuable information on how a robot should navigate in order to behave in a socially acceptable way.

2.3.6 Field of View and Attention

The visual perception is the most important sensory information for a human in a human-robot interaction context (auditory perception can also be considered essential but it is always followed by a visual perception). The visual perception can be modeled as a field, called field of view, which presents the angular extent of the observable world that is seen at any given moment. The field of view represents the area in the environment that the human can see at a given head/eye orientation. It also represents the direction where the human’s attention is oriented.

Attention can be modeled as a vector [Traver 00] and the visible region can be modeled as a cone both extending from the head or from the eyes towards infinity. The latter model will be described in detail in the following chapters since it corresponds to the model used in this work.

2.3.7 Activity

Another important addition to human model is to consider human activity. During the normal flow of a day, we conduct many activities that are not only tasks to do but also are protocols that imply some rules. The activity can vary from basic motions (sitting, standing, walking, etc.) to more complicated multi motion tasks (assembling, repairing, etc). But the common aspect of all these activities is that they comply a set of social rules. In order to share the environment with a person who is conducting an activity, one should comply with these rules. For example, one should be quite and should not disturb a person who is reading a book or who is sleeping; or should not pass between a person and a TV set if the person is watching television.

In order to equip the robot with a social behavior and socially acceptable motions, human activities should be included in human model and should be taken into account explicitly. Even though in literature these inclusion is generally done at symbolic level [Clodic 07b], we still notice simple human models that differentiate a walking person from a still person [Yoda 97].

2.3.8 Preferences

One interesting notion that may be included to the human model is human's preferences. These preferences can be communicated to the robot explicitly, e.g. a person preferring not to take an object from the robot but to take it from a surface, or implicitly, e.g. a left-handed person will feel more comfortable if the robot hands over the object to his/her left hand.

The preferences can contain any information the person would like to impose on the robot. In our work, the left and right-handedness are taken into account and will be detailed in the following chapters.

2.4 Navigation in Human Presence

Various methods for motion generation exist in literature. Most of these methods are designed for a specific task even though some can be generalized to a broader utilization. Research on motion synthesis in human presence can be categorized in various ways. We decided to put into use case categories to have a better understanding of the environment and the task that these methods are designed for. Major use cases in this field are (1) hallway crossing where a person and robot crosses in a hallway; (2) following where the robot is following a person; (3) obeying and maintaining a certain formation, and finally (4) free navigation where the robot navigates from one place to another.

2.4.1 Hallway Crossing

By definition hallways are walled corridors inside a building or structure. Hallways direct and restrict movement. Distances between people on a hallway are reduced from public space to personal space [Hall 66]. A robot living in a human environment will most probably have to take these passages to travel between rooms or staircases. As the distance between persons and robot gets shorter in a hallway, explicit methods dealing especially with these types of environments become necessary in order to ensure safety of the humans in the environment and also of the robot.

Early methods of hallway crossing encode human-human hallway passage behavior [Yoda 97] where the robot deviates its path when it approaches to a person. The distances between human and the robot, where it begins to deviate and where it passes next to the human, are obtained from user studies. This behavior is encoded to the robot and activated when a person is detected with ultrasonic sensors. The same method is used with different parameters for a standing, a walking and a running (with short steps) person.

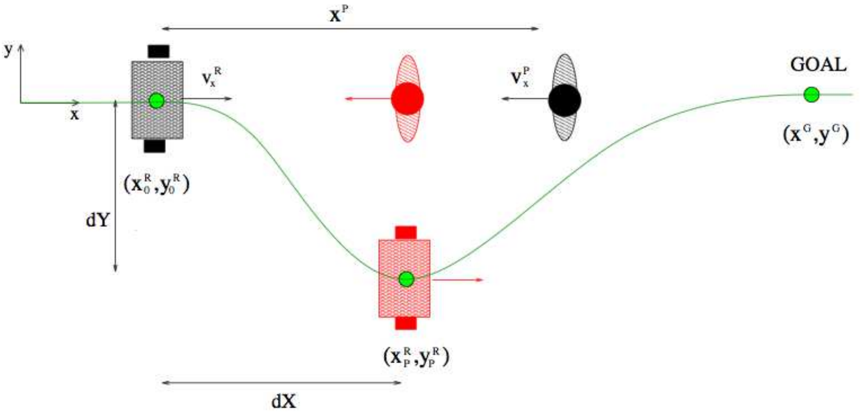


Figure 2.3: *Robot is passing a person in a hallway scenario. When the human is* x_P *close, the robot deviates its path, reaching the maximal lateral distance. Once the person is stayed behind, the robot returns to its previous lane. Figure taken from [Pacchierotti 06b].*

In [Pacchierotti 06b], Pacchierotti et al. extended this method with the capability to adapt to the changes in the speed of the person that allowed a dynamic interaction between the robot and the person. The following control strategy used for a hallway passage scenario illustrated in Figure 2.3:

$$dY = LD + w_R/2 - (y^P - y_0^R)$$

$$dX = v_x^R/(v_x^R - v_x^P) \times (x^P - x_0^R)$$

The motion of the robot is also controlled by 2 parameters based of definition borrowed from proxemics: Signaling distance (x^P), representing the distance between the human and the robot that the robot begins to deviate its path; and lateral distance (LD), representing the distance that the robot should be placed when it is passing the human.

Even though these methods allow a safe passage through a hallway, they are limited to plan the motion of the robot with only one person and with the assumption that the hallway is large enough to allow a safe passage for the robot and for the human next to each other. If the robot is blocked by the human or by an obstacle, it stops and waits for the human to have the initiative to clear the path. Once the path is clear, the robot resumes its motion.

Although these approaches work in perfect scenarios, in a real implementation they will likely fail because of the possible errors in perception. None of these methods offers a solution when an obstacle close to human is detected or when a human appears in a very close distance to the robot.

2.4.2 Following

A robot following a person is a very common scenario widely studied in this field. A user willing to show and teach something to the robot will ask the robot to follow. The robot should follow the human respecting his/her personal space, i.e. not approaching too much and not staying too much behind. The robot should also be reactive in order to avoid unseen obstacles or unexpected human behavior.

In [Yoshimi 06], Yoshimi et al. presented ApriAttenda^{TM}, a person following robot. Besides its human friendly form, this robot is equipped with a pair of camera that detects and track people around the robot. Once the "follow" behavior is activated, the robot tries to maintain a distance of 1.5 m to human with a basic navigation behavior: move towards the human if the distance is greater than 1.5 m, if it is lesser, then go backwards.

Takemura and al. introduced an improved following behavior in [Takemura 07] by using potential field based navigation to follow a person and to avoid obstacles. In this method the person is assigned an attractor function and the obstacles are modeled by repulsive forces. The attractive force is defined as the reciprocal of the distance between the robot and the goal, while the repulsive force is the one between the robot and the obstacles. The robot then moves the steepest descent direction to approach its goal by avoiding obstacles.

Gockley, Forlizzi and Simmons compared two following behavior: direction following and path following. In direction following, the robot programmed to go towards the same direction as the human. On the other hand for path following, the robot is programmed to follow the exact same path taken by the human. Even though the latter behavior guaranties a successful following (assuming that the person walks normally and does not try to pass through narrow places that the robot cannot), the first, direction following behavior is found to be more natural. The authors concluded their work [Gockley 07] with a future plan to incorporate a learning process to allow the robot switch between two behaviors.

In [Zender 07], Zender et al. incorporated a person following behavior with "making room for human to open/close a door" behavior in order to not only follow the human but also not to block doors and passageways. The robot uses a two layered map [Kruijff 07] containing a roadmap covering the free-space and a topological map containing distinct rooms that people adopt. In following mode, the robot navigates on its roadmap towards 0.5 m far from the human. When approaching to a door, this distance is increased to 2 m in order to avoid any blockage that can be caused by the robot. The robot's head is also commanded to look 1.70 m high towards the human in order to give an impression of social awareness.

A more complete method dealing with human following problem is introduced by Hoeller et al. in [Hoeller 07]. In this work the robot plans a constructs an extensive space tree from it current position to the human. A collision free minimum cost path towards the human is found with an $A^\star$ search in this tree (Figure 2.4.a). The state transition in constructed tree is described by:

$$
\begin{aligned}
x_k &= f(x_{k-1}, u_k) \\
&= \begin{pmatrix} x_{k-1} - \frac{v_k}{w_k}\sin\theta_{k-1} + \frac{v_k}{w_k}\sin(\theta_{k-1} + w_k\Delta t) \\ y_{k-1} - \frac{v_k}{w_k}\cos\theta_{k-1} + \frac{v_k}{w_k}\cos(\theta_{k-1} + w_k\Delta t) \\ \theta_{k-1} + w_k\Delta t \\ v_k \\ w_k \end{pmatrix}
\end{aligned}
\tag{2.1}
$$

where x_k is a configuration $(x, y, \theta, v, w)_k^T$ of the robot at time k consisting the position and the orientation of the robot along with its linear and angular velocity.

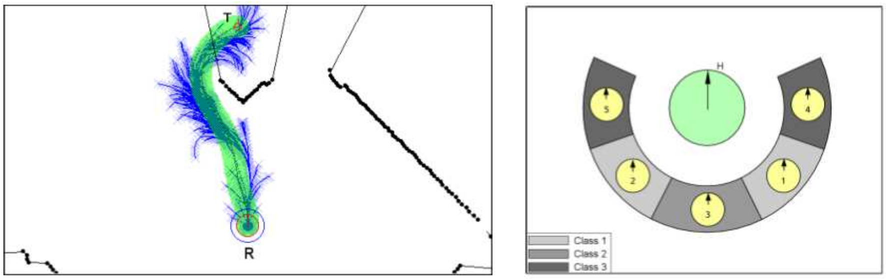
An interesting part of this work was the inclusion of human motion estimation. The robot guesses a short term potential target where human goes towards and calculates a probable human path using potential field method with obstacles as repulsers and target point as an attractor. The robot constantly updates its tree according to the human path found previously. The goal position of the robot is calculated automatically according to human's orientation. The zone behind the human is pre-divided to 5 hierarchical zones (Figure 2.4.b). The zone, which is collision free and highest in the hierarchy, is chosen to be the point where the robot plans to attain. In each planning stage, this choice is re-evaluated according to human's position in the environment (Figure 2.4.c).

Until now we considered the “following” as the robot following the human. But in a robot guide scenario, it will be the robot that will have the initiative of the motion and the human should follow it. As the robot is in front of the human and leads the way, the interaction is only at communication level and does not go to motion level. Nevertheless Martinez-Garcia et al. presents an interesting scenario of robot guide in [Martinez-Garcia 05] where a team of robot guiding a group of humans. The robots monitor constantly human group's center of gravity. In order to guide the group, robots change formation and control the angular acceleration of the group's center of gravity. The novelty of this work from human-robot interaction point of view is that no explicit communication occurs between humans and robots, thus only the motion and the formation of the robots communicate implicitly with humans. This behavior can be seen as similar to shepherd dogs guiding a herd without barking.

2.4.3 Maintaining Formation

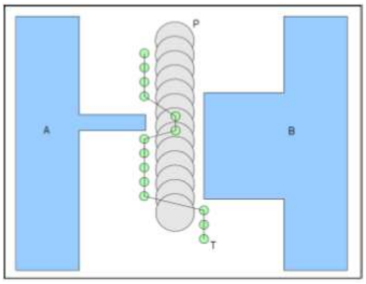
In our daily lives, there are many situations that we need to form a specific formation and maintain it for a period of time. This formation can be a precondition of our activity (e.g. standing in line) as well as a social rule (e.g. maintaining a circular formation in a group in order to allow everyone talk to everyone). So we should expect the same behavior from a robot that will live among us. This robot should observe and be a part of the formation in a natural manner.

In [Nakauchi 02], Nakauchi and Simmons present a robot that stands in line. The goal of the robot, which is placed far from the line, is to take a place in line and maintain a correct formation. The head of the line and its rough extension direction is given to the robot. With these informations the robot goes to the head of the line and begins to move towards the end of the line with its cameras facing humans. During this motion, each person in the line is detected and their orientation is found by comparing



(a) An extensive space tree is built between the robot and the person and a minimum distance path is found.

(b) Robot's goal position is chosen among 5 predefined position forming a half circle at the back of the human.



(c) In each replanning stage, the robot reevaluates its goal position in order to be collision free and as comfortable as possible.

Figure 2.4: *Robot following a person. Figures taken from [Hoeller 07].*

the position of head, neck and body. Once the end of the line is found, the robot places itself behind the last person.

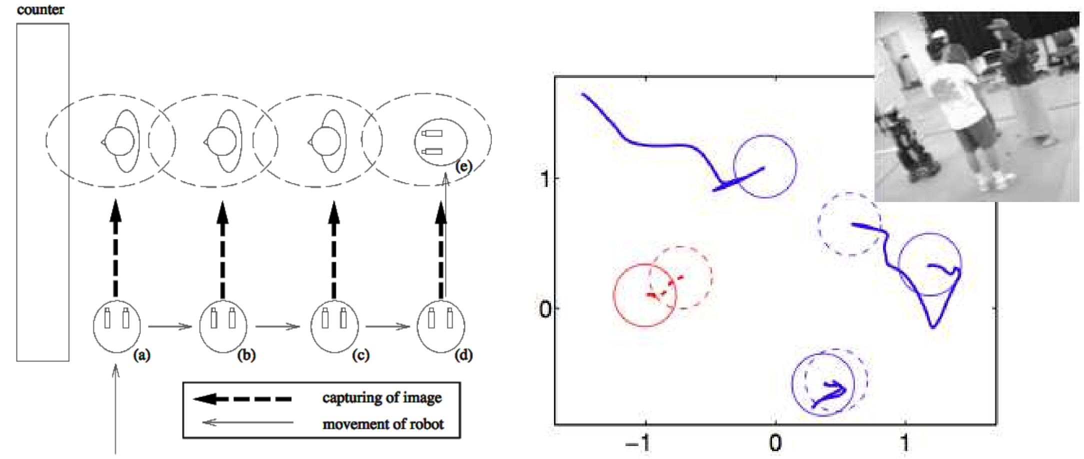
In this work humans are modeled not only with their positions and orientations, but also with their surrounding personal distances. The personal distance of a person is modeled as an ellipse wider towards the front. By taking into account this model, the robot tries to maintain a distance far enough to the person in front to avoid any disturbance and close enough to avoid people cutting the line (Figure 2.5.a).

Another example of robot maintaining a social formation can be observed in a group of chatting people. We generally notice that they shape a circular formation for that every member of the group can see and talk with everyone. The work presented in [Althaus 04] by Althaus et al. presented a robot that joins to the circular formation in a natural way, and maintains this formation all along its activity. Robot's heading direction is expressed by:

$$\psi = \frac{1}{N} \sum_{j} \psi_j e^{-c(d_j - D_{inter})}$$

where ψ_j is the direction of the person j, D_{inter} is the predefined interaction distance and N the number of human in the group. And the speed of the robot to maintain the formation is described by:

$$v = \frac{1}{N} \sum_{j} v_j e^{-c(d_j - D_{inter})}$$



(a) Robot that stands in line by following the line and putting itself at the end of it.

(b) Robot joins to a circular formation. The robot maintains this formation when humans move and another person joins.

Figure 2.5: *Robots taking and maintaining a specific formation. Figures taken from [Nakauchi 02] and [Althaus 04].*

where v_j represents the speed of each person.

With these representations, the robot joins to a group of people and maintains a circular formation according to the human's position, newcomers and leavers (as illustrated in Figure 2.5.b).

The methods used for robot that maintains a formation are simple basic rules that are specifically designed to work in a particular scenario (a particular formation). Robot behaviors in these types of scenarios are purely reactive and lack a planning approach. Although the information of the type of formation is necessary, the robots motion can be provided by more general schemes.

2.4.4 Free Navigation

Even though a social robot is expected to interact with people, in many cases an interaction is not necessary.

Bennewitz et al. presented a navigation strategy using motion patterns of people [Bennewitz 05]. In this work, the robot predicts the motion behavior of the people around and tries to avoid these predicted paths. The aim of the robot is to minimize the risk of interference with persons given the knowledge about their typical motion patterns.

A minimum cost path is found in 3D space-time occupancy map where the costs of the cells are calculated according to the estimated coordinates of a human at a time t. Some parts of the environment, like rooms, are matched with human motion patterns for the robot to anticipate their destinations. If the calculated robot path intersects with human's and causes a blockage, the robot slows down and/or stops in order to let the passage to the human.

Another work that take into account the human factor is [Sasaki 06]. In this work, Sasaki et al. presented a smart room scenario where multiple cameras and a 3D ultrasonic positioning system is placed. Human motions in this room are observed and walking paths are calculated with a 3-stage algorithm: extraction of important points (entry, exit, stop points), path clustering and path averaging. Unlike the previous method (by Bennewitz et al.) where the robot avoids predicted human path, in this work the robot takes the same paths as humans thus behaving more human like. Even though the robot can disturb other people, this behavior is very useful where environment and the activity attached to it put constraints on navigation. For example in a theater, where people avoid to pass in front of the screen to not disturb other spectators, a robot following the same path as spectators will move in the same natural way.

Although in these two works the robot plans motions by either trying to not cross or trying to follow human paths, they do not take into measure and reason on the "quantity of disturbance" that the robot can cause if their paths cross and minimize this quantity.

In [Madhava Krishna 06], Madhava-Krishna, Alami and Siméon presented a proactive path planning method where the safety is guaranteed during robot's motion. The planner takes into account the sensor capabilities of the robot (omnidirectional sensor with a limited range), the model of the environment containing polygonal obstacles and bounded velocities of possible moving objects hidden by static obstacles. A typical example would be to consider a door in a hallway and a robot navigating in this hallway. Because of the possibility of a sudden appearance of a person from the door, the robot can either slow down when passing in front of the door, or can try to pass as far as possible from the door. The presented method calculates a robot trajectory that minimizes the execution time of the path and finds a time optimal solution without compromising the safety by reasoning on the seen/unseen parts of the environment along the path.

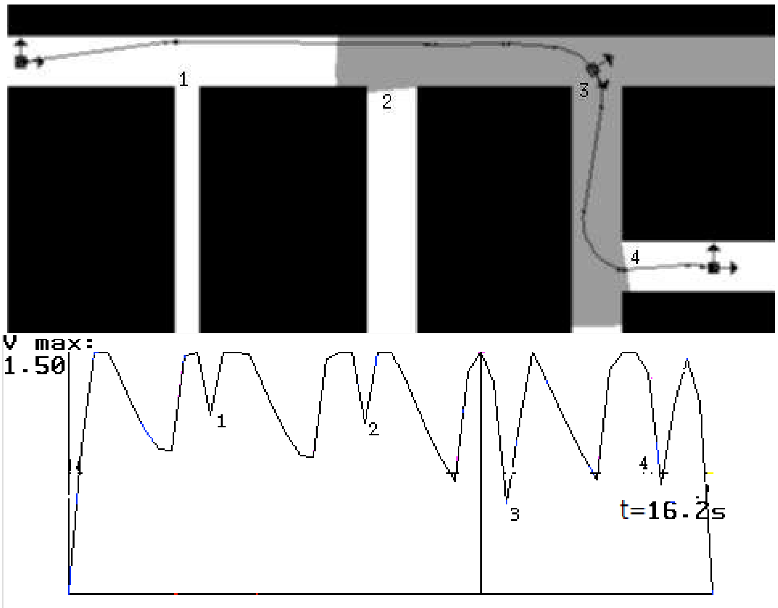


Figure 2.6: *The robot plans a time optimal path by getting farther from invisible and risky zones and by allowing higher speeds. Figure taken from [Madhava Krishna 06].*

Despite the absence of human in this method, the robot ensures that no collision will occur even a person appears at the most unexpected time (Figure 2.6). As humans do not approach to doors to minimize the risk of collision or slow down until they see no danger, paths produced by this planner are very similar to the ones taken by humans.

A recent work on intelligent wheelchairs, parallel to proactive path planning, is introduced by Gulati and Kuipers in [Gulati 08] where a “graceful” motion is presented and defined as a motion being safe, comfortable, fast and intuitive. In this work “safety” is defined as the absence of collision, “comfort” as maintaining sufficient clearances from surroundings with smooth and bounded motions, and “intuitive” as the naturalness of the motion to the driver. The wheelchair is placed in a doorway passage scenario where a graceful passage is needed.

With the presented control scheme, the velocity of the wheelchair reflects the curvature of the path rather than the closeness to the door edges thus finds a balance between the curvature of the path and its velocity.

The proactive motion and graceful motion presents two interesting aspects that ensure the comfort and the safety of the human at the same time. Although humans in the environment are not explicitly taken into account, resulting motions showed up to be natural. In pro-active planning case, these motions also guarantee robot (and also human) safety.

2.5 Manipulation in Human Presence

A helper robot equipped with manipulation capabilities will be expected to interact very closely with humans. In this interaction, the robot will hand objects to humans, transport loads for humans and manipulate objects together with humans. The closeness of such interactions will oblige the robot to adapt more complete control methods where the motion of the robotic arm and the human body must be taken into account.

Research towards manipulation in human presence is a new sub-field of Human-Robot Interaction which is a relatively young research field compared to robotics. Despite this, interesting works towards generating manipulation motions in HRI context exist in literature. Unlike the previous section (Navigation in Human Presence, §2.4), in this section, we will classify state of the art approaches not by their use cases but by their approaches to the problem.

2.5.1 Danger Index

In [Ikuta 03], Ikuta et al. presented the first general method of evaluating safety for human-care robots. In this work, factors affecting human safety are studied and classified into two categories: “design” and “control”. Design strategies consist of minimizing the weight of the robot to minimize the force of a possible impact and designing cover, surface, shape and joints to attenuate diffusion of the force.

To evaluate the danger, a total danger index based on various safety strategies is calculated. This danger index contains safety evaluation for design strategies like reducing weight, soft material, joint flexibility, shape and surface friction as well as control strategies like keeping the distance and approaching velocity. The general danger-index for a control strategy is defined by:

$$\alpha^* = \frac{F^*}{F_c}$$

where F^* is producible impact force and F_c is minimal impact force that causes an injury to human. The time until collision Δt for an approaching robot with mass m at reduced acceleration a from a distance l is calculated by:

$$l = v\Delta t - \frac{a\Delta t^2}{2}$$

$$\Delta t = \frac{v}{a} - \sqrt{\left(\frac{v}{a}\right)^2 - \frac{2l}{a}}$$

and the danger-index for keeping the distance can be expressed as:

$$\alpha^* = \frac{F^*}{F_c} = m\frac{(v - a\Delta t) - v'}{F_c dt}$$

which v' presenting the velocity after the collision.

On the other hand, the danger index for approach velocity is expressed as a time change of the momentum:

$$F^* = \frac{mv - mv'}{dt} \qquad \alpha^* = \frac{mv - mv'}{F_c dt}$$

These control safety strategies along with the design strategies form a total danger index α_{all}:

$$\alpha_{all} = \prod_{i=1}^{n} \alpha_i$$

The total danger index represents an evaluation of the safety for the physical structure and for the motions of the robot. In [Nokata 02], minimization of these danger indexes is used to generate safe motions. Even though the method produces safe paths, only the position of robot's end effector is controlled.

A more recent work following the same direction is described in [Kulić 05] by Kulić and Croft. In this work not only the end effector but the whole structure of the robot is controlled to generate safer motions. Two danger criteria are defined: "inertia criterion" and "relative distance criterion". A scalar value for robot's inertia is extracted by calculating the inertia about an axis originating at the robot base and normal to the robot's sagittal plane: $I_s = \overrightarrow{v}^T I \overrightarrow{v}$, where I_s is the inertia about the v axis, v is the vector normal to the robot sagittal plane and I is the robot inertia tensor about the base. The inertia criterion is expressed as:

$$f_I(I_s) = \frac{I_s}{m}$$

This function can be interpreted as an attractive function, pulling each link towards the robot base.

The relative distance criterion, which is a repulsive function between the human and the robot, is defined by:

$$f_{CM} = \begin{cases} \frac{1}{2\epsilon}, & D_{CMO} \leq \epsilon \\ \frac{1}{2}\left(\frac{1}{D_{CMO}} - \frac{1}{D_{Max}}\right), & \epsilon < D_{CMO} < D_{Max} \\ 0, & D_{CMO} \geq D_{Max} \end{cases}$$

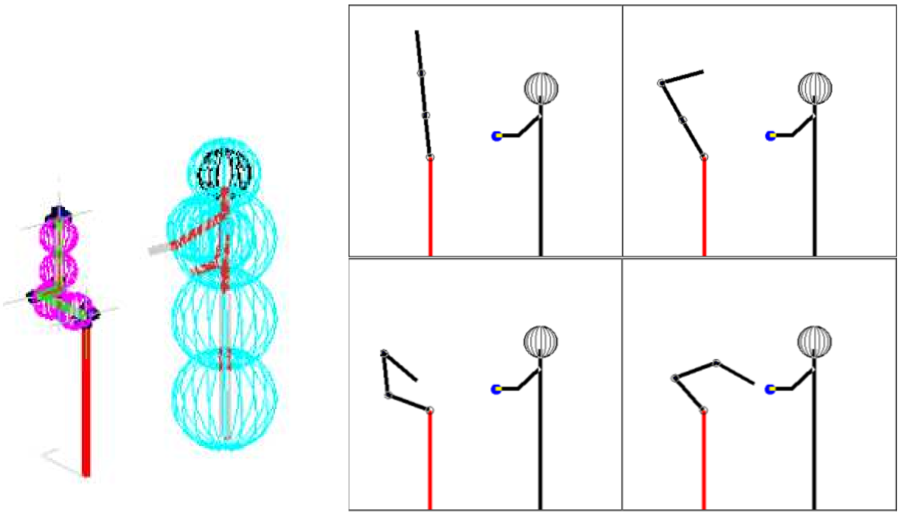
with $D_{CMO} = D_{CM} - D_{Min}$ where D_{CM} is the distance between robot and human, D_{Min} is the minimum allowable distance, and D_{Max} is the distance which this criterion is no longer contributes.

The global danger criterion cost function is defined by the weighted sum of the relative distance and inertia criteria:

$$DC = W_i f_I(I_s) + W_d f_{CM}(D_{CM})$$

With a goal seeking attractor function towards the hand of the human, a path is planned by searching robot configurations minimizing DC. The human is modeled with englobing spheres iteratively placed from the biggest to smallest until a complete minimal coverage is found (Figure 2.7.a).

As seen in figure 2.7.b, robot arm advances towards the human hand by conserving a posture as compact as possible, thus resulting a safe robot motion.



(a) Human is modeled by a set of englobing spheres.

(b) The robot moves in a compact way by the minimizing the Danger Criterion.

Figure 2.7: *Robot arm motion minimizing the danger index calculated by the robot inertia and human-robot relative distance. Figures taken from [Kulić 05].*

2.5.2 Human Vision

Another important information about the human, other that his/her placement, is his/her looking direction. What the person is seeing or what he/she is looking at gives us clue about his awareness of the robot. The level of danger decreases if the human is aware of the robot.

In [Traver 00], Traver et al. presents a danger index based on not only distance and speed but also on human's looking direction. With this last item, the danger index in this work, η, is calculated by:

$$\eta = \eta(q,t) = G\left(d_{hr}, \frac{\partial d_{hr}}{\partial t}, r(q), d_{TH}, \overrightarrow{l}\right)$$

where the function G is a grouping function (a weighted sum in this case), d_{hr} the distance between robot base and human, $r(q)$ the posture danger (based on how apart the end effector is from the base), d_{TH} the distance between human and the tip of the robotic arm and finally $\overrightarrow{l}$ is the look vector presenting the looking direction of the person.

With the minimization of the danger index η, the path computed by the robot not only ensures safety but also stays as visible as possible to the human.

2.6 Discussion

This chapter presented a brief survey on robot motion generation methods that consider explicitly the existence of human. User studies have given important information and metrics on how a robot should move and place itself around humans. These studies are very important step to understand the notion of "safety" and also more precisely the notion of "comfort". Evaluating comfort is a very difficult task because of the subjectiveness of this notion and the lack of clear expressiveness of human reactions.

From a simple 2D position to a more complete $(position, speed, state, activity, path, preferences)$, human models are described. Even though a complete human model is beneficial in almost all scenarios, it is highly dependent on the sensor capabilities of the robot. Correct data interpretation and fusion are vital steps to obtain a more complete human model.

Navigation methods focused on robot base motion in human presence are briefly described in use case scenario categories. Despite the efficacy of these methods, they are hardly expandable and stay very specific to the use case they belong. These methods stay local and reactive, and a more global planning approach is missing. The following chapter will present the Human-Aware Navigation Planner which fills this gap.

The last section presented manipulation methods composed of utilization of danger indexes and human gaze vector. These methods ensure the safety of the person but do not deal with the comfort issue. The Human-Aware Manipulation Planner, described in following chapters, offers a planning schema where not only the safety of the person but also its comfort is taken into account explicitly.

The methods of motion generation in the literature are mainly reactive approaches working in specific scenarios. The robot reasons only on the present situation to generate a motion for the moment. A planning approach, that will not only reason on present but also on future motions of the robot, of the human and of the state of the environment, is missing. With this work, we believe to fill this gap.

Following chapters will describe methods and algorithms that we use towards a human-aware robot motion.

CHAPTER
THREE

Human-Aware Navigation Planner

This chapter introduces the Human-Aware Navigation Planner (HANP), a planner that takes into account explicitly the presence of humans in the environment. Section 3.1 opens this chapter by stating the problem and the reasons why such a planner is necessary. Section 3.2 describes how the humans and the environment are modeled and introduces the base of our approach: the grids. In Sections 3.3, 3.4 and 3.5, three interaction criteria, called "Safety Criterion","Visibility Criterion" and "Hidden Zones Criterion", and their grid representations are presented respectively. Section 3.6 explains how the planner takes into account obstacles and illustrates the "Obstacle Grid". The planning process along with the usage of the grids are described in Section 3.7, and Section 3.8 shows the paths generated by the Human-Aware Navigation Planner in six different types of scenarios. Two extensions that may add to the social behavior of the robot and their underlying ideas are described in Section 3.9. Finally this chapter is concluded with a discussion (Section 3.10) that summarizes and discusses the methods and algorithms as well as the results of the planner.

3.1 Introduction

With the introduction of mobile robots in our daily lives, we began to see robots that "live" among us. IRobot's very successful vacuum cleaner Roomba [IRobot] or Sony's small dog AIBO [Sony] are two of many popular examples of robots that take a role in our lives. The future of such robots is more and more bright with the possibility of automatization of our "boring" or "not so pleasant" tasks.

Yet this introduction brings new challenges to robotics research that need to be studied explicitly. One of these challenges is the necessity of methods and algorithms of motion planning in human presence. As the robot lives and moves in an environment where humans exist, its motions need to be as comfortable and safe as possible to humans around. Even though safety and comfort can be partially assured with better hardware designs, it cannot be completely satisfied without an explicit reasoning in motion planning. In order to ensure safety and comfort of the humans around the robot, new motion planning methods, that take into account humans explicitly, are required.

As mentioned in previous chapter (§ 2.2.1), studies in human space sharing, e.g. proxemics theory, and human-robot user studies provide rules and protocols that the robot should respect in order to behave in a socially acceptable manner. A planner that includes these notion in the path planning level will produce paths not only safe by also comfortable and acceptable to humans.

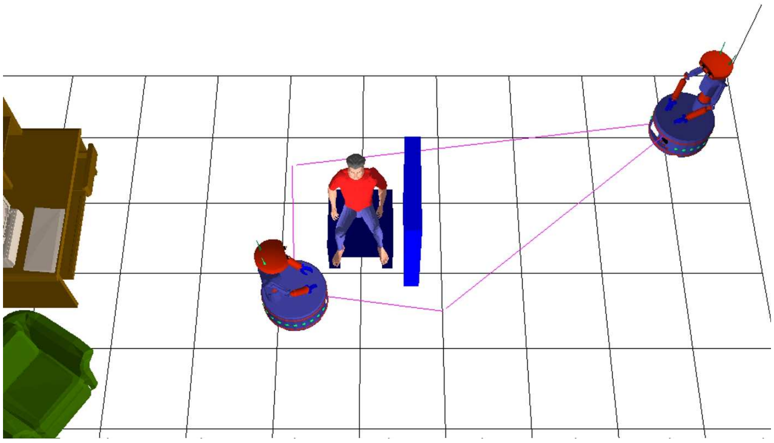


Figure 3.1: *2 paths generated by a "classical" motion planner. The robot follows the shortest path towards its goal by either passing behind and too near of the human or appears suddenly behind the wall. Clearly, these paths are inconvenient and will cause some discomfort to the person.*

Figure 3.1 illustrates scenario in order to understand the importance of such a planner. In this example, a robot and a human share the same environment. The robot's goal is to go next to the human for further interaction. As seen in the figure, a "classical" motion planner ends up with two possible paths, one passing behind and the other in front of the human. Even though these paths are completely "feasible" in classical motion planning [Latombe 91] point of view, if we put ourselves to this human's place, we can realize that the robot behavior is not very comfortable in neither of the paths. This discomfort is caused because:

- If the robot follows the path that passes behind the human, it will cause discomfort because it passes too near. As the robot will be invisible for most of its path, the human will also feel insecure of its actions and activities,
- The path passing in front of the human is also uncomfortable because of the existence of a wall. When took this path, the robot will burst into view from the obstacle and as it will also be too close to human, it will cause fear and surprise.

So it is vital for a robot, that will navigate among humans, to move in a way that not only ensures the safety but also should shows a certain level of social behavior by reasoning on humans.

In order to plan safe and socially acceptable mobile robot motions, a planner that explicitly takes into account humans with their safety, their field of view, their states and preferences, is developed during this thesis. As this planner is aware of the humans around the robot, it is called Human-Aware Navigation Planner.

A cost based approach similar to rough terrain navigation methods [Iagnemma 04] constitutes the basis of this planner. In the context that we are placed upon, roughness of the terrain is the safety and the comfort of humans and presented by various cost grids. These cost grids allow the planner to evaluate the situation (the environment, the humans and the robot) and generate safe and comfortable paths.

The following sections will describe in detail methods and algorithms of Human-Aware Navigation Planner, as well as its resulting paths[1] and possible extensions.

3.2 Environment, Humans and Grids

The representation of robot's environment is an important notion in motion planning. This representation defines the search space for the planner and contains the robot and the obstacles as well as the accessibility of the robot. In this search space, a continuous path connecting an initial point to a goal point is found respecting the collision-freeness as well as obeying the constraints that are imposed by the geometry of the robot, of the environment or of the task.

3.2.1 The Environment

In highly constrained environments, the robot motions need to be calculated very precisely and a 3D motion planning method is required in order to ensure the collision-freeness of robot's path. The environment needs to be represented in every detail and collisions between robot and the obstacles in the environment need to be checked in 3D, by taking into account the whole structure of the robot.

Even though planning in 3D configuration space[2](3D in case of a robot having only two translational and one rotational degree of freedom, for a robot having more degrees of freedom, the dimension of the configuration space is equal to the number of degrees of freedom) will have the benefit of generating precise motions for a complicated robot and for a constrained environment, the longevity of the planning presents a setback in environments that change too fast.

Yet in Human-Robot Interaction context, where the robot is moving in an environment shared with humans, the general shape of robot's motion yields more importance than its exact path. The small differences in robot path have a small influence on human's comfort unlike the shape of the path, which is the main actor on human's feeling of safety. In such a context, the planner needs to adapt the motions of the robot according to the change on not only the environment but also on the humans as well.

In Human-Aware Navigation Planner, a 2D projection of the 3D environment is used (Figure 3.2). The robot and the obstacles are represented by 2D projections of their bounding boxes. This simplification of the environment allows the planner to generate paths fast enough to follow the changes in the environment in sufficient time. The loss in precision because of this simplification is decreased by testing collision between the robot and the environment in 3D.

Thus it is accurate to say that Human-Aware Navigation Planner is a 2.5D planner with 2D representation of the environment searching in $SE(2)$[3] configuration space

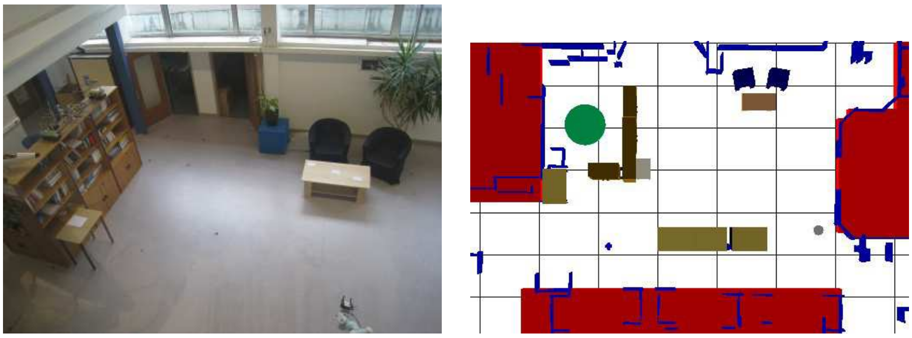
---

[1] It is important to make the distinction between a path and a trajectory. A path is a geometric curve while a trajectory is time parameterized path. If not explicitly stated, Human-Aware Navigation Planner generates path and robot follows this path with an external execution module that transforms it to a trajectory.

[2] Configuration Space is the space of possible positions that a robot may attain.

[3] $SE(2)$ is the special Euclidean group $SE(2) = \Re^2 \times SO(2)$, where $SO(2)$ is the special orthogonal group of 2D rotations. A configuration in this space can be represented using 3 parameters (x, y, θ).

with 3D collision detection.



(a) Real Environment. (b) 2D projection of the real environment.

Figure 3.2: *Representation of the environment.*

3.2.2 Humans in the Environment

As mentioned in § 2.3, modeling the human is an important step of the planning. In HANP, the position+orientation (x, y, θ) model is adopted and extended to include human's current states and preferences.

A human H_i is represented by

$$H_i = (St, State_1 \ldots State_n, Pref) \tag{3.1}$$

where St is the structure and kinematics of the human including (x, y, θ), $State_i$ is a human state defined by a number of cost parameters and $Pref$ represents various preferences of a person. A state is defined by:

$$State_i = (Name, Conf, Param) \tag{3.2}$$

where $Name$ is the name of a posture state (e.g. $Name = \texttt{SITTING}, \texttt{STANDING}$), $Conf$ is the human's configuration in that state (if applicable) and $Param$ represents the data needed to compute costs according to that state (the notion of "costs" will be explained in following sections).

The state of a person is mostly related to the activity that he/she is conducting. If his activity imposes some geometric constraints that the robot should obey (e.g not to approach too much to a person in `SLEEPING` state), it can be represented as a $State_i$. $Conf$ in this case will contain the posture of the human when conducting his activity (e.g. in a horizontal position for the `SLEEPING` state).

3.2.3 Cost Grids

HANP is constituted of various two dimensional grids. Each grid is composed of cells covering the whole environment. In order to cover all the space, the environment is sampled. A homogeneous sampling strategy is applied and a grid is obtained. Each cell in this grid corresponds a square zone in the environment. The number of the cells in a grid depends on the dimensions of the environment as well as the sampling resolution. For example, an environment whose size is 10 m to 15 m sampled homogeneously each

0.1 m will have $100 \times 150 = 15000$ cells and a point having the coordinates $(3, 6)$ will correspond to the cell whose coordinates are $(30, 60)$ in the grid.

Besides a position in the grid and a corresponding zone in the environment, a cell contains also a cost derived from the relative positions of humans, humans' states and preferences. Thus a grid G is defined by:

$$G = (M_{p,r}, H_1 \ldots H_n, f_{cost}, Sp) \tag{3.3}$$

where $M_{p,r}$ is a matrix containing $p \times r$ cells represented by $a_{i,j}$, the cost of the coordinate (i, j) in the grid, $H_1 \ldots H_n$ is the list of humans in the environment and Sp represents the sampling rate that used for the projection of the environment into this grid. The function f_{cost} calculates the value of each cell according to its coordinate by taking into account only one human. The matrix M is constructed by the equation:

$$a_{i,j} = \max_k(f_{cost}(H_k, i, j)) \tag{3.4}$$

The function f_{cost} constitutes the core of a grid and makes the difference between different types of grids. This function returns a scalar cost value for a human according to his position, orientation, looking direction, state, preferences and the environment. Figure 3.3 illustrates a grid covering the environment.

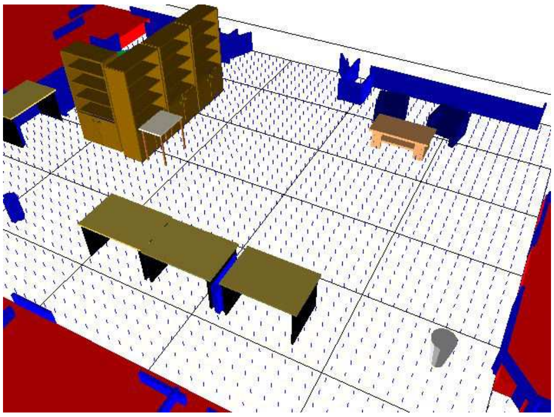


Figure 3.3: *Example of a grid covering the whole environment.*

Placing the robot on a grid allows to associate a cost to its position in the environment (not the orientation, not the dynamic of the motion). This cost measures how mush a human appreciates or dislikes the presence of the robot at a given position.

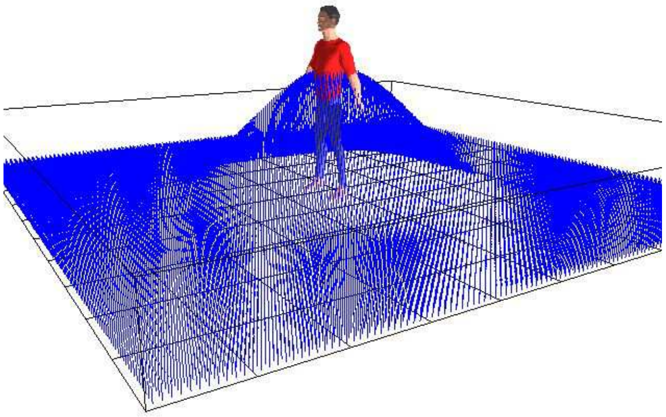
User studies on robot motion and approach direction with respect to humans [Walters 05a][Dautenhahn 06] provided us a number of properties and non-written rules and protocols [Hall 66] of human-robot or human-human interactions. From these studies, two additional criteria, named "Safety Criterion" and "Visibility Criterion" are extracted and represented by grids.

The following sections give details on these interaction criteria as well as their representation in the grids defined above.

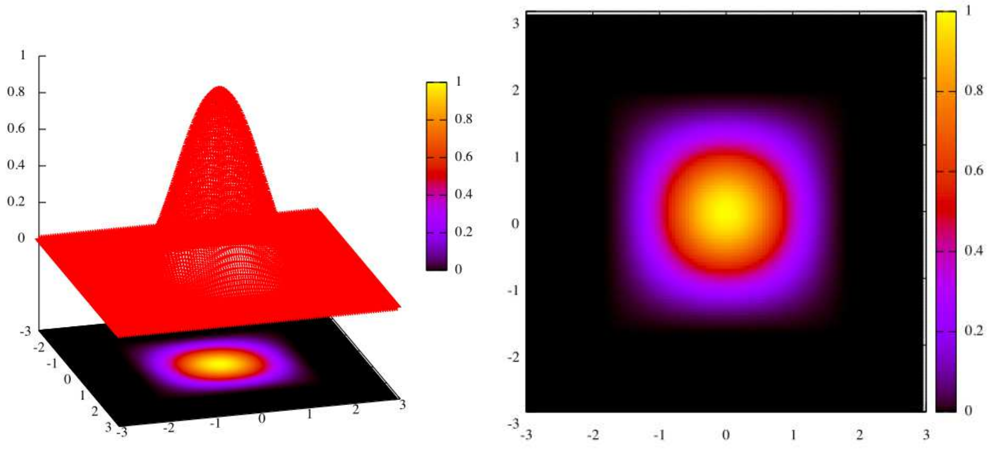
3.3 Safety Criterion

The first criterion, called "safety criterion", focuses on the safety of humans. In an environment where a robot and humans coexist, the threat to the safety of a person can be thought as parallel to the closeness of the robot to that person.

This property aims to keep a distance between the robot and the humans in the environment. However in some cases, as in the necessity of a close interaction (e.g. handing over an object), the robot has to approach the person whom it wants to interact with. Therefore, the distance between the robot and the human is neither uniform nor fixed and depends on the interaction. The feeling of safety is highly dependent on the human's personality, his physical capabilities and his actual states; for example, safety differs highly in a sitting position compared to standing. When the human is sitting, his mobility is reduced and he/she tends to have a low tolerance to the robot getting close. On the contrary, when standing he/she has a higher mobility, thus allowing the robot to come closer.



(a) The Safety Grid, containing a Gaussian cost function, is built around every human in the environment. The height of the vertical lines represents the cost associated with each cell. The nearer a point is placed from the human, the higher will be its cost.



(b) Same grid with colored 2D planar representation. The human is placed at the center of the grid.

Figure 3.4: *The "Safety Criterion" represented by the "Safety Grid"*

The Safety Criterion presents a scalar value to evaluate the safety of a point in the environment for a human. If a point (x, y) is close to a human H, it has higher value unlike a far away point which will have a smaller value. This property is treated in the current system by the "Safety Grid" G_{Safety} described as:

$$G_{Safety} = (M_{n,p}, H_1 \dots H_n, f_{safety}, Sp) \tag{3.5}$$

Each coordinate (i, j) in this grid contains a cost inversely proportional to the distance to the human. When the distance between the human H and a point (i_1, j_1) in the environment (in the grid) $Dist(H, i_1, j_1)$ is greater than the distance of another point $((i_2, j_2))$ $Dist(H, i_2, j_2)$, we have $f_{safety}(H, i_2, j_2) > f_{safety}(H, i_1, j_1)$. Since the safety concerns loose their importance when the robot is far away from the human, the cost also decreases when getting farther from the human, until some maximal distance $D_{Safety_{Max}}$ at which it becomes null. The cost function that serves as a metric for safety is defined as:

$$f_{safety}(H_k, i, j) = \begin{cases} \omega_{safety}\ g(H_k, i, j) & \text{if } Dist(H_k, i, j) \leq D_{Safety_{Max}} \\ 0 & \text{if } Dist(H_k, i, j) > D_{Safety_{Max}} \end{cases} \tag{3.6}$$

$$g(H_k, i, j) = \frac{(\cos(r \times (i - i_{H_k})) + 1)(\cos(r \times (j - j_{H_k})) + 1)}{2}$$

where r is represented by:

$$r = \frac{2D_{Safety_{Max}}}{\pi} \tag{3.7}$$

and serves to smooth descend of f_{safety} function from human to the maximal distance $D_{Safety_{Max}}$. (i_H, j_H) represents human coordinates on the grid, $Dist(H, i, j)$ is the function calculating the linear distance between the point (i, j) and the human H and finally ω_{safety} is a variable controlling the amplitude of the cost values.

Figure 3.4 illustrates a computed safety grid attached to a sitting/standing human. The height of the vertical lines represents the cost associated with each cell.

As mentioned earlier, the state of a person can have effects on his safety and comfort. While a point near a standing person can be considered safe, the same point can be evaluated as "not so safe" if the person is sitting because of the decrease of his mobility. This difference is represented in the *State* field of the person, and is reflected in grids with:

$$\omega_{safety_{SITTING}} > \omega_{safety_{STANDING}} \text{ and } D_{Safety_{Max_{SITTING}}} > D_{Safety_{Max_{STANDING}}} \tag{3.8}$$

As presented in the figure 3.5, the safety criterion difference between a standing and sitting person is reflected in grids as a cost amplitude and range difference.

In a scenario where there exist multiple humans in robot's environment, the costs of a point for each human are calculated and combined into one value with equation 3.4:

$$f_{Safety}(i, j) = max_k(f_{safety}(H_k, i, j)) \tag{3.9}$$

Once this grid is computed, searching for a minimum cost path will result in a motion that avoids moving too close to humans unless it is necessary (Figure 3.5). However, if the environment is constrained or if the task requires so, the robot is allowed to approach to people. Only very close proximity of a person is strictly prohibited to avoid collisions.

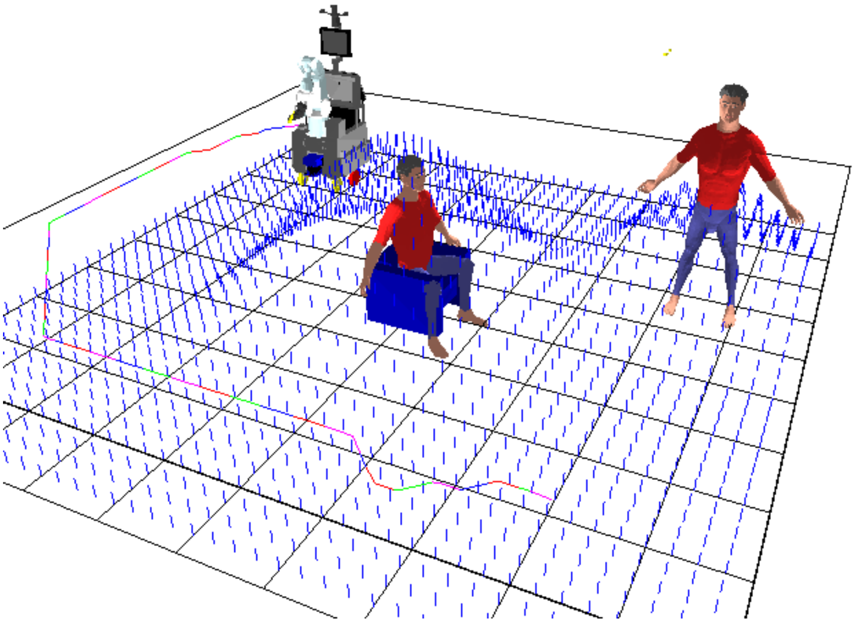


Figure 3.5: *The costs of safety grid attached to a person vary according to his state. As a sitting person will have a low mobility, the safety concerns increase as well as the amplitude and range. Finding a minimum cost path between two point in Safety Grid will result a path that maximizes the distance from humans.*

3.4 Visibility Criterion

The second criterion, called "Visibility Criterion", addresses the comfort of humans in robot's environment. What a person is looking at is an important information about the awareness of that person. Even though the awareness of a person is generally larger than the zone that is seeing, we can only be sure that the human is aware of what he/she is seeing.

Humans generally feel more comfortable when they are aware of robot's position and robot's activity. This means that a person feels safer (mental safety - comfort) when it sees the robot. In other words, when the robot is in the field of view of the person, the comfort increases.

This criterion evaluates a person's level of comfort related to his field of view. As human's attention is not the same all along his field of view, adopting a visible/invisible binary approach for the field of view does not reflect accurately the comfort. Unlike this binary approach, we represent the comfort (coming from human's field of view) of a point, as a measure of the effort required to see that point.

Like Safety Criterion, Visibility Criterion is also represented with a grid named "Visibility Grid". This grid is built according to general definition to the grids:

$$G_{Visibility} = (M_{n,p}, H_1 \dots H_n, f_{visibility}, Sp) \tag{3.10}$$

The cost of each coordinate (i, j) in the environment is calculated by the cost function $f_{Visibility}$ which represents the effort required by the human to get (i, j) in his field of view:

$$f_{visibility}(H,i,j) = \begin{cases} \omega_{vis}\Delta_{\Psi}\; g(H,i,j) & \text{if } (Dist(H,i,j) \leq D_{Vis_{Max}}) \wedge (\Delta_{\Psi} \geq \Psi) \\ 0 & \text{if } (Dist(H,i,j) > D_{Vis_{Max}}) \vee (\Delta_{\Psi} < \Psi) \end{cases} \tag{3.11}$$

$$g(H,i,j) = \frac{(\cos(r \times (i - i_H)) + 1)(\cos(r \times (j - j_H)) + 1)}{2}$$

r is expressed in equation 3.7 and serves to smooth descend of $f_{visibility}$ function from human to the maximal distance D_{VisMax}. ω_{vis} controls the amplitude of the cost values and Ψ is the angular tolerance of the angular extend of the word for a person. The human's field of view is approximately 180°, this means that a person is aware of the environment situated in front. In this case, with the assignment of $\Psi = 180°$, the points in front of the human will have no cost and be most comfortable. For a point (i,j) in this grid, Δ_{Ψ} represents the angular difference between this point and human's looking direction and is calculated by:

$$\Delta_{\Psi} = |\arccos(\overrightarrow{HP_{i,j}} \cdot \overrightarrow{L})|$$

where $\overrightarrow{HP_{i,j}}$ represents the vector from human position (i_H, j_H) to (i,j), $\overrightarrow{L}$ is the vector of human's looking direction. For example, grid points located in a direction to which the human only has to move his eyes have lower costs than positions requiring him/her to move his head in order to get the robot in his field of view. Also, for points far away from the human, the effect of the visibility must decrease.

In a scenario where there exist multiple humans in robot's environment, the costs of a point for each human are calculated and combined into one value with equation 3.4:

$$f_{Visibility}(i,j) = max_k(f_{visibility}(H_k,i,j)) \tag{3.12}$$

The computed visibility costs are shown in figure 3.6. The zone situated in front of the human has very low costs. On the contrary, the zone situated behind the human has higher costs. A minimum cost path found connecting one point to another in this grid, will maximize the visibility. A robot following this path will be as visible as possible and be less "disturbing" contributing to the comfort of the human (Figure 3.7).

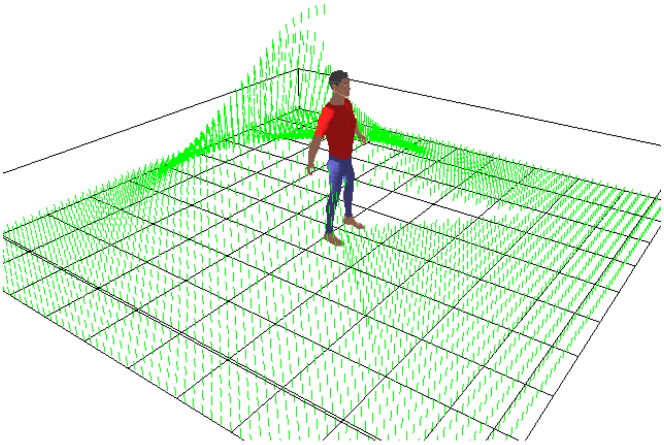
3.5 Hidden Zones

The visibility criterion, which addresses to human's comfort, takes into account only the humans and their looking directions. But in a home-like environment, the field of view of a human is partially blocked by obstacles. So the features in the environment, like walls, doors, furniture, have an important effect on what a person sees. In this case the looking direction cannot accurately match with the awareness because a person cannot[4] be aware of things obstructed by an obstacle.

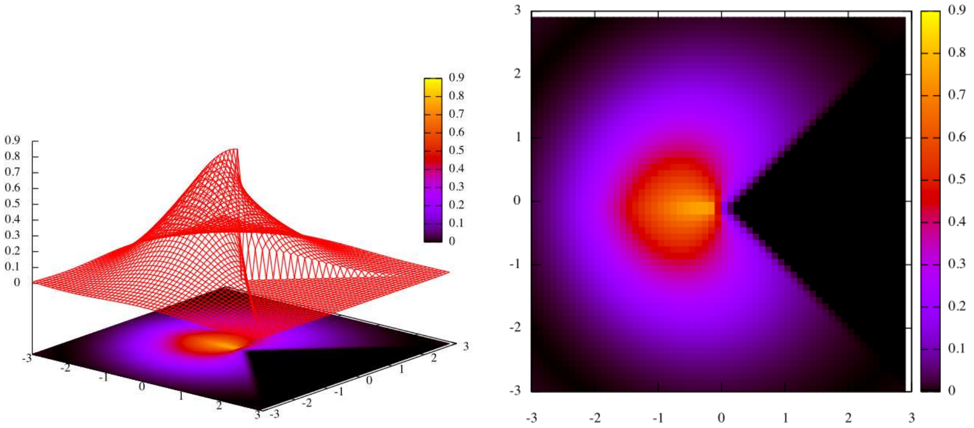
In Safety and Visibility grids, the costs are calculated without taking into account the obstacles in the environment. However, obstacles in close vicinity of the human have various effects on safety and comfort. If the robot is behind an obstacle, the human would feel secure because the obstacle blocks the direct path between the human and

---

[4] Here, we consider that a person is only aware of what he/she sees.



(a) Each point in the Visibility Grid is represented with costs reflecting the effort required by the human to get that point in his field of view



(b) Same grid with colored 2D planar representation. The human is placed at the center of the grid.

Figure 3.6: *The “Visibility Criterion” represented by the “Visibility Grid”*

the robot. So the Safety Criterion must be canceled in the zones located behind the obstacles.

On the other hand, when the robot becomes hidden by an obstacle, the visibility costs lose their meanings. To handle this issue, we introduce an extension to visibility and safety, called “Hidden Zones” criterion. This criterion evaluates points situated behind obstacles and helps to determine more accurately costs for positions hidden by the obstacles.

Another important effect of obstacles to human comfort is the surprise factor. When the robot is hidden by an obstacle and suddenly appears in the human field of view, it can cause surprise and fear, especially if it is close to the human. To avoid this effect, we must discourage the robot from appearing from the behind of an obstacle too closely, and must constrain it to enter human’s field of view sufficiently far away.

This property is reflected to grids by “Hidden Zones” grid. This grid contains costs for points situated in zones obstructed by obstacles regarding to humans. The Hidden

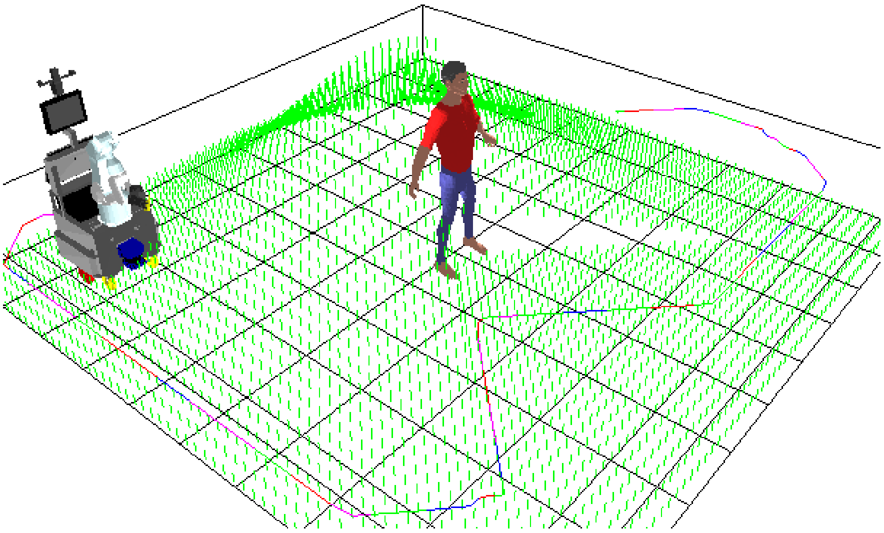


Figure 3.7: *Path found in the Visibility Grid. A minimum cost path connecting one point to an other will result a path maximizing the visibility. A robot following this path will be as visible as possible to a human thus will increase his comfort.*

Zones grid follows the same guidelines of grid construction with the addition of the environment:

$$G_{HiddenZones} = (M_{n,p}, H_1 \dots H_n, f_{hiddenzones}, Sp, \mathcal{E}) \tag{3.13}$$

the cost function $f_{hiddenzones}$ is defined by:

$$f_{hiddenzones}(H,i,j) = \begin{cases} 1 - \frac{\|\overrightarrow{HP_{i,j}}\|}{D_{HZ_{Max}}} & \text{if } Cond_{HiddenZones} \\ 0 & \text{if } \neg Cond_{HiddenZones} \end{cases} \tag{3.14}$$

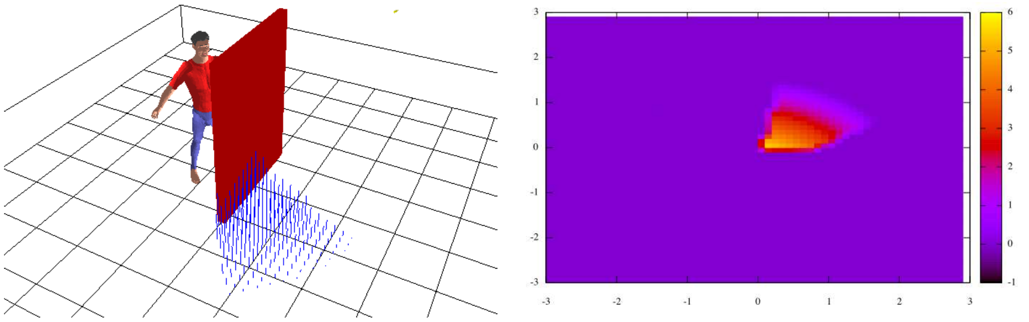
$$Cond_{HiddenZones} = (Dist(H,i,j) \leq D_{HZ_{Max}}) \wedge (\Delta_{\Psi} \geq \Psi) \wedge (\overline{H_h P_{i,j}} \cap \mathcal{E})$$

where $\|\overrightarrow{HP_{i,j}}\|$ is the distance between human H and point $P_{i,j}$ with coordinates (i,j); $D_{HZ_{Max}}$ is the minimum distance that a point is evaluated as most comfortable and is ignored. The costs calculated by $f_{hiddenzones}(H,i,j)$ in Hidden Zones grid are inversely proportional to the distance between the human and the robot (Figure 3.8) if the point P is in the limits of $D_{HZ_{Max}}$, is in the field of view of human ($\Delta_{\Psi} \geq \Psi$) and the line segment between this point and human's head ($\overline{H_h P_{i,j}}$) intersects with the environment $\mathcal{E}$ (in other words, the point is obstructed by an obstacle).

In a scenario where there exist multiple humans in robot's environment, the costs of a point for each human are calculated and combined into one value with equation 3.4:

$$f_{HiddenZones}(i,j) = max_k(f_{hiddenzones}(H_k,i,j)) \tag{3.15}$$

A robot following a minimum cost path calculated in Hidden Zones Grid will avoid to pass too near from the obstructed sides of the obstacles thus will not suddenly appear, causing fear or surprise (figure 3.9).



(a) Hidden Zones Grid contains costs situated in zones obstructed by obstacles regarding to humans.

(b) Same grid with colored 2D planar representation. The human is placed at the center of the grid.

Figure 3.8: *The "Hidden Zones Criterion" represented by the "Hidden Zones Grid"*

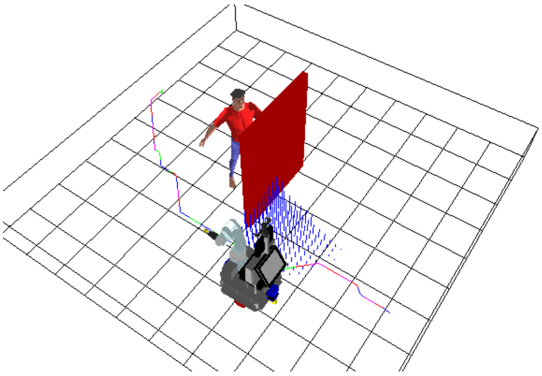


Figure 3.9: *Path found in the Hidden Zones Grid. A robot following a minimum cost path will avoid to pass too near from the hidden zones behind obstacles.*

3.6 Obstacles

As the robot is placed in an home-like environment, it will be expected that many obstacles will block its way. These obstacles can be relatively small, like furniture, or can be relatively big like the features of a building (doors, walls, etc). In order to have a safe motion, the environment cannot be ignored and should be taken into account explicitly.

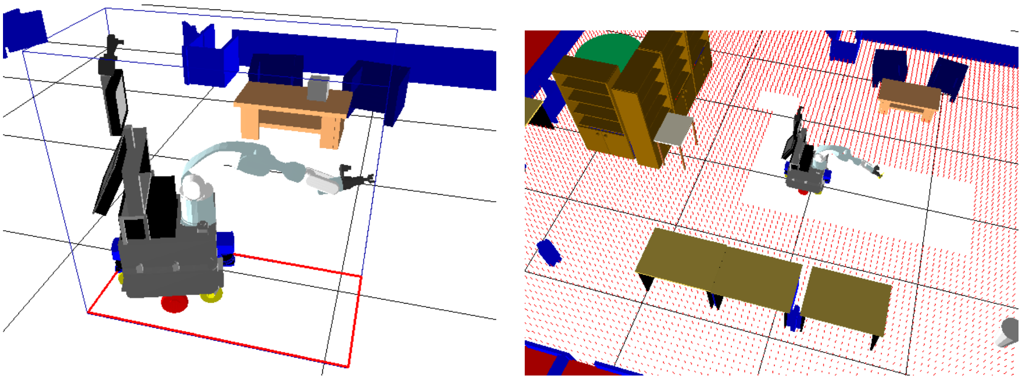
Like the interaction criteria described earlier, the obstacles in robot's environment are represented by a grid. But unlike the others, this grid does not represent a property of the interaction but only a property of the environment. So this "Obstacle Grid" does not depend on humans and their characteristics but depends on the features of the environment as well as the shape of the robot and can be defined as:

$$G_{Obstacle} = (M_{n,p}, \mathcal{R}, f_{obstacles}, Sp, \mathcal{E}) \tag{3.16}$$

where $\mathcal{R}$ represents the robot, $\mathcal{E}$ is the environment, Sp is the sampling rate and $f_{obstacles}$ is the cost function denoting the existence of an obstacle for a point in the grid.

In this grid, the cost function $f_{obstacles}$, marks the cells as an obstacle if the robot is in collision when placed into their corresponding real world positions. As the grid is a 2D structure, the collision test between the robot and the environment all along the cells can be done with the 2D projection of obstacles and 2D projection of the bounding box of the robot.

As the robot can be complicated in shape, its 2D representation should englobe the whole structure of the robot extending as far as its maximal dimension. In case of a non-cylindrical robot, a larger bounding "disc", with sufficient dimension to include the whole robot, needs to be created. So a problem arises when the collision tests are done with an upper bounding disc; if a point is found to be collision free then, its corresponding point in real environment is guaranteed to be collision free. But if a point is in collision, according to this type of collision checks, we can't be sure that in reality this point collides with obstacles. The resulting grid will have many false obstacle marks and will constrain too much the environment.



(a) The bounding box of a mobile manipulator extending its arm.

(b) Computed Obstacle Grid contains cells marked as an obstacle.

Figure 3.10: *Using an upper bounded 2D box for robot collision checks constrains the environment unnecessarily.*

Figure 3.10 illustrates this problem with a mobile manipulator robot. The robot is in a configuration where its arm is fully extended, and the bounding box calculated is big enough to englobe its whole structure. The Obstacle Grid is filled by checking the collisions between this bounding box and the environment. This resulting grid unnecessarily constrains the environment so that a path connecting the point A and B is blocked even though in reality it exists.

In order to have a better presentation of the obstacles in the grid, we use a ternary[5] function $f_{obstacles}$ that returns "collision free", "in collision" and "in possible collision" instead of the classic binary function returning if a point is in collision or not. The cost function of Obstacle Grid is defined by:

$$f_{obstacles}(\mathcal{R}, \mathcal{E}, i, j) = \begin{cases} -2 & \text{if } \mathcal{E}_{2D} \cap BB\mathcal{R}_{Min_{2D}} \\ -1 & \text{if } \neg(\mathcal{E}_{2D} \cap BB\mathcal{R}_{Min_{2D}}) \wedge (\mathcal{E}_{2D} \cap BB\mathcal{R}_{2D}) \\ 0 & \text{if } \neg(\mathcal{E}_{2D} \cap BB\mathcal{R}_{2D}) \end{cases} \tag{3.17}$$

where $\mathcal{E}_{2D}$ represents the 2D projection of the environment (basically it is where the obstacle are on a plan), $BB\mathcal{R}$ the classic 2D upper bounding box of the robot where

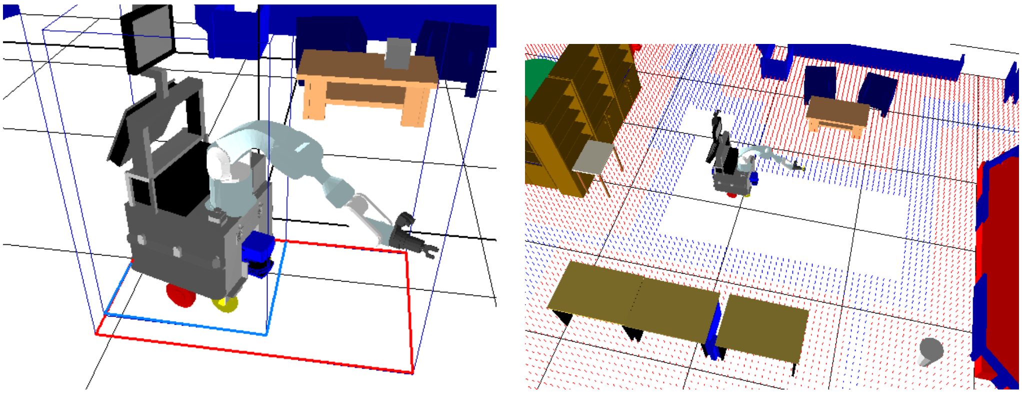
---

[5] Ternary or trinary is the base-3 numeral system. A ternary function returns 3 separate results unlike a binary function, which returns two.

$BB\mathcal{R}_{2D_{Min}}$ represents the 2D "minimal bounding box" covering only the base of the robot.

With this cost function, the grid will have 3 separate values: -2, representing a sure collision between the base of the robot and an obstacle; -1, representing a "possible" collision, where a 3D collision check is required (for further stages on planning) in order to be sure of a collision and finally 0 in the case of a point free of obstacles.

Figure 3.11 illustrates the two different bounding boxes as well as the constructed Obstacle Grid.



(a) The classic and minimal bounding boxes of a mobile manipulator extending its arm.

(b) Computed Obstacle Grid contains cells marked as an obstacle (red), a possible obstacle(blue) and obstacle free (white space)

Figure 3.11: *Using an minimal bounding 2D box covering only the base of the robot and ternary cost function eliminates unnecessary constraints and marks them as to be checked in 3D.*

3.7 Finding a Robot Path

Each of the 3 criterion grid addresses a different human safety and comfort property. A path found connecting two points in their corresponding grids will take into account the interaction property that they present. A path in Safety Grid will result a safe path, a path in Visibility Grid will produce a most visible path and a path in Hidden Zones Grid will produce a path that stays far from obstructed regions.

Even though these paths present individual properties, they can't be used individually. The robot following a path coming from Visibility Grid will ignore human's safety, unlike a path coming from Safety Grid which will ignore human's comfort.

Thus a fusion of these grids is necessary in order to find a balance between these criteria and produce safe "and" comfortable paths.

3.7.1 Combining Grids

Once the Safety, Visibility and Hidden Zones grids have been calculated, they are merged into a single grid (G_{Final} with a cost function f_{Final}) in which the robot will search for a minimum cost path. Various methods can be used to merge the grids; in this work we use two different methods of grid fusion.

The first method is to compute the overall cost from the weighted sum of the elementary costs[6]:

$$f_{Merged}(i,j) = w_1 f_{Safety}(i,j) + w_2 f_{Visibility}(i,j) \tag{3.18}$$

where (i,j) is a point in the grid, w_1 is the weight of the safety grid and w_2 is the weight of the visibility grid. With this type of combination, the weights can be adjusted according to the user's preferences.

The second method is to consider the maximum cost values when merging the grids:

$$f_{Merged}(i,j) = max(f_{Safety}(i,j), f_{Visibility}(i,j)) \tag{3.19}$$

Note that Hidden Zones Grid is not merged with the other two grids. That is mainly because the hidden zones grid serves as a replacement of these two grids for positions where the robot could not be seen because of an obstacle. The cost of a point (i,j) in the final grid is computed by:

$$f_{Final}(i,j) = max(f_{HiddenZones}(i,j), f_{Merged}(i,j)) \tag{3.20}$$

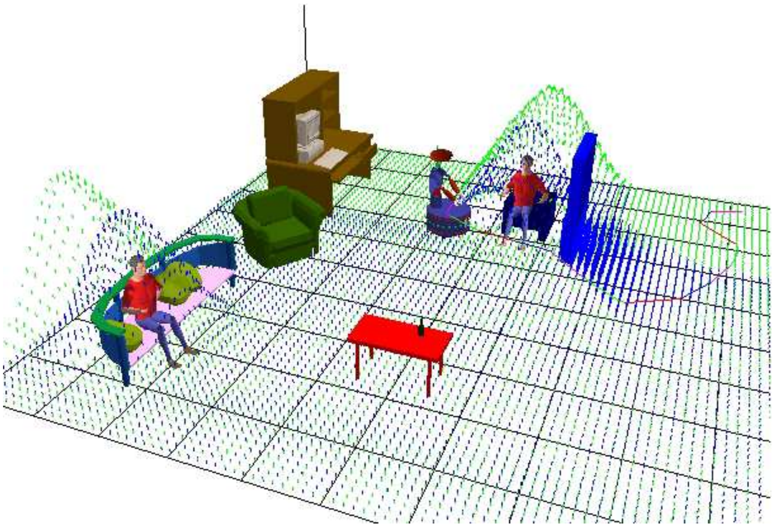


Figure 3.12: *Safety, Visibility and Hidden Zones grids illustrated in the same scenario.*

Figure 3.12 illustrates three interaction grids in the same scenario. The Human-Aware Navigation Planner can use both merging methods depending on the task and on the balance between all 3 interaction criteria. Yet, w_1,w_2 and w_3 parameters can be tuned according to the properties of the task and preferences of the user, e.g. a person familiar with robots can reduce the effect of Safety criteria because he/she is less threatened from the robot than a person having a first experience with a robot.

The final grid can also contain null costs by using null parameters. When a path is searched in such a grid, the resulting path will minimize its length and will be the shortest path. This flexibility coming from the usage of a cost based approach give us the opportunity to use the planner not only for human-robot interaction scenarios but also in classical navigation task.

---

[6] This combination method will be used all along the thesis.

3.7.2 Searching for a Path

To find a path for the robot to navigate between two coordinates in the environment, the planner searches for a set of cells in the final grid, G_{Final}, that minimizes the sum of their costs connecting the cells corresponding to these two coordinates. The minimum cost search is conducted with an $A^\star$ algorithm adapted to our problem.

The classical $A^\star$[Hart 68] is an efficient algorithm that guarantees to return a path of minimum cost whenever it exists. The costs are assigned to the edges of the searched graph and cost of a path is found by the sum of the costs of connecting cells' edges. $A^\star$ assigns a cost function $f(N)$ to a node N represented by $f(N) = g(N) + h(N)$ where $g(N)$ is the cost of the path between initial node and N, and $h(N)$ is the heuristic estimate of the remaining path's cost.

Unlike classical $A^\star$, the costs in the final grid, G_{Final}, are placed not on edges but on cells (corresponding to nodes in a graph representation). So the cost representation $f(C)$ is modified to include the cost values of the cells by:

$$f(C) = g(C) + f_{Final}(i_C, j_C) + h(C) \tag{3.21}$$

Another aspect that needs to be modified in the search algorithm comes from the treatment of the obstacles in the environment which is represented by the Obstacle Grid ($G_{Obstacle}$). As defined by equation 3.17, the cells in obstacle grid contains 3 different types tags: “obstacle”, ”not obstacle” and “possible obstacle”. The first two tags are parallel to the obstacle notion in classical $A^\star$ where they present the cells that are or not allowed be explored during the search. On the contrary, cells with “possible obstacle” tag cannot be treated in the same way and further evaluation on their collision status has to be done.

As defined in section 3.6, cells are tagged as “possible obstacle” only if a 2D collision check is not sufficient and a 3D check is necessary. The 3D collision checks for such cells are performed during $A^\star$ search. When passing from a cell (i, j) to its neighboring cell (k, l), a linear motion of the robot from (i, j, θ) to (k, l, θ) is checked against collisions. The orientation of the robot is assigned towards the destination cell (k, l): $\theta = \arctan(\frac{l-j}{k-i})$.

A very brief description of the $A^\star$ algorithm adapted to interaction grids is given in algorithm 1. Two lists, OPEN and CLOSED, are permanently kept and maintained with OPEN containing cells to be explored and CLOSE containing cells that are already explored. The function “lowest($List$)” returns and removes the cell with the lowest cost f from the $List$. The function $dispCost(C, C')$ return the cost of Euclidean displacement from the cell C to the cell C':

$$dispCost(C, C') = \begin{cases} \epsilon & \text{if } \neg(i_C - i_{C'}) \vee \neg(j_C - j_{C'}) \\ \epsilon\sqrt{2} & \text{else} \end{cases} \tag{3.22}$$

The displacement cost function returns the Euclidean distance between two cells in the grid. The value of the ϵ must be carefully chosen. The bigger value is assigned to ϵ the more important the Euclidean distance will be in path search. As our current objective is not to minimize the length of a path, but to minimize the grid costs, a small value for ϵ must be used.

A very important part of $A^\star$ algorithm is the heuristic function $h(C)$. This function calculates an estimated cost for the minimum cost path connecting the given cell, C, to the goal cell, C_{goal}. In order to guarantee that a minimum cost path will be returned by

Algorithm 1 $A^\star$ algorithm with modified cost representation and 3D collision tests by request: $A^\star(G, C_{start}, C_{goal}, h, k, G_{Obstacle}, \mathcal{R})$

```
insert($C_{start}$,OPEN);
while ¬empty(OPEN) do
  $C \leftarrow$ lowest(OPEN);
  insert($C$,$CLOSED$);
  if C = $C_{goal}$ then
    return Success;
  end if
  for all $C'$ adjacent to $C$ do
    if member($C'$,CLOSE) then
      ignore $C'$ ;
    end if
    if $f_{Obstacle}(i_{C'}, j_{C'}) = -2$ then
      ignore $C'$ ;
    end if
    if $(f_{Obstacle}(i_{C'}, j_{C'}) = -1) \wedge$ 3Dcollision($i_C, j_C, i_{C'}, j_{C'}, \mathcal{R}$)=TRUE then
      ignore $C'$ ;
    end if
    $g_{temp} \leftarrow g(C) + f_{Final}(i_{C'}, j_{C'}) + dispCost(C, C')$
    temp_is_better ← FALSE ;
    if ¬member($C'$,OPEN) then
      add($C'$,OPEN) ;
      compute $h(C', C_{goal})$ ;
      temp_is_better ← TRUE ;
    else if $g(C') > g_{temp}$ then
      temp_is_better ← TRUE ;
    end if
    if temp_is_better then
      parent($C'$) ← $C$;
      $g(C') \leftarrow g_{temp}$
      $f(C') \leftarrow g(C') + h(C')$
    end if
  end for
end while
return Failure;
```

$A^\star$, the heuristic used must be an underestimate of the remaining path, thus requires $h(C) \leq realPath(C, C_{goal})$. Even though a very low value can always be returned by the heuristic function, big differences between estimated cost value and the real cost value can cause an exploration of a larger space and reduce the speed of the algorithms. In this case the best heuristic will be the one that either returns the exact cost of the remaining path or a very close underestimate of it. The heuristic function for a cell C is defined as:

$$h(C) = \begin{cases} \epsilon(\sqrt{2}|i_{C_{goal}} - i_C| + |j_{C_{goal}} - j_C| - |i_{C_{goal}} - i_C|) & \text{if } |i_{C_{goal}} - i_C| < |j_{C_{goal}} - j_C| \\ \epsilon(\sqrt{2}|j_{C_{goal}} - j_C| + |i_{C_{goal}} - i_C| - |j_{C_{goal}} - j_C|) & \text{if } |i_{C_{goal}} - i_C| \geq |j_{C_{goal}} - j_C| \end{cases} \tag{3.23}$$

Neither the final grid nor 3 criterion grids are constructed explicitly but the values of the cells are calculated for the ones explored during $A^\star$ search. As humans in the environments can change their positions and orientations often, avoiding explicit grid construction gives us the possibility to replan a new path if a change in the environment occurs (i.e. change in human positions, orientations, or states).

Utilization of grids and a cost minimizing search allows the robot to navigate safely in an environment where humans exist by respecting the human's safety and comfort with safety, visibility and hidden zones criteria and respecting robot's safety by a collision-free motion.

3.8 Results

The Human-Aware Navigation Planner is implemented in Move3D [Siméon 01] software platform. It uses Move3D's collision checker as well as its graphical environment to simulate robot motions.

In this section, we present resulting paths generated by the Human-Aware Navigation Planner in various scenarios. A small robot with a humanoid upper body and a wheeled mobile base is placed in different environments where one or multiple humans exist. The task of the robot also varies according to the scenarios. Paths planned by the Human-Aware Navigation Planner are illustrated in five different scenarios: "Joining conversation", "Hallway Crossing", "Approaching", "Free Navigation" and "Home Environment".

3.8.1 Scenario 1: Joining a Conversation

In this scenario the robot is placed in an environment where two people are in a conversation. It's task is to approach and join them without causing any disturbance in a safe and comfortable manner. The humans have their back turned to the robot and neither of them sees or is aware of the presence of it.

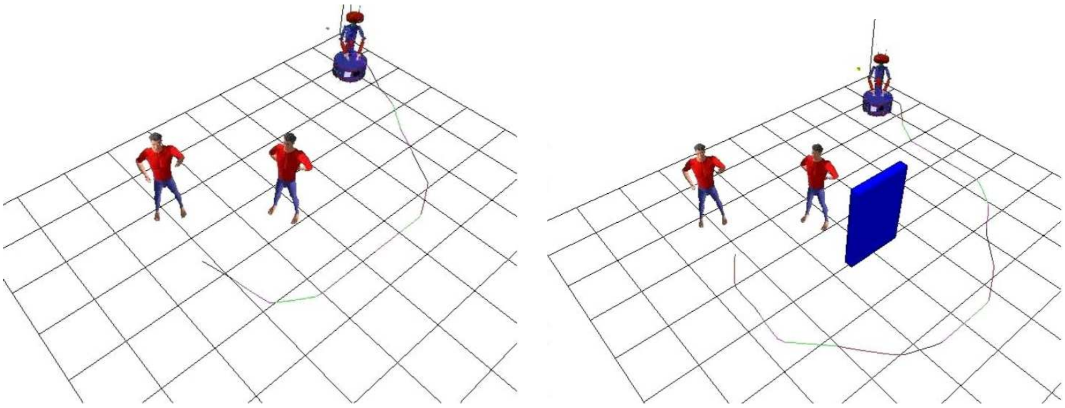
Figure 3.13 illustrates two similar situations in such context. Although the robot can take the shortest path and pass between humans, the planner calculates a path longer but safer and more comfortable for both humans (Figure 3.13.a). By following this path, the robot does not approach too close to the humans when it is invisible, and joins the conversation in a more natural way by making a frontal approach.

To illustrate the effect of obstacles in the environment, we place a wall in the same scenario, next to the human on the right (Figure 3.13.b). Although the obstacle is not blocking the path of the robot and the path is still valid from a classical planning point of view, the robot calculates a new path. Because of the obstacle blocking a part of the human's field of view, the previous path becomes undesirable by making the robot suddenly appear too close. With this new path the robot enters smoothly into the view.

3.8.2 Scenario 2: Hallway Crossing

The behavior of the Human-Aware Navigation Planner in a hallway is illustrated in figure 3.14. In this scenario, the robot and a person cross in a hallway.

The planner calculates a path to avoid a collision. Although the motion possibilities are restricted because of the environment, a friendly behavior appears. The robot avoids the human by moving to the right. After passing the human, instead of taking immediately its previous lane, the robot stays at a certain distance from the human



(a) Robot joining a conversation between 2 persons. The planner generates a path that avoids the robot to pass between humans or too near to their backs.

(b) Even the path is still feasible, it becomes uncomfortable with the insertion of an obstacle. The planner plans a path that avoids the robot to suddenly appear from behind the wall.

Figure 3.13: *Scenario 1: Joining to a conversation*

and thus avoids a possible collision from an unexpected human motion and creates less stress.

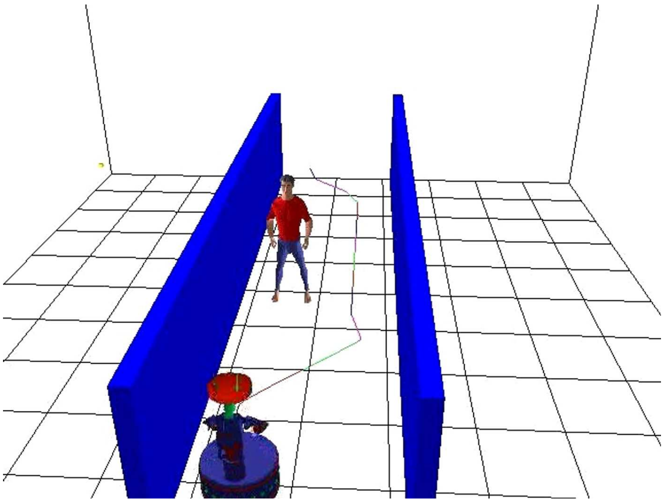


Figure 3.14: *Scenario 2: Hallway crossing. When crossed in a hallway, the robot deviates its path in order to avoid collisions. Once it passes the human, it does not immediately resume its previous lane but put a little distance in order to avoid any collision/fear. Robot's path is very similar to a catenary observed in user studies ([Yoda 97]) § 2.2.1.*

3.8.3 Scenario 3: Approaching a Person

Figure 3.15 illustrates the scenario introduced in section 3.1 with figure 3.1 with a person sitting in a room. The robot is initially located in the right corner of the room and its task is to move next to the human hidden by the wall obstacle. The wall obstacle is not only a physical obstacle but also an obstacle to the vision of the human.

The path computed by the Human-Aware Navigation Planner also illustrated in the same figure. On contrast to the path produced by a classical motion planner (Figure 3.1), this path has the following characteristics:

- The robot does not approach too close to the humans unless necessary (thus complies with signaling distance §2.2.1). It chooses a solution that only enters in the humans 3 m zone in the last portion of the path.
- The robot remains as visible as possible along the path. Because of the hidden start position, there is no possibility to be in the human field of view at the beginning of path. Therefore the planner chooses to pass behind the wall instead of passing behind the human.
- The robot is not too close to the human when it appears in his field of view. The transition from the invisible zone behind the wall to the visible one is sufficiently far from the human to avoid any surprise effect. Then the robot approaches to the human to reach its final position.

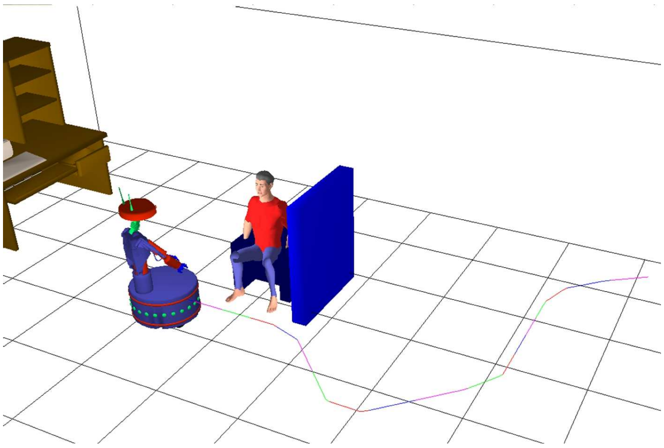


Figure 3.15: *Scenario 3: Approaching a person. The robot approaches a sitting person by avoiding to pass from his behind and by avoiding to appear suddenly behind the obstacle.*

3.8.4 Scenario 4: Free Navigation

In this scenario robot performs a free navigation from one place of the environment to another. 5 people are present in the environment, each having different positions and various orientations.

Figure 3.16 illustrates this scenario with calculated robot path. The path found has a shape that allows the robot not to pass behind the first human; not to appear suddenly for the second and the fourth and pass in front of the third person.

3.8.5 Scenario 5: Home Environment

A last example of the paths generated by the Human-Aware Navigation Planner is illustrated in figure 3.17 representing an apartment scenario with two persons: Clark (with red shirt) and Bruce (with black shirt). Four different paths are calculated between the living room and the kitchen in four different situations.

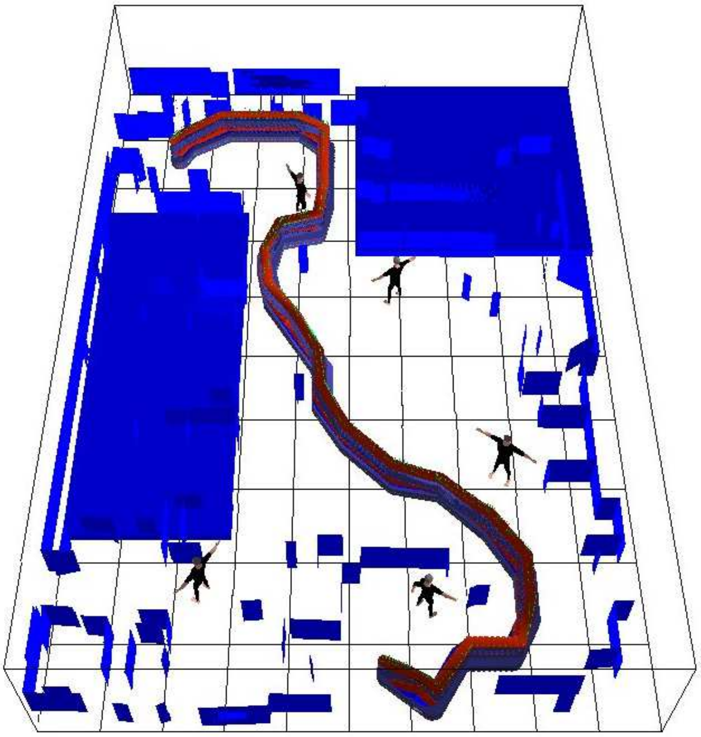


Figure 3.16: *Scenario 4: Free navigation*

In figure 3.17-a, we show the path generated by the navigation planner for a situation in which Clark orders the robot to bring a sandwich from the kitchen. The computed motion takes into account the safety and the comfort of both humans by trying to stay in the visibility fields.

We can see in figure 3.17-b a computed path that avoids “looming” from behind the kitchen wall. Instead the robot chooses a path that keeps a certain distance to this wall.

In figure 3.17-c, we can see that Bruce came to talk to Clark, so the robot calculates a different path which stays in Clark’s field of view and also avoids passing too near to Bruce’s back.

The minimum cost approach of our navigation planner allows the robot to choose an alternative path if the path is blocked by an obstacle or a person as shown in figure 3.17-d where Bruce is blocking the passage.

Human-Aware Navigation Planner is fast enough to replan and adapt its path along the execution. If a grid change occurs, like a change in human state, position, orientation or appearance of a an obstacle, fast computation times allow online replanning and a smooth switch to the new path. Table 3.1 shows the processing CPU-times on an AMD Athlon 1.8 GHz processor of the paths shown in figure 3.17 for 3 different grid resolutions.

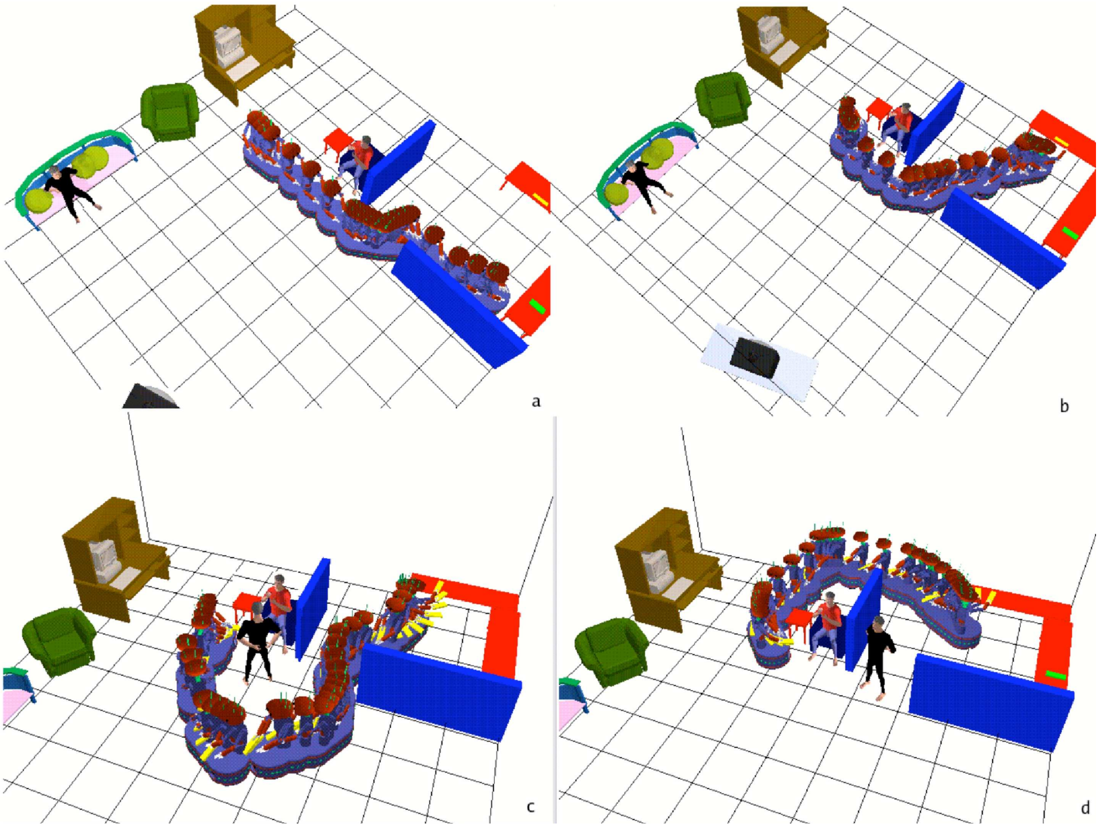


Figure 3.17: *Scenario 5: Home Environment.*

Table 3.1: *Computation times of the paths in figure 3.17*

Grid Resolution	Figure 3.17.a	Figure 3.17.b	Figure 3.17.c	Figure 3.17.d
0.2m	0.07	0.09	0.06	0.15
0.1m	0.21	0.25	0.23	0.50
0.05m	0.44	0.78	0.49	0.92

3.9 Extensions

Utilization of grids in Human-Aware Navigation Planner allows the planner to be expandable to further concepts and properties of human-robot interaction. If a new interaction property can be represented with a spatial cost function or with a pre-calculated grid, then this property can be integrated into the planner as an additional grid G_{New} with a combination weight ω_{new}. The found path will include this property with an importance of ω_{new}.

This approach gives the planner a level of flexibility to present different types of properties of human-robot interaction. In this section, 2 additional interaction properties are presented: “Human Activity” and “Human Highways”. These properties are linked to humans and to the characteristics of the environment and should be explored and evaluated further with user studies in order to include as a grid to Human-Aware

Navigation Planner.

3.9.1 Activity Representation

As mentioned in section 2.3, the activity, that a person is conducting, can contain social rules and protocols as well as geometrical constraints that may have effects to the motion of the robot.

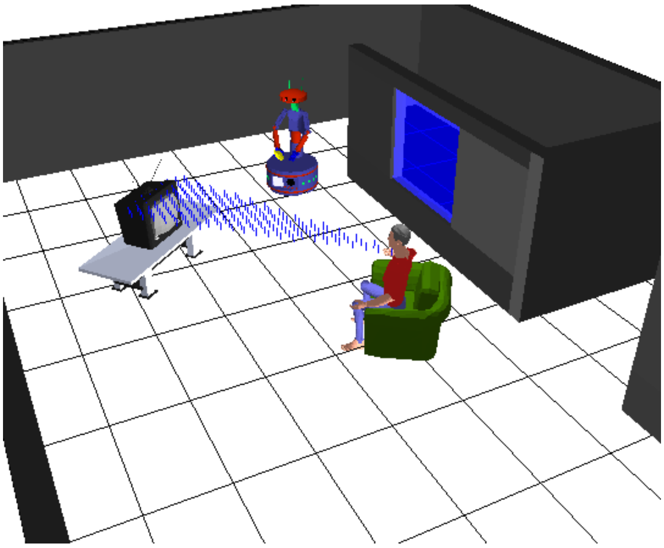


Figure 3.18: *A home environment with a person watching television. Clearly the socially aware behavior for the robot would be not to pass between this person and the television. This behavior may be obtained by placing costs between the human and the TV.*

Figure 3.18 illustrates a home scenario where a robot and a person share the environment. The activity, that the person is occupied of, is watching television. Although the motions of a person watching TV are not constraining the environment, it infers a social rule: it is not appreciated to pass between a television watching person and the television.

An example of grid containing costs according to this rule is illustrated in the same figure (Figure 3.18). Placing higher costs between the human and his point of attention will result path that avoid (as much as possible) to cross that zone.

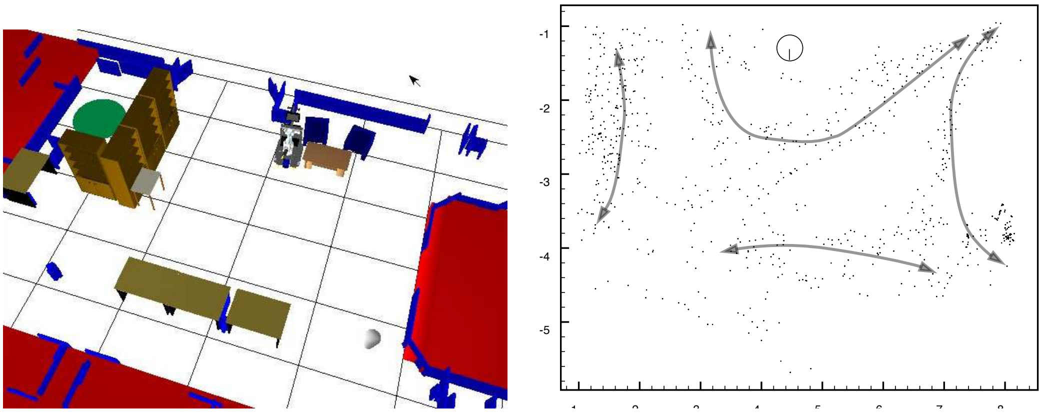
This rule is related both to the human and to the environment, specifically in this example to the TV. But other types of geometric constraints linked to human's activity may be linked only to the environment, e.g. in a theatre putting cost to the zone between the stage and the first row can avoid the robot to pass from that zone, or may be linked only to human, e.g. putting costs around a tennis playing person to avoid the robot to pass towards the field.

3.9.2 Human Highways

Another extension that we introduce for our planner is to take into account human paths that are usually taken in an environment. Depending on the shape, on the interesting points and on the features of the environment, the paths taken by the humans may show similarities in shape.

Like the approach described by Sasaki et al. in [Sasaki 06] where a human motions are tracked in a smart room, paths taken by people can be regrouped to obtain "highways" in an environment.

Figure 3.19.a illustrates the Robotic Hall of LAAS/CNRS. With robot placed next to the table, human positions are recorded with the laser of the robot and presented in figure 3.19.b. As can be noticed in this figure, some points form paths that focalize into specific parts of the environment. In Robotic Hall, these paths connect generally doors and are mostly taken by people. This notion can be integrated into the planner by representing these point with additional costs. By placing higher costs to human positions, a planned path will avoid the robot to interfere with the pat verhs usually taken by humans thus will results a non disturbing behavior. Yet an inverse approach can be used by putting higher costs to positions "not" taken by humans. This way, the robot will behave as a person by taking the same paths that people do generally.



(a) The environment *Grande Salle de Robotique* (b) Detected human positions in Grande Salle. These positions make appear paths that are generally followed by people.

Figure 3.19: *Paths generally taken by people can be integrated into the planner.*

3.10 Discussion

In this chapter, we presented a framework of the methods and algorithms to generate motion in presence of humans, which constitutes half of the algorithmic work of this thesis. These methods are assembled and form Human-Aware Navigation Planner. This planner incorporates geometric constraints extracted from user studies into the motion planning stage. Based on human/human and human/robot user studies and human/human social studies, the notions of human safety, visibility, states and preferences are defined and represented in form of cost grids. Paths generated by the planner are also illustrated in several situations and scenarios. Although these resulting paths are not the shortest ones, they are safe and comfortable for all the humans sharing the environment with the robot.

Utilization of 2D grids allows the planner to generate paths fast enough to be used in a real robot with a quasi-real home scenario[7]. During robot's motion, the planner

---

[7] The planner is integrated into two mobile robots. Chapter 5 is dedicated to the details of this integration as well as the resulting robot motions.

replans and switches to the new plan swiftly if a change in the environment or in humans occur.

One of the most important notion that we haven't taken into account and haven't reasoned about is the speed of the robot. The speed is a very important notion that has a direct effect to the safety and comfort of the human. A path cannot be considered fully safe, if the speed is left out in the reasoning: if the paths generated by the Human-Aware Navigation Planner are executed at a very high speed, it can still cause a risk to safety and comfort even though the shape of the path is satisfactory. A future step towards human aware motions, will be to include the speed of the robot in the planning loop in order to produce safe and comfortable "trajectories".

CHAPTER
FOUR

Human-Aware Manipulation Planner

This chapter introduces a motion planning framework for manipulation problems in presence of human. The methods and algorithms are materialized in Human-Aware Manipulation Planner, a planner that takes into account human safety and comfort for a “handing over an object” scenario. An introduction section (§ 4.1) opens this chapter by giving the context that we are placed on and the problem description with various examples. Section 4.2 presents roughly the 3-stage approach to the problem and gives an overview of the solution. Section 4.3 describes in detail the first stage, finding a suitable place where the transfer of the object from the robot to the human will take place. Section 4.4 shows the second stage where the planner computes a path for the object as if it is a free flying body. The final and third stage is explained in section 4.5 where the path of the robot is generated. Resulting paths are illustrated in section 4.6 with different types of robots in different scenarios. Before closing this chapter, in section 4.7, we present two extensions to improve the manipulation planner. Finally a discussion section (§ 4.8) concludes and closes this chapter.

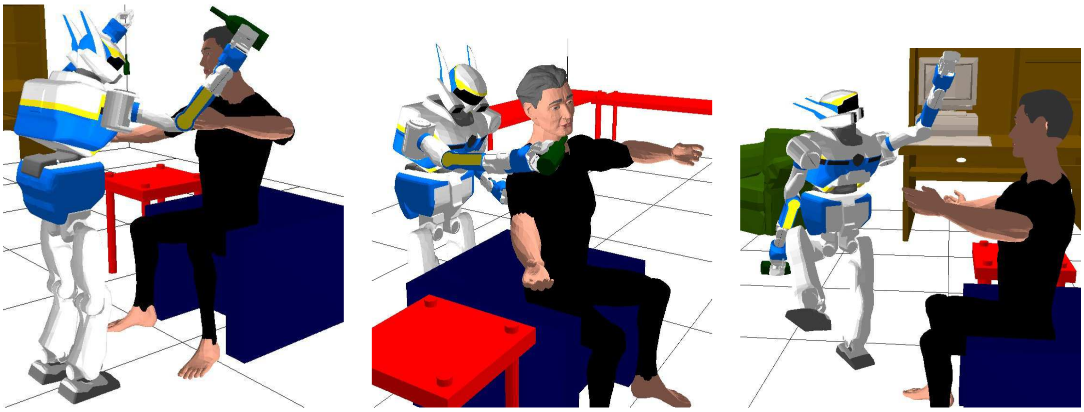
4.1 Introduction

The most important property for a robot is the ability to act and change the environment. A robot perceiving and reasoning would be a computer if it doesn’t act according to its reasoning. A manipulator robot, as its name indicates, is capable of changing its environment, displace and assemble objects with the help of its arm. This ability enables it a large field of utility for tasks requiring precision, speed, repeatability and in some cases for tasks involving a certain amount of danger.

Robot hardware becoming more and more safe and compliant will soon allow the robots and humans to work together side by side. The fade of this safety barrier will also introduce manipulator robots in our homes. We are not far from having robots that will help us in our daily lives (Examples of such robots began to emerge in recent years, like the robot helper and person carrier RI-MAN [Odashima 06] or Denso’s robot bartender).

Yet, in order to guarantee the safety of humans, safe and compliant hardware is not enough. The robots cognitive capabilities should also designed to take into account these notions. For a robot that "lives" among humans, the notion of safety gains a broader meaning and should be studied in detail.

User studies in this field showed a number of properties that need to be taken into account in robots behaviors. As the robot will be among humans, it needs not only physically safe but also respects the comfort and social rules of humans. In this chapter, we present a general framework for the motion generation to produce safe, comfortable and socially acceptable manipulation motions. We also place ourselves in a "robot handing over an object" scenario and present Human-Aware Manipulation Planner that generates comfortable robot motions for this scenario.



(a) The position of the object needs to be chosen according to the comfort of the human.

(b) Although the position of the bottle is acceptable, the robot is badly placed and not comfortable.

(c) The robot performs a motion that doesn't show its intention.

Figure 4.1: *Humanoid robot HRP-2 hands over a bottle to a sitting person.*

Figure 4.1 illustrates a scenario where a humanoid robot, HRP-2, hand over a bottle to a person. One of the important aspects of the hand over task is the choice of the place where the object transfer will occur. In order to remove the cognitive weight of the interaction, the robot should have the possibility to take the initiative. In the example, this initiative is represented by computing automatically the place of the object transfer. This place needs to be chosen not only by considering robot's accessibility but also human's safety, comfort and accessibility. Figure 4.1.a illustrates a situation where the robot chooses a "not so comfortable" place to hand the object.

Another aspect of this interaction is the choice of robot's placement and its motions. Figure 4.1.b illustrates a negative example of this choice. Even though the place of the object is well chosen, the position of the robot as well as its motion need to be considered as well. In the example, HRP-2 places itself on a position where it needs to hand over the object from the person's behind.

The legibility is another important issue in all human-robot interaction scenarios. When performing a task, the motions of the robot should be comprehensible. The human partner should understand clearly the intention of the robot without further communication. Figure 4.1.c illustrates the importance of the legibility with a non legible motion. In this example, even if the robots position and objects position will be "good", an unclear motion that doesn't reflect the robot's intention can surprise human and cause discomfort.

So in order the robot to behave in a acceptable way, the robot should not only consider the feasibility of the task but also the safety, comfort, legibility of its motions. In following sections, we describe methods and algorithms for a manipulator robot that performs human acceptable motions.

4.2 Handing Over an Object

If we think of ourselves when handing over an object to a person, we realize that reasoning about the object, our accessibility as well as target person's accessibility occurs. Before handing over, we have an idea about the rough coordinates of where our hand will reach out and where the person, whom the object is referred, will take the object. These coordinates depend not only on our reach but also on the other person's reach as well as the objects shape and the environment.

In a scenario where a person A hands and object to another person B, we call "object transfer point", the spatial point in 3D workspace where A and B will reach and hold the object together momentarily for its transfer. After the transfer, both hands will retract and the object will be on B's hand.

So it is very important that, a robot handing over an object to a human should not only take into account its kinematic structure and the object but also should consider the human part of the task. Before beginning its motion, the robot should decide the object transfer point which needs to be safe and "comfortable" for the human.

Yet finding a "good" transfer position does not completely solve the problem because the robot's motion to reach that point should also satisfy some conditions. As mentioned in previous section, robot should not only move in a safe manner but also need to ensure the comfort of the human as well as the "legibility" of its motions. In order to ensure the legibility and to make the intentions of the robot sufficiently clear, the handing over motion should be followed by complementary motions based on social rules and protocols (e.g. during the arm motion looking to the object).

To produce comfortable robot motions that take into account all the aspects mentioned above, a 3-stage approach is adopted in Human-Aware Manipulation Planner:

- **Finding Object Transfer Point:** The planner finds a safe and comfortable point for the robot to reach out with the object,

- **Calculating Object Path:** From its current position to Object Transfer Point, a path for the object is found as it is a free flying body,

- **Generating Robot Path:** With the object path obtained, the planner finalized the process by generating robot motion that will follow this path.

This decomposition allows us to reduce the complexity of the problem at the cost of the completeness of the planner. With this decomposition, the algorithm has risks of missing a safe & comfortable motion even if it exists. In practical situation, this does not cause a problem because a missing path will probably be too complicated and will require sophisticated motion which can cause the lost of its legibility.

These 3 stages are run sequentially with the output of one being the input of the next (illustrated in Figure 4.2). In case of a failure in one stage, the planner returns to the previous and rerun it with failed object point or object path forbidden. Next sections will explain in detail these stages with underlying HRI notions.

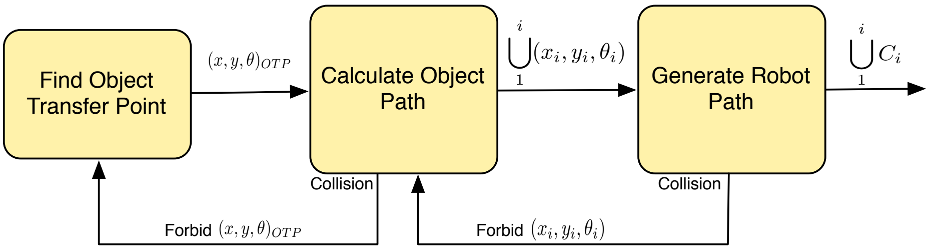


Figure 4.2: *Human-Aware Manipulation Planner uses a 3-stage approach where the output of a block is the input of the next.*

4.3 Object Transfer Point

For a hand over task, one of the key points in the planning is to decide where robot, human and the object meet. In standard motion planners, this decision is made implicitly by only reasoning about robot's and the object's structure. The absence of human is compensated by letting him adapt himself to the robot's motion, thus making the duty of the human more important and the motions of the robot less predictable.

User studies [Koay 07] and human anatomy allowed us to extract properties that should be taken into account to find a suitable point for object transfer. Like Human-Aware Navigation Planner, these properties are represented in form of grids in the manipulation planner. Although grids used in manipulation have similar structure as used in navigation, i.e. equation 3.3, there are few differences.

The main difference between a navigation grid and a manipulation grid is its dimension. Unlike the navigation where the problem can be simplified with a 2D projection of the real environment, in manipulation scenarios this simplification cannot be used because of the complexity of robot's motions. As the robot is in close proximity of a human and as its motions will involve its upper body, the environment as well as the human should be modeled in 3D.

Another difference is the number of humans taken into account by the planner. Unlike navigation planner that takes into account every human in robot's environment, as the robot is a very close proximity of the human whom it will interact, the manipulation grids takes into account only the interacted person.

With these differences, a grid for Human-Aware Manipulation Planner is defined as:

$$G = (M_{p,r,s}, H, f_{cost}, Sp) \tag{4.1}$$

where $M_{p,r,s}$ is a 3D matrix containing $p \times r \times s$ cells represented by $a_{i,j,k}$, the cost of the coordinate (i, j, k) in the grid, H the human, Sp the resolution and finally f_{cost} represents the cost function attached to the grid. The human representation H contains the kinematic structure of the human, his configuration as well as his states and preferences.

These grids are always built around the human and the human is always placed at the center of the grid. Although there is no limit for the dimensions of the grid, for practical raisons, in most scenarios 2x2x1.5 m grids with 0.1 m resolution are used. Each cell in these grids contains a cost that represents and measures an interaction property for the cell's spatial coordinates.

In Human-Aware Manipulation Planner, three different interaction properties, called “Safety”, “Visibility” and “Human arm comfort”, are represented as grids with their corresponding cost functions and are used to determine the Object Transfer Point.

4.3.1 Safety

The first of the 3 properties is the "safety". Ensuring human safety is the absolute need of any human-robot interaction scenario. It gains a higher importance in manipulation scenario where the robot places itself close proximity of the human.

Like the way that safety is modeled in HANP, in manipulation, the measure of safety of a point in human's vicinity is also highly related to its distance to the human. Each point around the human can be evaluated in a safety point of view and a cost can be associated with it: as farther a point is from human, the safer it is.

The safety property is represented with the "Safety Grid", G_{Safety}, with the cost function f_{Safety}:

$$G_{Safety} = (M_{p,r,s}, H, f_{Safety}, Sp) \tag{4.2}$$

The cost of a point in safety grid represents the measure of safety for the object placed in that particular point. As farther the object is placed from human, safer the interaction is. The safety cost function $f_{Safety}(H, i, j, k)$ is a decreasing function according to the distance between the human H and object coordinates (i, j, k) in the grid. For the object O placed in (i, j, k), which we will note as $O_{i,j,k}$, the safety cost function is defined as:

$$f_{Safety}(H, i, j, k) = \begin{cases} \omega_{safety}\ g(H, i, j, k) & \text{if } Dist(H, i, j, k) \leq D_{Safety_{Max}} \\ 0 & \text{if } Dist(H, i, j, k) > D_{Safety_{Max}} \end{cases} \tag{4.3}$$

$$g(H, i, j, k) = \frac{(\cos(r \times mDis_H(i) + 1)(\cos(r \times mDis_H(j) + 1)(\cos(r \times mDis_H(k) + 1)}{3}$$

where r is represented by:

$$r = \frac{2D_{Safety_{Max}}}{\pi} \tag{4.4}$$

and serves to smooth descend of f_{Safety} function from human to the maximal distance $D_{Safety_{Max}}$. (i_H, j_H, k_H) represents humans coordinates on the grid, $Dist(H, i, j, k)$ is the function calculating the linear distance between the point (i, j, k) and the human H and finally ω_{safety} is a variable controlling the amplitude of the cost values.

The safety function depends on the value of $mDis_H$ function. This function, defined as:

$$mDis_H(i) = min(|i_{Head_H} - i|, |i_{Chest_H} - i|) \tag{4.5}$$

represents the minimum distance between the point (i, j, k) and human head and chest in one axis. As the points in the grids are placed in a 3D world, the safety risk is evaluated according to human's head and chest which are the most vulnerable parts of the human body.

The safety function and the Safety Grid are illustrated in figure 4.3 with 0.05 m between neighboring points. It's clear that from a safety point of view, the farther the object is placed, the farther the robot will be placed, so the more safe will the interaction become.

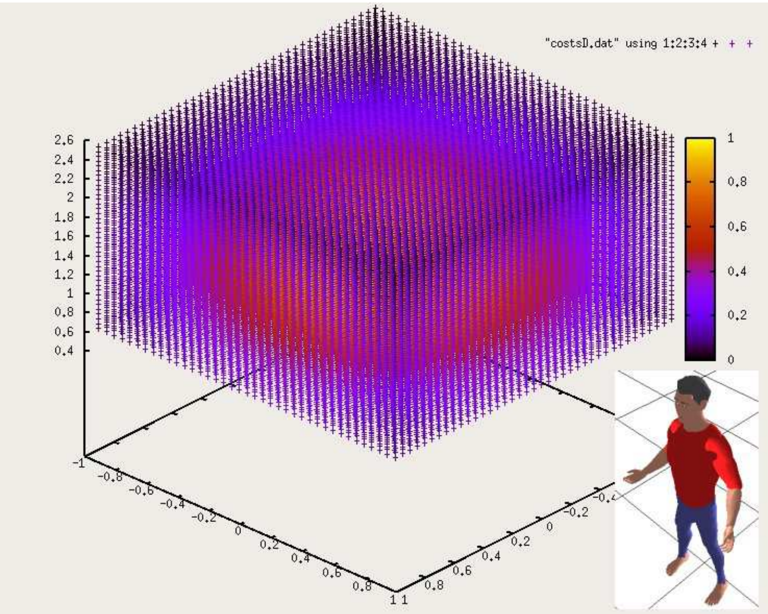


Figure 4.3: *The costs of Safety function mapped around the human at 0.05 m resolution. The human is placed at the center of this grid but illustrated at the lower corner for the clarity of the figure.*

4.3.2 Visibility

The visibility of the object is an important property of human robot manipulation scenarios. A user study[Koay 07] is conducted to understand human-robot spatial relationship. In this study a humanoid robot is assigned to hand over a can to a sitting person. The results of this study showed that 75% of study subjects preferred the robot to hand over the object in front of them where the object is totally visible. So the robot have to choose a place for the object where it will be as visible as possible to the human.

We represent this property with a visibility cost function $f_{Visibility}$. Alone this function represents the effort required by the human head and body to get the object in his field of view. If the object is placed directly in front of the human, as the object is complete visible and no effort is required, the resulting cost of objects placement will be null. On the contrary, when placed behind the human, as in order to see that object the human needs to turn his head and his body, the effort is higher, thus results a higher cost.

With a given eye motion tolerance, a point (i, j, k) that has a minimum cost is situated in the cone situated directly in front of human's gaze direction. For this property, the eye tolerance for human as well as any preferences or disabilities that he/she can have are used to compute $f_{Visibility}$ defined as:

$$f_{Visibility}(H, i, j, k) = \begin{cases} \omega_{vis}(\Delta_{\Psi} + \Delta_{\Phi}) & \text{if } Cond_{Visibility} \\ 0 & \text{if } \neg Cond_{Visibility} \end{cases} \tag{4.6}$$

$$Cond_{Visibility} = (Dist(H, i, j, k) \leq D_{Vis_{Max}}) \wedge (\Delta_{\Psi} \geq \Psi) \wedge (\Delta_{\Phi} \geq \Phi)$$

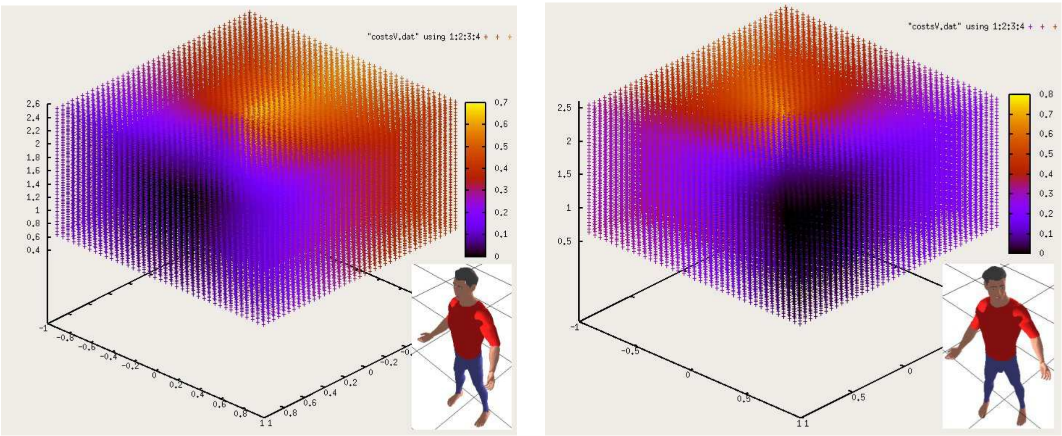
where Ψ and Φ represent the comfort limits of the eye as well as $D_{Vis_{Max}}$ the maximal distance both given as limits to the cost function. For a point (i, j, k) in this grid, Δ_Ψ and Δ_Φ represent angular differences between this point and human's looking direction and are calculated by:

$$\Delta_\Psi = |\arccos(proj_{\overrightarrow{u}}\overrightarrow{HP_{i,j,k}} \cdot \overrightarrow{L})| \tag{4.7}$$

$$\Delta_\Phi = |\arccos(proj_{\overrightarrow{v}}\overrightarrow{HP_{i,j,k}} \cdot \overrightarrow{L})| \tag{4.8}$$

where $\overrightarrow{HP_{i,j,k}}$ represents the vector from human head position $(i_{Head}, j_{Head}, k_{Head})$ to (i, j, k), $\overrightarrow{L}$ is the vector of human's looking direction, $\overrightarrow{u}$ and $\overrightarrow{v}$ are unit vectors of the environment frame and $proj_{\overrightarrow{u}}\overrightarrow{HP_{i,j,k}}$ represents the projection vector of $\overrightarrow{HP_{i,j,k}}$ onto vector $\overrightarrow{u}$.

With the cost function defined above, the Visibility Grid is illustrated in figure 4.4 with 0.05 m between neighboring points. We can see that points at the direction of human's gaze have lower costs. The more the human has to turn his head to see a point, the higher gets the cost.



(a) Human, placed at the center of the grid, is looking directly in front.

(b) Human with gaze direction towards his left

Figure 4.4: *The costs of Visibility function distributed around the human with 0.05 m resolution. The human is placed at the center of this grid but illustrated at the lower corner for the clarity of the figure. Points that the human have difficulty to see have higher costs.*

4.3.3 Arm Comfort

The last property of the placement of the object is the comfort of human's arm configuration when he/she tries to reach to the object. It is a key notion to take into account for a comfortable handing over motion. The robot should reason about human's accessibility and his kinematics to find a Object Transfer Point which is not only reachable by the human but also comfortable to reach.

Human ergonomics [Marler 05][Abdel-Malek 05] and user studies [Katayama 03] [Kolsch 03] provide very detailed work on the posture of human arm. In [Katayama 03], Katayama and Hasuura proposed 5 criteria that play more or less important roles for a human arm posture: (1) Medium joint angle index, representing the difference between joint values and medium angle values, (2) Minimum muscle activation index , representing the energy consumption and exhaustion, (3) Minimum muscle activation density index, representing the activation index divided by a cross section of each muscle, (4) Minimum joint torque index, representing the sum of toques of joints and finally (5) Minimum muscle stress index, representing the muscle tension divided by the cross section of each muscle. With studies conducted further, minimization of muscle stress proved to play the biggest role in the comfort of an arm. Kang et al. [Kang 03] showed a natural reach motion that minimizes the work done by the arm.

Although muscle stress and the work are the main property for the human arm when reaching, they are highly related to arm motion and its load and it needs a detailed knowledge of the physiology of interacted human. In our case, when we want to evaluate only a posture without any motion, these two properties do not evaluate the comfort. In Katayama's order of the 5 criteria, the "Medium joint angle index", representing the angular difference in joints for an arm posture, came up as second. For a comfort-wise evaluation of the arm posture when reaching to a point (i, j, k), we define $f_{displacement}$ as:

$$f_{displacement}(H, i, j, k) = \sum_{j=1}^{n} (\theta_{rest,j} - \theta_j)^2 \tag{4.9}$$

where θ_j is a joint angle of the jth joint, n is the number of arm joints and θ_{rest} is angle of the joint in the rest position. This function evaluates the comfort of human arm when reaching to a point in space with measuring the angular change in arm's degrees of freedom. This definition implies that the displacement cost ($f_{displacement}$) will be null when the reaching position is the resting position.

Another notion that we will take into account is the comfort related to the potential energy of the arm when performing a reaching motion. It is clear that when reaching an object it is more comfortable to reach a low point than a high point because when reaching high, the muscles need to support and bring all the weight of the arm to that point. This notion is represented with the cost function $f_{potential}$ and defined as :

$$f_{potential}(H, i, j, k) = \sum_{j=1}^{m} m_j g r_j \tag{4.10}$$

where m_j is the mass of the jth mass, m is the number of arm masses and r_j is the coordinates of the center of gravity of jth mass in environment frame.

Both of the cost functions, $f_{displacement}$ and $f_{potential}$, receive spatial coordinates of a point and return an evaluation of comfort of the arm's posture when reaching that point. This means that an inverse kinematics is applied in order to find a valid (respecting the joint limits) arm configuration that reaches to a point.

The arm of the human is modeled with 7 degrees of freedom (d.o.f), 3 on shoulder, 3 on waist and 1 placed on elbow. As the number of controlled d.o.f's (7) is greater than the dimensions of the point to reach (3), it results a redundant system with infinite number of solutions. To overcome this redundancy and solve the inverse kinematics of human arm, IKAN [Tolani 00] algorithm is used in Human-Aware Manipulation Planner. This algorithm proved to be very fast and generates ergonomic arm postures which is suitable in our case.

These 2 cost functions are merged to one function. As there is no restriction on which hand the human prefers to use, separate functions, $f_{ArmComfort_{L/R}}$, for left and right arm are defined as:

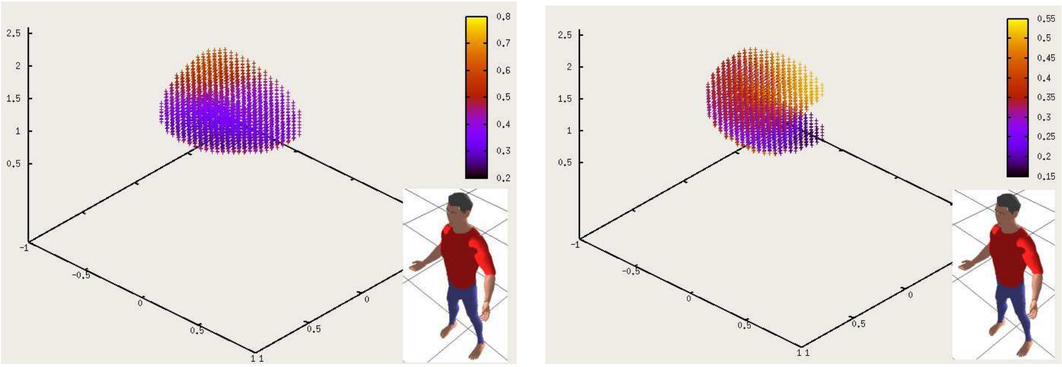
$$f_{ArmComfort_{L/R}}(H,i,j,k) = \beta_1 f_{displacement_{L/R}}(H,i,j,k) + \beta_2 f_{potential_{L/R}}(H,i,j,k) \tag{4.11}$$

where β_1 and β_2 representing weights that can be given to arm displacement and energy properties.

The separate "Arm Comfort" cost functions for left and right hand are merged and the final Arm Comfort cost function that represents the comfort of human arms represented with $f_{ArmComfort}(H,i,j,k)$ and defined as:

$$\begin{aligned} f_{ArmComfort}(H,i,j,k) = \quad min \quad & (f_{ArmComfort_L}(H,i,j,k) + P_{left}, \\ & f_{ArmComfort_R}(H,i,j,k) + P_{right}) \end{aligned} \tag{4.12}$$

where P_{left},P_{right} represents the penalties coming from left/right handedness. For example for a left handed person, his preference of reaching to a point would be with is left hand, thus the cost function will have $P_{left} < P_{right}$. The Arm Comfort functions for left and right arms for "left handed" person are illustrated in figure 4.5. Note that only the accessible and more comfortable points are shown in these figures. All other points (the unreachable points with human arm) are evaluated as not comfortable and their costs are highest.



(a) Arm Comfort Grid of human's left arm (b) Arm Comfort Grid of human's right arm

Figure 4.5: *Arm Comfort Grid for a left handed person. Although the shapes of left and right arm functions are same, a penalty is applied to the right arm thus increasing its costs. Note that only the accessible and more comfortable points are illustrated. Other points around the human have highest costs in this grid.*

With Arm Comfort cost function calculated, "Arm Comfort Grid" ($G_{ArmComfort}$) is built to contain costs representing the comfort of both of human's arms.

4.3.4 Finding Object Transfer Point

After the construction of all previous 3 grids representing the safety, the visibility and the comfort of human's arms, the final grid, called Object Transfer Grid G_{OT}, the one that the search for the Object Transfer Point, is built. Following the same definition of a grid (equation 4.1), this grid is obtained by merging the 3 previous grids and is defined with the cost function $f_{OT}(H, i, j, k)$ characterized as:

$$\begin{aligned} f_{OT}(H,i,j,k) &= w_{Safety} f_{Safety}(H,i,j,k) + \\ &\quad w_{Visibility} f_{Visibility}(H,i,j,k) + \\ &\quad w_{ArmComfort} f_{ArmComfort}(H,i,j,k) \end{aligned} \tag{4.13}$$

With the weighted sum of all 3 grids, the costs in the final grid obtain a balance between safety, visibility and human arm comfort. In order to find the Object Transfer Point, the cells are scanned and the cell with the minimum f_{OT} is assigned to be the Object Transfer Point:

$$OTP = ((i, j, k,)| \min_{i,j,k}(f_{OT}(H, i, j, k))) \tag{4.14}$$

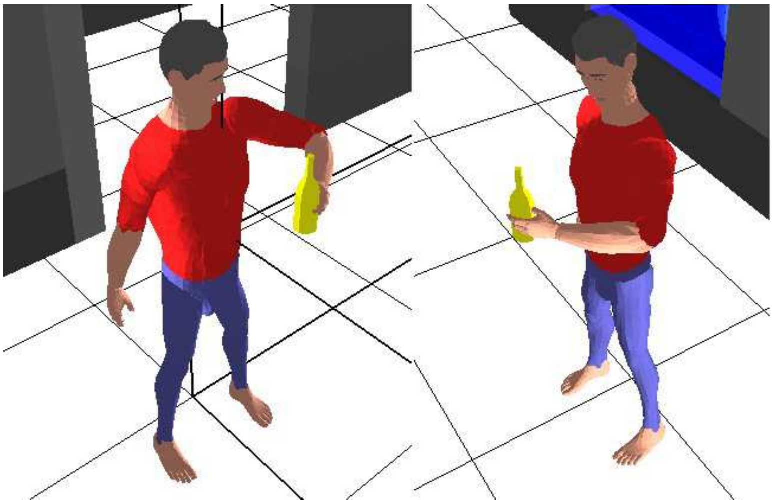
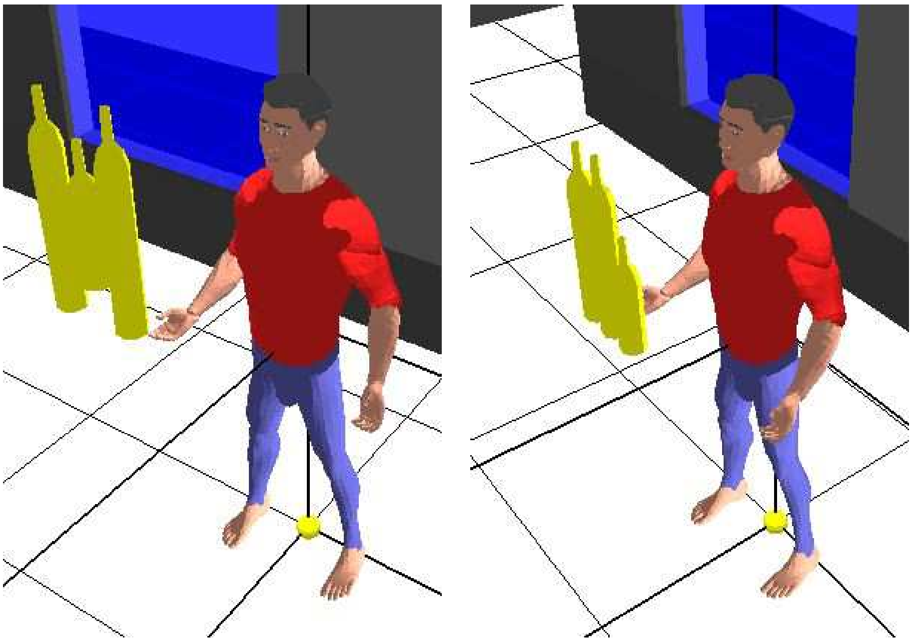


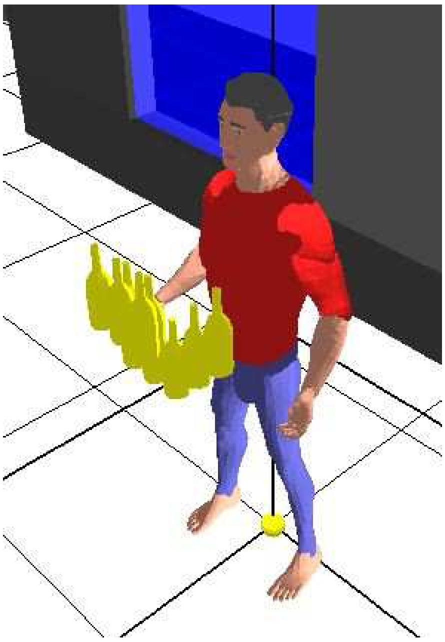
Figure 4.6: *Object's final placement, the Object Transfer Point, is found according to the minimization of safety, visibility and arm comfort cost functions. In this case, the weights applied to each f these functions are equal.*

Figure 4.6 illustrated calculated Object Transfer Point, the place where the robot will carry the object in its hand. This point is safe, visible and easily accessible to the human. Although this process depends only the human and his characteristics, combining weights of grids (equation 4.13) need to be chosen carefully. Different choices of these weights result different O.T.P's. In figure 4.7, ten points having lowest costs with different weights are illustrated. When combining grids, if $w_{Safety} > w_{Visibility} > w_{ArmComfort}$, the the resulting O.T.P will be as safe as possible (means far as possible, figure 4.7.a), unlike $w_{Visibility} > w_{Safety} > w_{ArmComfort}$ which results O.T.P visible and far (figure 4.7.a) or $w_{ArmComfort} > w_{Visibility} > w_{Safety}$ which results O.T.P accessible and comfortable for human's arm.



(a) $w_{Safety} > w_{Visibility} > w_{ArmComfort}$ With fusion weights ordered as above, the O.T.P is at the safest point, the one being as farthest possible to the human.

(b) $w_{Visibility} > w_{Safety} > w_{ArmComfort}$ In this case, the O.T.P is at the most visible point. As the safety is also more prioritized than the comfort of the arm, the O.T.P is far from the human.



(c) $w_{ArmComfort} > w_{Visibility} > w_{Safety}$ In this case, the O.T.P is accessible and most comfortable as possible for the arm of the human.

Figure 4.7: *10 points around the human having lowest costs of f_{OT}. The order of weights change drastically the result of the system.*

4.4 Object Path

The previous stage produced the Object Transfer Point, a safe and comfortable place around human that the robot will reach with the object. In this stage, a path connecting objects current position (at robot hands) to Object Transfer Point will be found. To find this path, a 3D grid built around the human, resulting from grids defined in the previous section, will be used.

In this section the object is considered as a free flying body[1], and the path found is for the object which is considered to be able to fly from its actual position to its final position. In order to compute such a path, the grid, Object Path Grid $G_{ObjectPath}$, is built with the same grid definition as previous sections and the cost function $f_{ObjectPath}$ is attached. This cost function is defined as:

$$f_{ObjectPath}(H,i,j,k) = \alpha_{Safety} f_{Safety}(H,i,j,k) + \alpha_{Visibility} f_{Visibility}(H,i,j,k) \quad (4.15)$$

where functions f_{Safety} and $f_{Visibility}$ are already obtained in the previous section, during the Object Transfer Point determination process. α_{Safety} and $\alpha_{Visibility}$ represent weights that can be attributed to cost functions in order to increase or reduce their effect.

With this definition the Object Path Grid represents a combination of Visibility and Safety Grids. After its construction, an $A^\star$ search (similar to algorithm 1 but in 3D) is used to find a minimum cost path that will be safe and visible at the same time. The $A^\star$ algorithm, described in chapter 3, is slightly modified to be used in a 3D grid where each cell has 26 neighboring cells instead of 8 in a 2D grid. The heuristics function, $h(C)$, is modified to be suitable and efficient for a 3D search and computed with the algorithm 2.

Algorithm 2 $A^\star$ Heuristics function for a cell C: $h(C)$

```
$h_{diag} \leftarrow \min(min(|i_{C_{OTP}} - i_C|, |j_{C_{OTP}} - j_C|), |k_{C_{OTP}} - k_C|)$
$h \leftarrow h_{diag} \times \sqrt{3} \times \epsilon$
if $\min(|i_{C_{OTP}} - i_C|, |j_{C_{OTP}} - j_C|) > |k_{C_{OTP}} - k_C|$ then
    $h_{2Ddiag} \leftarrow \min(|i_{C_{OTP}} - i_C| - h_{diag}, |j_{C_{OTP}} - j_C| - h_{diag})$
    $h_{2Dmanh} \leftarrow |i_{C_{OTP}} - i_C| - h_{diag} + |j_{C_{OTP}} - j_C| - h_{diag}$
    $h \leftarrow \epsilon(\sqrt{2} \times h_{2Ddiag} + 1. \times (h_{2Dmanh} - 2 \times h_{2Ddiag}))$
else if $|i_{C_{OTP}} - i_C| > |j_{C_{OTP}} - j_C|$ then
    $h_{2Ddiag} \leftarrow \min(|i_{C_{OTP}} - i_C| - h_{diag}, |k_{C_{OTP}} - k_C| - h_{diag})$
    $h_{2Dmanh} \leftarrow |i_{C_{OTP}} - i_C| - h_{diag} + |k_{C_{OTP}} - k_C| - h_{diag}$
    $h \leftarrow \epsilon(\sqrt{2} \times h_{2Ddiag} + 1. \times (h_{2Dmanh} - 2 \times h_{2Ddiag}))$
else
    $h_{2Ddiag} \leftarrow \min(|k_{C_{OTP}} - k_C| - h_{diag}, |j_{C_{OTP}} - j_C| - h_{diag})$
    $h_{2Dmanh} \leftarrow |k_{C_{OTP}} - k_C| - h_{diag} + |j_{C_{OTP}} - j_C| - h_{diag}$
    $h \leftarrow \epsilon(\sqrt{2} \times h_{2Ddiag} + 1. \times (h_{2Dmanh} - 2 \times h_{2Ddiag}))$
end if
return $h$
```

The resulting path, illustrated in figure 4.8, will be the path that the object and the robot's hand will follow. Obtaining a safe and visible path for the object helps the

---

[1] A body having free flying characteristic can move in the environment freely without being subject to constraints. Basically it can rotate and translate to any direction.

robot more easily adapt its structure to follow this path and generate a comfortable motion.

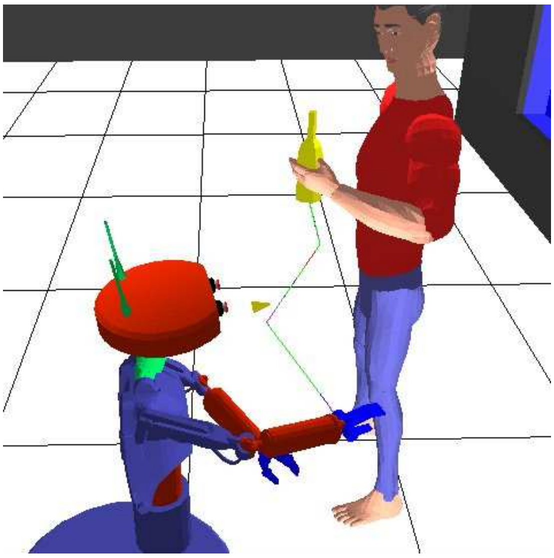


Figure 4.8: *Object's motion is planned as it is a freeflying object. This path is the result of a $A^\star$ search that minimizes Safety and Visibility cost functions all along object's motion.*

4.5 Robot Path

Even though we found a path for the object (and robot's hand) to follow, it is not enough to produce an acceptable robot motion in HRI context where the motion should be safe, comfortable and predictable. With this motion the robot must make clear of its intention.

The third and final stage of Human-Aware Manipulation Planner consists of finding a path for the robot that will follow object's motion. The object's path is computed as it was a free flying object. But in reality it is the robot who holds the object and who will make the object follow it's path.

To adapt the robot structure to the object's motion, we use Generalized Inverse Kinematics[Nakamura 90][Baerlocher 04][Yamane 03] algorithm. Although this method is computationally expensive, it has certain advantages:

- **Not dependent to the robot structure**: The Generalized Inverse kinematics method only needs a Jacobian matrix easily obtainable from robot's structure. This property makes this method easily portable from one robot to another.

- **Multiple tasks with priorities**: This method allows us to define additional tasks next to the main task. Therefore the robot not only accomplishes its task but also can takes into account additional tasks during its motion.

- **Customizable according to various criteria**: Various costs, potentials or postures can be used as additional criteria to the main task.

The general form of Generalized Inverse Kinematics (G.I.K) is defined as:

$$\Delta\theta = J^{+}\Delta x + P_{N(J)}\Delta\alpha \tag{4.16}$$

$$P_{N(J)=I_n - J^{+}J} \tag{4.17}$$

where $\Delta\theta$ represents n-dimensional posture variation vector, Δx m-dimensional high priority constraints, J $m \times n$ Jacobian matrix, J^{+} $n \times m$ pseudo-inverse of J, I_n $n \times n$ identity matrix, $\Delta\alpha$ n-dimensional posture variation vector and $P_{N(J)}$ represents $n \times n$ projector operator on $N(J)$, the null space of J.

Using a standard pseudo-inverse operator, J^{+}, causes an instability around singularities because the norm of J^{+} goes to infinity for position near singularities. To avoid this instability, we use "singularity robust" pseudo-inverse of J, $J^{+\lambda}$, defined as:

$$J = \sum_{i=1}^{r} \sigma_i u_i v_i^T \tag{4.18}$$

$$J^{+} = \sum_{i=1}^{r} \frac{1}{\sigma_i} v_i u_i^T \tag{4.19}$$

$$J^{+\lambda} = \sum_{i=1}^{r} \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T \tag{4.20}$$

where σ_i, u_i and v_i obtained from the Singular Value Decomposition (S.V.D) of the Jacobian matrix J. σ_i represents the singular values. The representation of the pseudo-inverse, J^{+}, shows the instability coming from $\frac{1}{\sigma_i}$. When σ_i goes to zero, which is the case near singularities, the pseudo-inverse results a value of infinity.

To overcome this problem of infinity, a constant value λ is added to σ_i to change the first expression of the pseudo-inverse to $\frac{\sigma_i}{\sigma_i^2+\lambda^2}$. Choosing a high value for λ will guarantee the stability but causes slow convergence on contrast to choosing a small value which speed up the convergence at the cost of loosing the stability.

The general formulation [Baerlocher 04] of Generalized Inverse Kinematics with two tasks is expressed as:

$$\Delta\theta = J_1^{+\lambda_1}\Delta x_1 + [J_2 P_{N(J_1)}]^{+\lambda_2}(\Delta x_2 - J_2(J_1^{+\lambda_1}\Delta x_1)) \tag{4.21}$$

where J_1 and J_2 are the Jacobian matrixes of two tasks, $+\lambda_1$ is the singularity robust pseudo-inverse operator, Δx_1 and Δx_2 are goal points for two tasks, and finally $\Delta\theta$ represents the resulting configuration of the robot. This definition means that the solution for the second task (the one with low priority) is chosen among the solutions in null space of J_1, solutions satisfying the task with high priority.

We use two tasks with different priorities to find an acceptable posture. The first task with higher priority contains the joints that affect the hand of the robot (shoulder, elbow,wrist,waist). This task aims to reach to a given position in object's path. The second task, with lower priority, controls robot's gaze direction (camera joints) including all the joints that affects to robot's head (waist, neck). The main purpose of this latter task is to increase the legibility of robot's motion by expressing explicitly its intention by looking at the object.

Finally, to generate robot motions that will follow object's path, the object path is divided into samples. The path sampling rate used in this stage is equal to the sampling rate of the Object Path Grid. The sampling rate is chosen arbitrarily and according to the scenarios. A higher sampling rate will better represent the object's path at the cost of slowing the planning process.

For each point of the sampled object path, G.I.K is executed and a robot posture is found to reach to that point. The robot motions between 2 samples are generated with a linear interpolation of robot configurations.

The Generalized Inverse Kinematics is an iterative method that needs to be executed until the robot converges to the target position or until a threshold is reached. With small steps and small displacement the problem becomes linear. Another problem that appears is the treatment of joint limits. As the robot is a mechanical structure that has limits for its joints, these limits need to be taken into account during the inverse kinematics computation. The limits of robot's joint are managed in G.I.K with the clamping algorithm presented in [Baerlocher 04]. The general algorithm of G.I.K including the joint limits management is illustrated in algorithm 3.

Algorithm 3 Generalized Inverse Kinematics (GIK) algorithm with joint limits management for p tasks with θ_{C_j} representing the current value and θ_{L_j} the limit value of jth joint

```
while constraints not met do
   Compute $J_i$
   $P_{N(J_0)} \leftarrow I_n$
   $\Delta\theta_0 \leftarrow 0$
   set joints state to Free
   while Clamping detected do
      $i \leftarrow 1$
      while $i \leq p$ do
         Compute SVD of $J$
         $\Delta x'_i \leftarrow \Delta x - J_i \Delta\theta_{i-1}$
         $\tilde{J}_i \leftarrow J_i P_{N(J_{i-1})}$
         $\Delta\theta_i \leftarrow \Delta\theta_{i-1} + \tilde{J}_i^{+\lambda_i} \Delta x'_i$
         $P_{N(J_i)} \leftarrow P_{N(J_{i-1})} - \tilde{J}_i^{+} \tilde{J}_i$
         $i \leftarrow i + 1$
      end while
      $\Delta\theta \leftarrow \Delta\theta_p + P_{N(J_p)} \Delta\alpha$
      set Clamping detected to NO
      for all Free joints $j$ do
         if $\theta_j$ update is over the limit $\theta_{L_j}$ then
            set Clamping detected to Yes
            $\Delta\theta_0 \leftarrow \Delta\theta_{C_j}$
            $\theta_j \leftarrow \theta_{L_j}$
            Diagonal term of $P_{N(J_0)} \leftarrow 0$
            set joint $j$ state to Locked
         end if
      end for
   end while
end while
return $\Delta\theta$
```

This algorithm contains 3 loops: first, the convergence loop verifying if the robot reached to its destination position; second the clamping loop verifying if a clamping occurred in one of the joints and third the priority loop computing the necessary angular variations respecting the priorities of the tasks.

In the examples illustrated above, we used a Jacobian matrix where we do not control the orientations of the robot's end effector and control only the position. Although the system is completely capable of controlling the 6 coordinates, the orientations are left out in order to increase the redundancy and the null space.

With this method, the robot's posture is adapted to object's motion. Although the first task (motion of robot's arm) is enough to follow the object's path, the supplementary task of moving the head helps the robot express its intention clearly, thus makes the interaction more comfortable.

At the end of this stage a path is obtained, shown in figure 4.9 for the robot which is safe, visible and comfortable to the human as we took into account his accessibility, field of view and his preferences.

The number of task can be increased to include more properties or to represent other type of interaction protocols (for example, in case of a humanoid robot, during the motion of robot's arm and its head, the robot can also point the object with its other hand to make robot's motion more legible).

The algorithm 4 makes a summary on the inner workings of Human-Aware Manipulation Planner containing all steps from finding object's placement to the robot's motion in parallel to the figure 4.2.

4.6 Results

In this section, we will illustrate the Human Aware Manipulation Planner in multiple scenarios with various robots. The planner is implemented in Move3D[Siméon 01] software platform. It uses Move3D's collision checker as well as its graphical environment to simulate robot motions.

Figure 4.10 illustrates a scenario where the robot hands over an object to a standing person. The human is looking towards the robot. The robot's base is placed in a position where the human is accessible. The planner calculates a path for the upper body of the robot by ensuring human's safety and comfort. The motion of the robot is easily understandable with its head and arm moving together.

A comparison between a standard motion planner and Human-Aware planner is illustrated in figures 4.11 and 4.12 where a humanoid robot handed over a bottle to a right handed sitting person. In figure 4.11 the robots motion is calculated with a classical motion planner (yet, the Object Transfer Point is found by HAMP). As seen in this figure, only the arm of the robot moves. The object and robot's arm enter suddenly in human's field of view with a direct motion and block right arm of the human during the movement. The intention of the robot is not easily comprehensible and the motion is not comfortable.

Figure 4.12, on the other hand, illustrated the robot path calculated by the Human-Aware Manipulation Planner. At the beginning of its motion, the robot looks at the object. To ensure the safety of the human, it pulls the object towards its body far

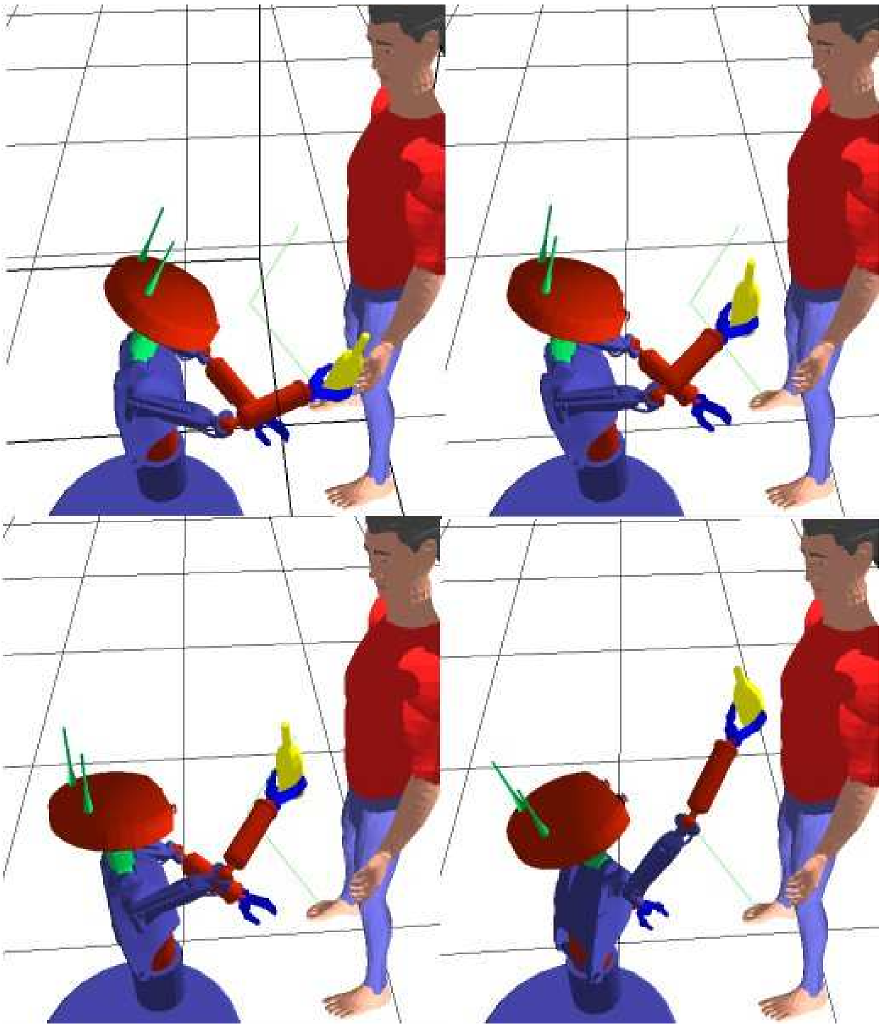


Figure 4.9: *Calculated path for a "handing over an object" scenario. The robot looks at the object during this motion, ensuring the clarity of its intention to its human partner.*

from the person. When the object enters the human's field of view, the robot begins to bring it forward until reaching O.T.P. During its arm motion the whole upper body of the robot moves and its head follows the object. This behavior ensures the legibility of robot's motion.

Human-Aware Manipulation Planner depends mainly on the human and not the robot. Usage of generalized inverse kinematics allows the planner to be used with different type of robots. On the programmer's perspective, the only information that needs to be given to the system, when changing the robot, is a list of joints that the inverse kinematics is allowed to move (the joints that we allow the robot to move during its motion).

Figure 4.13 and 4.14 illustrate a scenario with two different robots. The robots are kinematically very different, one being the humanoid robot, HRP-2, and one Jido, a mobile industrial manipulator. The goal for both of the robots is to give the bottle to the standing left-handed person. The point where object transfer will occur is found for both robots the same way because of the fact that it doesn't rely on robot

Algorithm 4 Generation of a safe and comfortable handing over path with Human-Aware Manipulation Planner

```
Compute $G_{Safety}$, $G_{Visibility}$ and $G_{ArmComfort}$
OTP_Failure ← FALSE
while ¬OTP_Failure ∧ ($Counter_{OTP} < Counter_{OTP_{Max}}$) do
  Path_Failure ← FALSE
  $Counter_{Path} \leftarrow 0$
  Find Object Transfer Point → $O_{OTP}$
  while ¬Path_Failure ∧ ($Counter_{Path} < Counter_{Path_{Max}}$) do
    GIK_Failure ← FALSE
    Find Object Path → $Path_{Object}$
    Sample $Path_{Object}$ with sampling rate $Sp$
    while ¬GIK_Failure ∧ ($i < \frac{Length_{Path_{Object}}}{Sp}$) do
      Find a robot configuration with GIK to make its arm reach to $C_i$
      if GIK Fails then
        GIK_Failure ← TRUE
        Block $C_i$ for Object Path Finding
      else
        $i \leftarrow i + 1$
      end if
    end while
    Unblock all blocked cells
    OTP_Failure ← TRUE
  end while
end while
```

structure. Even though the structures of the robots are significantly different, the planner generates a robot path for both of the robots. For HRP-2 (figure 4.13) the path generated by the planner take into account two task: follow object's path and look at the object. On the other hand, as Jido does not have a head the planner produces a path followed only by its manipulator arm.

4.7 Extensions

4.7.1 PSP - PerSpective Placement

Perspective Placement is mainly developed by Luis Felipe Marin during his thesis work.

One of the preconditions for the manipulation planner to work is that the robot should be placed in a configuration where the human is accessible and visible. This configuration should also be collision free and accessible for the navigation planner. For a robot, which is not near the human, and needs to hand over an object, the navigation planner has to linked with the manipulation planner with a suitable configuration. This configuration is found with the PersPective Placement system, PSP.

In order to interact with human, the robot has to find a configuration where it can have an “eye contact” with the human. This constraint helps to restrain the search space for a destination point of the navigation planner and can be divided into two phases: (1) finding positions that belongs to human “attentional” field of view and

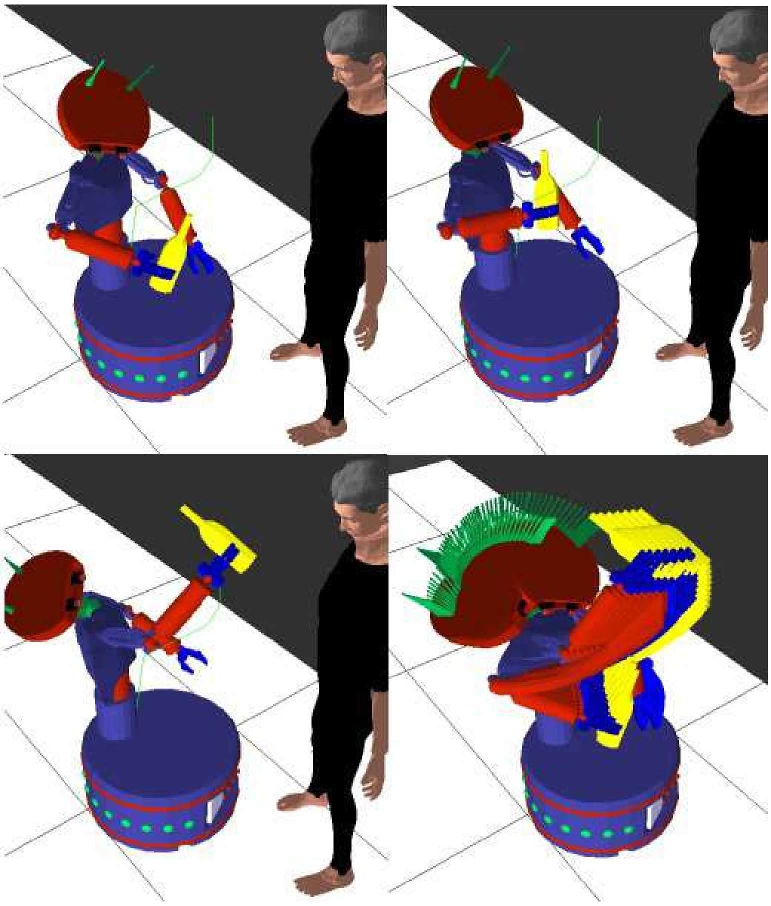


Figure 4.10: *Path planned by Human-Aware Manipulation Planner. The object placement and robot posture are safe, comfortable and also shows clearly the robot's intention.*

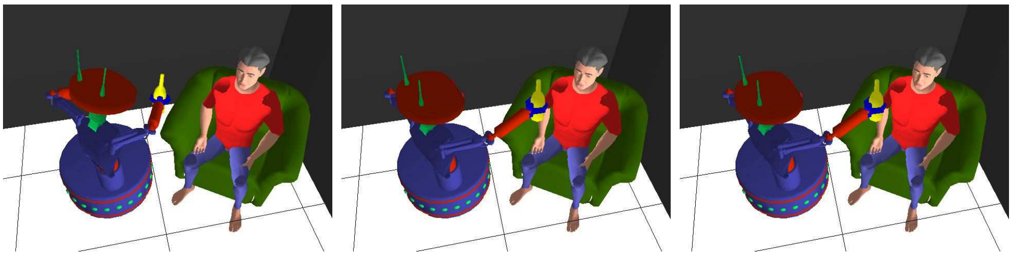
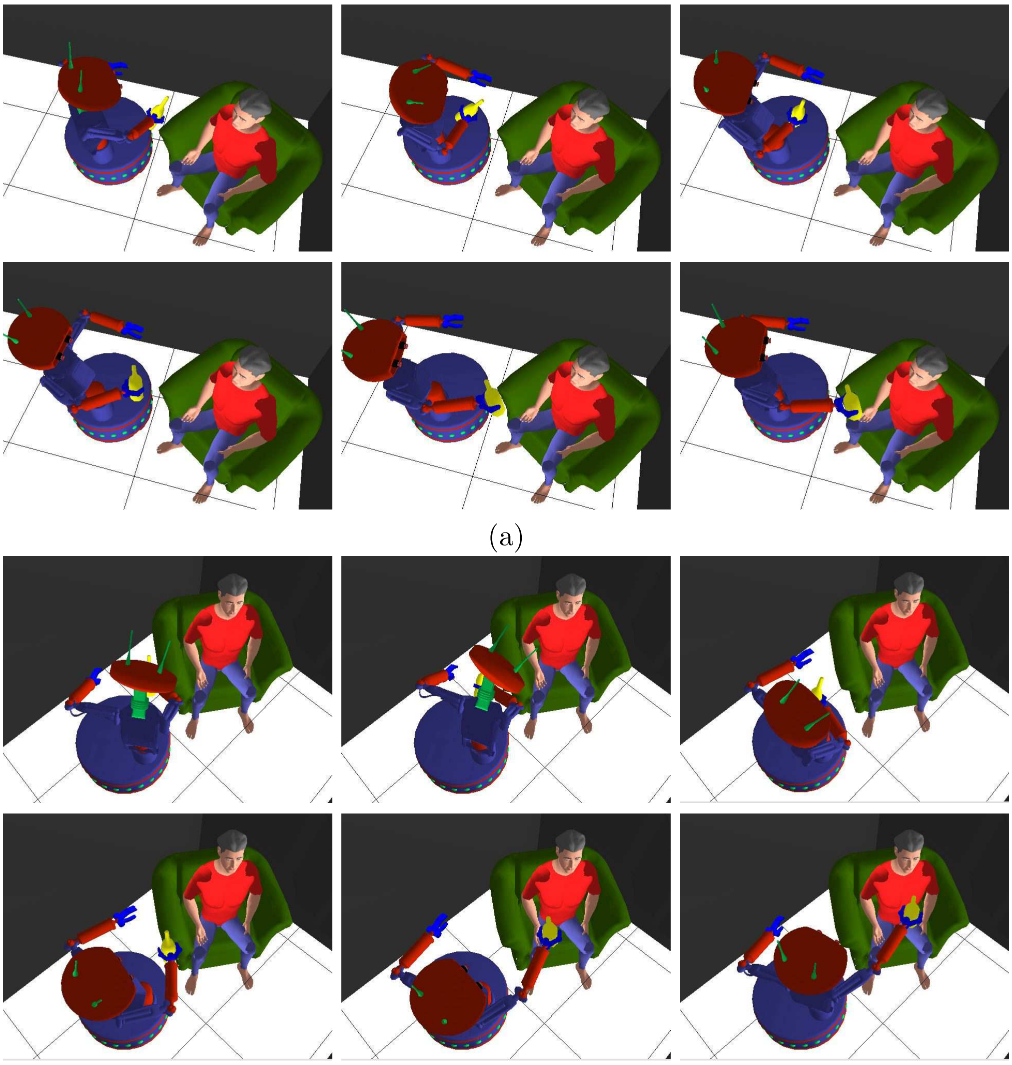


Figure 4.11: *Path planned by a "classical" motion planner. The final configuration needs to be given explicitly to the system. In these motions, robot arm makes a direct motion towards its goal, appearing suddenly in human's f.o.v and blocking his right arm.*

validating these positions in order to have a visual contact and (2) preventing big visual obstructions from blocking robot perception.

The area in human attentional field of view, called "Interaction Area", is defined as the zone in front of the person limited by an angle ($\alpha_{view} \mid 0° \geq \alpha_{view} \leq 180°$) and by a radius $Rad_{min} < Rad < Rad_{max}$ depending on the characteristics of the interaction, robot sensor capabilities and human preferences.

In figure 4.15.a, the Interaction Area is represented by a green arc and human's



(a)

(b)

Figure 4.12: *A scenario where the robot hands over a bottle to a sitting person with Human Aware Manipulation Planner. The object placement and robot posture are safe, comfortable and also show clearly the robot's intention.*

field of view out of the Interaction Area is shown red color. Once interaction area is defined, random points are generated and selected based on following properties:

- **Collision Free**: robot in this position must not have collision with objects in the environment.
- **Sensor Oriented**: one or multiple sensors must be oriented towards the human in order to perceive it.
- **Without Visual Obstructions**: in sensor's acquisition, human has to be present with a predefined percentage.

To determine what camera perceives, we use 2D perspective projection of the 3D environment. This is acquired from sensor's relative position in the space to the robot

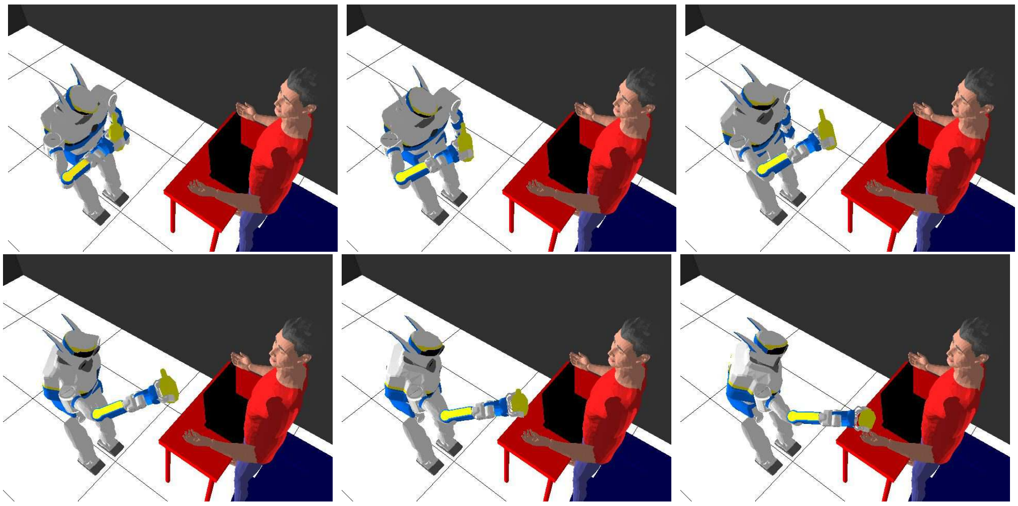


Figure 4.13: *The planner is easily portable and adaptable to different type of robots. In this example HRP-2 uses HAMP to hand over a bottle to a standing left-handed person.*

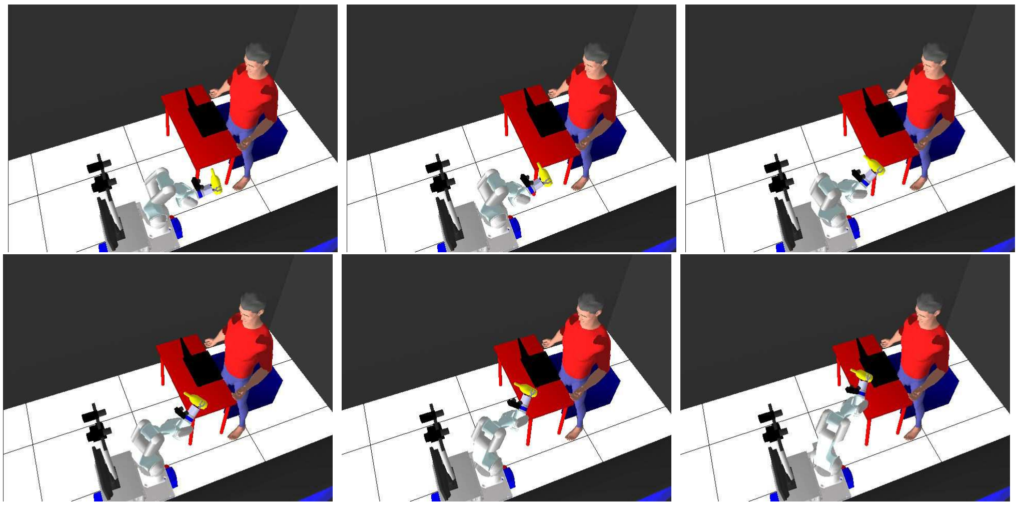


Figure 4.14: *Although Jido has a very different kinematics than the previous robots, the planner plans a path where the Generalized Inverse Kinematics solver is given only one task: follow the object path (because of the absence of a head on the robot)*

global desired position. The obtained projection is a 2D vector $MatP$ where the value of a position (x, y) represents one point in the projection image of objects in camera's field of view. In the figure 4.15, 2D projection is illustrated.

We define "relative projection" Pr as the quantity of an element of the environment represented in $MatP$, obtained by:

$$Pr(Ob) = \Sigma MatP(x, y) \mid (x, y) \in Ob \tag{4.22}$$

The relative projection of an element that is not projected Pr_{hidden} can be obtained as:

$$Pr_{hidden}(Ob) = Pr_{desired}(Ob) - Pr_{visible}(Ob) \tag{4.23}$$

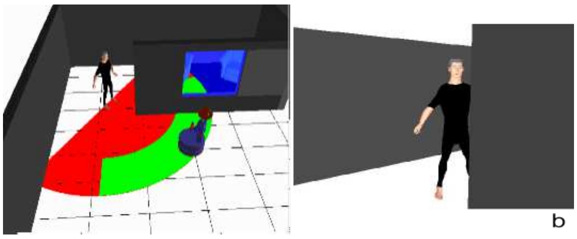


Figure 4.15: *a) Robot positioned in the interaction area (Green). b) Robot's computed perception (2D projection)*

where $Pr_{visible}$ is the relative projection that considers visual obstructions (only visible projection). On the other hand, $Pr_{desired}$ is relative projection obtained without considering objects in the environment (as it should look without visual obstacles). In figure 4.16, we can observe the difference between desired and visible relative projections.

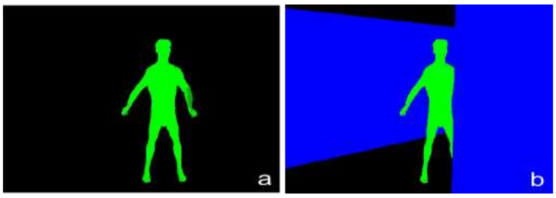


Figure 4.16: *Relative projections in green the objective, in blue and black elements and/or environment. a) Desired relative projection b) Visible relative projection (what actually the robot perceives)*

Objective's Ob visibility percentage, $Watch$, is determined by:

$$Watch(Ob) = Pr_{visible}(Ob)/Pr_{desired}(Ob) \tag{4.24}$$

Finally, the selection for a perspective placement is done with $Watch(Ob) \geq \mu$ where μ is a threshold that corresponds to a desired percentage.

To illustrate the effect of PerSpective Placement and how it links the two planners, let's take the environment illustrated in figure 4.17 containing a human and a robot. The human is in a room waiting for the robot to carry his bottle. The robot has the bottle at its hand and all it has to do is to carry and give it to human.

To find a correct placement for manipulation, the robot must see the state and the place of the human. So the first reasoning should be on where to place itself to see the human. In this example, there are 2 possible ways to do so: by looking thorough window and by entering to the room. The PerSpective Placement mechanism finds a collision free configuration in front of the window with human in the field of view. In figure 4.17, we can see this configuration with the path to reach it planned by the Human-Aware Navigation Planner.

After reaching its goal, the robot sees where the human is and finds a good start configuration (being reachable by human and visible) for the Human-Aware Manipulation Planner. PerSpective Placement mechanism then reasks to the navigation planner to produce a collision free, human aware path to reach this new configuration (figure 4.18).

PerSpective Placement is an important aspect in human-robot interaction where a task is composed of more that one action. This system interprets abstract orders coming from a supervisor and converts them as configurations for the manipulation and navigation planners thus behaving as a bridge between task planners and motion planners.

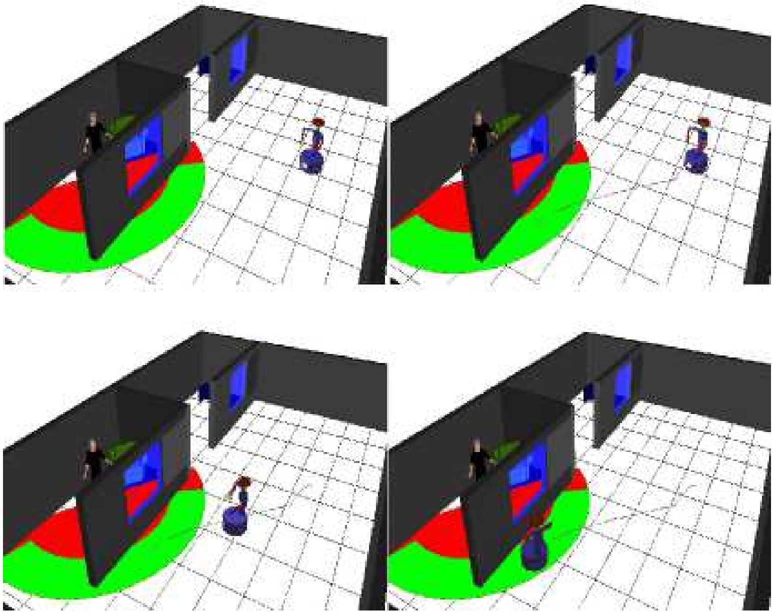


Figure 4.17: *PerSpective Placement mechanism finds a collision free configuration by the window that allows to see the human for further manipulation. The robot then plans a path towards this target configuration.*

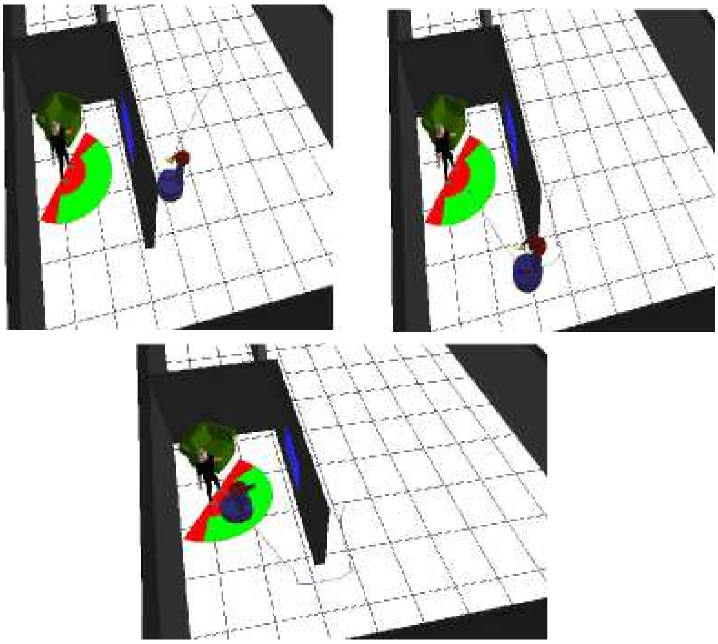


Figure 4.18: *After seeing where the human is, PerSpective Placement finds another configuration that will be suitable for manipulation.*

4.7.2 Probabilistic Approaches

A limitation of the Human-Aware Manipulation Planner is the necessity to have a fixed robot base. When the planner generates a path, the base of the robot is considered as not moving. This results a motion that only involves the upper body of the robot when handing over an object. Using the navigation and manipulation planner successively will cause the robot move towards the human, stop and then hand the object. Yet in the user study that we have conducted [Koay 07], most of the subjects evaluated a the motion of the arm during the motion of the base as most comfortable.

In order to have a coordinated arm and base motion, we propose to use probabilistic approaches at the final stage (finding robot path to follow object path) of our 3-stage hand over algorithm.

Let's consider a scenario with a robot being 3 m far from the human. At that distance the human is not reachable and a base motion is necessary.

The first and second stage of the hand over algorithm, where the Object Transfer Point is found and a path for the object is calculated, can be applied directly by extending the grids to a greater distance to cover all the space between the robot and the human. However, the final stage where the robot path is found, this cannot be applied because of the lack of robot base motions.

At this stage, the problem that we consider can be interpreted by "following a pre-computed robot gripper path". With this interpretation the problem becomes a closed chain motion planning problem where the end effector of the robot is subjected to constraints.

Among different methods proposed in the literature we have adopted an RRT based approach [Tang 07] [Oriolo 02] to coordinate base and arm motions.

The algorithm 5 describes the method that can be used for such a problem. In this algorithm, $RANDOM\_CONF(Path_{Object}(i))$ is a function returning a random

Algorithm 5 Following an end effector path using RRT based algorithm

```
Sample $Path_{Object}$ with sampling rate $Sp$
$i \leftarrow 0$
while $i < \frac{Length_{Path_{Object}}}{Sp}$ do
    $j \leftarrow 0$
    while $\neg(FAILURE \vee j < MAX\_ITERATION)$ do
        $q_{base_{rand}} \leftarrow RANDOM\_CONF(Path_{Object}(i), \mu)$
        $q_{near} \leftarrow NEAR\_NODE(q_{base_{rand}}, G)$
        $q_{new} \leftarrow GIK(q_{base_{rand}}, Path_{Object}(i), FAILURE)$
        $j \leftarrow j + 1$
    end while
    $ADD\_NODE(q_{new}, G)$
    $ADD\_EDGE(q_{new}, q_{near}, G)$
    $i \leftarrow i + 1$
end while
return $G$
```

configuration for the base of the robot staying in a pre-defined perimeter μ from the (x, y) coordinates of the ith sampled point on object's path, $Path_{Object}$. The function $NEAR\_NODE(q_{base_{rand}}, G)$ return the nearest node to the configuration $q_{base_{rand}}$ accessing $i - 1$th point in object's path in the graph G.

With this algorithm the robot's base and arm motions are coordinated and satisfy the result of mentioned user study. The biggest challenge, yet to be solved, in this algorithm is the choice of base position. A fully randomized choice results unnecessary robot movements as well as orientation changes and causes a severe impact to the legibility issue. One solution to this problem can be to select base configurations not in workspace but in control space by minimizing robot rotations, but yet still stays to be explored.

As the base of the robot is moving with this approach, we can also use the cost grids of the Human-Aware Navigation Planner in the choice of base configuration. Jaillet et al. [Jaillet 08] proposed an approach to integrate cost spaces into the RRT algorithm. This method, proved to be efficient in terms of quality of the paths, is yet to be explored.

4.8 Discussion

In this chapter, we presented a general framework for manipulation planning in human presence. The presence of humans in robot environments forces the robot to obey social rules and to generate motions that are not only safe but also socially acceptable. Besides the notions of safety and visibility, we introduced the notion of "legibility" into the motion planning to make the robot clearly express its intentions to the human partner.

We presented a Human-Aware Manipulation Planner designed for object hand over tasks. Dividing the problem into 3-stage allowed to simplify the problem at the cost of the completeness of the planner. One of the planners novel characteristics is to find automatically the position where the object transfer between the robot and the human will take place. This place, called Object Transfer Point, is calculated according to human's safety, visibility and comfort. With this property breaks the human-centric of the human-robot interaction and allow the robot to take the initiative (at least in an object hand over scenario).

Using Generalized Inverse Kinematics at the last stage of the algorithm allows the planner to integrate not only an arm motion but also a full upper body motion that increases the legibility.

We illustrated a number of simulation results to demonstrate our planner in various scenarios.

We believe, the two extensions represented in this chapter will contribute greatly to the efficiency of the planner as well as the quality of the resulting paths.

The major contribution of the Human-Aware Manipulation Planner is to transform the motion planning problem of finding a path from one configuration to an other, to "finding a path for a task".

CHAPTER
FIVE

Integration to a Real Robot

This chapter describes the integration of the Human-Aware planners into two real robotic platforms. Section 5.1 opens this chapter by stating the challenges of this integration on robots where multiple hardware and software components exist. Brief descriptions of these two robots, Rackham and Jido, as well as their control architectures fill the section 5.2. Section 5.3 describes MHP, Motion in Human Presence, module that encapsulates the navigation and the manipulation planners. The internal architectures as well as their places in the global architecture are illustrated. Two supporting modules, HumPos, human detection & tracking; and Xarm, soft trajectory planner, are described in section 5.4 since they provide and execute inputs and outputs of MHP. Section 5.5 illustrates the results of the planners on various scenarios with real robots. Finally, this chapter is concluded with a discussion section (§ 5.6).

5.1 Challenges

The integration of multiple modules, that manage, interpret and reason on various data coming from multiple sensors, is a practical challenge of any robotics system. A robot, that will interact with people and is expected to localize, detect and recognize its environment, should be equipped with software interpreting the data of the sensors. Such a robot is also expected to reason about the situation according to the acquired data, therefore it also needs to be equipped with reasoning software. Finally in order to act, move and manipulate according to its reasoning, the robot needs to be equipped with necessary hardware and software to manage the whole system..

An HRI robot, that will interact with people, has to include a large number of software and hardware components all inter-communicating. One of the challenges that needs to be overcome is to manage such a complex system efficiently and to allow sufficient level of flexibility, extensibility and efficiency.

In order to achieve a well working system, not only the quality of its individual components, but also the quality of their architecture and their management matter. The design of the architecture plays an important role in the design of the system.

Another important aspect of the integration is data interpretation. For example, for an HRI robot, probably containing multiple modules[1] of perception each dedicated to different objectives (e.g. one modules for human face detection, one module for human leg detection, etc.), fusion of the data coming from multiple sensors plays an important role to obtain a single coherent and unified result (e.g. unifying human head, legs and arm to obtain a human model).

This chapter presents the integration of the Human-Aware planners into two robotic platforms as well as the robot motions resulting from this integration. We also describe the control architecture and additional modules to provide a better understanding of this integration.

5.2 Robots and Architecture

The Human-Aware Navigation and Manipulation Planners are integrated to Rackham and Jido[2], two robotic platforms in LAAS/CNRS. Despite of their different structures and capabilities, the software architecture of these two robots is based on the LAAS architecture [Alami 98] having multiple Genom [Fleury 97] modules. The LAAS architecture, developed incrementally for many years, provides a great level of modularity and genericity that ease the programming load of integration. This architecture, originally decomposed in three layers (functional, decisional and executional layers) is revised to better adapt to HRI and transformed into an architecture of two layers:

- **Functional Layer** : This layer, also called the "low layer", contains the whole perception and action functions of the robot. Control loops and data interpretations are encapsulated into Genom modules. The modules in this layer have direct access to robot's hardware components (i.e sensors and motors) and offer services controllable via requests. Each module communicates by publishing its own attributed memory block, called "poster".

- **Decisional Layer** : The decisional layer, or the "high layer", contains components that provide decision capabilities to the robot. A task planner, a supervisor and a fact database are situated in this layer. The task planner, called Human-Aware Task Planner [Montreuil 07], is specifically designed for HRI scenarios where the robot and a human shares a joint task. The planner generates symbolic plans and sends it to supervisor. The supervision system [Clodic 07b], called SHARY[3], manages the execution of the received plan by the functional level.[4]

The following two sections will briefly describe architectures of Jido and Rackham as well as some of their underlying modules.

---

[1] In this chapter, a "module" will always represent an individual software component in a robot control architecture.

[2] The name Rackham comes from the pirate Rackham the Red in Tintin comic strip because of its color and missing eye. The name Jido is believed to come from Japanese and yet no Japanese speaking person could recognize it.

[3] SHARY stands for Supervision for Human Adapted Robot Y(I)nteraction.

[4] Although the planner and the supervisor are two very important steps towards a unified "Human-Aware" architecture, they are not within the scope of this work and will not be detailed further in this report. Yet, we refer the reader to [Montreuil 07],[Clodic 07b] and [Clodic 07a] for detailed descriptions of these systems.

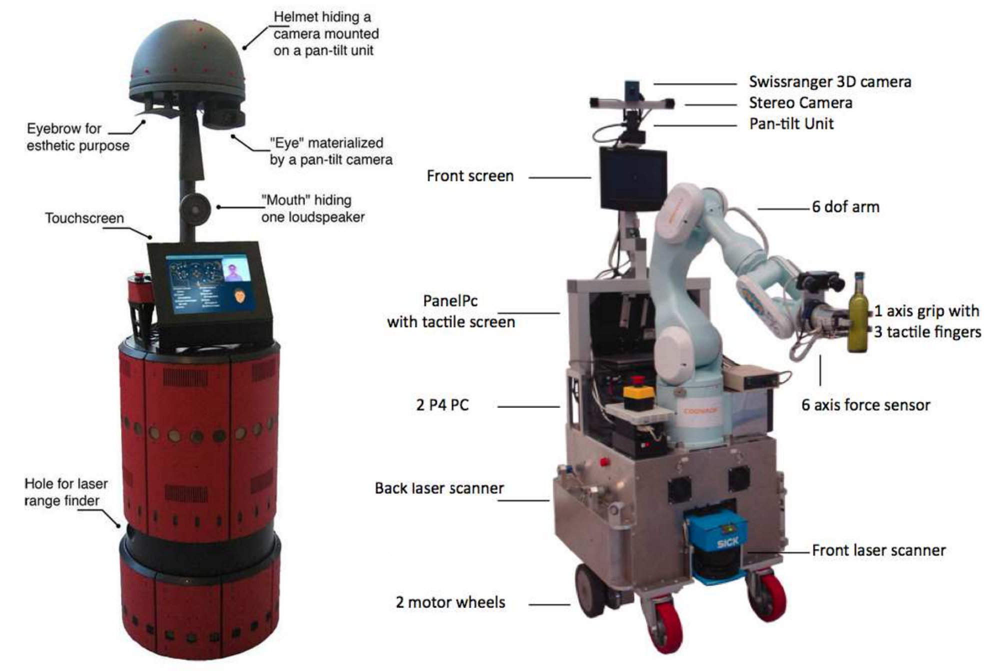


Figure 5.1: *Rackham (on the left) and Jido (on the right) robotic platforms that the motion planners are integrated.*

5.2.1 Rackham

The first robot that we would like to present is Rackham (figure 5.1). Rackham, being a B21r robot (iRobot), is a 52 cm wide and 118 cm tall cylinder topped with a mast supporting a kind of helmet. It integrates 2 PCs (one mono-CPU and one bi-CPUs running P3 at 850 MHz). The standard equipment is extended with a pan-tilt Sony camera EVI-D70 attached under the helmet, a digital camera mounted on a Directed Perception pan-tilt unit, an ELO touch screen, a pair of loudspeakers, an optical fiber gyroscope and wireless Ethernet.

Rackham is designed to be a guide robot and its main capabilities are people detection and navigation. Visual Human detection and tracking are performed by ISY [Germa 07] module. This module detects and provides positions of humans in the environment. Laser based human detection is performed by HumPos module, which will be explained later in this chapter. To execute the plans generated by the navigation planner, the robot uses the SFL module [Philippsen 04], a module that generates an executable trajectory from the received path.

5.2.2 Jido

The second robot on which the planners are integrated, is Jido (figure 5.1). Jido is a MP-L655 platform from Neobotix, equipped with a Mitsubishi PA-10 arm (with 6 degrees of freedom). Several sensors are available on the platform: sonars, 2 laser range finders, two stereo camera banks (one mounted on the arm and the other on a pan-tilt unit on the base platform), several contact sensors and a force sensor on the gripper. Four on-board computers (Two Intel Pentium 4 processors, at 3GHz, one panel pc and one Intel Pentium Core 2 Duo CPU at 2GHz), using the Linux operating system, provide processing power to the robot.

Jido is designed to be a home-helper robot thanks to its many sensors, mobility and manipulability capabilities. With over thirty modules running simultaneously, Jido is far more complicated and capable than Rackham. In order to make clear the vocabulary used in following sections, among these thirty modules five most important are as follows:

- GEST : This perception module is in charge if detecting and tracking human head as well as both hands by using the top camera of Jido. The detection of human parts are performed using their shapes, colors and positions [Fontmarty 07]. The working range of this module is ≈ 3 m.
- HumPos : HumPos modules detects humans though the robot's front laser sensor. This module will be explained in the next section since it represents an important part of the human detection process.
- Xarm : The plans generated by the manipulation planner are executed by this module. Xarm generates trajectories with bounded jerks for Jido's arm. This module will be explained in the next section since it represents an important part of friendly arm motions.
- PILO : Unlike Rackham, the paths of the base of Jido are transformed into trajectories and executed by PILO module [Fleury 95]. This module receives passage points as input and generates trajectories consisting straight lines and clothoids.

5.3 Motion in Human Presence (MHP) Module

The Human-Aware Navigation and Manipulation Planners are integrated into the LAAS architecture as single Genom module. As both of the planners rely on Move3D and share common/similar grid representations, they are implemented and represented in the architecture as one module named MHP, Motion in Human Presence. Inside the module, these two planners work independently sharing the same environment and communication through PerSpective Placement (§ 4.7.1) mechanism. Besides the computation and programming advantages, the choice of reassembling these three systems together allow them to share the environment including robot and human models, thus removes a possible risk of their mismatch.

MHP module is situated in the functional layer of the robot's architecture. It communicates directly with other modules, notably the execution and perception modules, via posters. The module contains a number of requests that enable the supervisor or the user to launch and to control the planning.

Figure 5.2 illustrates MHP with its three components, HANP, HAMP and PSP. The requests generated and sent by the supervisor are too abstract for the planners to execute. PSP behaves as an intermediate level between the supervisor and the planners. It transforms the high level requests of the supervisor to more concrete commands for planners. For example, the supervisor sends the request `GO TO THE HUMAN H`, yet this request stays too abstract to HANP because it needs to have a robot goal position (x, y, θ) to plan a path. That is why PSP reasons on the task asked by the supervisor and computes a goal position for the planner (in the case of HAMP, which computes

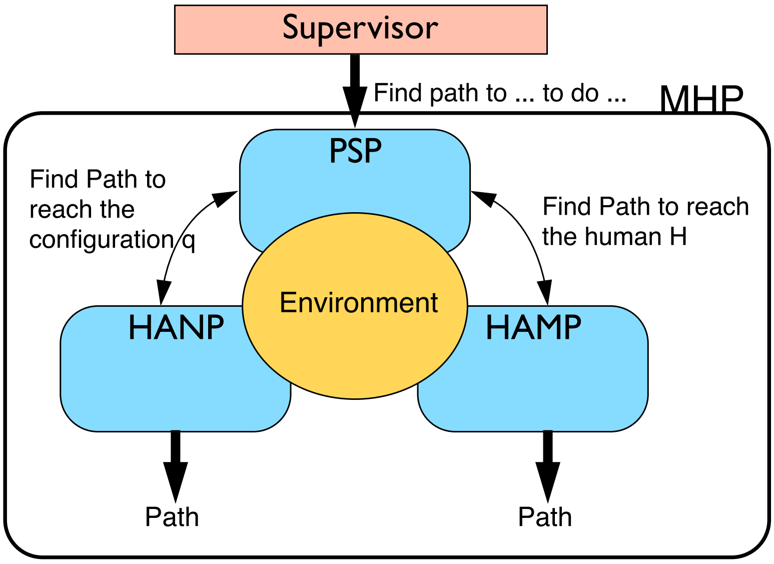


Figure 5.2: *Architecture of MHP module.*

the goal configuration automatically, PSP only makes the choice of the human to whom the robot need to hand the object).

After receiving the interpreted orders from PSP, both planners rely on their own algorithms and internal structures to generate robot paths.

5.3.1 Integration of HANP

The grids consist the core of the Human-Aware Navigation Planner. They interact with the collision checker and the environment on Move3D. Inside the modules the positions of humans are permanently updated in order to have an up-to-date representation of the real environment. The human models contain the 3D structure of Move3D's human model. Even though the navigation module does not need a 3D representation of the humans, in order to share the same environment with the manipulation module, human positions and state changes are maintained in 3D.

Figure 5.3 illustrates the internal architecture of the implementation of HANP. A fixed number of humans are present in the modules environment and they are created, updated or destroyed permanently. In the current implementation of the module, the number of the human is limited to 5. This limitation comes from practical reasons and Move3D and does not represent any algorithmic limitations in the planner.

The environment representation is partially generated by a script that transforms the 2D laser map shared by all the modules in the architecture to 3D environment of Move3D. As the laser data miss the information on heights of the objects, an arbitrary height needs to be given according to the real environment. In figure 5.4, 0.5 m height is given to laser data. Even though this height assignment is a sufficient approximation of the environment for HANP, the manipulation planner needs to have a more accurate model in order to compute complex motions. That is why a number of detailed 3D models of "important" objects are added to the 3D environment by hand. In figure 5.4.b, these objects are the ones that the robot's tasks may require a close interaction: tables, libraries and chairs.

As the planner requires many inputs related to the environment and produces out-

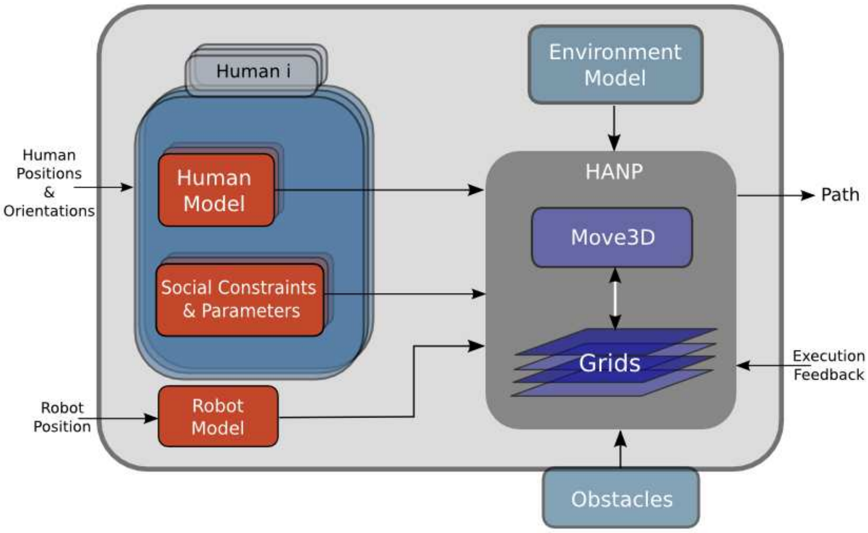
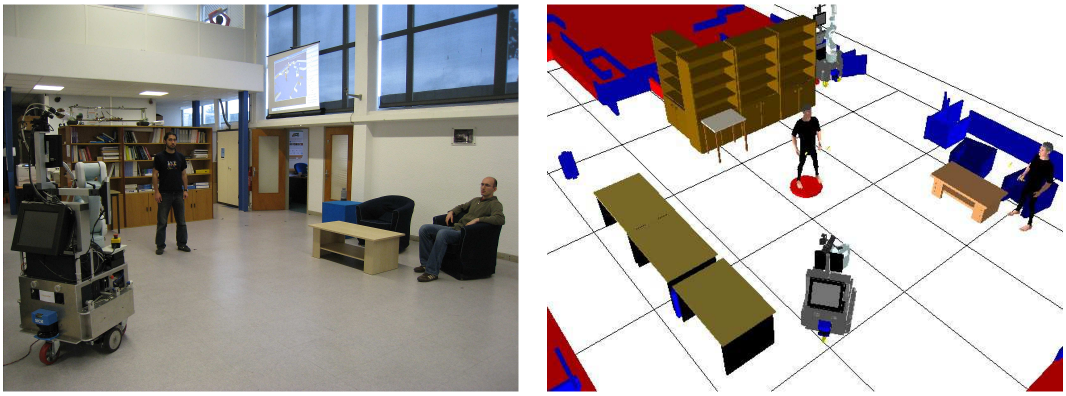


Figure 5.3: *Architecture of the Human-Aware Navigation Planner in MHP module*



(a) The real "Grande Salle" Environment.

(b) Representation of the "Grande Salle" environment along with the robot and humans.

Figure 5.4: *The representation of the environment in MHP module shared by both planners and PerSpective Placement.*

put to be executed, it communicates with a number of perceptive and executive modules. The main input of MHP is HumPos module. This module detects and tracks humans in the environment with the laser scanner. The laser data provides an accurate localization of the humans and also it provides a larger aperture thanks to its 180° field of view. The robot's position, which is permanently updated in the planner, is acquired from the POsition Manager module (POM). This module provides up-to-date values of the whole configuration of the robot.

Having all these inputs, according to the supervisor's request, HANP generates a path and sends it to the execution module SFL in the form of passage points. SFL smoothes and follows this path. When the execution starts, the planner constantly checks the positions and state of humans as well as the robot placement on the path. When a change occurs, the planner immediately plans a new path and sends it to SFL.

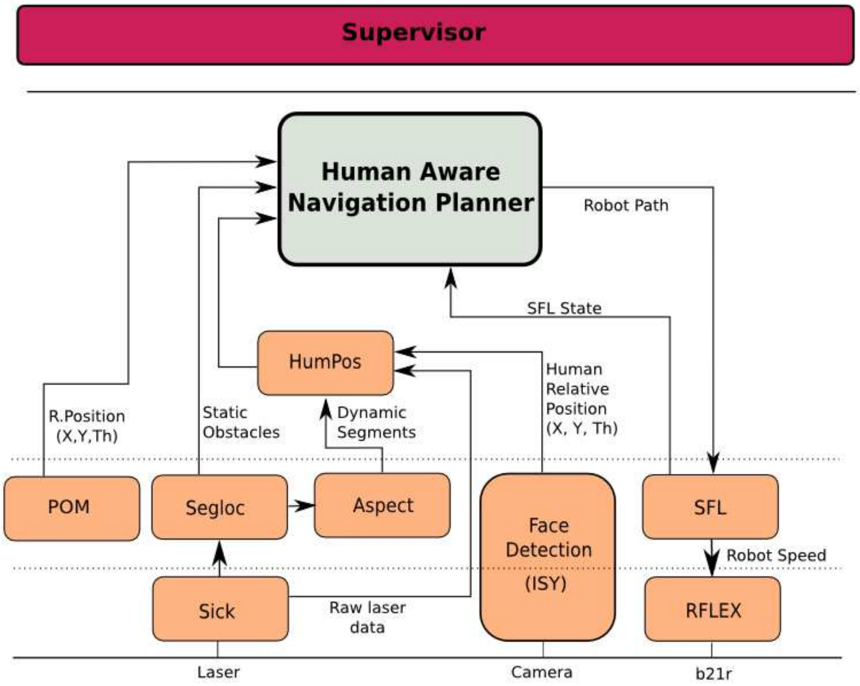


Figure 5.5: *Integration of HANP in LAAS architecture.*

To avoid any oscillations, replanned path does not begin from robot's current position but from the next passage points (The passage point where the planner needs to replan depends on the speed of the robot. The faster the robot advances, the farther will be the starting point of the new path).

Figure 5.5 illustrates connected modules to HANP.

5.3.2 Integration of HAMP

The Human-Aware Manipulation Planner is integrated only to Jido because of the absence of manipulation capabilities on Rackham. Like the navigation planner, the MHP module encapsulates the manipulation planner as well.

Despite the similarity of the internal architectures of navigation and manipulation planners, there are few major differences. On of these differences is the treatment of the humans. Unlike navigation planner, the manipulation planner takes into account only one person. That is why in its environment representation, only the nearest human is taken into account (The nearest person is shown with a red circle in figure 5.4.b).

Grids and Generalized Inverse Kinematics solver establish the core of the planner (Figure 5.6). Human configuration is permanently updated with the 3D head/hand position data coming from GEST and the leg position data coming from HumPos modules. Having two separate data for the same human requires the fusion of the data in order to keep the human model up-to-date as correct as possible. As the laser data is more accurate than the stereo cameras, a prioritized data fusion is executed inside the planner (Algorithm 6).

With the algorithm 6, HAMP maintains permanently the 3D structures of the humans in the environment. The `STANDING/SITTING` states are also applied in this stage with a simple assumption of human height: if a person is shorter than 1.40 m,

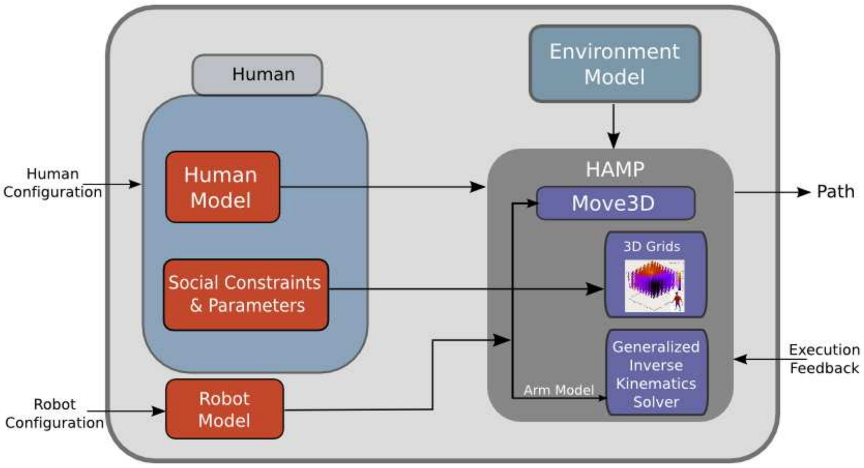


Figure 5.6: *Architecture of the Human Aware Manipulation Planner in MHP module*

Algorithm 6 Fusion of Hand/Head position with leg position.

```
$Legs_{x,y,\theta} \leftarrow$ Laser based leg detection
for $i = 0, i < n$ do
    if $H_i.id == Legs_{x,y,\theta}.id$ then
        if Distance($H_i, Legs_{x,y,\theta}$)> Update Dist then
            $H_i \leftarrow Legs_{x,y,\theta}$
            ChangeState($H_i$, STANDING)
        end if
    end if
end for
if i==n then
    addHuman($Legs_{x,y,\theta}$)
    ChangeState($H_{Legs_{x,y,\theta}}$, STANDING)
end if
Human 2 Interact ← Closest Human
$Head/Hand_{detect} \leftarrow$ Camera based Head and Hand detection
for $i = 0, i < n$ do
    if Distance($Head, H_i$) ¡ 0.2 then
        ${H_i}_{Head/Hand} \leftarrow Head/Hand_{detect}$
        if Height($H_i$) ¡ 1.40 then
            ChangeState($H_i$, SITTING)
        end if
    end if
end for
```

he/she is considered as `SITTING`. This is a very simple simplification that we had to make because of the absence of an activity/state recognition system [Losch 07] in perception modules.

With all these information, the planner module computes a path for the robot's arm and sends it to the execution module, Xarm. This module transforms the path

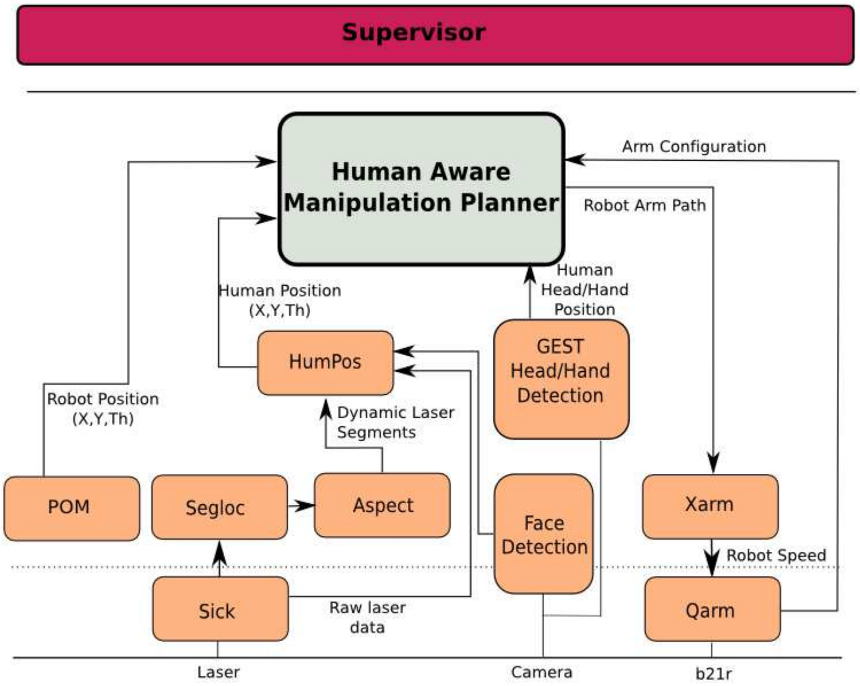


Figure 5.7: *Integration of the HAMP in LAAS architecture.*

to a limited jerk trajectory whose speed profiles are proved to be similar to humans' (Xarm module will be detailed in next section).

5.4 Supporting Modules

Among others, two modules proved to be very important for MHP and need to be detailed further.

In order the Human-Aware Navigation Planner to plan a "human-aware" path, it needs to know the exact positions and orientations of the humans. This information is vital for the planner's human awareness. As the visual detection modules can only detect and localize humans in short range (≈ 3 m) with an important error (± 0.3 m), we need a laser based detection system to detect humans situated far from the robot with a small error. We developed a simple, computationally light module, called HumPos, to fill the needs of the planner.

The other module that we detail in this section is Xarm, the execution module of robot arm's paths. Although the Human-Aware Manipulation Planner's paths can be executed with a standard point-to-point execution module, Xarm's limited jerk speed profiles adds greatly to the friendliness of the robot's motion and improves the resulting interaction.

5.4.1 HumPos - Human Detection & Tracking Module

Detecting humans is necessary for a robotic/computer [CHIL 04] system that involves interaction with humans. There are different methods depending on the robot's sensor capabilities. With camera and laser, the information can be used to detect more precisely humans in the robot's proximity [Kleinehagenbrock 02]. In the absence of cameras, the laser can be used to detect leg-like shapes [Xavier 05]. After the detection, tracking [Shulz 01][Baba 06] must be launched in order to follow the human motions and detect motion patterns.

For this purpose, we have developed[5] the "HumPos" module, a module that provides human detection and tracking services based mainly on laser (and additionally camera) data. HumPos provides a list of humans in the environment to the motion planners. This list contains positions and orientations of detected humans associated with a confidence index and an identifier.

The algorithm and methods used for laser based human detection and tracking are very simplistic and work under two assumptions:

- The gaze direction of a person is always the same as the direction of his body.
- A moving person is always moving forward looking at his motion direction.

The general algorithm consists of making two types of human detection (laser and visual), matching and tracking. At the end, an orientation assignment stage is performed on the results of the tracking. Figure 5.8 illustrates the overall mechanism of human detection and tracking.

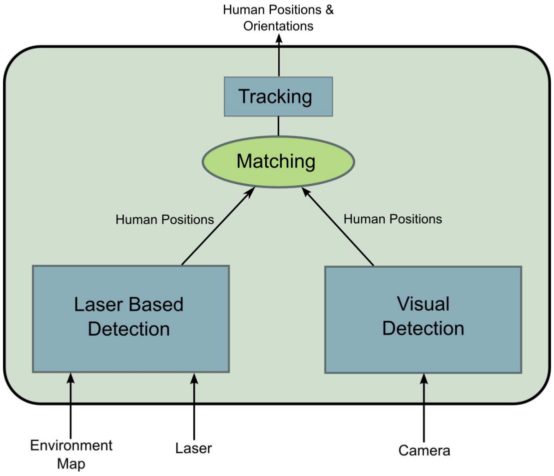


Figure 5.8: *Internal architecture of the HumPos module. The human detection process combines laser and visual data to detect and track humans.*

In laser based detection stage, static obstacles in the environment map are filtered from the sensor data. Resulting points are then used to detect leg-like shapes (a leg or pair of legs) according to their geometry and neighborhood. This process produces a list of detected humans with their positions and an attached confidence index.

On the other hand, the visual data coming from the camera are used to detect people in near proximity of the robot by the visual face detection module (figure 5.9.b). The visual face detection module provides a list of humans looking directly at the robot with their estimated distance based on facial size metrics within a range of ≈ 3 meters.

These two lists are then matched to produce only one list of humans with corresponding positions, orientations, and confidence index (figure 5.9.c). Finally, detected humans are tracked by the tracking stage. At the end of this stage, orientations are assigned to detected humans according to their motions, the visual detection result

---

[5] HumPos module is developed with Luis Marin-Urias.

and the two assumptions that we made above. The orientation assignment procedure is described in algorithm 7.

Algorithm 7 Orientation assignment in HumPos module

```
if a person $P$ detected by laser then
    if VisualFaceDetection detects $P$ then
        $Direction_P$ ← looking at the robot (body towards the robot)
    else
        if $P$ is moving then
            $Direction_P$ ← motion direction
        else
            if $P$ detected before then
                $Direction_P \leftarrow OldDirection_P$
            else
                $Direction_P$ ← looking at the robot (body towards the robot)
            end if
        end if
    end if
end if
```

If a person is looking at the robot and thus detected visually, we assign his orientation to the direction of the robot. If the visual face detection fails, then laser based leg detection decides humans' orientations. If a person is detected and he/she is moving, his motion direction is assigned as his/her head/body orientation. If a person stops, his/her last orientation is conserved and assigned to next detections until he/she moves, disappears, or is detected by visual face detection.

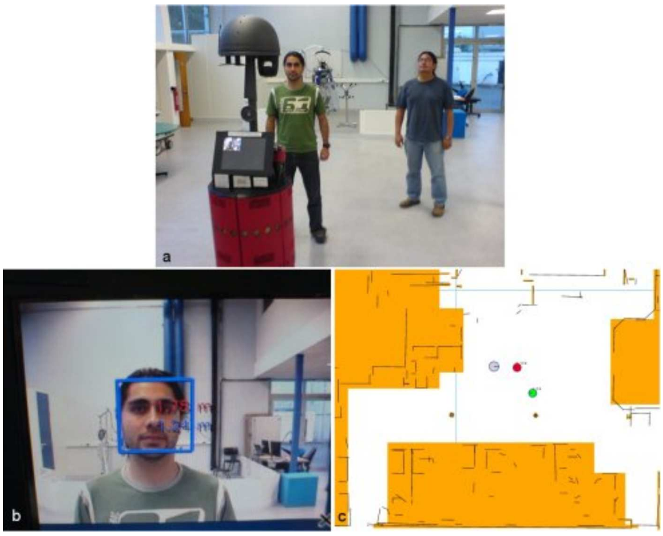


Figure 5.9: *a- Two persons have been detected based on laser data b- One of them is also detected using vision-based face detection c- The one detected by the camera has a high confidence index and is marked with red while the other person is marked with a lower confidence.*

5.4.2 Xarm - Soft Trajectory Planner

The Xarm module, developed by Herrera-Aguilar [Herrera-Aguilar 07] and Broquère [Broquère 08], is a soft motion trajectory planner limiting jerk, acceleration and velocity in cartesian space for the RA6 arm. In our architecture, Xarm is in charge of transforming the path produced by MHP into a trajectory.

The underlying idea of Xarm is the division of a point-to-point motion into seven elementary motions. These motions, illustrated in figure 5.10, can be combined to represent any type of motions having null start and goal jerk/acceleration/velocity conditions (basically a motion with null initial and final kinematics conditions).

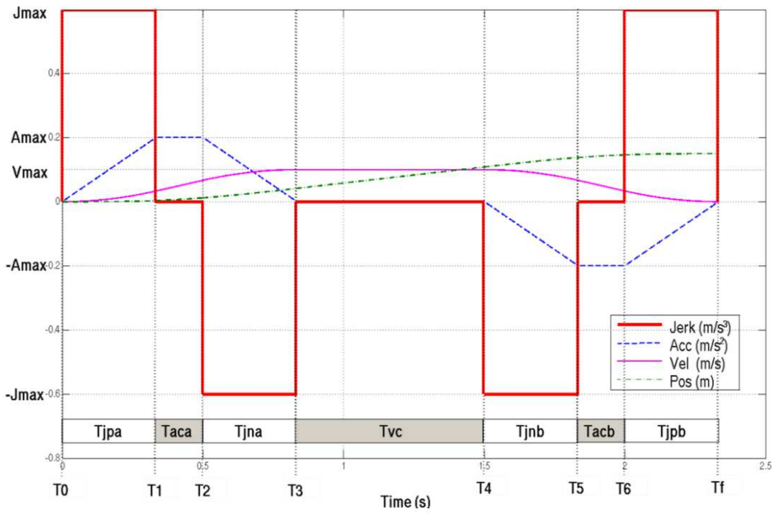


Figure 5.10: *The speed profile generated by Xarm having null start/goal kinematics conditions. A point-to-point motion is proved to be decomposable to 7 elementary segments which are illustrated in this figure.*

As the manipulation planner provides a set of passage points, the point-to-point motion is not preferable unless we want the robot to stop on every single passage point. To generate motions with non-null start/end conditions (figure 5.11), Xarm executes 3 steps:

Step 1: We compute the adjusted point to point motion between the current position (P0) and the intermediate point (P1). We compute also the adjusted point to point motion between the point (P1) and the final point (Pf). In this state, the motion is stopped at (P1).

Step 2: For the transition motion, we use as initial conditions the ones found at the end point of the *Tvc* segment of the first point-to-point motion (IC_T) (figure 5.11) and as final conditions the states at the beginning of the *Tvc* segment of the second point-to-point motion (FC_T). So we have for each axis :

$$\begin{aligned} A(IC_T) &= 0 \qquad & A(FC_T) &= 0 \\ V(IC_T) &= V_0 \qquad & V(FC_T) &= V_f \\ X(IC_T) &= X_0 \qquad & X(FC_T) &= X_f \end{aligned}$$

Step 3: Once the execution time of all 7 segments are computed, we obtain the optimal times T_{opt} for each axis. Then, we have to constrain the motion time duration of each axis considering the axis which has the largest duration.

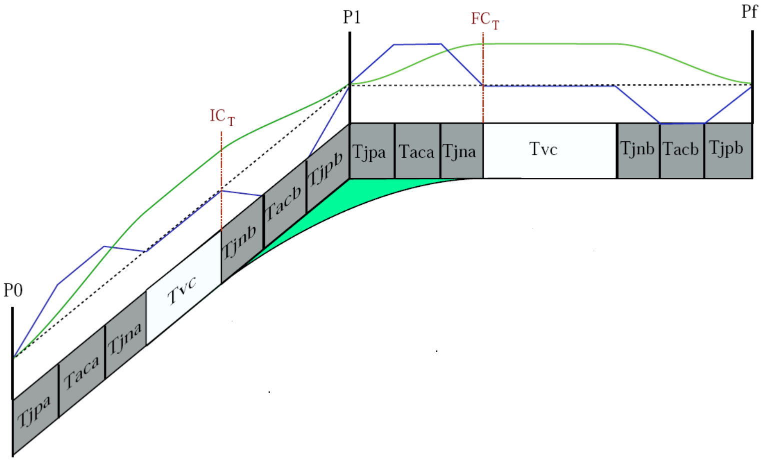


Figure 5.11: *The speed profiles and elementary motions for a motion that crosses an intermediate point with non-null kinematics conditions.*

This supporting module is an important part towards a friendly robot arm motion. As the HAMP does not generate trajectories and leaves the speed adjustments of the paths to the execution module, the comfort of the motion is not fully guaranteed. Even though the execution follows the path, fast accelerations can still cause fear and surprise. Xarm module removes this risk by generating natural speed profiles that proved to be friendly and comfortable by user studies [Huber 08].

The methods and algorithms used in this module is being transported to Move3D to allow HAMP generate directly comfortable trajectories and will improve the completeness of the comfort of the human partner.

5.5 Results

This section contains results of motions generated by Human-Aware Navigation and Manipulation Planners with real robots. The paths planned by the navigation planner are illustrated with the robot Rackham, and the path of the manipulation planner with Jido.

5.5.1 Human-Aware Navigation Planner

Figure 5.12 illustrates a scenario with two people having a conversation with a comparison between a standard motion planner and Human-Aware Navigation Planner. In this scenario, the robot aims to move from one corner of the room to the other. The direct path between these two points is blocked by two people. One of the humans has his back turned to the robot and thus can neither see it nor is aware of its presence.

Using a "classical" motion planner, the robot tries to follow the shortest path to get to its goal. The humans are considered as simple obstacles to avoid. When the robot approaches sufficiently close, it modifies its path just enough to avoid them (figure 5.12.a). When the robot passes next to them, it causes surprise and fear to the person who has not see the robot coming. Then it reaches its goal with a direct path.

Then, we replace the planner by HAMP. As can be seen in figure 5.12.b, the robot does not approach directly to the humans because it cannot be seen by one of them.

Figure 5.12: *A comparison between a "classic" motion planner and the Human-Aware Navigation Planner which produces a more acceptable path by taking into account the safety and visibility of each human in the environment.*

Thus it takes a greater distance to avoid any surprise and fear and it enters more smoothly into their field of view.

The initial path and replanned paths can be seen in figure 5.13. In this figure, detected humans are represented with green circles and robot with a gray circle. As the robot does not detect any one, it calculates a direct path. During its motion, it detects humans on its path and replans to adapt to their presence.

Figure 5.14 illustrates the effect of the Visibility Criterion (§ 3.4). By taking into account the visibility cost function, a path costs less when the robot passes in front of the human than a path situated behind. That is why Rackham "prefers" to pass in front of the human as shown in figure 5.14.a. However in situations where there is not enough free space in front of the human (or blocked by an obstacle or another human), the robot passes behind but puts a greater distance between itself and the human (figure 5.14.b).

Another scenario is illustrated in figure 5.15 where a person and robot move toward each other. The robot follows a straight path before detecting the human (figure 5.15.a). Once the human is detected, the replanning produces a new path to avoid him (figure 5.15.b). After passing the human, the robot doesn't immediately take its previous lane but it gives a little distance to the human's back (figure 5.15.c). This behavior avoids possible unpleasantness in case of a change in the human's motion.

The final example shows the effect of Hidden zones (§ 3.5). In this scenario (figure 5.16), a whiteboard separates the human and the robot. Because of this obstacle, the robot is completely hidden from the human. On the robot's side, the human is partially hidden because his legs are visible to robot's laser. The robot detects that there is a person behind the whiteboard. Since the human approached the whiteboard before, his correct orientation coming from the leg tracking is kept by the robot. So the robot plans a path that takes into account not only safety and visibility but also the surprise and fear effect coming from looming into the human's field of view. By following this path, the robot gets farther from the whiteboard. Then once it is visible, it moves toward its goal.

The Human-Aware Navigation Planner is also evaluated by neutral subjects in the

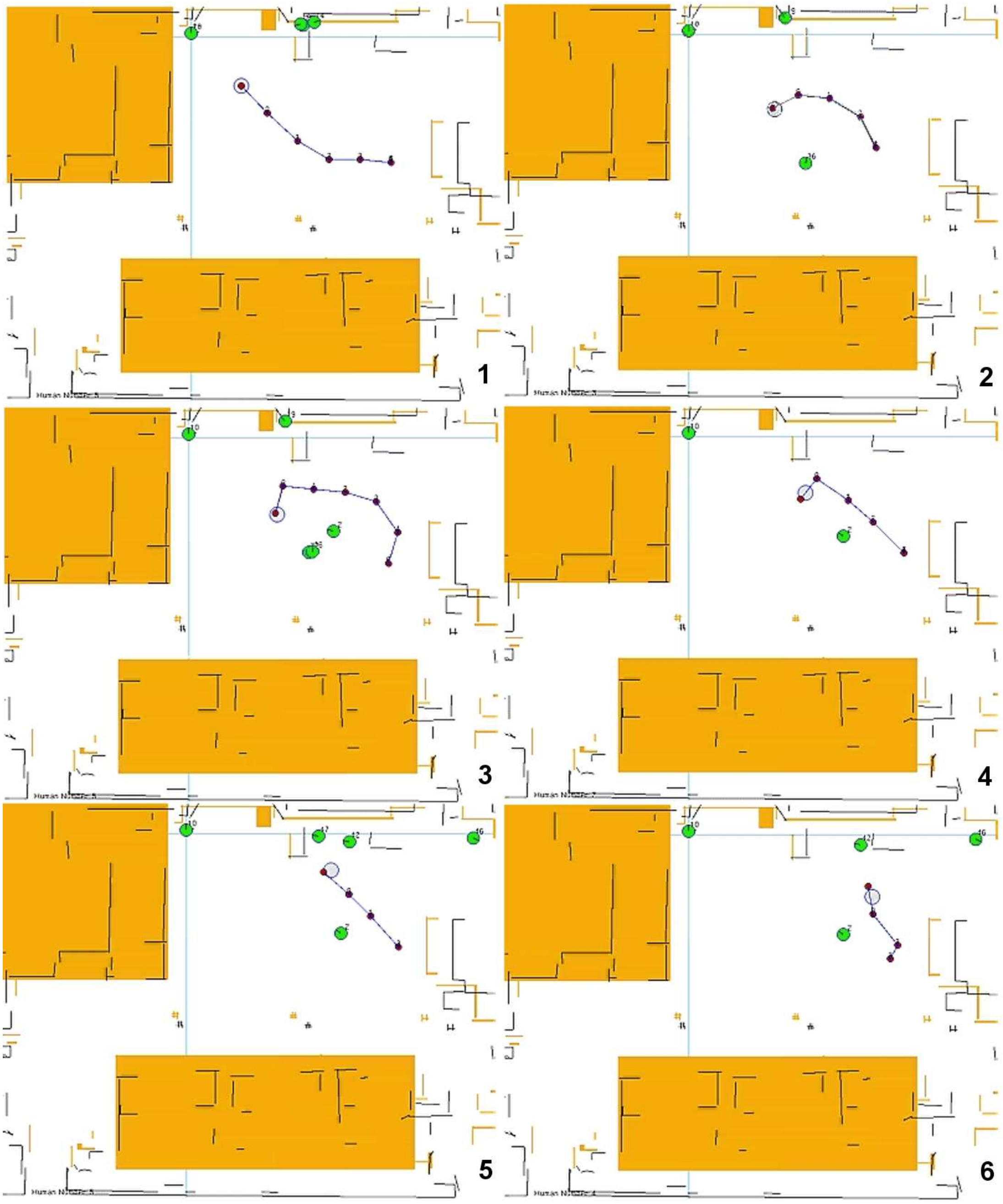


Figure 5.13: *Replanned path during the execution of a trajectory. In case of a change in the environment or humans' positions, the planner replans a path and sends it to the execution module.*

scope of the FP6 European COGNIRON Project. A home scenario is run by the robot and its behavior is videotaped for further evaluation. A team from University of Hertfordshire used these videos to conduct user studies by showing them to subjects and asking to rate the behavior of the robot. Most of the subjects evaluated [Cogniron 08] the motions of the robot as natural and comfortable without causing any unpleasantness.

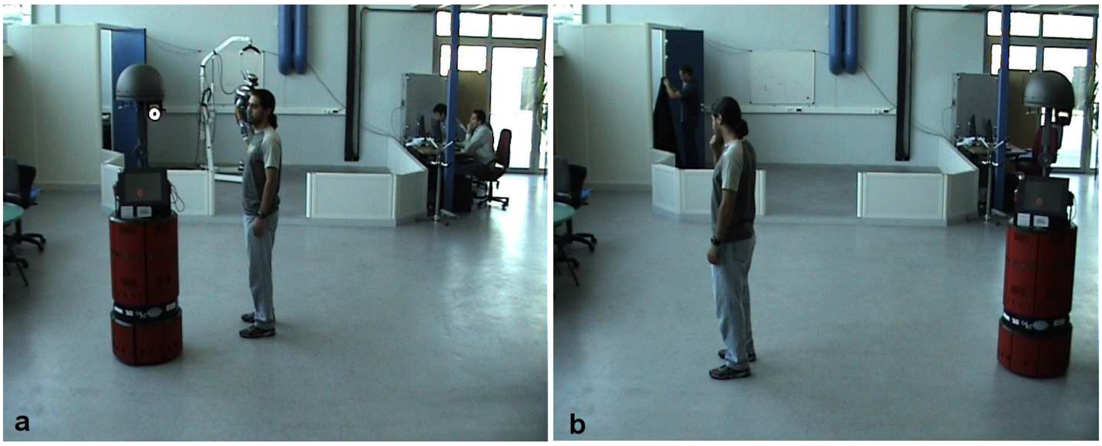


Figure 5.14: *The effect of the Visibility Criterion (§ 3.4) for a path passing in front or behind of a person.*

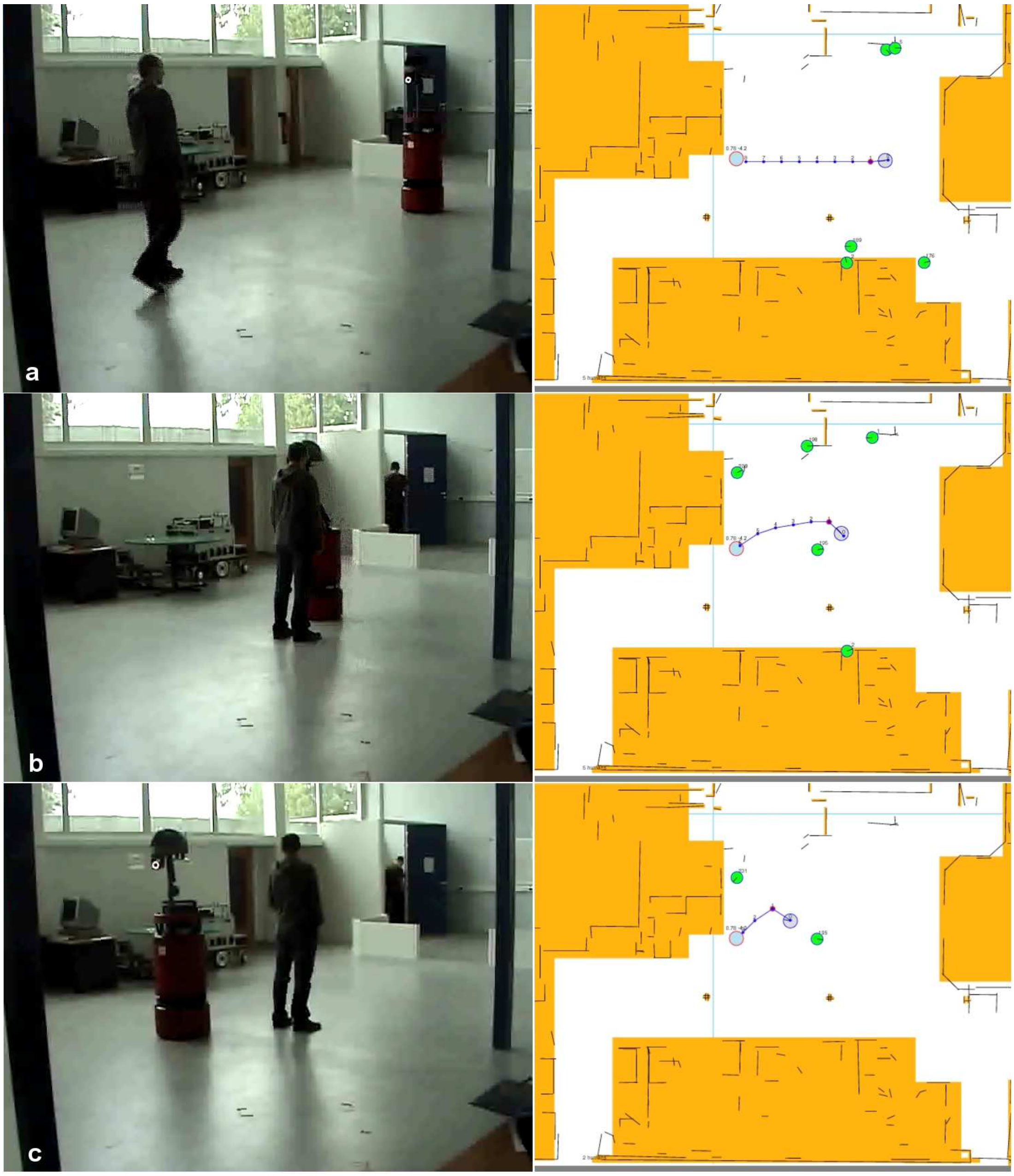


Figure 5.15: *A person and robot move towards each other. The robot changes it's path to avoid a collision. After passing next to the human, instead of taking immediately its previous lane, it puts a little distance to the human. This behavior ensures a comfortable and riskless motion for the human.*

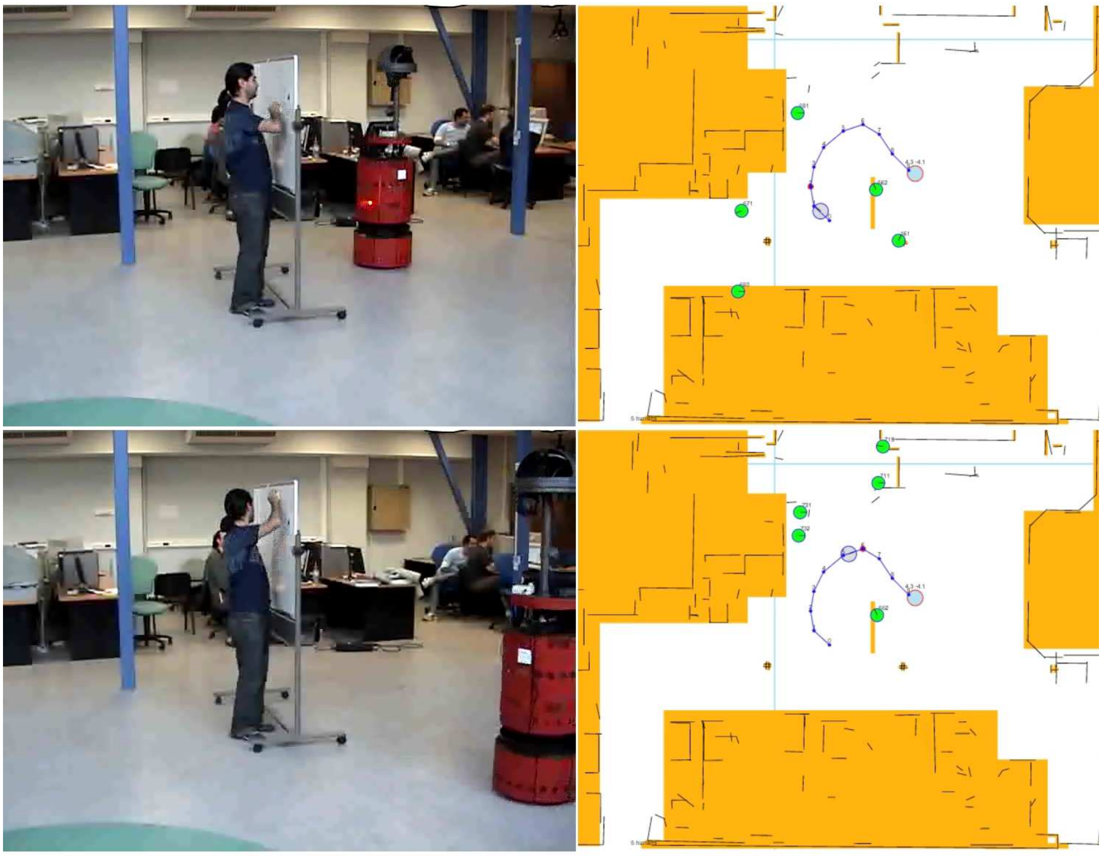


Figure 5.16: *Robot is invisible to the human. To avoid any unpleasantness coming from the sudden appearance of the robot, the planner generates a path which makes the robot appear farther.*

5.5.2 Human-Aware Manipulation Planner

Unlike the navigation planner which is integrated to both robots, the Human-Aware Manipulation Planner is only integrated to Jido because of its manipulation capabilities. Although Jido offers sufficient accessibility with its arm, it lacks the presence of a head. That is why in the implementation of the planner, when computing the robot path to follow the object path, only one task is taken into account in generalized inverse kinematics solver.

The motions of the robots are tested in various scenarios. The biggest setback in the evaluation of the robot's motions is the insufficiency of the perception modules. In order to be detected, localized and tracked correctly, the human partner needs to make the effort to maintain a good placement. Although this setback is easily overcome by an experienced user, an inexperienced user needs to make an effort.

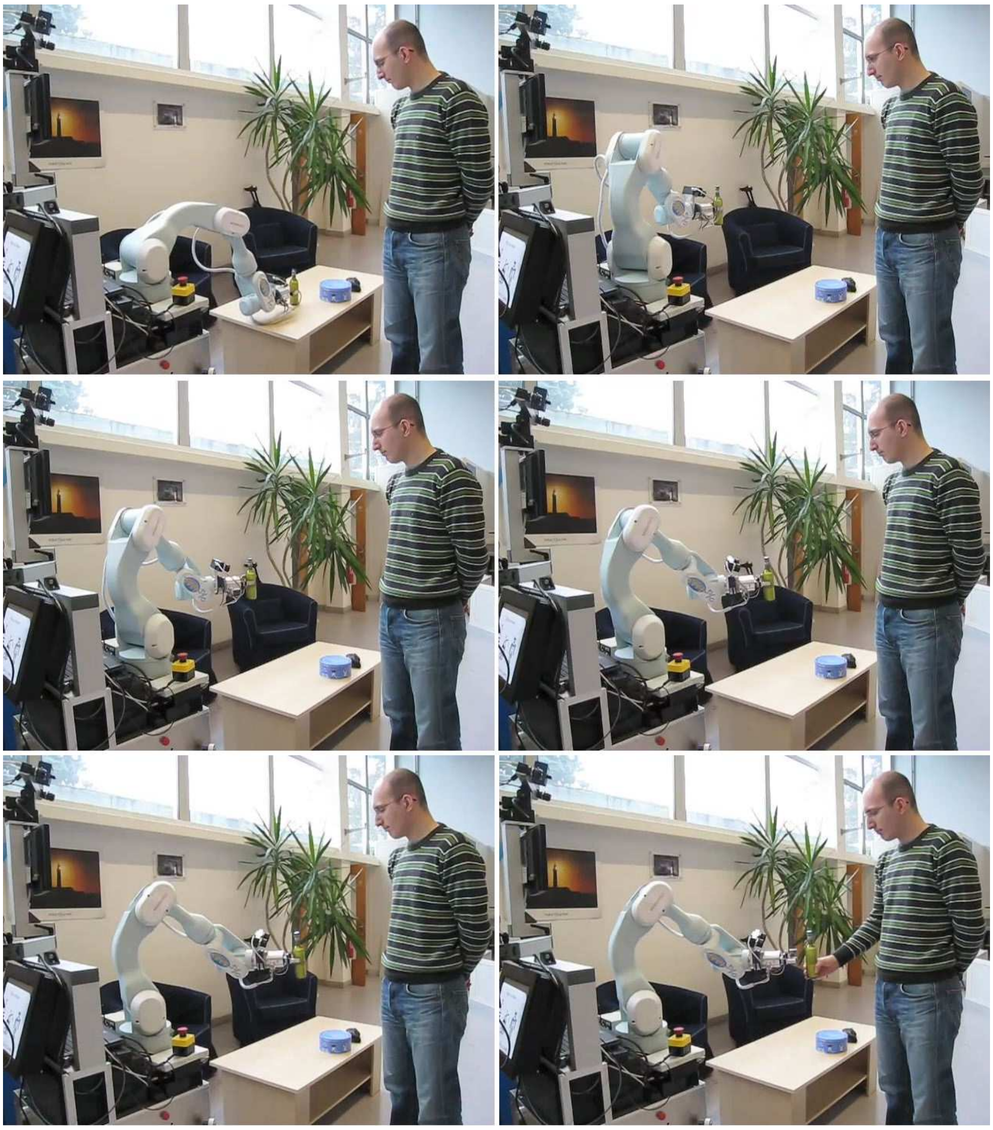


Figure 5.17: *Jido hands over a bottle to a standing person. The position where the object transfer will occur is chosen by the planner and motion towards this point is performed.*

Figure 5.17 illustrates a scenario where the robot hands a bottle to a standing person. With the leg and head/hand detection modules, the planner updates its human model and plans a path towards a position of its choosing (The Object Transfer Point) by considering that the person is standing and facing it.

The planner generates and sends the path in the form of successive configurations to the execution modules. Once the robot reaches its final position, it waits for the human to pull the bottle. Once the bottle is handed, the arm reaches to its original configuration. The sequence of moving arm, detecting object pull, opening the gripper and returning to original position, is performed either by tcl scripts passed to the robot by a human controller or the supervisor situated in the decisional level.

Figure 5.18: *When the actions of a "hand over an object" task are run sequentially, the robot only releases the object when the arm reaches its position. For a person who wants to take the object earlier, this behavior causes discomfort in the interaction.*

Figure 5.18 shows the importance of the sequencing of the actions in a handing over task. In this scenario, Jido reaches its arm to hand over a bottle. The person, being impatient, tries to take the bottle before to robot arm reaches its destination. As the robot only releases the bottle once it reaches its destination, this behavior causes discomfort.

In order to solve this problem the force detection and the arm motion actions are run in parallel and an interrupt is sent to the execution if a pulling force is applied to the object. This parallelization allows the robot hand over the object during its motion whenever the human desires. Scenarios illustrated in figure 5.18 and 5.19 are performed with this parallelization.

A final example of motions is given in figure 5.19 illustrating a scenario where two lazy persons sit around the robot. The person on the right asks the robot to take the bottle and pass it to the other person.

Another functionality that is added to the planner module is to ability to by pass the Object Transfer Point computation process and plan a path directly to human's

Figure 5.19: *A scenario where two lazy persons sit around the robot and ask the robot to pass a bottle from one to the other.*

hand. Also a monitor detecting human reaching motion is implemented in order to detect if the person performs a reaching gesture to ask for the object.

In this scenario, the planner generates a path towards the hand of the human. Once it grasps the bottle, it turns its cameras to the other side and waits for the reach gesture from the other person. Once the human asks for the object, the robot plans a path by taking into account safety and the visibility.

5.6 Discussion

In this chapter, we presented the integration of Human-Aware Navigation and Manipulation Planners into two real robots. These planners are encapsulated in a Genom module, called MHP, in LAAS architecture. This module communicates with the perception modules, notably human detection and localization, and execution modules.

Even though the module merges the data coming from visual and laser based perception module to increase the reliability, it still requires high quality results from perception modules. On of the main challenges stays at the perception stage to improve the quality of the interpreted data.

Both of the planners are evaluated internally[6] and externally in the frame of the FP6 Cogniron Project [Cogniron 08] with “naive” subjects. The behavior of the robot is evaluated as satisfactory and not disturbing for the most of the subjects via a video based user study. We plan to carry further studies in order to evaluate and improve the planners.

---

[6] Details of the internal evaluation can be found in annex B.

CHAPTER
SIX

Conclusion

Generating motions for a robot in close interaction with humans requires explicit reasoning capabilities on humans. New methods and algorithms for motion planning, that will take the human as a separate entity unlike classical motion planning methods that consider him as a moving obstacle, are necessary. The robot should not only reason on itself, its task and its environment, but also put into consideration that it is moving among humans and is expected to obey social rules and protocols that humans are subject to.

A detailed study to understand the notions of "safety" and "comfort" is necessary to equip the robot with algorithms that will allow a safe and comfortable co-existence for the robot among humans. This requires multi-disciplinary effort for both robotics and psychology societies. Further user studies need to be conducted and a metric to evaluate human reactions to robot motions need to be found.

This work presents a first approach of integrating human comfort into the planning loop by tackling the problem in a larger point of view than ad-hoc methods that are designed for specific scenarios and that can be found in the literature.

A short but focused survey on motion generation methods that "explicitly" takes into account humans is presented in this work. Few number of works exist in literature mostly being ad-hoc methods and necessitating too much effort from the human, thus staying still too "human-centric". We believe this work fills the absence of a general framework that approaches the motion planning in human presence problem from a larger point of view.

We presented the Human-Aware Navigation Planner, a motion planner that explicitly takes into account humans in robots environment. Three important criteria are extracted from user studies and are represented with cost functions to provide a spatial evaluation metric for robot's position. These criteria, called safety, visibility and hidden zones criteria, form good bases of navigation in any HRI scenarios.

Having adopted a cost based approach allows not only computational weight reduction but also offers high level of flexibility and extensibility. Two extensions, activity presentations and human highways, are proposed in this work to illustrate the extensibility of our approach.

Methods and algorithms towards human-aware manipulation motions are also presented in this work. We proposed a general framework of manipulation motion generation with 3D grids. The general motion planning problem to find a path from one configuration to an other, is transformed to find a path to accomplish a task. We materialized our approach with a Human-Aware Manipulation Planner designed for " handing over an object" scenarios.

The division of the whole problem into 3 stages allowed to reduce the complexity of the problem. Incorporating multiple tasks into the Generalized Inverse Kinematics solver during the planning resulted "legible" robot motions where the clarity of the robot's intention is ensured only by its motions.

An extension to connect navigation and manipulation planners, called PSP, is proposed along with a probabilistic approach to coordinate robot base and arm motions.

The two planners are integrated into two robots having a large number of modules. The planners are permanently in communication with perception and execution modules as well as the supervisor system. The robot motions produced by the planners are evaluated by an individual video based user study and found natural and comfortable by most of the subjects.

Perspectives

This work, as far as we know, is the first to tackle the motion generation problem of HRI in very large point of view by proposing a general framework. Being the first in its field, a large number of questions and perspectives resulted from this work.

The most important perspectives are listed as follows:

- One of the most vital notion to ensure safety and comfort in HRI is the notion of "speed". In this work, we have only interested in and proposed solutions for the shape of robot motions. In real world, we cannot fully guarantee the safety if the speed of the robot remains uncontrolled. Even though the classic approach of; "path planning → path to trajectory transformation → trajectory execution" can be used, this approach causes the loss of many solutions. In order to fully ensure safety and comfort, we have to by pass the intermediate level of path to trajectory transformation and incorporate the speed into the planning loop.

- The methods and algorithms described in this work consider the humans as static, not moving, entities. Yet in real world, humans are generally moving in a way that can be foreseen. Although in our system the lack of this notion is compensated by fast replanning in case of a human position change, the system is not designed to include human motions. Modeling human motions and planning by taking into account these models can reduce the replanning weight and end up with smoother paths.

- The point above raises an other notion that needs to be taken into account. In the simulation environment of the planner there is a strong assumption of the robot seeing every human in the environment. Yet for a real robot, this is not the case for most of the time. When a detected person disappears behind an obstacle, the robot should store the information of a possible existence of a person behind that obstacle. The underlying notion of hidden zones grid (§ 3.5), where the person behind an obstacle is taken into account; and the notion of pro-active paths [Madhava Krishna 06], where the obstructing environments are taken into account, can be merged with a dissipating probability given to the existence of humans behind obstacles.

- The extensions, proposed in chapters 3 and 4, present also perspectives that we want to explore.
- In manipulation scenarios, the shape of the environment can play a big role. The features of the environment can hide object from the robot or from the human. This role becomes more important in the probabilistic approach extension that we have proposed for the manipulation planner, where the distance between robot and the human is greater and can contain a large number of objects. So the robot put itself in human's place to model the things that human sees. This can improve the interaction and ease the communication between the robot and the human. PerSperctive Placement, mentioned as an extension, is a first step towards this direction.
- Another issue that we want to point out is the social rules imposed by the environment. In same places, like in a theater, the environment implicitly contains some rules for the motions of the people coming from the type of activity conducted in this environment (for ex. not passing in front of the stage in a theater). This can either modeled as a property of the environment or of the task. Taking into account this notion can produce a more socially acceptable motion.
- A final issue that we want to explore is planning for the human. For a robot helping a person, there will be moments that the robot will not find a solution satisfying both the safety and the feasibility of the task without moving the human. So the robot should also reason on how costly the human motions will be in order to plan for itself and for the human to find an optimal plan that satisfies the task.

Robot Human Interaction

CHAPTER
SEVEN

Vers la planification de mouvement avec prise en compte explicite de l'homme

(Résumé)

Introduction

Les machines articulées qui interagissent avec des personnes ont toujours suscité beaucoup d'attention et curiosité dans toutes les périodes de l'histoire. Bien que le mot "robot" ne soit apparu qu'au début du 20^{e} siècle, des exemples des machines articulées qui se comportent comme des animaux ou des humains remontent aux années 1000 av. J.-C. . Grâce aux développements en science et technologie, les automates qui interagissent avec les gens sont apparus en 18^{e} siècle.

Avec les développements technologique du siècle dernier, l'idée d'avoir des robots qui "coexistent" parmi nous commence à être une réalité. La notion d'autonomie est apparue permettant aux robots de percevoir, raisonner et agir avec leurs propres moyens. Avoir des robots dans notre vie quotidienne a fait surgir de nouvelles questions et de nouveaux défis pour la robotique et fait apparaître un nouveau domaine de recherche appelé HRI, Interaction Homme-Robot.

HRI est un domaine de recherche aujourd'hui très actif et en pleine expansion selon plusieurs directions pour atteindre au même but : un robot qui va percevoir son environnement, raisonner sur la situation et agir d'une façon sûre et confortable pour faciliter la vie des personnes.

Cette thèse aborde les capacités de raisonnement du robot dans ce contexte notamment pour répondre à des questions telles que : "comment les mouvements de robots doivent être influencés par le fait qu'il agit en présence et/ou en collaboration avec les humains ?", "comment le robot et l'homme doivent-ils partager l'espace ?". Ceci nous a conduit à proposer des méthodes et des algorithmes de planification qui engendrent des mouvements de robot en raisonnant non seulement sur l'environnement mais aussi en "explicitement" sur la présence et l'activité des personnes alentour.

La Problématique

L'introduction des robots dans la vie quotidienne apporte un problème important qui "s'ajoute" au "défi standard" des robots autonomes : la présence d'hommes dans son environnement et le besoin d'interagir avec eux. Dans la robotique industrielle, même s'il existe des opérateurs dans l'environnement, une distance de sécurité est toujours gardée entre les humains et les robots. Si cette approche apporte une garantie de bon fonctionnement et la sécurité du système, elle rend l'interaction entre les hommes et les robots pratiquement impossible.

Pour permettre une "coexistence" entre les robots et les hommes, nous devons considérer tous les aspects de l'interaction homme-robot et les comportements résultants qui doivent être pris en compte dans toutes les étapes de la conception du robot. Un robot qui va coexister avec les gens voire les aider ne doit pas être seulement une machine mais doit aussi respecter des règles "sociales [Chatila 02][Fong 03]".

Ce travail s'intéresse aux problèmes de l'interaction proche entre humains et robots, en se plaçant du point de vue des décisions de mouvement qui doivent être prises par le robot pour assurer un mouvement :

- sûr, où il ne soit pas possible de blesser l'homme,
- effectif et digne de confiance où la tâche commandée soit réalisée correctement en fonction des capacités du robot,
- "agréable" à l'homme, où les préférences, les besoins et les mouvements de l'homme sont pris en compte,
- compréhensible à l'homme, où l'intention du robot est exprimée clairement.

Le robot doit accomplir des actions de navigation et de manipulation et doit être capable de déterminer à quel endroit une tâche doit se produire, comment se placer par rapport à l'homme, comment l'approcher, comment lui tendre un objet et comment se déplacer dans un environnent relativement contraint en présence de l'homme.

Les Contributions

Cette thèse est la première dans le domaine qui répond aux problèmes mentionnés ci-dessus en proposant un cadre général pour la planification de tâches de navigation et de manipulation en prenant en compte explicitement la présence des personnes dans l'environnement du robot.

La première contribution de ce travail est de fournir un état de l'art détaillé et focalisé sur les méthodes de génération de mouvements en présence de l'homme et sur les études utilisateurs (user studies) portant sur le partage de l'espace homme-homme et homme-robot. Ces deux domaines, restant généralement bien séparés, sont analysés et intégrés en vue de la conception de planificateurs de mouvements.

La sécurité et le confort de l'homme sont étudiés et représentés par des fonctions des coûts. L'implémentation de ces coûts a donné naissance à un planificateur de tâches de navigation appelé "Human-Aware Navigation Planner", et qui génère des mouvements à la fois sûrs et confortables. L'introduction de la notion de "visibilité" ainsi que les préférences et la posture de l'homme à l'étape de planification est une des nouveautés qui transforme la notion de chemin "faisable" en "chemin acceptable". Enfin, avec l'intégration de la notion de perspective développée dans le cadre d'une algorithmique appelée "Perspective Placement", le problème initial de navigation est transformé en un problème beaucoup plus riche de recherche d'un chemin entre deux configurations "pour réaliser une tache", fournissant ainsi la possibilité de raisonner à un niveau d'abstraction supérieur.

La troisième contribution de ce travail est la conception d'un cadre formel pour le raisonnement sur la manipulation en présence de l'homme, Ceci a conduit à un planificateur appelé “Human-Aware Manipulation Planner”. Ce dernier est basé sur une décomposition de la génération mouvement en plusieurs étapes, permettant notamment de réduire la complexité et d'intégrer la notion d'initiative. Ainsi, tendre un objet à l'homme est traité en trois étapes. Le planificateur choisit d'abord l'endroit où l'interaction va se dérouler puis génère une trajectoire. Enfin, grâce à l'utilisation de la cinématique inverse généralisée et et l'exploration de l'espace nul engendré par une structure cinématique redondante, le robot exprime son intention clairement à l'homme en effectuant des mouvements complémentaires notamment de la tête du robot ou d'un deuxième bras non utilisé par la tâche principale.

Finalement les deux planificateurs sont intégrés dans deux plates-formes robotiques et validés par des études utilisateur.

État de l'Art

L'introduction des robots dans les environnements humains induit des préoccupations nouvelles portant sur la sécurité de l'interaction. Ceci a notamment conduit au développement de travaux sur la notion de “sécurité” ([Alami 06]) ainsi qu'à des évaluations menées dans le cadre d'études utilisateurs (user studies [Haddadin 08]).

En complément des travaux visant à concevoir, au niveau matériel, des robots plus sûrs, la sécurité peut aussi être renforcée par les stratégies de contrôle [Ikuta 03]. Cette section résume brièvement les stratégies de contrôle et de planification qui prennent en compte l'homme. Bien que les méthodes proposées dans cette thèse s'inspirent et utilise des méthodes standards de planification de mouvement, on a choisi de ne pas faire une état de l'art sur ces méthodes puisqu'elles sont maintenant connues et très bien décrites dans de plusieurs ouvrages [Latombe 91, Laumond 97, Lavalle 06, Laugier 07].

La Relation Spatiale Entre l'Homme et le Robot

Pour synthétiser des mouvements sûrs (et confortables), on doit d'abord comprendre quels types de comportements sociaux et quels types de mouvements sont acceptés par les hommes dans leur vie quotidienne, privée ou professionnelle. La théorie dite “de la proxémie” qui catégorise le partage de l'espace entre les personnes et les études de cas mettant en oeuvre des robots et des hommes fournissent des mesures, des règles et des préférences qu'il faut serait intéressant de prendre en compte dans les mouvements du robot.

Une des études principales sur le partage de l'espace entre les hommes a été conduite par Hall [Hall 66]. Cette étude a conduit au développement d'une théorie, la “proxémie”, qui catégorise les placements relatifs entre deux personnes en 4 zones, nommées intime, personnelle, sociale et publique.

Des études utilisateurs ont été réalisées pour un robot qui croise une personne dans un couloir [Yoda 95, Pacchierotti 05], un robot qui s'approche d'une personne [Walters 05a, Dautenhahn 06] ou encore un robot qui présente un objet à une personne [Yamaoka 08]. Ces études ont fourni des mesures et des indications sur les comportements du robot acceptés par les gens.

Alors que les scénarios de navigation peuvent être analysés dans un espace 2D, pour un robot manipulateur, le partage de l'environnement se déroule en 3D. Dans ce cas, non seulement le placement du robot mais aussi le placement de l'objet à manipuler ainsi que de toute la structure cinématique du robot jouent des rôles importants. Les mouvements d'un robot manipulateur ont été évalués dans le cas d'une tâche de prise et pose d'un objet [Sakata 04], ainsi que dans le cas d'un robot qui s'approche et donne une bouteille à une personne [Koay 07]. Ces études ont montré l'importance du placement spatial et de la coordination du mouvement du robot dans l'interaction avec une personne.

Ces études sont des points de départ pour le développement de stratégies de contrôle et de planification de mouvement destinées à produire des comportements de robots qui soient considérés comme socialement acceptables.

Modélisation de l'Homme

Pour engendrer des mouvements non seulement sûrs mais aussi socialement acceptables, les hommes dans l'environnement ne doivent pas être considérés comme des objets qui bougent mais doivent être pris en compte au moyen de modèles plus élaborés. Selon les capacités de planification ou de contrôle, les différentes caractéristiques de l'homme sont modélisées :

Position et orientation : Chaque personne dans l'environnement a une position et une orientation qui permet au robot de savoir où elle se trouve [Nakauchi 02, Takemura 07, Yoshimi 06].

Capacité de manipulation : Dans le cas de la manipulation, les mains de l'homme interagissent avec le robot doivent être prises en compte dans la représentation de l'homme que le robot manipule [Kulić 05].

Mouvement : Comme les hommes se déplacent souvent, un modèle qui représente leur mouvement [Pacchierotti 06b, Althaus 04, Martinez-Garcia 05, Hoeller 07, Zender 07] permet au robot de se déplacer en prenant en compte les emplacements futurs des personnes.

Attention et Champs de vue : En raisonnant sur le champ de vue de l'homme, le robot peut déduire où son attention est orientée [Traver 00].

Activité : La tâche qu'une personne est en train de réaliser est également à prendre en compte.

Préférences : Nous incluons ici les préférences ou particularités des hommes qui peuvent être pertinentes, par exemple le fait d'être gaucher ou droitier.

Chacune de ces caractéristiques peut avoir une influence sur les comportements et sur les mouvements du robot et il serait pertinent de les prendre en compte de manière explicite.

La Navigation en Présence de l'Homme

Plusieurs méthodes de contrôle et de planification existent dans la littérature. Ces méthodes sont souvent conçues pour des scénarios très spécifiques qui peuvent être catégorisées comme suit :

Croisement dans un couloir : En utilisant les mesures provenant des user studies, des systèmes robotiques qui imitent le comportement humain [Yoda 97], ou encodent un comportement explicite d'évitement [Pacchierotti 06b] ont été réalisés pour un robot qui croise une personne dans un couloir. Notons que ces systèmes sont spécifiques à cette tâche ; de plus, ils sont limités à une personne et ne fonctionnent que sous des hypothèses fortes sur l'environnement.

Suivi : Le robot qui suit une personne est un cas largement étudié dans le domaine. Les méthodes pour réaliser cette tâche varient du simple maintien de distance [Yoshimi 06] au suivi fondés sur des forces attractives [Takemura 07], à la découverte d'une roadmap [Zender 07] ou encore à la planification probabiliste [Hoeller 07].

Conserver une formation : Un robot qui va coexister avec les hommes doit aussi respecter certaines règles de formation qu'on suit dans nos vies quotidiennes. Dans [Nakauchi 02], Nakauchi et Simmons présentent un robot qui se met en queue comme une personne pour avancer vers son but. D'autres comportements, que l'on peut peut aussi considérer comme relevant des règles sociales, ont été étudiés tels que l'insertion du robot dans une formation circulaire pour s'adresser à un groupe de personnes [Althaus 04].

Navigation libre : Bien qu'on robot soit doté de capacités d'interaction sociale, certaines de ses tâches ne nécessitent pas une telle interaction. En revanche, le robot doit toujours prendre en compte l'existence des personnes même si l'interaction avec eux est minimale. Bennewitz et al. ont présenté dans [Bennewitz 05] un robot qui détecte et analyse les mouvements des personnes dans l'environnement. Pour aller d'un endroit à un autre, le robot prend en compte les mouvements des gens pour ne pas les perturber. Au contraire, [Sasaki 06] présente une pièce intelligente et un robot qui prend les chemins pris par les hommes pour produire une comportement plus proche aux hommes.

Dans [Madhava Krishna 06], Krishna et al. ont présenté un planificateur de mouvement qui raisonne sur les capacités sensorielles et dynamiques du robot et de l'environnement pour produire des chemins qui garantissent l'absence de collision. Ce planificateur permet de produire des chemins qui évitent des collisions possibles avec les personnes tout en restant optimal en temps d'exécution.

La Manipulation en Présence de l'Homme

La recherche sur la manipulation en présence de l'homme est un sous-domaine de l'interaction homme-robot relativement jeune et encore peu exploré. On trouve, toutefois, des travaux intéressants qui traitent des problèmes de manipulation en présence ou en collaboration avec l'homme. On peut catégoriser les méthodes existantes dans la littérature par rapport à leurs méthodes :

Fonction de danger : Une méthode utilisée pour générer des mouvements en interaction proche avec l'homme est d'évaluer le danger avec une fonction, et de trouver des postures de robot qui la minimise. Dans [Ikuta 03], Ikuta et al. ont présenté une stratégie complète pour assurer la sécurité de la personne avec qui le robot interagit en considérant une fonction dite "de danger" basée sur l'estimation de la force d'impact du robot. Cette stratégie est utilisée dans [Nokata 02] pour produire des mouvements sûrs de l'organe terminal du robot manipulateur.

Kulić et al. [Kulić 05] ont présenté une méthode similaire basée sur la distance entre l'homme et le robot en considérant toute la structure du robot.

La vision humaine : Une autre information qui contribue à la sécurité est le champ de vision de l'homme. On suppose que l'homme est conscient de ce qu'il regarde et que le danger diminue si le robot bouge tout en restant visible à l'homme. Dans [Traver 00], Traver et al. présentent une fonction de danger qui inclut la direction du regard de l'homme. Une technique de minimisation permet de choisir un chemin visible et sûr.

Cette section a brièvement présenté les différentes méthodes de génération de mouvement pour un robot qui partage l'environnement avec les hommes. Ces méthodes restent des approches réactives pour des scénarios souvent spécifiques. Une méthode de planification qui ne raisonne pas seulement sur le présent mais aussi sur le futur est une pièce manquante dans la littérature.

Un planificateur pour la navigation en présence de l'homme

L'introduction des robots dans la vie quotidienne apporte, comme nous l'avons vu plus haut, de nouveaux problèmes. Un de ces défis porte sur la nécessité d'élaborer des méthodes et des algorithmes de planification pour une navigation en présence de l'homme qui assure assure la sécurité et le confort.

Des études utilisateur sur le partage de l'espace homme-robot nous ont permis de déterminer trois critères pour la planification de mouvements sûrs et confortables : le "critère de sécurité", le "critère de visibilité" et le "critère des zones cachées". Chaque critère est représenté par un ensemble de valeurs numériques stockées dans une grille 2D combinant divers coûts. Ce valeurs dépendent de la position relative du robot à l'homme ainsi que de l'état, de la posture et des préférences de l'homme.

Le Critère de Sécurité

Le critère de sécurité permet de garder une distance acceptable entre l'homme et le robot pour diminuer le danger. Des coûts élevés sont attribués aux zones proches de l'homme. Par contre dans certains cas, si la tâche le nécessite le robot peut s'approcher de l'homme avec lequel il souhaite interagir. Cette propriété est représentée par "une grille de sécurité". Cette grille contient une distribution de coût de forme gaussienne centrée sur l'homme.

La figure 7.1 illustre une grille de sécurité construite par rapport à l'homme.

Le Critère de Visibilité

Ce critère concerne le confort de l'homme afin d'éviter les surprises et les gênes lors de l'interaction. En effet, on peut considérer que l'homme est plus en confiance lorsque le robot est visible. Pour cela ce critère permet au robot de rester dans le champ de vue de l'homme.

Cette propriété est représentée par "une grille de visibilité". Les coûts des cellules qui sont hors du champ de vue de l'homme sont plus importants que ceux qui sont dans le champ (figure 7.2). Le coût d'une position peut être interprété comme l'effort que l'homme doit faire pour voir ce point.

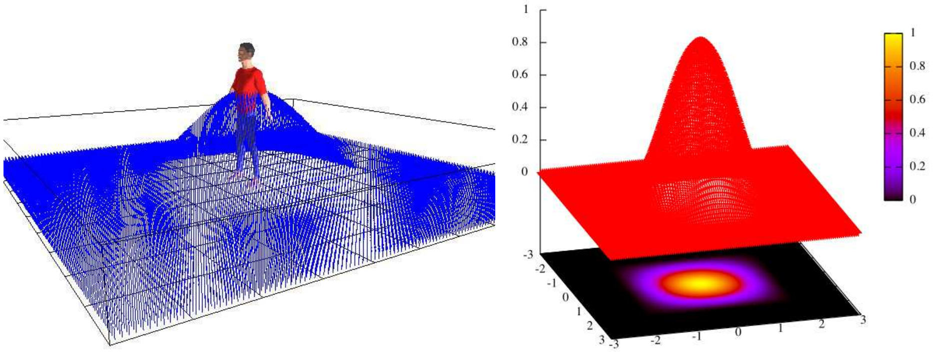


Fig. 7.1: *La grille de sécurité est construite autour de chaque homme dans l'environnement. Les des petites lignes verticales montrent les emplacements des cellules. Leurs hauteurs sont proportionnelles aux coûts.*

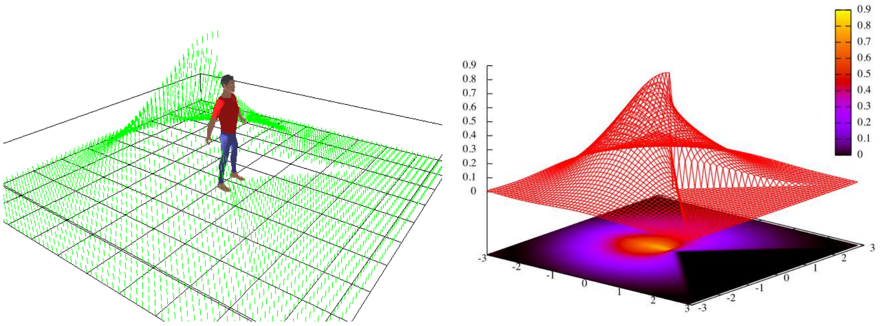


Fig. 7.2: *La grille de visibilité basée sur le champ de vision.*

Le Critère des Zones Cachées

Dans les grilles mentionnées dans les sections précédentes, les coûts sont calculés sans prendre en compte les obstacles qui se trouvent dans l'environnement. Par contre les obstacles qui se trouvent près de l'homme, peuvent avoir des effets différents sur la sécurité et le confort. En effet, ce critère traite les zones cachées par les obstacles. Lorsque le robot apparaît à l'homme alors qu'il était masqué par un obstacle, il peut le surprendre.

La trajectoire du robot doit donc éviter au maximum ces zones cachées qui sont représentées par des coûts décroissants placés derrière les obstacles (figure 7.3).

Calcul un chemin pour le robot

Les trois grilles précédemment calculées sont fusionnées en leurs accordant des importances différentes. Dans cette nouvelle grille, l'algorithme $A^\star$ est utilisé pour trouver le chemin le moins coûteux et sans collision dans l'environnement. Ce chemin prend en compte la sécurité de l'homme, son champ de vue et aussi les effets des obstacles environnants.

Une comparaison entre des chemins planifiés par un planificateur classique et le "Human-Aware Navigation Planner" (HANP) est illustrée par la figure 7.4. Les chemins produits par un planificateur classique sont faisables mais pas acceptables puisque le robot risque de passer trop près de l'homme ou de surgir brusquement derrière un

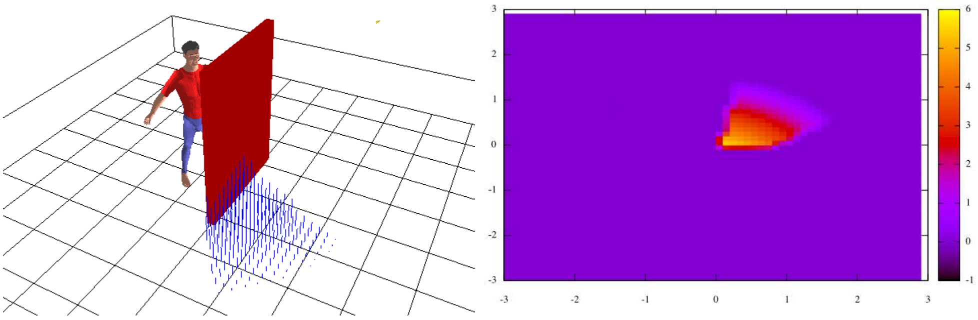


Fig. 7.3: *Définifition des coûts attribués aux zones cachées par les obstacles. Ces coûts supplémentaires découragent le robot de s'approcher trop près des obstacles.*

obstacle. Pour le même scénario, le chemin planifié par HANP prend en compte la sécurité, la visibilité de l'homme et évite les apparitions soudaines du robot à l'homme.

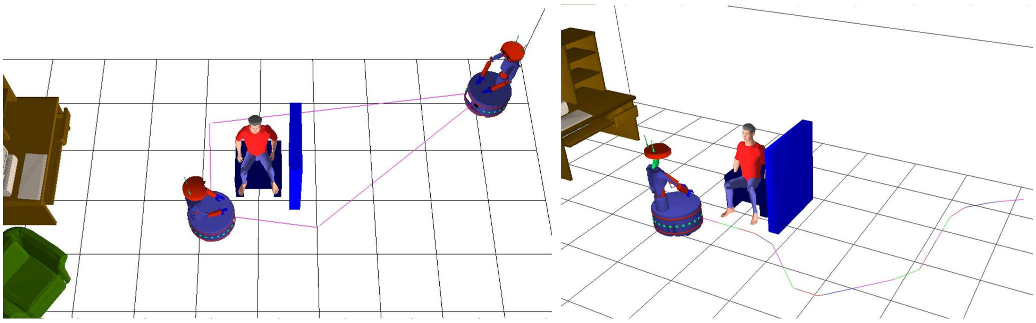


Fig. 7.4: *La comparaison des chemins planifiés par un planificateur classique (gauche) et par HANP (Human-Aware Navigation Planner) (droite).*

Un planificateur pour la manipulation en présence de l'homme

Comme le robot et l'homme sont en interaction proche, les problèmes de manipulation ne peuvent pas être traités en 2D comme pour la navigation. Le robot ainsi que l'homme doivent être représentés en 3D. Cette section présente une structure d'accueil pour la planification de la manipulation et son implémentation appelée "Human-Aware Manipulation Planner" (HAMP). Ce planificateur est conçu pour un robot dont le but est de tendre un objet à l'homme.

Les études sur les comportements sociaux entre l'homme et le robot permettent d'extraire certaines propriétés sur lesquelles se fonde notre planificateur. On a choisi de décomposer le problème de la planification en trois étapes ; (1) trouver les coordonnées spatiales du point le plus pertinent où le transfert d'objet va se dérouler, (2) calculer le chemin que l'objet va suivre comme s'il était un objet volant, (3) et enfin, calculer le mouvement du corps entier du robot en s'adaptant au mouvement de l'objet pour satisfaire plusieurs tâches à la fois en bénéficiant notamment de la forte redondance de certains robots afin de rendre le mouvement plus expressif.

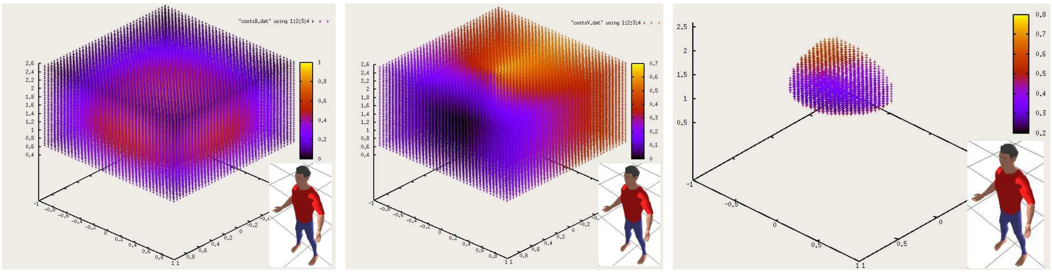


Fig. 7.5: *Les trois fonctions de coûts. De gauche à droite : sécurité, visibilité et confort.*

Détermination du Point de Transfert

La première étape de la planification consiste à déterminer les coordonnées de l'endroit où le robot va positionner l'objet. Ce point sera le point où l'objet va changer de main et passer du robot à l'homme. Pour calculer ce point, on adopte une approche similaire à celle de la navigation : les fonctions de coûts.

L'espace autour de l'homme est évalué au moyen de trois fonctions de coûts :

Fonction de sécurité : Cette fonction évalue un point autour de l'homme par rapport à sa distance à l'homme. Un point est coûteux s'il se trouve proche de l'homme.

Fonction de visibilité : Cette fonction estime le champ de vue de l'homme. Plus l'endroit est difficile à voir, plus le coût est élevé.

Fonction de confort : Le confort du bras de l'homme est représenté par cette fonction. Le coût d'un point est calculé en combinant la variation des degrés de liberté et l'énergie potentielle de la posture du bras une fois la position atteinte.

Ces trois fonctions sont illustrées dans la figure 7.5 pour les points qui se trouvent autour de l'homme. Ces fonctions sont combinées et le point ayant le coût minimal est choisi pour être le point de transfert de l'objet.

Le chemin de l'objet

Après avoir trouvé le point de transfert, c'est-à-dire l'endroit où le robot va positionner l'objet, la deuxième étape consiste à calculer le chemin que l'objet va emprunter. Dans cette étape, on considère que l'objet est un corps volant ("free-flying") qui va se déplacer de sa position initiale, la pince du robot également à sa position initiale, à sa position finale, le point de transfert.

Pour calculer le chemin de l'objet, on fusionne la fonction de sécurité et la fonction de visibilité. Une recherche $A^\star$ est conduite pour trouver le meilleur le chemin en fonction de critère.

Le chemin du robot

La dernière étape de planification consiste à produire le chemin du robot pour tendre l'objet. Comme on connaît le chemin que l'objet doit emprunter, cette étape consiste à adapter la posture du robot pour réaliser ce chemin.

Parmi différentes méthodes d'adaptation de posture, nous avons choisi la Cinématique Inverse Généralisée [Nakamura 90][Baerlocher 04][Yamane 03] car elle permet :

- d'inclure plusieurs tâches avec des priorités en assurant la réalisation de la tâche la plus prioritaire,
- d'adapter aisément l'algorithme à différents types de robots.

Le chemin de l'objet est échantillonné, et pour chaque échantillon le résolveur de cinématique inverse est exécuté. On utilise deux tâches pour la cinématique inverse. La tâche la plus prioritaire consiste à suivre le chemin de l'objet avec les articulations qui ont un effet sur la pince du robot. La tâche la moins prioritaire consiste à orienter le regard du robot vers l'objet pour bien exprimer l'intention du robot. Ainsi le robot ne va pas seulement tendre l'objet mais aussi le suivre du regard pour bien exprimer son intention.

Ainsi, HAMP trouve automatiquement une position sûre, visible et confortable pour positionner l'objet et génère un mouvement à la fois sûr et lisible (figure 7.6). Grâce à ce planificateur, le robot a plus d'initiative dans cette interaction en décidant où le transfert va se dérouler. Ainsi, la problématique de planification consistant à simplement trouver un chemin d'une configuration à une autre est transformé en “trouver un chemin pour réaliser une tâche”.

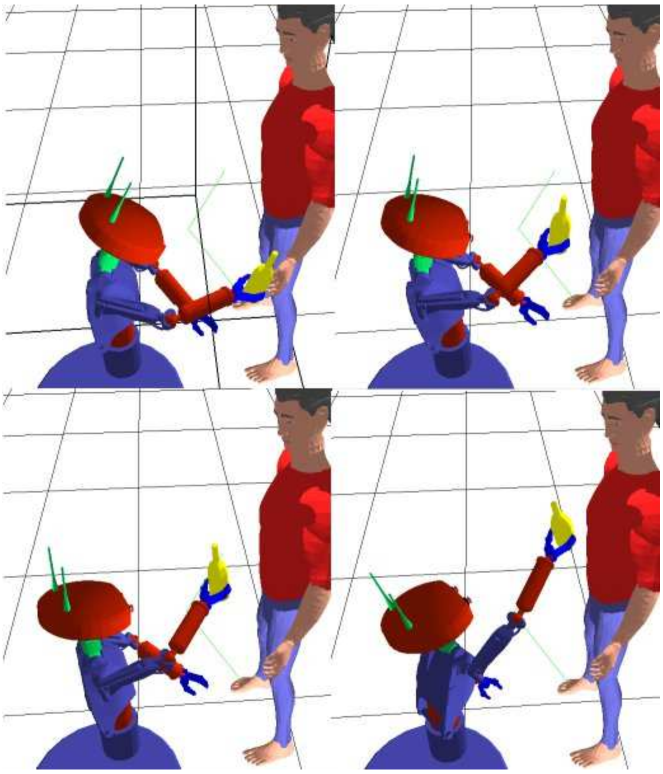


Fig. 7.6: *Le chemin calculé pour le scénario de “tendre un objet à l'homme”. Le mouvement du robot assure la sécurité et visibilité, et exprime bien son intention.*

Bien que ce planificateur produise des mouvements pour le haut du corps du robot, l'emplacement initial du robot joue un rôle très important sur la qualité et la faisabilité du mouvement. Pour sélectionner des configurations de départ convenables pour la tâche, nous avons mis en oeuvre un mécanisme dit de “Prise de Perspective”. Ce

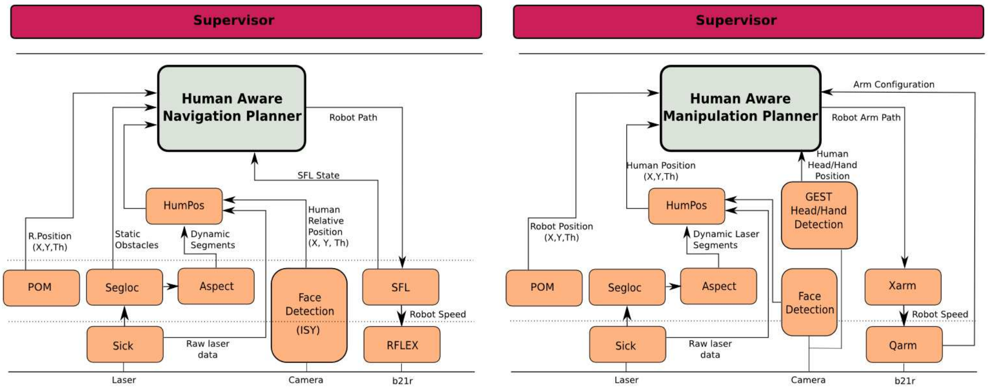


Fig. 7.7: *Intégration des deux planificateurs dans l'architecture LAAS.*

mécanisme sert à produire et évaluer des configurations en fonction du pourcentage de visibilité et d'accessibilité de l'homme et du robot, qui seront utilisés comme entrées du planificateur de mouvement.

Intégration et Résultats

Les deux planificateurs de mouvement ont été intégrés dans deux robots sous la forme de modules fonctionnels Genom [Fleury 97] compatible avec l'architecture LAAS [Alami 98]. Comme le planificateur de navigation et le planificateur de manipulation partagent le même environnement et utilisent la même bibliothèque de raisonnement géométrique Move3D [Siméon 01], ils ont été intégrés en un seul module nommé MHP (Manipulation in Human Presence).

Ce module reçoit les commandes de superviseur et produit les mouvements qui vont être exécutés par les modules de contrôle des robots physiques.

Intégration et évaluation de HANP

Comme le planificateur de navigation est fortement dépendant des positions et des orientations des personnes dans l'environnement, il est connecté aux modules de perception. Pour localiser les personnes, on utilise le télémètre de laser et les caméras qui se trouvent sur le robot. Le module HumPos, chargé de la détection et du suivi des jambes par laser, fournit à MHP les positions et les orientations des personnes dans l'environnement. Cette information est renforcée par un autre module, appelé ISY, qui est en charge de détecter les personnes à l'aide des caméras.

A partir de la configuration du robot et de la carte d'environnement, le planificateur de navigation génère un chemin et l'envoie au module d'exécution en forme de points de passage (figure 7.7).

Le figure 7.8 illustre un mouvement du robot. Dans ce scénario, le robot veut aller à coté de l'homme qui est dans une position où il est supposé ne pas être conscient de la présence du robot. Le planificateur planifie un chemin sûr et qui évite de surgir brusquement derrière le tableau.

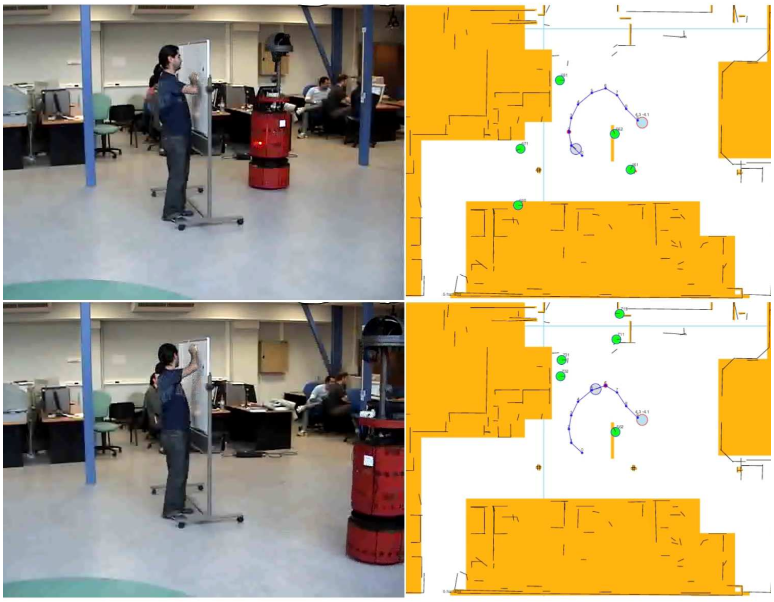


Fig. 7.8: *Le robot est invisible à l'homme au départ. Il prend un chemin qui n'est pas seulement sûr mais aussi confortable en évitant de surgir brusquement derrière le tableau.*

Intégration et évaluation de HAMP

Le planificateur de manipulation nécessite des informations plus détaillées des hommes dans l'environnement puisqu'il a besoin de construire un modèle 3D. Pour construire un modèle 3D de l'homme avec lequel le robot interagit, le planificateur est lié à un module de perception, appelé GEST, qui fournit les positions et les orientations des mains et de la tête.

Avec cette information, MHP construit un modèle 3D de l'homme, génère un chemin et l'envoie au module d'exécution en forme de points de passage (figure 7.7).

L'exemple, illustré dans le figure 7.9, montre un robot qui tend une bouteille à une personne. Le point où le transfert de l'objet va se dérouler est choisi par MHP et est atteint avec un mouvement sûr, visible et confortable.

Conclusion

Pour produire les mouvements d'un robot “social”, on a besoin de capacités de raisonnement explicite sur la présence des hommes. De nouveaux algorithmes de planification de mouvement sont donc sont nécessaires. Le robot ne doit pas seulement considérer ses propres caractéristiques même mais aussi prendre en compte son environnement, les personnes présentes ainsi que des règles sociales et des protocoles de comportement.

Ce travail représente une première approche qui intègre l'homme dans la boucle de planification en attaquant au problème avec un point de vue plus global que des méthodes spécifiques de génération de mouvement.

Un état de l'art succinct mais très focalisé sur les différentes méthodes de génération de mouvement en présence de l'homme est décrit dans ce travail. Ces méthodes sont souvent fondées sur des approches ad-hoc, réactives ou demandent trop d'effort de la

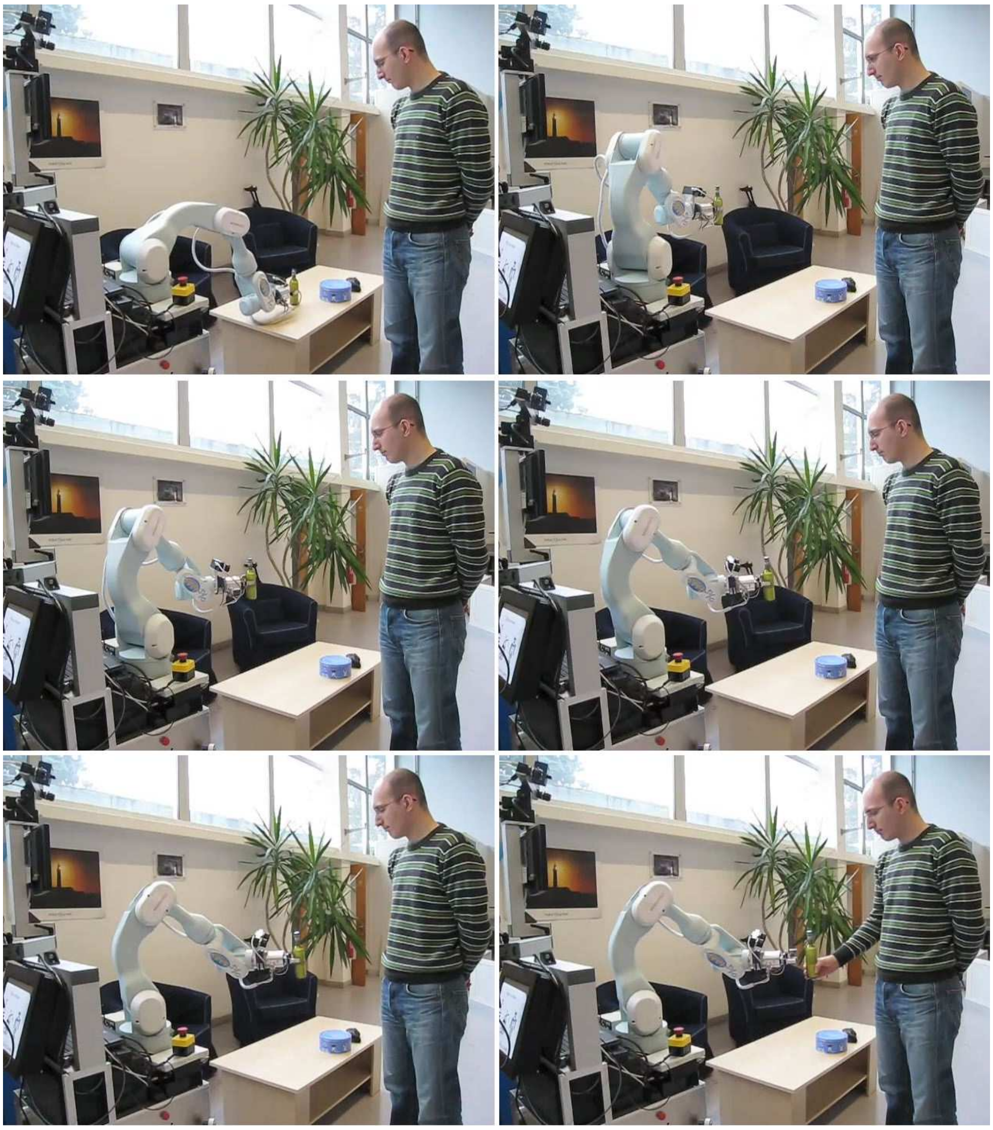


Fig. 7.9: *Jido tend la bouteille à une personne. L'endroit où le transfert de l'objet va se passer est choisi par le robot et est atteint avec un mouvement sûr, visible et confortable.*

part de l'homme.

On a présenté un premier planificateur appelé "Human-Aware Navigation Planner", un planificateur de navigation qui prend en compte explicitement les hommes dans l'environnement. Trois critères importants sont extraits des études utilisateurs et représentés par des fonctions de coûts qui fournissent une évaluation spatiale à la position du robot. Ces critères, nommés sécurité, visibilité et zones cachées, maintiennent une bonne base pour une large classe de scénarios d'interaction.

L'approche basée sur les coûts et les grilles a permis une flexibilité et extensibilité ainsi qu'un réduction de la complexité computationnelle.

Les méthodes et algorithmes pour un planificateur de manipulation sont aussi présentés dans ce travail. On a proposé un cadre général permettant d'exprimer le problème de la manipulation en présence de l'homme. Ainsi, la problématique générale de planification de mouvement - trouver un chemin d'une configuration à une autre

- est transformée en une question plus riche : trouver un chemin pour réaliser une tâche interactive. Cette approche a donné lieu au développement d'un planificateur, Human-Aware Manipulation Planner, qui répond au problème de tendre un objet à l'homme.

La division du problème en trois étapes successives a permis de réduire la complexité. Enfin, l'utilisation de la cinématique inverse généralisée pendant la planification a permis de produire des mouvements intentionnels.

Les deux planificateurs sont intégrés et illustrés dans deux plates-formes robotiques sous la forme de composants logiciels interagissant avec d'autres modules de perception, de supervision et de contrôle. ont présents. Les mouvements des robots ont également été évalués à travers des études utilisateurs et ont été considérés comme pertinents, naturels et confortables par la plupart des participants.

Perspectives

Ce travail est le premier qui traite le problème de planification de mouvement en présence de l'homme en proposant une approche globale générique. Étant le premier, plusieurs questions et perspectives sont découvertes.

Les perspectives les plus importantes sont suivantes :

- Une des notions les plus importantes qui a un effet direct sur la sécurité et le confort de l'homme est la dynamique du mouvement du robot. Dans ce travail, nous nous sommes intéressés à la forme des chemins. Dans le monde réel, on ne peut pas garantir la sécurité si on ne prend pas en compte la vitesse de robot. Ainsi, il serait intéressant de reconsidérer l'approche classique "planifier un chemin → transformer le chemin à un trajectoire → exécuter la trajectoire" en intégrant un raisonnement sur les vitesses dès l'étape de planification.
- Les méthodes et les algorithmes présentés dans ce travail considèrent l'homme comme statique bien que dans la vie réelle, les gens sont en général en mouvement. Bien que les planificateurs "Human-Aware" permettent de contourner partiellement ce problème grâce à des calculs rapides, le système n'inclut pas les mouvements humains. Modéliser ces mouvements et les intégrer dans le planificateur est également un chantier pour le futur.
- Le point précédent fait apparaître une autre notion qu'il serait intéressant de prendre en compte. Dans l'environnement de simulation, on suppose que le robot voit toutes les personnes. Par contre pour un robot réel, cette hypothèse ne peut pas être applicable puisque la perception du robot dépend fortement de ses capteurs et de l'environnement. Pour mieux s'adapter aux scénarios réels, le robot doit gérer explicitement l'incertitude sur la présence, la position et l'orientation des hommes alentour.
- Le planificateur de navigation et manipulation sont exécutés séquentiellement. Cela résulte en un comportement de robot non optimal ; le robot avance à sa destination, s'arrête et tend l'objet. Avec l'intégration d'un approche probabiliste (comme les Rapid-Random Trees ou RRT), le robot peut commencer à démarrer le mouvement de manipulation avant que sa base arrive à sa destination.
- Une dernière perspective qu'il faut souligner est l'exploration de la planification pour l'homme. Pour un robot qui aide une personne, il y aura des situations où le robot ne trouvera pas de solution sans causer de l'inconfort. Le robot devra alors raisonner sur les coûts des mouvements de l'homme pour lui proposer un plan coordonné dans lequel le robot et l'homme participent ensemble à la réalisation

de la tâche.

APPENDIX
A

Object Hand Over User Study

In order to understand the space sharing between a robot and a human in a object hand over scenario, we have conducted a user study involving PeopleBot and 12 participants with the collaboration of University of Hertfordshire under FP6 COGNIRON Project.

Trial Setup

The Trials were conducted in University of Hertfordshire "Robot House" (dedicated to Human-Robot Interaction (HRI) Studies in a domestic environment, figure A.1) in the summer of 2006. The aim was to understand from the user's perspective how a robot with humanoid arms (see figure 1) should approach and hand over a can to a seated person.

Twelve participants aged between 21-41 (eight males and four females) were recruited for the study. They were recruited immediately after they finished taking part in a five-week long-term HRI experiment where they interacted with a robot twice a week on an hour per session basis.

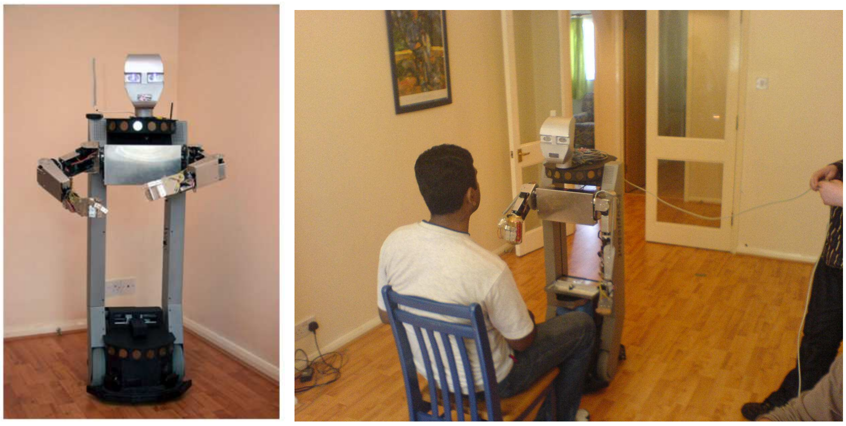


Figure A.1: *PeopleBot and UH's Robot House*

Trials

The first stage of the trials involved the subjects interacting actively with the experimenters and the robot, regarding their preferences of how the robot should approach and hand them an object. The purpose of this approach was to actively involve the subject in the study, in contrast to our previous experiments where the subjects passively experienced and later chose from a set of preprogrammed robot approach behaviors. For the current trials, subjects guided the creation of a handing over gesture for the robot arm at their preferred position for handing over a can of soft drink. This gesture was then coordinated with the approach movements of the robot's base in four different ways:

1. Robot starts moving towards the subject only after it completed its handing over gesture.
2. Robot starts moving towards the subject but only executes its handing over gesture coordinated from 1 m far from the subject.
3. Robot starts moving towards the subject but only executes its handing over gesture coordinated from 2 m far from the subject.
4. Robot starts executing its handing over gesture after it has stopped.

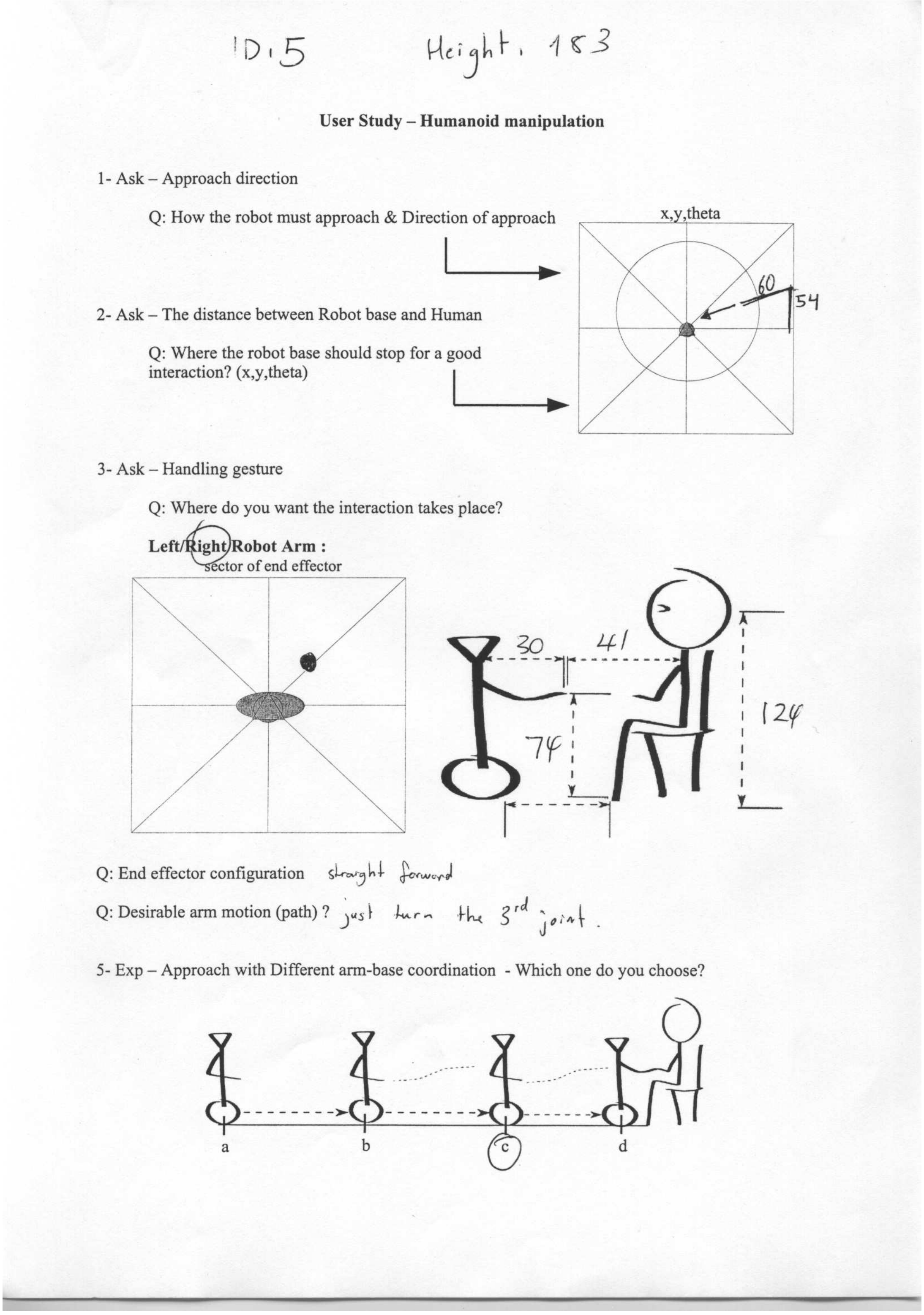
For each participant, a data sheet was used to acquire measures from the trials (figure A.2). The acquired data were:

- The direction of robot's approach.
- The distance between the robot base and the human.
- Distances between robot's (base, head, hand) and human's (head, feet, hand) body parts.
- The choice of the 4 coordinated hand over motions.

Results

The results show that 58.3% of the subjects prefer the robot to approach from the subject's front, 25% prefer the robot to approach from the subject's right front and 8.3% for each robot approach from subject's right and subject's left front. We found that 75% of the subjects prefer the robot to hand them the object from directly in front, 17% prefer the robot to hand the object at their right front and 8% prefer the robot to hand them the object to their left front. The summary of these two results shows that the direction where the robot should hand over an object has most influence on determining where the robot should approach.

The mean preferred robot base approach interaction distance for the whole sample was 66.8 cm. The minimum distance was 58 cm, and the maximum distance was 82 cm. Assuming the distances between the subjects and the robot should be measured from subjects' chest (i.e. centre of the chair), the results show that the subjects prefer to interact with the robot within their personal zone [Hall 66].



ID: 5 Height: 183

User Study – Humanoid manipulation

1- Ask – Approach direction

Q: How the robot must approach & Direction of approach

2- Ask – The distance between Robot base and Human

Q: Where the robot base should stop for a good interaction? (x,y,theta)

3- Ask – Handling gesture

Q: Where do you want the interaction takes place?

Left/[✓ Right] Robot Arm :
sector of end effector

Q: End effector configuration straight forward

Q: Desirable arm motion (path) ? just turn the 3rd joint.

5- Exp – Approach with Different arm-base coordination - Which one do you choose?

Figure A.2: *The data sheet used to acquire measures from the hand over trial.*

Two clusters of the preferred robot base approach interaction distances were found which centred at 72.42 cm and 61.25 cm, and were significantly different from each other. The results also show that the subjects preferred robot base approach interaction distances were positively correlated with subjects preferred robot handing over distances. This may imply that subjects who were comfortable with the robot being physically close to them prefer to interact closely, while subjects whom prefer to interact with the robot at a larger distance, prefer the robot to stay further away.

APPENDIX
B

Internal Evaluation User Study

The paths generated by Human-Aware Manipulation Planner, the speed profiles produced by limited-jerk execution module (Xarm, §5.4.2) and the utility of gripper force sensor are evaluated in an internal user study with the participation of 12 subjects. The participants are asked to evaluate the comfort of specific parts as well as the global interaction for the object hand over task. For the trials, mobile manipulator Jido (§5.2) is used.

Trial Setup

This user study took place in the robotics lab of LAAS. The subjects were composed of M.S and Ph.D students in the LAAS-CNRS robotics lab having at least a minimum familiarity with the robots (figure B.1). All the subjects happened to be right-handed except one who preferred using his right hand even though he was mixed-handed.

Each participant is placed on 1 m far from the robot's front laser facing toward the robot. The robot was holding a bottle in its grippers. The participants are told that the robot's base will not move, but the robot arm will hand the bottle and that they can grab the bottle when ever they want.

Figure B.1: *User Study setup*

The Trial

The robot, having a bottle in its grippers, is run in 6 trials with different types of hand over motions:

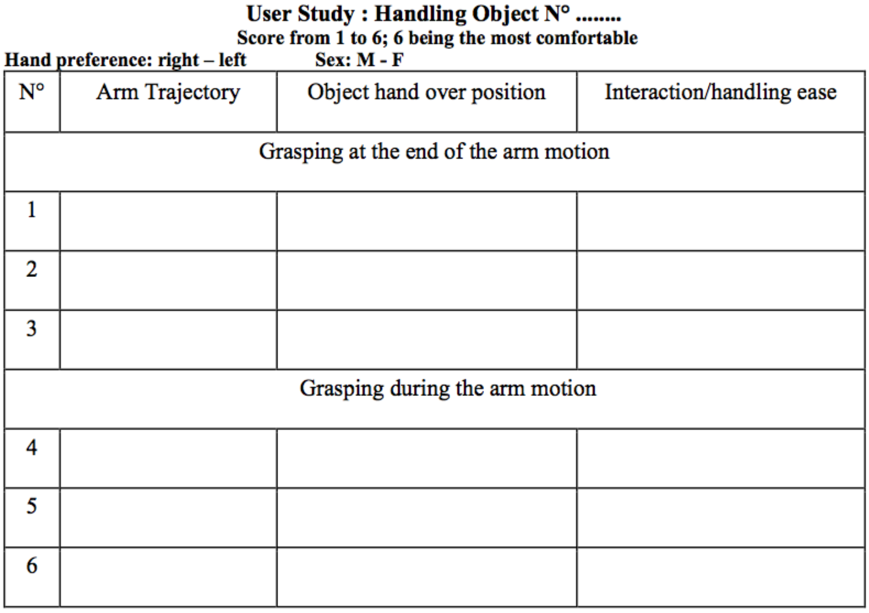
1. The robot arm makes a forward motion of 0.35 m without taking into account the position of the human. After reaching its destination, it waits for a pulling force applied to the bottle in order to release it.

2. This time the robot detects the human with its laser sensor.Then moves its arm towards the human to the transfer point calculated by HAMP. After reaching its destination it releases the bottle once a pulling force is detected.

3. This time whole body model is constructed in HAMP with leg/head/hand detection and tracking modules. The robot moves its arm towards the hand of the person using HAMP. After reaching its destination it releases the bottle once a pulling force is detected.

4. 5. 6. The second part of the study consists of playing same motions with the force detection enabled during arm motion. This option enables the participant to grasp the bottle whenever he/she wants during the hand over motion.

After the experiment, each subject is given a questionnaire (figure B.2) and asked to rate the trials by evaluating the motion of the arm and the handling position. The participants gave points from 1 to 6 to each trial, 1 being the less comfortable and 6 being the most.



User Study : Handling Object N° ....

Score from 1 to 6; 6 being the most comfortable

Hand preference: right – left **Sex: M - F**

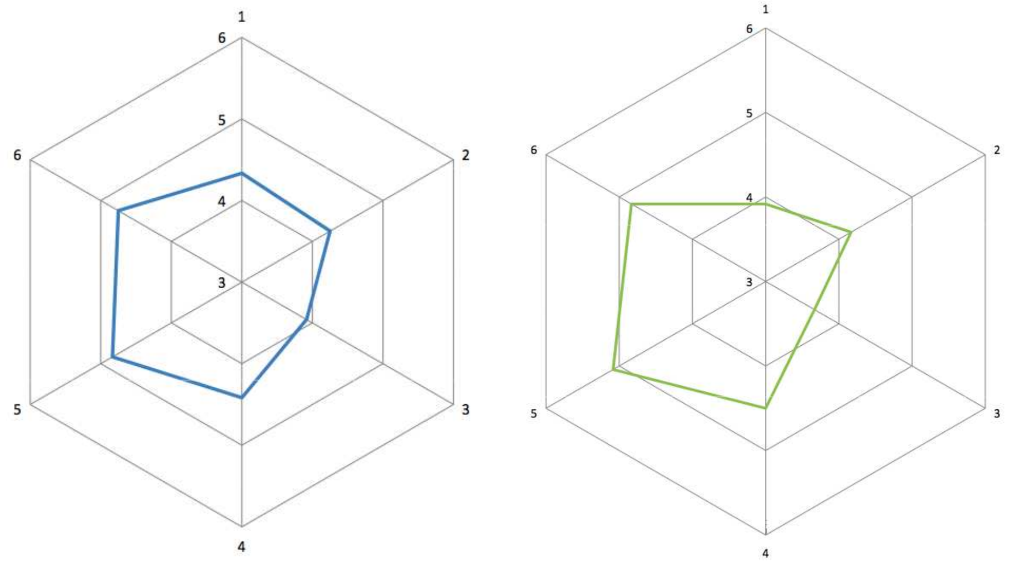
N°	Arm Trajectory	Object hand over position	Interaction/handling ease
Grasping at the end of the arm motion			
1			
2			
3			
Grasping during the arm motion			
4			
5			
6			

Figure B.2: *A questionnaire is distributed to each participant to rate the comfort of robot's motion.*

Results

The results for the evaluation of arm motion (figure B.3.a) show that the 6th and 5th trials are rated as the most comfortable with 4.83 and 4.75 points in average. The first 3 trials are rated as less comfortable then the last 3 trials showing the positive effect of allowing the user grasp the object whenever he/she wants. The results also show that the motions generated by Human-Aware Manipulation Planner are rated as more comfortable than the ones without.

The second part of the questionnaire consists of rating the position where the object handling occurs. The results of this evaluation (figure B.3.b) are parallel to the previous one as the motions of 5th and 6th trials are rated as most comfortable with 5.08 and 4.83 points. Also 79% of the participants rated the amount of force require to pull object from robot's gripper as comfortable enough.



(a) Trajectory of the arm. The arm motions in 6th and 5th trials are evaluated as most comfortable.

(b) Object Hand Over Position. The 6th and 5th trials are evaluated as most comfortable.

Figure B.3: *Results of the evaluation user study. Corners of the pentagon represent each trial and the distance between the center and the corners illustrate average of the scores given by participants*

BIBLIOGRAPHY

[Abdel-Malek 05] K. Abdel-Malek, Z. Mi, J. Yang and K. Nebel. *Optimization-based layout design.* Applied Bionics and Biomechanics, vol. 2, no. 3-4, pages 187–196, January 2005.

[Alami 98] R. Alami, R. Chatila, S. Fleury, M. Ghallab and F. Ingrand. *An Architecture for Autonomy.* International Journal of Robotics Research, vol. 17, pages 315–337, 1998.

[Alami 06] R. Alami, A. Albu-Schaeffer, A. Bicchi, R. Bischoff, R. Chatila, A. De Luca, A. De Santis, G. Giralt, J. Guiochet, G. Hirzinger, F. Ingrand, V. Lippiello, R. Mattone, D. Powell, S. Sen, B. Siciliano, G. Tonietti and L. Villani. *Safe and Dependable Physical Human-Robot Interaction in Anthropic Domains: State of the Art and Challenges.* In A. Bicchi and A. De Luca, editeurs, Procceedings IROS Workshop on pHRI - Physical Human-Robot Interaction in Anthropic Domains, Beijing, China, October 2006.

[Althaus 04] P. Althaus, H. Ishiguro, T. Kanda, T. Miyashita and H.I. Christensen. *Navigation for human-robot interaction tasks.* In IEEE International Conference on Robotics and Automation, ICRA, volume 2, pages 1894–1900, New Orleans, USA, April 2004.

[Baba 06] Abedallatif Baba and Raja Chatila. *Simultaneous environment mapping and mobile target tracking.* In International Conference on Intelligent Autonomous Systems, IAS, Tokyo, Japan, January 2006.

[Baerlocher 04] Paolo Baerlocher and Ronan Boulic. *An inverse kinematics architecture enforcing an arbitrary number of strict priority levels.* The Visual Computer: International Journal of Computer Graphics, vol. 20, no. 6, pages 402–417, 2004.

[Bennewitz 05] M. Bennewitz, Wolfram Burgard, G. Cielniak and S. Thrun. *Learning Motion Patterns of People for Compliant Robot Mo-*

tion. International Journal of Robotics Research, vol. 24, no. 1, 2005.

[Bicchi 04] A. Bicchi and G. Tonietti. *Fast and Soft Arm Tactics: Dealing with the Safety-Performance Trade-Off in Robot Arms Design and Control.* Robotics and Automation Magazine, vol. 11, no. 2, pages 22–33, 2004.

[Boekhorst 05] R. Boekhorst, M.L. Walters, Kheng Lee Koay, K. Dautenhahn and C.L. Nehaniv. *A study of a single robot interacting with groups of children in a rotation game scenario.* In IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA, pages 35–40, Espoo, Finland, June 2005.

[Broquère 08] X. Broquère, D. Sidobre and I. Herrera-Aguilar. *Soft motion trajectory planner for service manipulator robot.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Nice, France, September 2008.

[Chatila 02] R. Chatila, R. Alami, T. Simeon, J. Pettre and L. Jaillet. *Safe, reliable and friendly interaction between humans and humanoids.* In IARP International Workshop on Humanoid and Human Friendly Robotics, pages 83–87, Tsukuba, Japan, December 2002.

[CHIL 04] Project CHIL. *Computers In the Human Interaction Loop - CHIL Website*, 2004.

[Clodic 07a] A. Clodic, M. Ransan, R. Alami and V. Montreuil. *A management of mutual belief for human-robot interaction.* In IEEE International Conference on Systems, Man and Cybernetics, ISIC, pages 1551–1556, Montreal, Canada, October 2007.

[Clodic 07b] Aurelie Clodic. *Supervision pour un robot interactif: Action et interaction pour un robot autonome en environnement humain.* PhD thesis, Universite Paul Sabatier, June 2007.

[Cogniron 08] Cogniron. *COGNIRON Project RA3 Final Deliverable.* Rapport technique, LAAS/CNRS, April 2008.

[Dautenhahn 06] K. Dautenhahn, M. Walters, S. Woods, K. L. Koay, C. L. Nehaniv, E. A. Sisbot, R. Alami and T. Siméon. *How may I serve you?: a robot companion approaching a seated person in a helping context.* In ACM SIGCHI/SIGART International Conference on Human-Robot Interaction, HRI, pages 172–179, Utah, USA, March 2006.

[De Santis 08] Agostino De Santis, Bruno Siciliano, Alessandro De Luca and Antonio Bicchi. *An atlas of physical human-robot interaction.* Mechanism and Machine Theory, vol. 43, no. 3, pages 253–270, March 2008.

[Fleury 95] S. Fleury, P. Soueres, J.-P. Laumond and R. Chatila. *Primitives for smoothing mobile robot trajectories.* IEEE Transactions on Robotics, vol. 11, no. 3, pages 441–448, June 1995.

[Fleury 97] S. Fleury, M. Herrb and R. Chatila. *Genom: a tool for the specification and the implementation of operating modules in a distributed robot architecture.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 842–848, Grenoble, France, September 1997.

[Fong 03] T. Fong, I. Nourbakhsh and K. Dautenhahn. *A Survey of Socially Interactive Robots.* Robotics and Autonomous Systems, vol. 42, no. 3-4, 2003.

[Fontmarty 07] Mathias Fontmarty, Thierry Germa, Brice Burger, Luis-Felipe Marin and Steffen Knoop. *Implementation of human perception algorithms on a mobile robot.* In IFAC Symposium on Intelligent Autonomous Vehicles, IAV, Toulouse, France, September 2007.

[Germa 07] T. Germa, F. Lerasle, P. Danes and L. Brethes. *Human / robot visual interaction for a tour-guide robot.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 3448–3453, San Diego, CA, USA, November 2007.

[Gockley 07] Rachel Gockley, Jodi Forlizzi and Reid Simmons. *Natural person-following behavior for social robots.* In ACM/IEEE International Conference on Human-Robot Interaction, HRI, pages 17–24, Washington DC, USA, March 2007.

[Goodrich 07] Michael A. Goodrich and Alan C. Schultz. *Human-robot interaction: a survey.* Foundations and Trends in Human-Computer Interaction, vol. 1, no. 3, pages 203–275, 2007.

[Gulati 08] Shilpa Gulati and Benjamin Kuipers. *High Performance Control for Graceful Motion of an Intelligent Wheelchair.* In IEEE International Conference on Robotics and Automation, ICRA, pages 3932–3938, Pasadena, USA, May 2008.

[Haddadin 07] Sami Haddadin, Alin Albu-Schaffer and Gerd Hirzinger. *Safety Evaluation of Physical Human-Robot Interaction via Crash-Testing, Robotics.* In Robotics: Science and Systems Conference, RSS, Atlanta, USA, June 2007.

[Haddadin 08] Sami Haddadin, Alin Albu-Schaffer and Gerd Hirzinger. *The role of the robot mass and velocity in physical human-robot interaction - Part I: Non-constrained blunt impacts.* In IEEE International Conference on Robotics and Automation, ICRA, pages 1331–1338, Pasadena, USA, May 2008.

[Hall 66] E. T. Hall. The hidden dimension. Doubleday, Garden City, N.Y., 1966.

[Hart 68] P. E. Hart, N. J. Nilsson and B. Raphael. *A Formal Basis for the Heuristic Determination of Minimum Cost Paths.* IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pages 100–107, July 1968.

[Herrera-Aguilar 07] Ignacio Herrera-Aguilar. *Commande des bras manipulateurs et retour visuel pour des applications à la robotique de service.* PhD thesis, University of Toulouse, Toulouse, France, September 2007.

[Hoeller 07] F. Hoeller, D. Schulz, M. Moors and F.E. Schneider. *Accompanying persons with a mobile robot using motion prediction and probabilistic roadmaps.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 1260–1265, San Diego, CA, USA, November 2007.

[Huber 08] Markus Huber, Markus Rickert, Alois Knoll, Thomas Brandt and Stefan Glasauer. *Human-Robot Interaction in Handing-Over Tasks.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, Munich, Germany, August 2008.

[Iagnemma 04] K. Iagnemma and Steven Dubowsky. *Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers.* Springer Tracts in Advanced Robotics, vol. 12, 2004.

[Ikuta 03] Koji Ikuta, Hideki Ishii and Makoto Nokata. *Safety Evaluation Method of Design and Control for Human-Care Robots.* International Journal of Robotics Research, vol. 22, no. 5, pages 281–297, 2003.

[Inoue 05] K. Inoue, S. Nonaka, Y. Ujiie, T. Takubo and T. Arai. *Comparison of human psychology for real and virtual mobile manipulators.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 73–78, Nashville, USA, August 2005.

[IRobot] IRobot. *Roomba Vacuum Cleaner Robot.* www.irobot.com.

[Jaillet 08] L. Jaillet, J. Cortés and T. Siméon. *Transition-based RRT for path planning in continuous cost spaces.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Nice, France, September 2008.

[Kang 03] T. Kang, S.H. Tillery and J. He. *Determining natural arm configuration along reaching trajectory.* In International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, volume 2, pages 1444–1447, Cancun, Mexico, September 2003.

[Katayama 03] M. Katayama and H. Hasuura. *Optimization principle determines human arm postures and "comfort".* SICE 2003 Annual Conference, vol. 1, pages 1000–1005, August 2003.

[Kelley 83] J. F. Kelley. *An empirical methodology for writing user-friendly natural language computer applications.* In Proceedings of the SIGCHI International Conference on Human Factors in Computing Systems, CHI, pages 193–196, New York, NY, USA, 1983. ACM.

[Kemp 07] C.C. Kemp, A. Edsinger and E. Torres-Jara. *Challenges for robot manipulation in human environments [Grand Challenges of Robotics].* IEEE Robotics & Automation Magazine, vol. 14, no. 1, pages 20–29, March 2007.

[Kleinehagenbrock 02] M. Kleinehagenbrock, S. Lang, J. Fritsch, F. Lomker, G. A. Fink and G. Sagerer. *Person Tracking with a Mobile Robot based onMulti-Modal Anchoring.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, Berlin, Germany, September 2002.

[Koay 07] K. L. Koay, E. Akin Sisbot, D. A. Syrdal, M. L. Walters, K. Dautenhahn and R. Alami. *Exploratory Study of a Robot Approaching a Person in the Context of Handling Over an Object.* In Association for the Advancement of Artificial Intelligence Spring Symposia, AAAI, Palo Alto, CA, USA, March 2007.

[Kolsch 03] Mathias Kolsch, Andrew C. Beall and Matthew Turk. *The Postural Comfort Zone for Reaching Gestures.* Human Factors and Ergonomics Society Annual Meeting Proceedings, vol. 47, pages 787–791, 2003.

[Kruijff 07] Geert-Jan M. Kruijff, Hendrik Zender, Patric Jensfelt and Henrik I. Christensen. *Situated Dialogue and Spatial Organization: What, Where... and Why?* International Journal of Advanced Robotic Systems, vol. 4, no. 2, 2007.

[Kulić 05] Dana Kulić and Elizabeth A. Croft. *Safe planning for human-robot interaction.* Journal of Robotic Systems, vol. 22, no. 7, pages 383–396, 2005.

[Latombe 91] Jean-Claude Latombe. Robot motion planning. Kluwer Academic Publishers, Norwell, MA, USA, 1991.

[Laugier 07] Christian Laugier and Raja Chatila, editeurs. Autonomous navigation in dynamic environments, volume 35 of *Springer Tracts in Advanced Robotics.* Springer, 2007.

[Laugier 08] C. Laugier, Dizan Vasquez, M. Yguel, Th. Fraichard and O. Aycard. *Geometric and Bayesian models for safe navigation in dynamic environments.* Intelligent Service Robotics, vol. 1, no. 1, pages 51–72, 2008.

[Laumond 97] Jean-Paul Laumond. Robot motion planning and control. Springer-Verlag Telos, 1997.

[Lavalle 06] Steven Lavalle. Planning algorithms. Cambrisge University Press, 2006.

[Littlejohn 05] Stephen W. Littlejohn and Karen A. Foss. Theories of human communication. Thomson Wadsworth Communication, 2005.

[Losch 07] M. Losch, S. Schmidt-Rohr, S. Knoop, S. Vacek and R. Dillmann. *Feature Set Selection and Optimal Classifier for Human Activity Recognition.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 1022–1027, Jeju, Korea, August 2007.

[Low 03] Setha M. Low and Denise Lawrence-Zuniga. The anthropology of space and place: Locating culture. Blackwell Publishing, 2003.

[Lussier 05] Benjamin Lussier, Re Lampe, Raja Chatila, Jérémie Guiochet, Félix Ingr, Marc olivier Killijian and David Powell. *Fault Tolerance in Autonomous Systems: How and How Much.* In IARP/IEEE-RAS/EURON Joint Workshop on Technical Challenge for Dependable Robots in Human Environments, pages 16–18, Nagoya, Japan, June 2005.

[Madhava Krishna 06] K. Madhava Krishna, R. Alami and T. Simeon. *Safe proactive plans and their execution.* Robotics and Autonomous Systems, vol. 54, no. 3, pages 244–255, 2006.

[Marler 05] R. T. Marler, J. Yang, J. S. Arora and K. Abdel-Malek. *Study of Bi-Criterion Upper Body Posture Prediction using Pareto Optimal Sets.* In IASTED International Conference on Modeling, Simulation and Optimization, MSO, Oranjestad, Aruba, August 2005.

[Martinez-Garcia 05] E. A. Martinez-Garcia, Ohya Akihisa and Shin'ichi Yuta. *Crowding and guiding groups of humans by teams of mobile robots.* In IEEE Workshop on Advanced Robotics and its Social Impacts, ARSO, pages 91–96, Nagoya, Japan, June 2005.

[Montreuil 07] V. Montreuil, A. Clodic, M. Ransan and R. Alami. *Planning human centered robot activities.* In IEEE International Conference on Systems, Man and Cybernetics, ISIC, pages 2618–2623, Montreal, Canada, October 2007.

[Nakamura 90] Yoshihiko Nakamura. Advanced robotics: Redundancy and optimization. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1990.

[Nakauchi 02] Yasushi Nakauchi and Reid Simmons. *A Social Robot that Stands in Line.* Autonomous Robots, vol. 12, no. 3, pages 313 – 324, May 2002.

[Nokata 02] M. Nokata, K. Ikuta and H. Ishii. *Safety-optimizing method of human-care robot design and control.* In IEEE International Conference on Robotics and Automation, ICRA, volume 2, pages 1991–1996, Washington DC, USA, May 2002.

[Nonaka 04] S. Nonaka, K. Inoue, T. Arai and Y. Mae. *Evaluation of human sense of security for coexisting robots using virtual reality. 1st report: evaluation of pick and place motion of humanoid robots.* In IEEE International Conference on Robotics and Automation, ICRA, volume 3, pages 2770–2775, New Orleans, USA, April 2004.

[Odashima 06] Tadashi Odashima, Masaki Onishi, Kenji Tahara, Kentaro Takagi, Fumihiko Asano, Yo Kato, Hiromichi Nakashima, Yuichi Kobayashi, Toshiharu Mukai, Zhiwei Luo and S. Hosoe. *A Soft Human-Interactive Robot RI-MAN.* Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, page 1, October 2006.

[Oriolo 02] G. Oriolo, M. Ottavi and M. Vendittelli. *Probabilistic motion planning for redundant robots along given end-effector paths.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, volume 2, pages 1657–1662, Lausanne, Switzerland, October 2002.

[Pacchierotti 05] E. Pacchierotti, H.I. Christensen and P. Jensfelt. *Human-robot embodied interaction in hallway settings: a pilot user study.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 164–171, Nashville, USA, August 2005.

[Pacchierotti 06a] E. Pacchierotti, H.I. Christensen and P. Jensfelt. *Evaluation of Passing Distance for Social Robots.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 315–320, Hatfield, UK, September 2006.

[Pacchierotti 06b] Elena Pacchierotti, Henrik Christensen and Patric Jensfelt. *Embodied Social Interaction for Service Robots in Hallway Environments.* Field and Service Robotics, pages 293–304, 2006.

[Panangadan 04] Anand Panangadan, Maja J. Matarić and Gaurav S. Sukhatme. *Detecting Anomalous Human Interactions using Laser Rangefinders.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 2136–2141, Sendai, Japan, September 2004.

[Philippsen 04] Roland Philippsen. *Motion Planning and Obstacle Avoidance for Mobile Robots in Highly Cluttered Dynamic Environments.* PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 2004.

[Sakata 04] K. Sakata, T. Takubo, K. Inoue, S. Nonaka, Y. Mae and T. Arai. *Psychological evaluation on shape and motions of real humanoid* *robot.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 29–34, Okayama, Japan, September 2004.

[Sasaki 06] Takeshi Sasaki and Hideki Hashimoto. *Human Observation Based Mobile Robot Navigation in Intelligent Space.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 1044–1049, Beijing, China, October 2006.

[Shulz 01] D. Shulz, W. Burgard, D. Fox and A.B. Cremers. *Tracking multimple moving objects with a mobile robot.* In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, Kauai, Hawaii, December 2001.

[Siméon 01] T. Siméon, J.-P. Laumond and F. Lamiraux. *Move3D: A generic platform for path planning.* In IEEE International Symposium on Assembly and Task Planning, ISATP, pages 25–30, Fukuoka, Japan, May 2001.

[Sony] Sony. *AIBO, Artificial Intelligence roBOt.* http://support.sony-europe.com/aibo/.

[Takemura 07] H. Takemura, K. Ito and H. Mizoguchi. *Person following mobile robot under varying illumination based on distance and color information.* In IEEE International Conference on Robotics and Biomimetics, ROBIO, pages 1500–1505, Bangkok, Tailand, December 2007.

[Tang 07] Xinyu Tang, S. Thomas and N.M. Amato. *Planning with Reachable Distances: Fast Enforcement of Closure Constraints.* In IEEE International Conference on Robotics and Automation, ICRA, pages 2694–2699, Pasadena, USA, April 2007.

[Tolani 00] Deepak Tolani, Ambarish Goswami and Norman I. Badler. *Real-time inverse kinematics techniques for anthropomorphic limbs.* Graphical Models and Image Processing, vol. 62, no. 5, pages 353–388, 2000.

[Traver 00] V. J. Traver, A. P. del Pobil and M. Perez-Francisco. *Making service robots human-safe.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, volume 1, pages 696–701, Takamatsu, Japan, November 2000.

[Walters 05a] M. L. Walters, K. Dautenhahn, Koay K. L., C. Kaouri, R. Boekhorst, C. Nehaniv, I. Werry and D. Lee. *Close encounters: spatial distances between people and a robot of mechanistic appearance.* In IEEE-RAS International Conference on Humanoid Robots, Humanoids, pages 450–455, Tsukuba, Japan, August 2005.

[Walters 05b] M. L. Walters, K. Dautenhahn, R. te Boekhorst, Koay K. L., C. Kaouri, S. Woods, C. Nehaniv, D. Lee and I. Werry. *The* *influence of subjects' personality traits on personal spatial zones in a human-robot interaction experiment.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 347–352, Nashville, USA, August 2005.

[Walters 05c] M. L. Walters, S. N. Woods, K. L. Koay and K. Dautenhahn. *Practical and Methodological Challenges in Designing and Conducting Human-Robot Interaction Studies.* In Symposium on Robot Companions Hard Problems and Open Challenges in Human-Robot Interaction, AISB, pages 110–119, Hatfield, UK, April 2005.

[Xavier 05] Joao Xavier, Marco Pacheco, Daniel Castro and Antonio Ruano. *Last line, arc/circle and leg detection from laser scan data in a Player driver.* In IEEE International Conference on Robotics and Automation, ICRA, pages 3930–3935, Barcelona, Spain, April 2005.

[Yamane 03] K. Yamane and Y. Nakamura. *Natural motion animation through constraining and deconstraining at will.* IEEE Transactions on Visualization and Computer Graphics, vol. 9, no. 3, pages 352–360, July-September 2003.

[Yamaoka 08] Fumitaka Yamaoka, Takayuki Kanda, Hiroshi Ishiguro and Norihiro Hagita. *How close?: model of proximity control for information-presenting robots.* In ACM/IEEE International Conference on Human-Robot Interaction, HRI, pages 137–144, Amsterdam, The Netherlands, March 2008.

[Yoda 95] M. Yoda and Y. Shiota. *Basic study on avoidance motions for human behaviors.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 318–322, Tokyo, Japan, July 1995.

[Yoda 97] M. Yoda and Y. Shiota. *The mobile robot which passes a man.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 112–117, Sendai, Japan, October 1997.

[Yoshimi 06] T. Yoshimi, M. Nishiyama, T. Sonoura, H. Nakamoto, S. Tokura, H. Sato, F. Ozaki, N. Matsuhira and H. Mizoguchi. *Development of a Person Following Robot with Vision Based Target Detection.* In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pages 5286–5291, Beijing, China, October 2006.

[Zender 07] H. Zender, P. Jensfelt and G.-J.M. Kruijff. *Human- and Situation-Aware People Following.* In IEEE International Workshop on Robot and Human Interactive Communication, RO-MAN, pages 1131–1136, Jeju, Korea, August 2007.

[Zinn 04] M. Zinn, O. Khatib, B. Roth and J. K. Salisbury. *Playing it safe [human-friendly robots].* IEEE Robotics & Automation Magazine, vol. 11, no. 2, pages 12–21, 2004.

[Zollo 05] L. Zollo, B. Siciliano, A. De Luca, E. Guglielmelli and P. Dario. *Compliance Control for an Anthropomorphic Robot with Elastic Joints: Theory and Experiments.* Journal of Dynamic Systems, Measurement, and Control, vol. 127, no. 3, pages 321–328, September 2005.

AUTEUR: Emrah Akın ŞİŞBOT

TITRE : Towards Human-Aware Robot Motions

DIRECTEUR DE THESE : Rachid ALAMI

LIEU ET DATE DE SOUTENANCE : LAAS-CNRS, 17 OCTOBRE 2008

Résumé :

L’introduction des robots dans la vie quotidienne apporte un problème important qui “s’ajoute” au “défi standard” des robots autonomes : la présence d’hommes dans son environnement et le besoin d’interagir avec eux. Ce travail s’intéresse aux problèmes de l’interaction proche entre humains et robots, en se plaçant du point de vue des décisions de mouvement qui doivent être prises par le robot pour assurer un mouvement sûr, effectif, compréhensible et confortable pour l’homme. On présente un cadre général de planification de mouvement qui prend explicitement en compte la présence de l’homme. Ce cadre est matérialisé par deux planificateurs.

Le premier, « Human-Aware Navigation Planner », est un planificateur de navigation qui raisonne sur la sécurité, la visibilité, la posture et les préférences de l’homme pour générer des mouvements sûrs et confortables pour l’homme. Le deuxième, « Human-Aware Manipulation Planner », est un planificateur qui traite les problèmes de transfert d’objet entre l’homme et le robot. Ce planificateur transforme le problème initial de planification de mouvement en un problème beaucoup plus riche de recherche d'un chemin « pour réaliser une tache » fournissant ainsi la possibilité de raisonner à un niveau d'abstraction supérieur.

Les deux planificateurs sont intégrés dans deux plates-formes robotiques, Jido et Rackham, et validés à travers des études utilisateurs dans le cadre du projet européen COGNIRON.

MOTS-CLES :

Interaction Homme-Robot, Planification de mouvement, Robotique de service, Modélisation de l’homme, Navigation, Manipulation

DISCIPLINE ADMINISTRATIVE : Informatique